An Optimization Centered Approach to Multifidelity Aircraft Design

by

Cody Jacob Karcher

B.S., Aerospace Engineering, University of Maryland, College Park, 2014
M.S., Aeronautics and Astronautics, MIT, 2017

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Aeronautics and Astronautics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2022

Author .....
Cody Jacob Karcher
Department of Aeronautics and Astronautics
July 11, 2022

Certified by .....

Mark Drela
Terry J. Kohler Professor of Aeronautics and Astronautics
Massachusetts Institute of Technology
Thesis Supervisor

Certified by .....

R. John Hansman
T. Wilson (1953) Professor of Aeronautics and Astronautics
Massachusetts Institute of Technology
Thesis Committee

Certified by .....

Robert Haimes
Principal Research Engineer, Department of Aeronautics and Astronautics
Massachusetts Institute of Technology
Thesis Committee

Certified by .....

Joaquim R. R. A. Martins
Pauline M. Sherman Collegiate Professor of Aerospace Engineering
University of Michigan
Thesis Reader

Certified by .....

Justin Gray
Research Engineer
NASA Glenn Research Center
Thesis Reader

Certified by .....

Jonathan P. How
Richard Cockburn Maclaurin Professor of Aeronautics and Astronautics
Massachusetts Institute of Technology
Chair, Graduate Program Committee

An Optimization Centered Approach to Multifidelity Aircraft Design

by

Cody Jacob Karcher

Submitted to the Department of Aeronautics and Astronautics
on July 11, 2022, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Aeronautics and Astronautics

Abstract

A common thread across all of engineering is the concept of design optimization. When a new product is in development, it is the job of the design team to ensure the final result performs well as measured by some set of evaluative criteria, and design decisions are made throughout the process in an effort to maximize this final performance objective. Parallel to this idea is the notion of numerical optimization, where various mathematical methods are used to construct closed form optimization problems that are then solved using computing resources. The past few decades have seen extensive effort at applying numerical optimization to design optimization given their similar nature, and in aircraft design these efforts are primarily categorized as techniques of Multi-Disciplinary Analysis and Optimization (MDAO). For a number of compelling historical reasons, the MDAO community tends to assume the presence of large integrated analysis models that are pieced together in an MDAO framework before being integrated with numerical optimization. But recent work has begun to challenge this approach, instead assuming that aircraft design problems are fundamentally compatible with a fast and efficient form of optimization called Geometric Programming (GP) and adapting all analysis models to fit this form. Both the analysis centered (MDAO) and optimization centered (GP) approaches have considerable merit, but to date remain fundamentally incompatible. This work takes the best elements from both approaches, the efficient mathematical structure from GP and the natural ability to integrate existing black box analysis models from traditional MDAO, and develops two new optimization algorithms that enable an optimization centered, multifidelity approach to aircraft design. The new algorithms are benchmarked against a current state of the art algorithm using a set of example problems, and the new design approach is applied to two representative aircraft design case studies.

Thesis Supervisor: Mark Drela
Title: Terry J. Kohler Professor of Aeronautics and Astronautics
Massachusetts Institute of Technology

Thesis Committee: R. John Hansman
Title: T. Wilson (1953) Professor of Aeronautics and Astronautics
Massachusetts Institute of Technology

Thesis Committee: Robert Haimes
Title: Principal Research Engineer, Department of Aeronautics and Astronautics
Massachusetts Institute of Technology

Thesis Reader: Joaquim R. R. A. Martins
Title: Pauline M. Sherman Collegiate Professor of Aerospace Engineering
University of Michigan

Thesis Reader: Justin Gray
Title: Research Engineer
NASA Glenn Research Center

Contents

Chapter 1

The Analysis First Approach to Aircraft Design

1.1 Why Use Numerical Optimization in Aircraft Design?

In the early days of aviation, the critical challenge was designing an aircraft that could get off the ground. These so called 'feasible designs' were on the cusp of being impossible to achieve, requiring state of the art technology in aerodynamics, structures, control and propulsion (and perhaps a bit of luck) to even takeoff. But in today's design environment, flight physics is now so well understood that the question "Is there a feasible design?" has largely been increasingly replaced by the question "What is the best aircraft design?" This subtle change in focus implies that there are many possible aircraft designs that can fulfill a given set of design requirements, and that tradeoffs exist between these various feasible designs. The design team is responsible for searching through this set of feasible options and choosing the one that has the 'best' performance. In the classical sense, this process of selecting the 'best' design is commonly referred to as *design optimization*.

A key tool used in design optimization is *numerical optimization*. Imagine there exists an oracle capable of taking in any potential aircraft design and instantly predicting with perfect accuracy how the aircraft will perform when it is eventually built. If given this oracle, the design process can be modeled as follows (Figure 1-1):

1. A human chooses any candidate aircraft design
2. The oracle is queried to produce the performance data associated with the candidate design
3. The human examines the performance data and decides if any changes should be made to the design
4. If changes are necessary then produce a new design to pass to the oracle, if not then proceed with the selected design

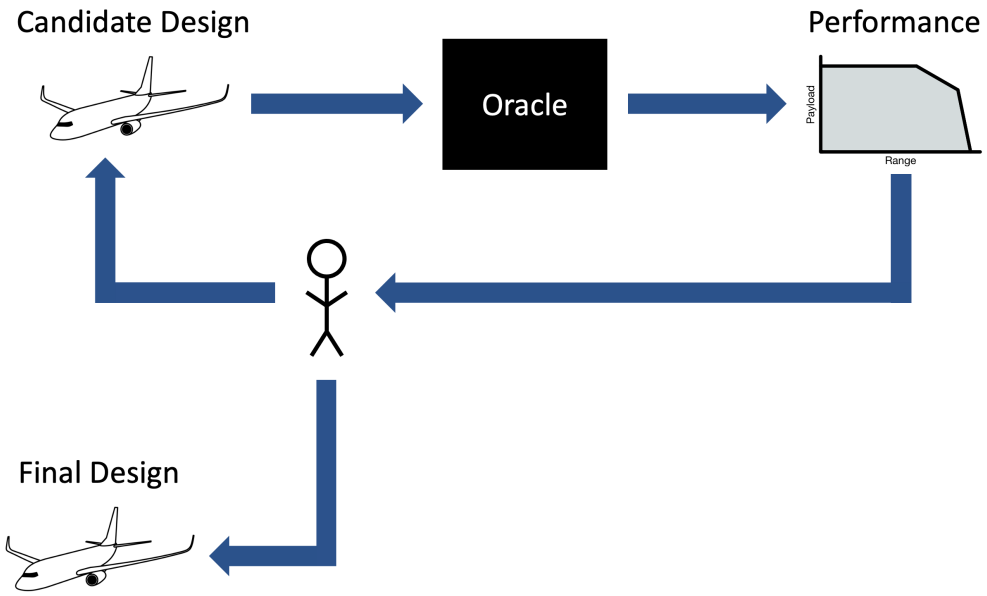


Figure 1-1: The idealized design process if given a perfect aircraft analysis oracle

If the oracle is free to evaluate (ie, if it has no financial cost and returns data instantaneously) then the bandwidth of the human decision maker is the fundamental limiter on the speed of the design cycle. But by implementing numerical optimization, that is a set of mathematical

rules for systematically interpreting the performance data to produce a new candidate design, the burden of the design cycle can be transferred to a computing architecture (Figure 1-2). The human decision maker now serves a supervisory role in the core design loop and eventually selects the final design.

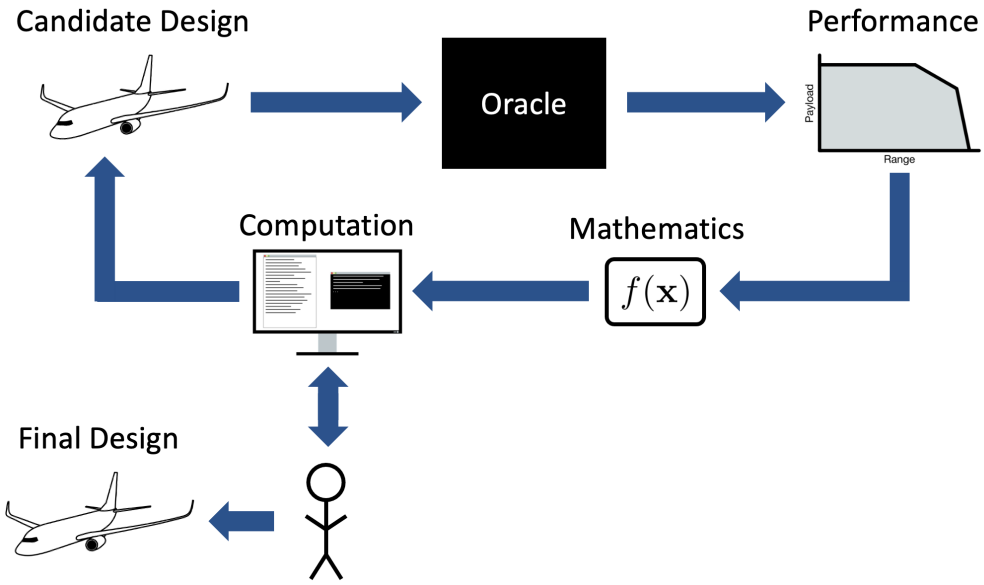


Figure 1-2: The idealized design process if given a perfect aircraft analysis oracle accelerated using modern computational tools

Though the human input is significantly reduced during the actual cycling, significant effort must now be expended in developing and implementing the numerical optimization techniques that close the design loop.

1.2 The Basics of Numerical Optimization

Numerical optimization in the general sense seeks to identify a set of 'optimal' choices that minimize or maximize some objective function. Real world objectives are most often economic in nature: minimize cost, maximize profit, and similar, though in practice, objective functions based in economics are challenging to write in meaningful mathematical form, and so engineering surrogates are often used instead. In aviation, these surrogates often include travel time, vehicle weight, or fuel consumption.

An optimization problem can be written mathematically as the minimization of the objective function:

$$\text{minimize} \quad f(\mathbf{x}) \tag{1.1}$$

where the vector $\mathbf{x}$ represents the set of choices that can be made to minimize the objective function. In the context of aircraft design, these choices are typically called *design variables* and frequently include quantities like wingspan, engine diameter, fuselage length, and similar.

Often in engineering design, the problem is bounded by *constraints* which prevent the absolute minimum of the objective function from being obtained:

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & \mathbf{g}_i(\mathbf{x}) \leq 0, \ i = 1, \ldots, N \\ & \mathbf{h}_j(\mathbf{x}) = 0, \ j = 1, \ldots, M \end{aligned} \tag{1.2}$$

These constraints are usually divided into *inequality* constraints and *equality* constraints, as shown above by $\mathbf{g}_i(\mathbf{x})$ and $\mathbf{h}_j(\mathbf{x})$ respectively.

Design variables x_i can either be in the continuous space $\mathbb{R}$, integers in $\mathbb{Z}$, or some combination of the two, leading to complex mathematical subspaces that must be traversed when solving the optimization problem. The dimensionality of the variable vector $\mathbf{x}$ can also be exceedingly large, making the computational tractability of these problems border on the impossible. All things considered, the problems posed by Equations 1.1 and 1.2 are not easy to solve. This work somewhat reduces the level of difficulty by considering only constrained, continuous optimization problems.

Note that a constrained optimization problem can be re-written as an *unconstrained* opti-

mization problem by means of a Lagrangian function:

$$\text{minimize} \quad \mathcal{L}(\mathbf{x}, \mathbf{u}, \mathbf{v}) = f(\mathbf{x}) + \sum_{i=1}^{N} u_i g_i(\mathbf{x}) + \sum_{j=1}^{M} v_j h_j(\mathbf{x}) \tag{1.3}$$

where the variables $\mathbf{u}$ and $\mathbf{v}$ are new variables called Lagrange Multipliers. Consider for the moment only the equality constraints $\mathbf{h}_j$ and the corresponding Lagrange multipliers v_i. If the Lagrange multipliers are all set to some large value, say $v_i = 10^{100}$, then the objective function will likely be minimized only when all of the equality constraints are minimized, regardless of the function $f(\mathbf{x})$.

Given the existence of the Lagrangian function it is valid to ask, why bother solving constrained optimization problems at all? The answer is that often times the constrained form of the problem can be solved more efficiently than the unconstrained problem defined by the Lagrangian [1].

1.3 Analysis Models

1.3.1 The Problem with Analysis Oracles

Even armed with the most advanced techniques of numerical optimization, the idealized design process of Figure 1-2 is impossible to achieve since, unsurprisingly, idealized analysis oracles do not exist. Instead, designers rely on *analysis models* to fill this role in the design process.

At the most basic level, an analysis model is a composite function that maps a vector of inputs to a vector of outputs. A typical analysis model might be a simple equation that can be evaluated by hand (ex: the lift or drag equations), or a complicated software package (ex: CFD, FEA), but the fundamental nature of analysis remains the same regardless of the complexity of the actual composite function itself.

An analysis model generally consists of three major elements (Figure 1-3). First is a pre-processor that interprets the input into meaningful parameters for use in the analysis model. One of the most common examples of this step would be meshing for CFD, where the variables that define aircraft geometry are translated into a volume mesh that can be used to solve the appropriate fluid dynamics equations. But for a simpler model, this might mean extracting the meaningful parameters from a larger input vector. For example, a wing defined by a series of airfoil slices might need to have the individual sub-areas summed to compute the planform area for use in the lift equation. In all almost all situations where an analysis model is utilized, some degree of geometry translation must occur.

Second is the core set of mathematics that make up the analysis model. These mathematics can take nearly any form, the most common being a series of equations solved in a cascade, a matrix solve, or a set of differential equations. In general, the mathematics in any given analysis model are complicated, and require a great deal of discipline specific expertise to understand.

Finally is the computational architecture used to run through the mathematical expressions. For simple analysis models (ex: thin airfoil theory, lifting line theory, etc.) the human engineer may perform the analysis manually, though the abundance of computational power in the modern era means that most analysis models are now implemented as a software package.

1.3.2 Analysis Model Fidelity

From their construction, it is easy to see the ways that real analysis models can differ from an idealized analysis oracle. First, real analysis models implemented as a software tool require that inputs be read in some specific form, and therefore translation of the relevant design variables must be done by some larger framework. Though much progress in software integration has been made in recent years, the fact remains that for most tools, an tool specific input file must be generated to represent each candidate design.

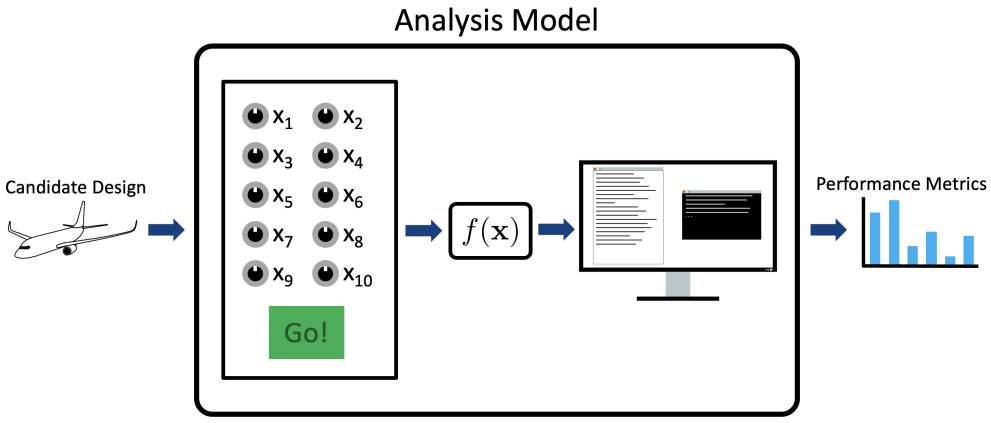


Figure 1-3: The anatomy of a typical analysis model: an interpreter, a set of mathematics, and a computational architecture

Second, the complicated nature of the mathematics in a typical analysis model generally means that a typical analysis model will only produce some subset of performance metrics, not the entire set. For example, FEA codes may provide information on the maximum load factor an aircraft can sustain without structural failure, but seldom provide any insight into the range or fuel efficiency of the aircraft in question. Thus, many analysis models must be used in combination to produce the require set of performance metrics.

Third, a real analysis model has some cost to evaluate. Taken to the extreme, one way of mapping a candidate design to its performance metrics is to actually build and test the design. This approach usually involves an impractical monetary expenditure, combined with a significant time delay. But even computational models require resources to evaluate in the form of engineering hours and computational time.

Finally, real analysis models produce performance metrics that have some amount of *uncertainty*, due to the nature of the assumptions that have to be made in modeling real physics. For example, a 2D airfoil code assumes that 3D flow effects on the wing are negligible, wind tunnel models assume scaling relative to the final flight article, and CFD assumes the discretization of a continuous fluid. Again taking the extreme example of this, a full build and test program will typically result in a very small degree of uncertainty in the result,

as the primary source of uncertainty will be measurement error of the sensors being used to gather the data. However, simple models will often have a high degree of uncertainty since the simplicity of the model generally comes from making a large number of simplifying assumptions.

This perceived relationship between uncertainty and evaluation cost generally holds true: as uncertainty is decreased, evaluation cost increases, giving rise to the notion of analysis model *fidelity*. A low fidelity model is one that makes many simplifying assumptions in order to capture only the most critical elements of the true underlying physics. On the other hand, a high fidelity model is one that strives to model the physics as exactly as is practical. As a result, low fidelity models tend to be very inexpensive to evaluate, while high fidelity models can only be queried at significant cost.

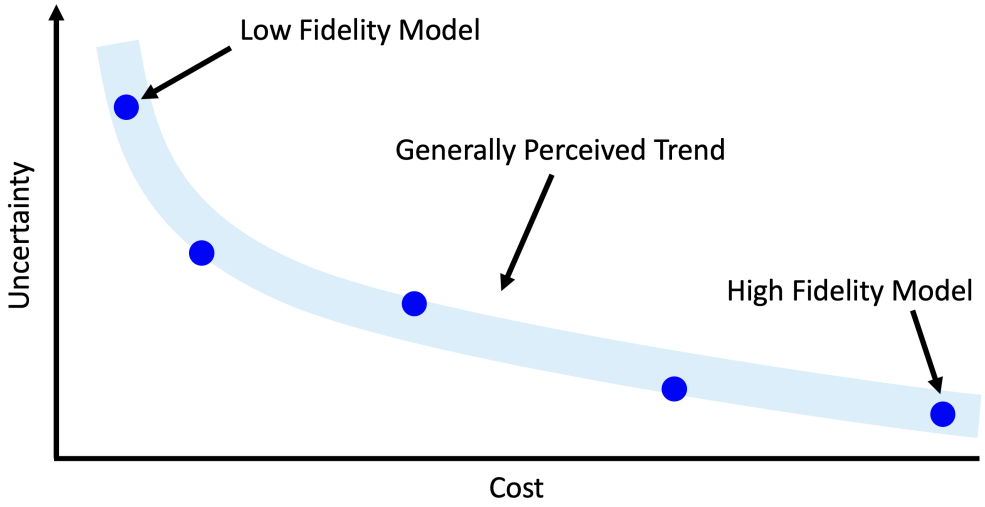


Figure 1-4: The perceived trend in analysis model fidelity, cost of evaluation increases as uncertainty is reduced, with diminishing returns as more cost is paid

1.3.3 Black Box Analysis Models

As has been alluded to, the core mathematics in an individual analysis model tend to be highly complicated and require a high degree of discipline specific expertise to utilize effectively. As a result, analysis models beyond the lowest levels of fidelity tend to be developed by highly knowledgeable subject matter experts and packaged as self contained software

tools. A software tool can then be operated by a less knowledgeable user in a *black box* fashion, passing in inputs and obtaining outputs with no knowledge of the actual mathematics contained in the software tool.

Black box analysis models serve a critical role in organizations that perform aircraft design activities. Simply as a practical matter, black box analyses remove the requirement that every engineer who touches an analysis model have detailed knowledge of the mathematics being used to represent the flight physics. Few designers have sufficient technical depth to handle high fidelity CFD, FEA, and engine deck analysis all at the level required on a real aircraft design program. In addition, black box analysis tools can be adapted for a wide variety of design efforts. Rather than developing a new CFD tool as a part of each new design effort, an aircraft design company can maintain a single set of CFD tools with a core group of CFD experts. But perhaps most importantly, black box analysis tools undergo verification and validation by being utilized across multiple design efforts, resulting in a high degree of trust in these black box tools. And a tool is only as useful as the degree of trust an engineer has in the result.

For these reasons, any approach to design optimization *must* utilize black box analysis software as a matter of practicality.

1.4 Multi-Disciplinary Analysis and Optimization (MDAO)

1.4.1 Introduction to MDAO

Thousands of analysis models and software tools exist for aircraft analysis, but consider selecting a single tool for aerodynamic analysis. By setting up the proper input file, a wing geometry can be analyzed using this chosen model to produce performance metrics like lift, drag, and moment. However, a wing under load will flex to some new shape, fundamentally changing the original geometry that was being analyzed by the aerodynamic analysis tool. To account for this shape change, the output of the aerodynamic model (in this case the

surface pressures on the wing) can be passed as the input to a structural deflection model that computes the new wing shape. This new wing shape is now passed back to the aerodynamic analysis model, producing a new set of surface pressures, and the process repeats ad nauseam until the difference between inputs in successive cycles is less than some predefined tolerance. The result is a coupled aeroelastic analysis model that has been constructed from one analysis model of each discipline (Figure 1-5).

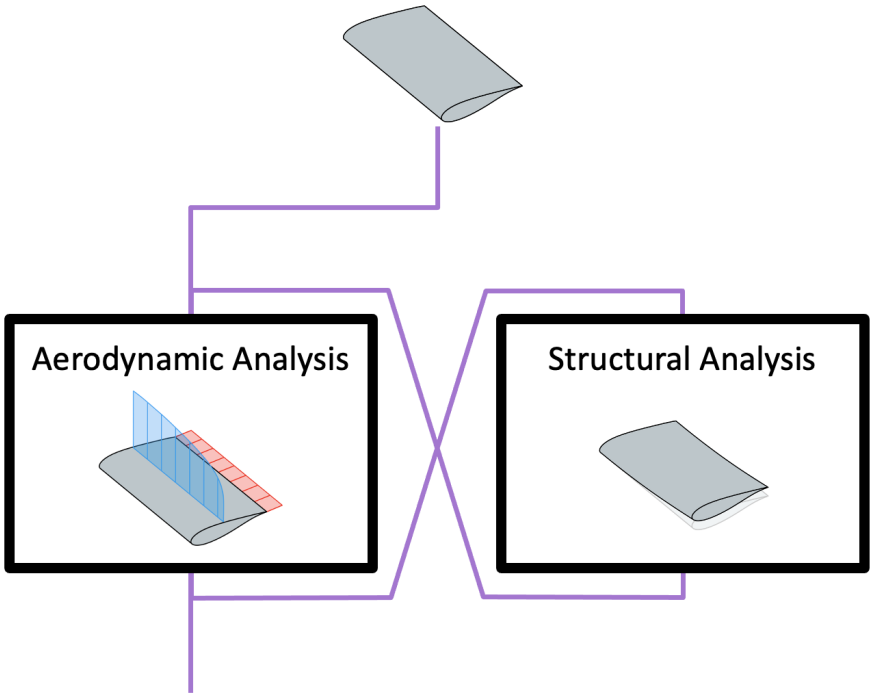


Figure 1-5: A diagram of a coupled aeroelastic model consisting of two separate analysis models

This newly coupled model looks a lot like the analysis oracle! At least it does for wing design. And so, an numerical optimizer is wrapped around this coupled model in order to find the wing design that minimizes weight, drag, or some other desired objective (Figure 1-6).

As early as the 1960s, aircraft designers realized the potential for using numerical optimization and coupled analysis models in a fashion similar to Figure 1-6 [2] to help inform the larger design optimization process. And by the early 1990s, the study of such topics had matured into the field known today as Multi-Disciplinary Analysis and Optimization

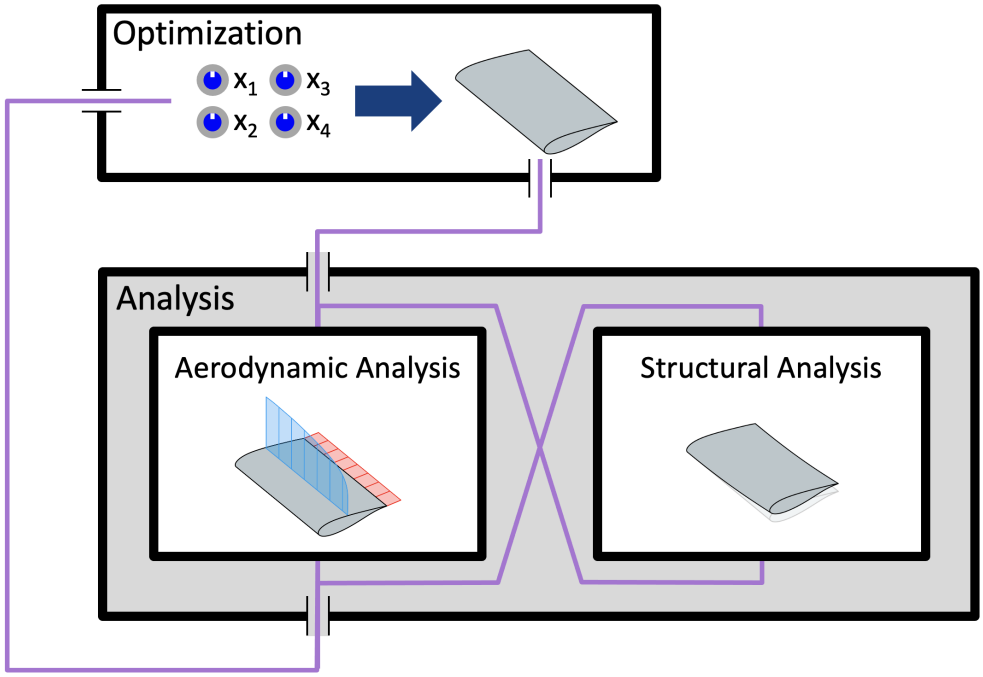


Figure 1-6: Optimization wrapped around a coupled aeroelastic analysis model

(MDAO)[1].

MDAO primarily focuses on optimizing complex systems that are made up of sub-disciplines that are in active conflict with each other. Take the aeroelastic wing example. Considering only the aerodynamics of a wing will yield a long skinny wing that minimizes drag, but that must be made with Unobtainium to be structurally feasible. Similarly, considering only the structural analysis outputs of a wing will yield a very thick I-beam capable of generating no lift (see Figure 1-7 for a humorous interpretation). And so, the field of MDAO has largely focused on understanding these types of systems by pursuing how to best model the complex discipline interactions, how to tie the various models together, and how to best integrate numerical optimization with these large networks of models.

[1]Though this is my preferred terminology, the field is also called Multidisciplinary Design Optimization (MDO) or less commonly Multi-Disciplinary Optimization (MDO). All refer to the same area of work.

Figure 1-7: A drawing of each discipline's ideal airplane, drawn by C. W. Miller, original publication unknown

1.4.2 MDAO Architectures

Central to MDAO is the concept of an *architecture*. Given the need for black box analyses, most of the active research in MDAO focuses on schemes to cleverly connect the various analysis models and optimizers to obtain the 'best' design for the least computational expense. Though dozens of architectures exist [3], three major categories have emerged. The technical details of a number of MDAO architectures are outlined by Martins and Lambe [3], but the intent here is to provide intuition to the reader and to motivate the work, and so discussion will be at the overview level. The discussed concepts come primarily from [3] and [4].

Multi-Discipline Feasible (MDF) Architectures

Multi-Discipline Feasible (MDF) architectures are by far the simplest and easiest to implement. In fact, the MDF architecture has already been shown in Figure 1-6 [4]. Existing analysis models (typically black box software tools) are combined together in a large, coupled analysis architecture creating in essence a non-idealized analysis oracle that is expensive to evaluate and has some degree of uncertainty.

MDF architectures have a number of key advantages that make them by far the most popular in practice. First, they are intellectually simple, requiring only the wiring of a bunch of inputs and outputs together in some kind of automated software package. Many packages exist to assist in this process, the most notable perhaps being Phoenix Integration's ModelCenter, though flexible programming languages like Python are also frequently used to create this massive integrated tool. With regards to development (and not evaluation), this wiring between models is by far the most time consuming element of the MDF approach.

Second, MDF architectures produce a feasible aircraft design at each evaluation. This property is why MDF architectures are named multi-discipline feasible, because the design that comes out of the integrated analysis tool will be feasible in each of the connected disciplines, a property not shared by the remaining architecture types. Since each evaluation produces a feasible aircraft, MDF architectures are perceived to be the easiest to interpret by the humans on the design team.

Despite these two significant advantages, MDF architectures are by far the most inefficient computationally. Ensuring feasibility of each design requires cycling through the coupled analysis model to completion at each iteration of the optimizer. Even for the simple aerostructural case (Figure 1-6), one call to the coupled model may be 10 or more calls to both the aerodynamic and structural models. Once models for propulsion, control, stability, performance, and other disciplines are added, even one evaluation of the coupled model can be prohibitively expensive. And so the interpretability of MDF comes at the cost of tremendous computational expense.

Individual Discipline Feasible (IDF) Architectures

Individual Discipline Feasible (IDF) architectures are by far the most challenging to understand because they occupy a cloudy middle ground between two extremes. Key to understanding IDF architectures is to realize that numerical optimization algorithms (in stark contrast with design optimization cycles) do not require that a design be feasible at each iteration of the optimizer, only the final design must be feasible.

Recall the simple aeroelastic model. Imagine that rather than thinking of deflections as outputs from the structural model, that a set of deflection variables are introduced in the optimization problem. Now, when the first call is made to the aerodynamic analysis tool, these deflection variables are also passed in, resulting in a (randomly initialized) deflected geometry. Obviously, information from the structural analysis model must be introduced to inform the optimizer how to choose the set of deflection variables that are feasible with respect to flight physics. So, a set of surface pressure variables is introduced to the optimization problem (randomly initialized), and passed into the structural model along with the undeformed wing shape. The optimizer then enforces a set of constraints that ensure the deflection variables being passed into the aerodynamic analysis model are equal to the output of the structural model, and that the pressure variables being passed into the structural model are equal to the output of the aerodynamic model.

This is the core principle of IDF, that any variables shared between disciplines have a copy unique to each discipline that uses them, and the optimizer simply enforces that all of these variables must be equal when the algorithm terminates (Figure 1-8)

Clearly, IDF architectures are far less intuitive than MDF architectures. Rather than simply wiring together inputs and outputs, an IDF architecture requires that optimization play a key role in determining *feasibility* not just optimality. At each iteration of the optimizer, the individual analysis models are operating on a feasible design (ie, each individual discipline is feasible), but each analysis model is not operating on the *same* design, making the full multi-disciplinary design problem infeasible. This traversal of infeasible space tends to lead

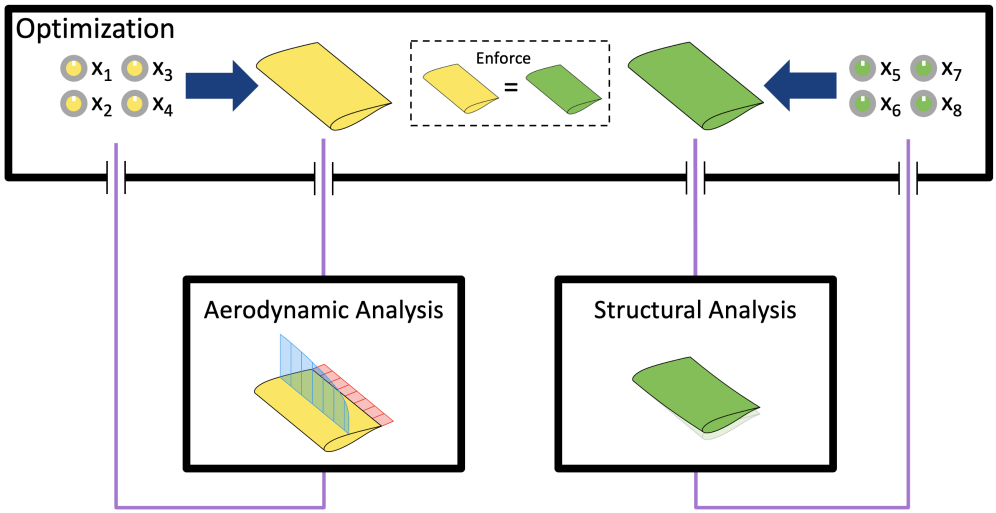


Figure 1-8: The IDF architecture applied to the aeroelastic wing optimization problem

to considerable confusion among engineering experts not familiar with the concept, because it impossible to interpret the candidate designs at each iteration of the optimizer, as the design is likely to actually be infeasible.

Despite this drawback, IDF architectures can be constructed in ways that are extremely computationally efficient due to the ability to run analysis models entirely in parallel. As seen in the diagram (Figure 1-8), discipline analysis models are naturally siloed and are easy to separate into different computing blocks. Additionally the issue of internal closure of a coupled analysis model has disappeared and is instead being handled by the optimizer. Though this in theory should increase the number of optimization iterations, on balance it dramatically decreases the number of calls that must be made to the individual discipline analysis models.

Simultaneous Analysis aNd Design (SAND) Architectures

Simultaneous Analysis aNd Design (SAND) architectures encode each and every analysis equation as a constraint in the optimization problem. It is challenging to convey the scope of this statement, so imagine using an aerodynamic model that utilizes a CFD mesh with a few million cells. Every flow parameter in every cell would become a variable in the optimization

problem. Boundary conditions are enforced as constraints, as one might assume, but the actual discretized differential equations that govern the flow physics also are enforced as constraints, as are the equations that translate the flow field data to global parameters like lift, drag, and moment. By the time this exercise is complete, tens of millions of design variables are being bounded by tens of millions of constraints, and this is just for a single discipline.

The implications of the SAND approach are staggering, because each and every black box analysis tool must be unpacked in this way and directly integrated into the optimization problem. Representing this visually is challenging, as the whole problem is essentially a monolith, but Figure 1-9 may provide some insight.

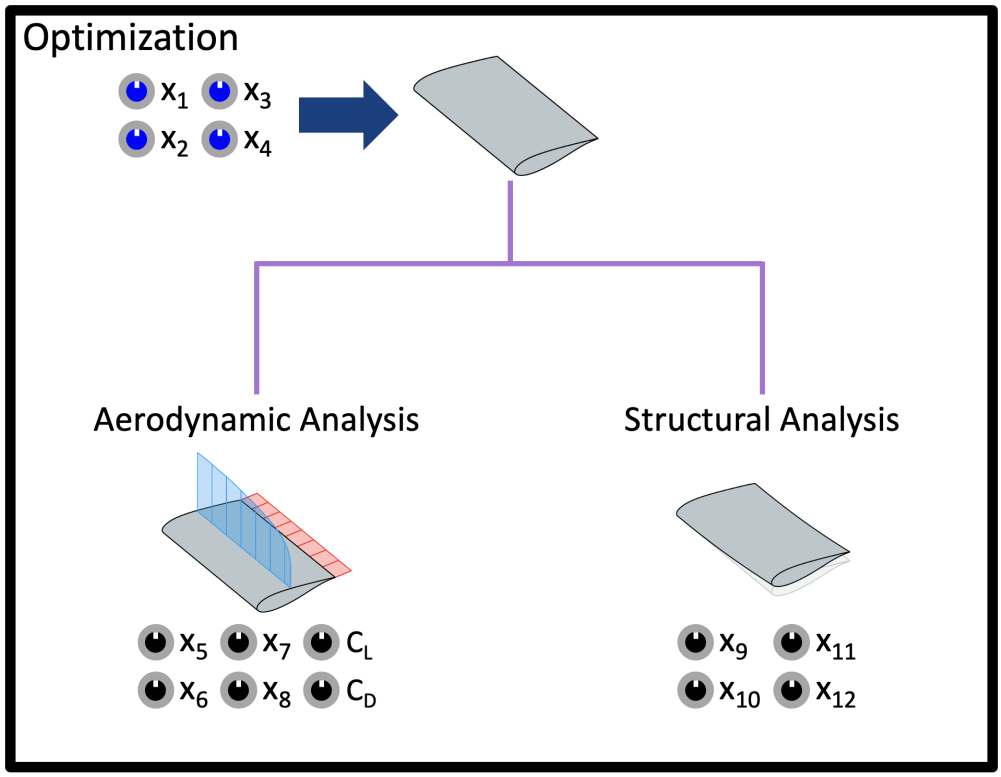


Figure 1-9: Optimization wrapped around a coupled aeroelastic analysis model

Despite the staggering scale of SAND architectures, they are theoretically the most computationally efficient. In the same way that IDF removed the loop closures that coupled

analysis models together in MDF, SAND removes the internal closure loops that are often present in IDF analysis models that solve residual equations.

Although SAND architectures are attractive in theory, practicality has essentially removed them from serious consideration in the MDAO community due to the rejection of existing black box analysis models. As stated in 1993 by Cramer et. al., "We feel that the [SAND] approach remains theoretically attractive because of the probability that it will be the least expensive computationally. Unfortunately, it requires a higher degree of software integration than is likely to be achieved in the near future for realistic applications." [4] Martins and Lambe echo a similar sentiment, "In engineering design, software for discipline analysis often operates in a black-box fashion, directly computing the coupling variables while hiding the discipline-analysis residuals and state variables. Even if the software can be modified to return the residuals, the cost and effort required may be excessive. Therefore, most practical MDO problems require an architecture that can take advantage of existing discipline analysis software." [3]

By rejecting the SAND approach, traditional MDAO has become what could be called an *analysis first* approach to aircraft design optimization, since optimization is only integrated around existing analysis models that have been predefined.

Table 1.1 summarizes the pros and cons of the MDF, IDF, and SAND MDAO architectures (adopted from Cramer et. al. [4])

Table 1.1: A summary of the pros and cons of various MDAO architectures

	MDF	IDF	SAND
Implementation	Easy	Medium	Hard
Optimization	Very Inefficient	Inefficient	Efficient
Number of Variables	Few	Many	Very Many
Parallelization	Not Really	Yes	Yes
Existing Analysis Tools	Yes	Yes	No

1.5 Reviving Old Ideas

While MDAO has been highly successful in many areas, there is also a widespread belief that the promise of MDAO has under-delivered on expectations from the early 1990s. The summary paper by Martins and Lambe [3] shows that for all of the various architectures, no clear winner emerges. Instead, each architecture was typically created in response to some specific design problem, and performs poorly as a general approach to MDAO.

Recently, there have been challenges to fundamental assumptions that originally drove the field to MDF and to a lesser extent IDF architectures, especially in cases focused on system level optimization using relatively low-order or low-fidelity physical models. Examples are tools like PASS [5], WingMOD [6], TASOPT [7], and SUAVE [8] take a far more integrated approach to design, slowly chipping away at the barriers surrounding black box analysis tools. But more importantly, computational capacity has made leaps and bounds since the early 1990s when the MDAO community largely abandoned SAND architectures. Previous concerns of challenges to implementation have lessened due to the widespread availability of flexible coding architectures, and the massive increases in available RAM and disk memory has significantly reduced concerns that come from optimization problems with large numbers of variables.

And so, perhaps it is time to take another look at SAND architectures and see if computational tools have advanced to the point that they are now a viable method for MDAO. Indeed, this idea is the genesis of a new approach to aircraft design centered on Geometric Programming.

Chapter 2

The Optimization First Approach to Aircraft Design

2.1 Aircraft Design as Optimization

Optimization is a natural mathematical language for expressing aircraft design. Frequently, the goal is to create a design that minimizes the weight of fuel consumed during flight:

$$\text{minimize} \quad W_{\text{fuel}} \tag{2.1}$$

Implicitly weight cannot be negative, so solving this specific optimization problem will always result in an aircraft that burns zero fuel, ie, no aircraft at all.

Thus, there will need to be constraints on the problem that ensure an aircraft is designed. First, fuel consumed by an aircraft flying a certain range can be computed via the Breguet Range equation [9]:

$$W_{\text{fuel}} \geq W_{\text{initial}} \left(1 - \exp\left(\frac{-g \quad R \quad \text{SFC}}{(L/D) \quad V}\right)\right) \tag{2.2}$$

Note that this constraint utilizes an inequality relational operator rather than an equality. There is no reason physical reason that, if given only this constraint, the fuel weight cannot be greater than the amount required as specified by the Breguet Range equation. However, the objective function will strive to minimize the fuel weight and keep this constraint tight. There are other benefits to using inequality relationships here that will be discussed along with convexity in a later section.

The introduction of the Breguet Range equation has also introduced new variables. As before, leaving these variables free allows for the fuel weight to reach its minimum of zero, either through initial weight being zero, range being zero, lift to drag ration being infinite, or similar. And so more constraints must be introduced that bound these newly introduced variables.

The initial weight of the aircraft can be computed as the sum of all the individual weight components

$$W_{\text{initial}} \geq W_{\text{wings}} + W_{\text{fuselage}} + W_{\text{engines}} + W_{\text{fuel}} + ... + \sum_{i=1}^{N} W_i \tag{2.3}$$

Range can be bounded to be greater than some predefined requirement:

$$R \geq R_{\text{required}} \tag{2.4}$$

Lift can be computed via the lift equation:

$$L = \frac{1}{2}\rho V^2 S C_L \tag{2.5}$$

but must also be sufficient for the aircraft to fly:

$$L \geq W_{\text{initial}} \tag{2.6}$$

And drag can be computed via the drag equation:

$$D \geq \frac{1}{2}\rho V^2 S C_D \tag{2.7}$$

Once again, new variables have been introduced in these new constraints and must be similarly bounded. Drag coefficient can be modeled as the sum of fuselage drag coefficient, profile drag coefficient, and induced drag coefficient:

$$C_D \geq \frac{A_{C_{D_0}}}{S} + kC_f\frac{S_{\text{wet}}}{S} + \frac{C_L^2}{\pi Ae} \tag{2.8}$$

and so on. Continuing this process eventually yields an optimization problem for aircraft design that can be written as follows [10]:

Minimize:

$$W_{\text{fuel,out}} + W_{\text{fuel,ret}} \tag{2.9}$$

Subject to the following constraints, which are classified for readability.

Steady level flight relations:

$$\begin{aligned} W &= \frac{1}{2}\rho V^2 C_L S \\ T &\geq \frac{1}{2}\rho V^2 C_D S \\ Re &= \frac{\rho V S^{1/2}}{A^{1/2}\mu} \end{aligned} \tag{2.10}$$

Landing flight condition:

$$\begin{aligned} W_{\text{MTO}} &\leq \frac{1}{2}\rho_{\text{sl}} V_{\text{stall}}^2 C_{L,\max} S \\ V_{\text{stall}} &\leq 38 \end{aligned} \tag{2.11}$$

Sprint flight condition:

$$\begin{aligned} P_{\max} &\geq \frac{T_{\text{sprint}} V_{\text{sprint}}}{\eta_{0,\text{sprint}}} \\ V_{\text{sprint}} &\geq 150 \end{aligned} \tag{2.12}$$

Drag model:

$$\begin{aligned} C_D \geq & \frac{0.5}{S} + C_{D_p} + \frac{C_L^2}{\pi e A} \\ 1 \geq & 2.56 \frac{C_L^{5.88}}{\tau^{3.32} Re^{1.54} C_{D_p}^{2.26}} + 3.80 \times 10^{-9} \frac{\tau^{6.23}}{C_L^{0.92} Re^{1.38} C_{D_p}^{9.57}} + \\ & 2.20 \times 10^{-3} \frac{\tau^{0.03} Re^{0.14}}{C_L^{0.01} C_{D_p}^{0.73}} + 1.19 \times 10^{4} \frac{C_L^{9.78} \tau^{1.76}}{Re^{1.00} C_{D_p}^{0.91}} + \\ & 6.14 \times 10^{-6} \frac{C_L^{6.53}}{\tau^{0.52} Re^{0.99} C_{D_p}^{5.19}} \end{aligned} \tag{2.13}$$

Propulsive efficiency:

$$\begin{aligned} \eta_0 &\leq \eta_{\text{eng}} \eta_{\text{prop}} \\ \eta_{\text{prop}} &\leq \eta_i \eta_v \\ 4\eta_i + \frac{T \eta_i^2}{\frac{1}{2} \rho V^2 A_{\text{prop}}} &\leq 4 \end{aligned} \tag{2.14}$$

Range constraints:

$$\begin{aligned} R &\geq 5000 \times 10^3 \\ z_{\text{bre}} &\geq \frac{gRT}{h_{\text{fuel}} \eta_0 W} \\ \frac{W_{\text{fuel}}}{W} &\geq z_{\text{bre}} + \frac{z_{\text{bre}}^2}{2} + \frac{z_{\text{bre}}^3}{6} + \frac{z_{\text{bre}}^4}{24} \end{aligned} \tag{2.15}$$

Weight relations:

$$
\begin{aligned}
W_{\text{pay}} &\geq 500g \\
\tilde{W} &\geq W_{\text{fixed}} + W_{\text{pay}} + W_{\text{eng}} \\
W_{\text{zfw}} &\geq \tilde{W} + W_{\text{wing}} \\
W_{\text{eng}} &\geq 0.0372 P_{\text{max}}^{0.803} \\
\frac{W_{\text{wing}}}{f_{\text{wadd}}} &\geq W_{\text{web}} + W_{\text{cap}} \\
W_{\text{out}} &\geq W_{\text{zfw}} + W_{\text{fuel,ret}} \\
W_{\text{MTO}} &\geq W_{\text{out}} + W_{\text{fuel,out}} \\
W_{\text{sprint}} &= W_{\text{out}}
\end{aligned}
\tag{2.16}
$$

Wing structural model:

$$
\begin{aligned}
2q &\geq 1 + p \\
p &\geq 1.9 \\
\tau &\leq 0.15 \\
\bar{M}_r &\geq \frac{\tilde{W} A p}{24} \\
0.92 \bar{w} \tau \bar{t}_{\text{cap}}^2 + \bar{I}_{\text{cap}} &\leq \frac{0.92^2}{2} \bar{w} \tau^2 \bar{t}_{\text{cap}} \\
8 &\geq \frac{N_{\text{lift}} \bar{M}_r A q^2 \tau}{S \bar{I}_{\text{cap}} \sigma_{\text{max}}} \\
12 &\geq \frac{A \tilde{W} N_{\text{lift}} q^2}{\tau S \bar{t}_{\text{web}} \sigma_{\text{max,shear}}} \\
\nu^{3.94} &\geq 0.86 p^{-2.38} + 0.14 p^{0.56} \\
W_{\text{cap}} &\geq \frac{8 \rho_{\text{cap}} g \tilde{w} \bar{t}_{\text{cap}} S^{3/2} \nu}{3 A^{1/2}} \\
W_{\text{web}} &\geq \frac{8 \rho_{\text{web}} g r_h \tau \bar{t}_{\text{web}} S^{3/2} \nu}{3 A^{1/2}}
\end{aligned}
\tag{2.17}
$$

The optimization problem defined by Equations 2.9 through 2.17 represents a SAND approach to aircraft design, since analysis models have been directly encoded as constraints

in the optimization problem. But surprisingly, this optimization problem is a Geometric Program.

2.2 Geometric Programming

Geometric Programming (GP) is a type of constrained, continuous optimization with a number of unique properties [11]. Geometric Programming has received attention from a wide range of fields [12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27], but Hoburg [10] and Torenbeek [28] were the first to note the potential of geometric programming for aircraft design. Perhaps the most interesting property of geometric programs is that they become convex upon transformation to log-log space[1] [11, 10] [11]. Since convex optimization problems can be solved rapidly for a guaranteed global optimum, geometric programming is an extremely powerful method for aircraft design.

Mathematical Formulation

Geometric Programs are built upon the two fundamental building blocks of monomials and posynomials. A monomial function is defined as the product of a leading constant with each design variable raised to a real power [11]:

$$m(\mathbf{x}) = c{x_1}^{a_1}{x_2}^{a_2}... \ {x_n}^{a_n} = c\prod_{i=1}^{N} x_i^{a_i} \tag{2.18}$$

A posynomial is simply the sum of monomials, which can be defined in notation as[2] [11]:

$$p(\mathbf{x}) = m_1(\mathbf{x}) + m_2(\mathbf{x}) + ... + m_n(\mathbf{x}) = \sum_{k=1}^{K} c_k \prod_{i=1}^{N} x_i^{a_{ik}} \tag{2.19}$$

[1] Convexity will be discussed in detail in Section 2.3

[2] A monomial is a special case of a posynomial with only a single term. This note is relevant for some of the more condensed notations.

From these two building blocks, it is possible to construct the definition of a GP in standard form[3] [11]:

$$\begin{aligned} \text{minimize} \quad & p_0(\mathbf{x}) \\ \text{subject to} \quad & m_i(\mathbf{x}) = 1,\ i = 1, \ldots, N \\ & p_j(\mathbf{x}) \leq 1,\ j = 1, \ldots, M \end{aligned} \tag{2.20}$$

When constraints and objectives can be written in the form specified in Equation 2.20 the problem is said to be *GP compatible*.

Solution Method

In general, the formulation defined by Equation 2.20 is a non-linear and non-convex optimization problem, making it extremely difficult to solve. However, a GP can be transformed into a convex optimization problem by undergoing a logarithmic transformation [11].

Consider the transformation of the standard variables x_i into logspace variables y_i:

$$x_i = e^{y_i} \qquad \text{or} \qquad y_i = \log x_i \tag{2.21}$$

And the transformed GP:

$$\begin{aligned} \text{minimize} \quad & \log p_0(e^{\mathbf{y}}) \\ \text{subject to} \quad & \log m_i(e^{\mathbf{y}}) = 0,\ i = 1, \ldots, N \\ & \log p_j(e^{\mathbf{y}}) \leq 0,\ j = 1, \ldots, M \end{aligned} \tag{2.22}$$

Upon first glance, the formulation in Equation 2.22 looks far more difficult than the formulation in Equation 2.20. However, consider the monomial constraints under the log-log

[3]The monomial equality constraints are redundant and not necessary to fully define the problem, but are useful for clarity.

transformation[4]:

$$\log m(e^{\mathbf{y}}) = \log\left(ce^{a_1 y_1} e^{a_2 y_2} ... \; e^{a_n y_n}\right) \tag{2.23}$$

Using the algebraic properties of logarithms, this becomes:

$$\log m(e^{\mathbf{y}}) = \log c + \log e^{a_1 y_1} + \log e^{a_2 y_2} + ... + \log e^{a_n y_n} \tag{2.24}$$

$$\log m(e^{\mathbf{y}}) = \log c + a_1 y_1 + a_2 y_2 + ... + a_n y_n \tag{2.25}$$

Which is a *linear* constraint with respect to the transformed variables y_i. Similarly, consider a posynomial under transformation:

$$\log p(e^{\mathbf{y}}) = \log\left(\sum_{k=1}^{K} c_k \prod_{i=1}^{N} e^{a_{ik} y_i}\right) \tag{2.26}$$

Equation 2.26 is exactly a weighted Log-Sum-Exp function, which is known to be *convex* for non-negative values of c_i. Thus, it has been shown that the formulation in Equation 2.22 has a convex objective with a convex feasible set, making it a *convex optimization* problem.

As convex programs, GPs can be solved by a wide variety of algorithms, but most are currently solved using primal/dual methods [11]. Solvers such as MOSEK [29] and CVXOPT [30] are available and return reliable results under most circumstances.

2.3 Convexity and the KKT Conditions

Consider the case of unconstrained minimization of a scalar continuous objective function $f(\mathbf{x})$. From the principles of basic calculus, a point $\mathbf{x}^*$ is a *local minimum* of function $f(\mathbf{x})$ if both of the following conditions are true:

1. $\nabla f(\mathbf{x}) = 0$

[4]Note that this transformation is most commonly referred to as a log-log transformation because both the dependent and independent variables are transformed, but for conciseness, this is sometimes abbreviated to just a log transformation, logspace, and similar.

2. $\nabla^2 f(\mathbf{x}) > 0$

In other words, a candidate point is a local minimum of a scalar continuous objective function if the first derivative of the function, evaluated at the candidate point, is equal to zero, and if the second derivative of the function, evaluated at the candidate point, is greater than zero. The first condition is called a *necessary* condition, because though all local minima will satisfy this condition, local maxima satisfy this condition as well. The second condition is a *sufficient* condition because the condition is only true at local minima and not local maxima.

As was mentioned in the previous chapter, a constrained optimization problem:

$$\begin{aligned} &\text{minimize} && f(\mathbf{x}) \\ &\text{subject to} && \mathbf{g}_i(\mathbf{x}) \leq 0,\ i = 1, \ldots, N \\ &&& \mathbf{h}_j(\mathbf{x}) = 0,\ j = 1, \ldots, M \end{aligned} \tag{2.27}$$

can be transformed into an unconstrained minimization problem via the Lagrangian function:

$$\text{minimize} \quad \mathcal{L}(\mathbf{x}, \mathbf{u}, \mathbf{v}) = f(\mathbf{x}) + \sum_{i=1}^{N} u_i g_i(\mathbf{x}) + \sum_{j=1}^{M} v_j h_j(\mathbf{x}) \tag{2.28}$$

Applying the same necessary condition to the Lagrangian function yields:

$$\frac{\partial \mathcal{L}(\mathbf{x}, \mathbf{u}, \mathbf{v})}{\partial \mathbf{x}} = \nabla f(\mathbf{x}) + \sum_{i=1}^{N} u_i \nabla g_i(\mathbf{x}) + \sum_{j=1}^{M} v_j \nabla h_j(\mathbf{x}) = 0 \tag{2.29}$$

Which is indeed one of the necessary conditions for constrained optimization.

Two other necessary conditions exist, asserted here without proof or explanation (see [1, 31, 32]). The resulting set, combined with the original constraints for feasibility, are known as

the Karush-Kuhn-Tucker, or KKT Conditions:

$$\begin{aligned} \mathbf{g}_i(\mathbf{x}) &\leq 0,\ i = 1, \ldots, N \\ \mathbf{h}_j(\mathbf{x}) &= 0,\ j = 1, \ldots, M \\ \nabla f(\mathbf{x}) + \sum_{i=1}^{N} u_i \nabla g_i(\mathbf{x}) + \sum_{j=1}^{M} v_j \nabla h_j(\mathbf{x}) &= 0 \\ v_i &\geq 0 \\ v_i g_i &= 0 \end{aligned} \tag{2.30}$$

The last condition $v_i g_i = 0$ ensures that slack inequality constraints do not worsen the value of the Lagrangian, and the remaining condition $v_i \geq 0$ addresses the inherent asymmetry in the inequalities. Though a second order sufficient condition does exist for constrained optimization, computing $\nabla^2 \mathcal{L}(\mathbf{x}, \mathbf{u}, \mathbf{v})$ is generally a significant computational expense, and is therefore not used in practice.

At first glance, the KKT conditions seem to provide a set of simultaneous equations that can be used to solve directly for the optimum solution. And indeed, this is sometimes true. Consider:

$$\begin{aligned} \text{minimize} \quad & x + y \\ \text{subject to} \quad & x^2 + y^2 \leq 1 \\ & y = -2x \end{aligned} \tag{2.31}$$

the Lagrangian is:

$$\mathcal{L}(x, y, u, v) = x + y + u(x^2 + y^2 - 1) + v(y + 2x) \tag{2.32}$$

Which leads to KKT conditions:

$$\begin{aligned} &1 + 2ux + 2v = 0 \\ &1 + 2uy + v = 0 \\ &y + 2x = 0 \\ &u(x^2 + y^2 - 1) = 0 \quad u \geq 0 \\ &x^2 + y^2 \leq 1 \\ &y = -2x \end{aligned} \tag{2.33}$$

The complimentary slackness constraint $u(x^2 + y^2 - 1) = 0$ implies two cases, first either u must be zero, in which case the constraint is not equal to zero and is therefore inactive, or the quantity (x^2+y^2-1) is zero, implying that the constraint is active. Taking $u = 0$ results in the first two constraints being:

$$\begin{aligned} 1 + 2v = 0 \\ 1 + v = 0 \end{aligned} \tag{2.34}$$

which cannot be simultaneously satisfied, thus the constraint must be active. So the two equations:

$$\begin{aligned} y + 2x = 0 \\ x^2 + y^2 - 1 = 0 \end{aligned} \tag{2.35}$$

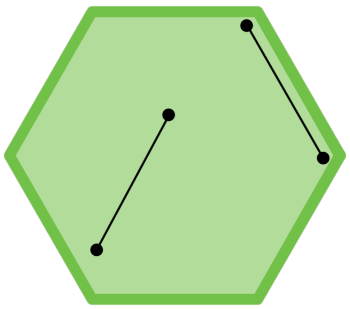
can be solved to obtain $x = \sqrt{1/5}$ and $y = -2\sqrt{1/5}$, which is the optimal solution!

However, cases where analytical solutions are rare, and no general method exists for solving the KKT conditions. And even if there was, recall the KKT conditions are only *necessary* conditions: just because the KKT conditions can be solved does not mean that the solution is the global minimum, or even a minimum at all.

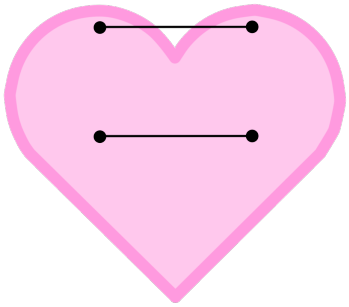
But there is one exception. If a problem is *convex*, then the KKT conditions are *sufficient* for proving global optimality [31, 32]. Furthermore, it is valid to conceptualize convex optimization problems as having direct solutions of the KKT conditions [31]. It is therefore generally

valid to assume that convex optimization problems can be solved reliably for roughly the expense of a matrix solve, though this statement should be read with heavy caution depending on the application and the specific question being asked. Regardless, it is clear that convexity is a critical element in efficient optimization.

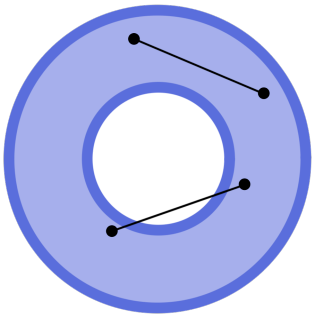
So what is convexity? Convexity is a geometric property of sets and functions. A convex set is one where any two points in the set can be connected by a line, and the line is also entirely contained in the set (Figure 2-1).



(a) A regular hexagon is convex



(b) A heart is not convex



(c) A donut is not convex

Figure 2-1: Examples of convex and non-convex sets

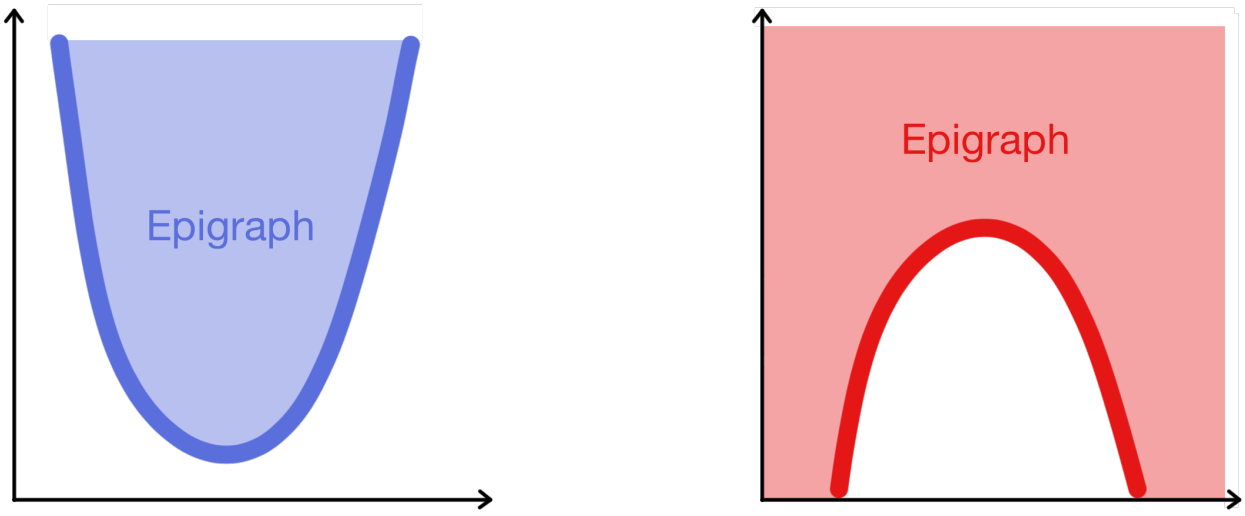
Expressed mathematically, the linear combination of any two points must also be in the set:

$$\begin{aligned} &\lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2 \in \mathcal{S} \\ &\forall \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{S} \\ &\forall \lambda \in [0,1] \end{aligned} \tag{2.36}$$

A convex function is one where the *epigraph* is a convex set. The epigraph of a function is defined as the set of points lower bounded by that function:

$$\mathcal{E} = \{(\mathbf{x}, y) \in X : y \geq f(\mathbf{x})\} \tag{2.37}$$

This is much more simple to visualize graphically:



(a) Epigraph of a convex function

(b) Epigraph of a concave function

Figure 2-2: The epigraph is the set that resides above a function

Thus, a convex function can be defined as one which satisfies Jensen's Inequality:

$$\lambda f(\mathbf{x}_1) + (1-\lambda) f(\mathbf{x}_2) \geq f(\lambda \mathbf{x}_1) + (1-\lambda)\mathbf{x}_2) \tag{2.38}$$

which states that line segment connecting any two points on a function (the linear combination of these two points) must be entirely contained in the epigraph of the function (Figure 2-3).

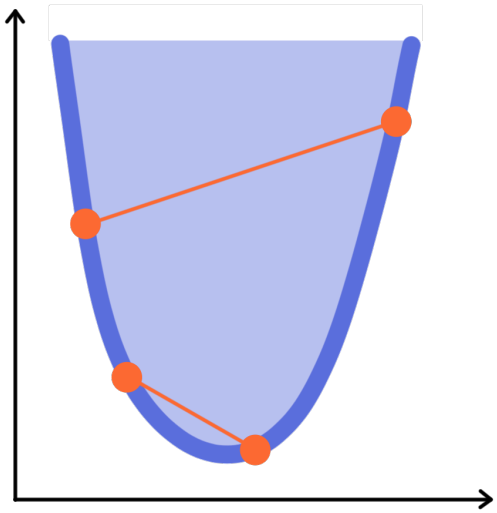


Figure 2-3: Jensen's inequality states that a convex function must have an epigraph that is a convex set. Line segments connecting points on the function must be contained in this epigraph.

A *convex optimization problem*, or convex program, is defined as a minimization problem where the objective function is convex, and constraints form a convex set. In practice, this means that inequality constraints must be convex, and equality constraints must be linear [31]. Many types of convex programs exist, including Linear Programs (LP), Quadratic Programs (QP), Geometric Programs (GP) upon transformation, Second Order Cone Programs (SOCP), and Semi-Definite Programs (SDP) [31, 32]. For the purposes of this work, it is assumed that convex optimization problems can be reliably solved for trivial expense, made possible by solvers like MOSEK [29] and CVXOPT [30].

In Section 2.1, the design optimization problem used inequality relationships to bound analysis models that are traditionally thought of as strict equalities. These inequality constraints also appear frequently in the GP literature. Though this use of inequalities may be somewhat non-intuitive, the definition of a convex optimization problem now gives the most compelling reason to use inequality relationships. A convex function like the one in Figure 2-3 only creates a convex set when an inequality is used in the constraint construction. If an equality constraint were to be implemented instead, then the points in the epigraph of the function would not be included in the feasible set, and the resulting set formed only by the points

along the equality constraint itself would not be a convex set. Since a convex optimization problem must be bounded by a convex set (and not by convex functions), the use of an equality operator on a constraint defined by a convex function would not result in a convex optimization problem. Thus, to obtain the benefits of convex optimization, certain analysis models must be bounded as inequalities to ensure the model retains its exploitable convex structure.

2.4 Limitations of Geometric Programming

The mathematical form of geometric programming imposes a number of unfortunate limitations on the the types of problems that can be considered. First, the log transformations limit all design variables to be strictly positive in the untransformed space. For engineering design problems, this ends up being a workable limitation, as quantities like mass, weight, length, and time cannot be negative. No aircraft has ever been constructed with a negative wingspan or zero mass. In cases where values are always negative, the absolute value of the variable can be optimized. And in cases where the variable is expected to change sign, simple additive shifts can be implemented, though it is worth noting that the shape of the log function is substantially different near $x = 1$ than as x approaches infinity, and so large additive shifts may substantially alter the underlying shape of the design space.

Second, the log transformations also enforce that all constraint values must be strictly greater than zero. As formulated, this limitation tends not to be particularly troublesome, especially given the methods for constructing standard form that come from Signomial Programming (Section 2.5). But it is a failure mode that sometimes shows up in practice.

Third, all posynomials must be upper bounded to create a convex problem. This limitation relates to the nature of the convex epigraph. As was seen in Equation 2.25, monomials become linear upon log transformation. Conceptually, a linear or affine function can have any relational operator and remain a convex set. But posynomials are convex functions under transformation (Equation 2.26) and therefore only the epigraph is a convex set. Thus,

only upper bounded posynomials meet the requirements for convexity.

This limitation on posynomial relationships does prove challenging to accommodate. For example, if a wing is discretized in to individual lift bins, as might be the case with lifting line theory, then the sum of these lift bins must be greater than the weight of the aircraft. But this relationship is a lower bounded posynomial, and therefore is not GP compatible. Clever tricks can sometimes overcome these challenges [33], but often the fundamental mathematical structure is not compatible with this limitations on polynomials.

Finally, the implicit limitation of GP is that all objectives and constraints must be written in GP compatible form, that is a posynomial objective and upper bounded posynomial constraints. Though the other limitations can all be handled to some degree, this one often proves insurmountable for two reasons. First, even simple functions like $\sin(x)$, $\cos(x)$, e^x, and ironically $\log(x)$ are fundamentally incompatible with this form. Sometimes, the Taylor Series expansions of these functions can be implemented as posynomial constraints, but Taylor Series expansions of $\sin(x)$ and $\cos(x)$ become non-posynomial with the second term. Second, the entire approach to using GP for design is built on the foundation of a SAND architecture for design optimization, and the notion that high fidelity black box analysis models can be written in GP compatible form is likely to be impossible.

2.5 Signomial Programming

Many of the limitations of Geometric Programming can be overcome by Signomial Programming [34, 11], which is a natural extension to the mathematical GP form. The first use of Signomial Programs (SPs) for aircraft design was work by Kirshen [35], which extended the work by Hoburg [10] to signomial programming. Work from York [36] later completed much of the work initiated by Kirschen [35]. From this core, a number of branching publications have emerged showing the versatility and effectiveness of the GP and SP formulations in solving aircraft design optimization problems [33, 37, 38, 39, 40, 41, 42, 43].

Signomial Programs allow for a significant increase in modeling flexibility when compared to GP, but this comes at a high cost: the SP formulation is no longer convex. Thus, convergence is not guaranteed, the final solution can only be assumed to be a local optimum, and the convergence rate can be problem dependent. In essence most of the benefits of using GP are eliminated, but abundance of successful implementations in the literature suggests that the consequences of this loss may be minimal.

Mathematical Formulation

The signomial programming formulation utilizes one additional function type called a *signomial*, which is defined as the difference between two posynomials $p(\mathbf{x})$ and $n(\mathbf{x})$:

$$s(\mathbf{x}) = p(\mathbf{x}) - n(\mathbf{x}) = \sum_{k=1}^{K} c_k \prod_{i=1}^{N} x_i^{a_{ik}} - \sum_{p=1}^{P} d_p \prod_{i=1}^{N} x_i^{g_{ik}} \tag{2.39}$$

The posynomial $n(\mathbf{x})$ is referred to as a 'neginomial' in order to emphasize the distinction between the terms with positive coefficients and the terms with negative coefficients. The neginomial is best thought of as the concave component of the signomial.

Note that the signomial is an extremely flexible function class. Notably, Taylor Series approximations are signomials, and therefore almost any continuous and differentiable function can be well approximated by a signomial function with a sufficient number of terms.

Given the definition of a signomial, it is now possible to construct the standard form for a Signomial Program [35]:

$$\begin{aligned} \text{minimize} \quad & \frac{p_0(\mathbf{x})}{n_0(\mathbf{x})} \\ \text{subject to} \quad & s_i(\mathbf{x}) = 0, \ i = 1, \ldots, N \\ & s_j(\mathbf{x}) \leq 0, \ j = 1, \ldots, M \end{aligned} \tag{2.40}$$

However the following form is often superior [35]:

$$\begin{aligned} \text{minimize} \quad & \frac{p_0(\mathbf{x})}{n_0(\mathbf{x})} \\ \text{subject to} \quad & \frac{p_i(\mathbf{x})}{n_i(\mathbf{x})} = 1,\ i = 1, \ldots, N \\ & \frac{p_j(\mathbf{x})}{n_j(\mathbf{x})} \leq 1,\ j = 1, \ldots, M \end{aligned} \tag{2.41}$$

The neginomial $n(\mathbf{x})$ is added to both sides, and then used as a divisor to construct an expression either equal to or constrained by a value of one.

Solution Method

Signomial Programs are solved using the Difference of Convex Algorithm (DCA), sometimes called the Convex-Concave Procedure. In DCA, the representation from Equation 2.41 is used to express the problem. A candidate solution $\mathbf{x}^*$ is selected and then the neginomials are then approximated as monomials about this candidate solution [11].

$$\bar{n}(\mathbf{x})|_{\mathbf{x}^*} \approx n(\mathbf{x}^*) \prod_{n=1}^{N} \left(\frac{x_n}{x_n^*} \right)^{a_n} \tag{2.42}$$

$$a_n = \frac{x_n^*}{n(\mathbf{x}^*)} \frac{\partial n}{\partial x_n} \tag{2.43}$$

The result is a Geometric Program that can now be solved using standard interior point primal/dual methods for a new $\mathbf{x}^*$. Successive iterations are taken until a relative convergence criteria $|p_0(\mathbf{x}_{k+1}) - p_0(\mathbf{x}_k)| < \varepsilon_{rel}$ is reached.

While this method works in practice for the inequality constraints, work by Opgenrood et. al. [44] suggests an improved method for handling the equality constraints is to split each equality constraint into two inequalities, where both the posynomial and neginomial elements are replaced with their monomial approximation.

In practice, the DCA struggles due to a tendency to generate infeasible GPs during iteration,

a common problem for iterative algorithms [1]. The Penalty Convex-Concave Procedure (PCCP) constructs a relaxed problem in order to correct this issue [45]. In PCCP, signomial constraints and non GP-compatible posynomial constraints are taken into a separate subset of the constraints. The equality constraints in this subset are split into two inequality constraints, and slack variables[5] $\mathbf{s}$ are introduced. Thus, the signomial program:

$$\begin{aligned} \text{minimize} \quad & \frac{p_0(\mathbf{x})}{n_0(\mathbf{x})} \\ \text{subject to} \quad & \frac{p_i(\mathbf{x})}{n_i(\mathbf{x})} = 1,\ i = 1, \ldots, N \\ & \frac{p_j(\mathbf{x})}{n_j(\mathbf{x})} \leq 1,\ j = 1, \ldots, M \\ & p_k(\mathbf{x}) \leq 1,\ k = 1, \ldots, K \end{aligned} \tag{2.44}$$

becomes:

$$\begin{aligned} \text{minimize} \quad & \frac{p_0(\mathbf{x})}{n_0(\mathbf{x})} + \sum_{i=1}^{2N} s_i^2 + \sum_{j=1}^{M} s_j^2 \\ \text{subject to} \quad & p_i(\mathbf{x}) \leq s_i n_i(\mathbf{x}),\ i = 1, \ldots, N \\ & s_i p_i(\mathbf{x}) \geq n_i(\mathbf{x}),\ i = N+1, \ldots, 2N \\ & p_j(\mathbf{x}) \leq s_j n_j(\mathbf{x}),\ j = 1, \ldots, M \\ & p_k(\mathbf{x}) \leq 1,\ k = 1, \ldots, K \end{aligned} \tag{2.45}$$

The problem is now solved with the DCA algorithm, just as any other signomial program. The optimal solution is achieved when all slack variables $\mathbf{s} = 1$, thus ensuring all of the constraints are being enforced as desired. Though the PCCP method is more robust, it typically requires noticeably longer to solve than the unmodified DCA.

When compared to other iterative algorithms, the SP termination criteria $|p_0(\mathbf{x}_{k+1}) - p_0(\mathbf{x}_k)| < \varepsilon_{rel}$ is not terribly rigorous. As a result, it is difficult to compare the results of the algorithm to other benchmarks, or even to say for sure that an optimal solution has been reached. But

[5] not to be confused with the notation for signomials

in practice, this is a widely used convergence metric.

2.6 Data Fitting

Work in geometric programming has identified one final trick to overcome the inherent limitations of the mathematical form. Although the detailed mathematics of high fidelity analysis typically are not GP compatible, the relationship between inputs and outputs often is. For example, the airfoil analysis tool XFOIL [46] is not at all GP compatible, but drag polars in the form $C_{D_p} = f(C_L, Re, \tau)$ often are. Hoburg [47] proposed a method where a GP compatible surrogate model is fit to representative data from higher fidelity black box analysis tools like XFOIL. The actual underlying function does not matter in this approach, only the function $C_{D_p} = f(C_L, Re, \tau)$. As a result, XFOIL data, CFD data, and wind tunnel data, all of which model a very different set of physics, are treated as equal in this approach. In addition to the GP compatible data fitting methods, a generalization for use in signomial programming was also published by Karcher [48], and is discussed in Appendix A.

But data fitting suffers from many of the same issues present in traditional surrogate modeling techniques, notably challenges in obtaining the data set used to fit the function. It is rarely clear how large of a sample interval to consider for each input variable, and how many discretizations to take within each interval. The result is a level of uncertainty in the surrogate that is difficult to quantify and that varies significantly depending on the decisions that were made in selecting the data set. And even if a sufficient sample set is selected, evaluating all of these points in the high fidelity analysis model may be prohibitively expensive.

It is worth noting that large body of literature exists surrounding methods of surrogate modeling, which today includes the use of machine learning. These topics in a general sense are well beyond scope here, but the critical concept is that data fitting provides a 'shortcut' around high fidelity physics, instead allowing core GP compatible relationships to be preserved. It also has the secondary benefit of being the only way that data from black box analysis functions can be included in the GP formulation, breaking out of one of the

critical issues of SAND architectures.

2.7 The Optimization First Approach vs Traditionally Defined SAND

Until geometric and signomial programming were introduced as methods for aircraft design, there really was no notion of what a practical and successful SAND approach to MDAO might look like. And despite its many flaws, the use of GP in aircraft design has been a resounding success, measured in the numerous publications, industry interest, and most strikingly in the successful flight of the Jungle Hawk Owl UAV [49], which was designed almost entirely using this modern SAND approach.

Though GP and SP are indeed SAND approaches to design, there is one key distinguishing element. A traditional interpretation of SAND makes no assumptions on the kind of optimization being used, only that the analysis models must be entirely implemented in the optimization formulation. In this way, SAND remains an analysis first approach to design since the analysis models remain the fixed element in the process. The approach of GP and SP is different in that analysis models are *required* to fit some mathematical form that complies with the structure of a particular method of numerical optimization, making this an *optimization first* approach to aircraft design.

To take on this optimization first approach does have significant advantages: the actual numerical optimization runs are almost assured to be rapid, the entire problem is visible to the user, and modifications to the design problem can be made with just a few modifications to either the objectives or constraints. But placing optimization first does require a significant adjustment in thinking on the part of the human design team. Analysis fidelity is no longer the driving force behind the mathematics of a given analysis model, instead compatibility with the chosen optimization method drives all of the choices made in developing mathematical models. Furthermore, the bulk of the design effort now shifts from wiring models

together and waiting for a long slow optimization phase to complete to an active model development phase where the optimization routine is fast, but often produces results that are nonsense.

Though both approaches have merit, the design team must consider the pros and cons of the two approaches. And as it stands, these two approaches are fundamentally incompatible, and so to the new optimization first approach carries significant risk, despite its potential upside.

Chapter 3

Extending the Optimization First Approach with Non-Linear Optimization

3.1 The Limitations of the GP Based Optimization First Approach

When Geometric Programming was being rolled out to industry partners, the prevailing attitude was that the geometric programming had the potential to be a powerful tool, but the exclusion of existing black box analysis tools was a non-starter. This sentiment is stated in a paper published by a group of designers from Aurora Flight Sciences [43], one of the first industry partners to really begin dedicating funds and technical staff to use the optimization first approach to design. It became clear that in order for the optimization first approach to be made practical, some effort at including black box analysis tools would have to be made. Methods of fitting black box data (see Section 2.6) might work for lower fidelity analysis models, but high fidelity tools simply could not generate sufficient data to fit a meaningful

surrogate model.

Fortunately, the field of MDAO provides the key to unlocking black box analysis through Sequential Quadratic Programming (SQP).

3.2 Sequential Quadratic Programming

3.2.1 Newton's Method

Section 2.3, stated that the KKT conditions are necessary for determining optimality in constrained, continuous optimization problems, and that these conditions were sufficient for optimality if and only if the optimization problem is convex. Furthermore, in the case of convex optimization problems, efficient methods exist for (in essence) directly solving the KKT conditions[1]. But what about when optimization problems are not convex?

This question dates back to Newton, who was the first to tackle the problem for unconstrained optimization. Recall the unconstrained optimization formulation:

$$\text{minimize} \quad f(\mathbf{x}) \tag{3.1}$$

which is known in the general case to be difficult to solve. Rather than attempting a direct solution, Newton expanded the function $f(\mathbf{x})$ as a Taylor Series about some point $\mathbf{x}_k$ (done here to second order):

$$f(\mathbf{x}) \approx f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T (\mathbf{x} - \mathbf{x}_k) + \frac{1}{2} (\mathbf{x} - \mathbf{x}_k)^T \nabla^2 f(\mathbf{x}_k) (\mathbf{x} - \mathbf{x}_k) + \text{H.O.T.} \tag{3.2}$$

and after ignoring the Higher Order Terms instead chose to solve the problem:

$$\text{minimize} \quad f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T (\mathbf{x} - \mathbf{x}_k) + \frac{1}{2} (\mathbf{x} - \mathbf{x}_k)^T \nabla^2 f(\mathbf{x}_k) (\mathbf{x} - \mathbf{x}_k) \tag{3.3}$$

[1] See [31] Section 5.5.3 for discussion of this statement

which is trivial. First take the derivative with respect to $\mathbf{x}$ and set it equal to zero:

$$\nabla f(\mathbf{x}_k) + \nabla^2 f(\mathbf{x}_k)\,(\mathbf{x} - \mathbf{x}_k) = 0 \tag{3.4}$$

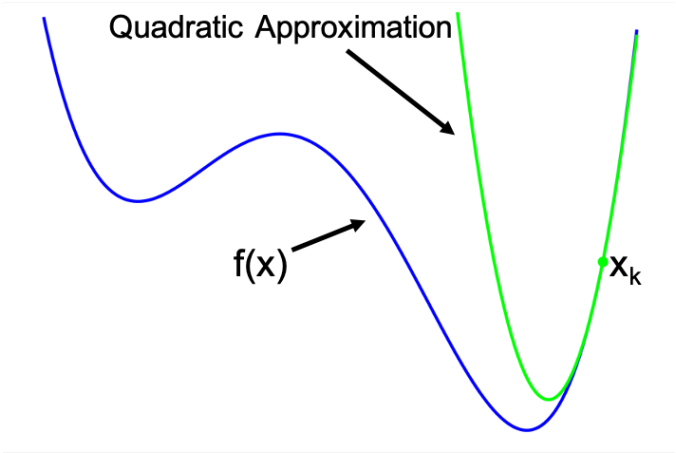
With some algebra this becomes:

$$\mathbf{x} - \mathbf{x}_k = -\left(\nabla^2 f(\mathbf{x}_k)\right)^{-1} \nabla f(\mathbf{x}_k) \tag{3.5}$$

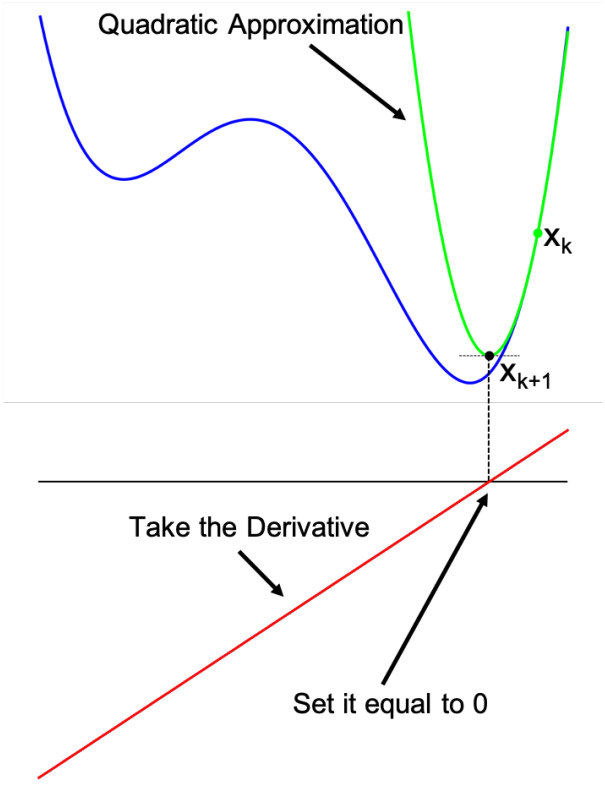
and finally using the notation for the Hessian matrix:

$$\mathbf{x} = \mathbf{x}_k - H(\mathbf{x}_k)^{-1} \nabla f(\mathbf{x}_k) \tag{3.6}$$

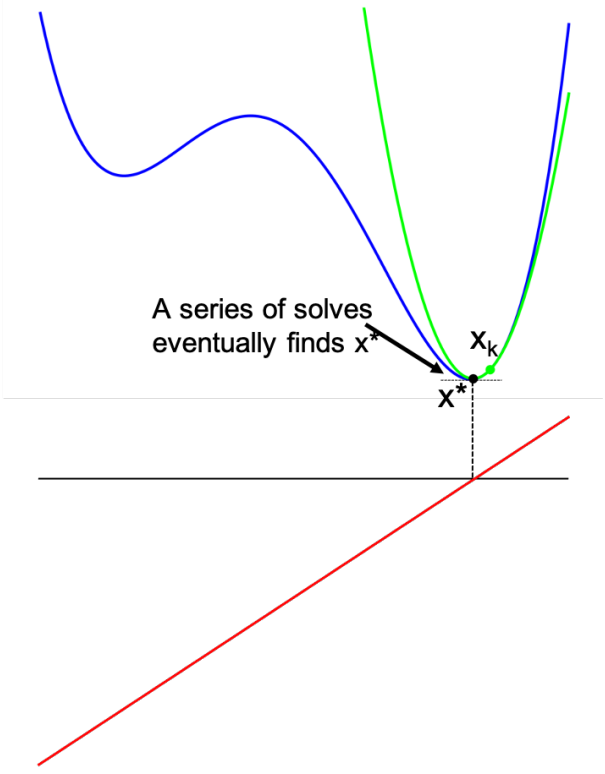
Equation 3.6 states that moving from $\mathbf{x}_k$ to this newly found point $\mathbf{x}_{k+1}$ will exactly minimize the quadratic approximation of $f(\mathbf{x})$ about the point $\mathbf{x}_k$.



(a) Quadratic approximation of $f(\mathbf{x})$



(b) Take the derivative, set it equal to zero



(c) Step to new $\mathbf{x_{k+1}} = \mathbf{x}_k - H(\mathbf{x}_k)^{-1}\nabla f(\mathbf{x}_k)$

Figure 3-1: Graphical representation of Newton's Method on a simple 2D case

If the function $f(\mathbf{x})$ is quadratic, then the approximation in Equation 3.2 is exact and a single application of Equation 3.6 will immediately find the optimal solution. When the approximation is not exact, the error between the approximation and the true function $f(\mathbf{x})$ will result in the next $\mathbf{x}_k$ not being the true optimum (Figure 3-1b). If Equation 3.6 is applied a second time, this error will decrease (Figure 3-1). Continued applications of Equation 3.6 have been shown to converge *quadratically* to the true optimum of the non-linear function $f(\mathbf{x})$ given a reasonable initial guess [32]. This iterative process has become known in modern terminology as Newton's Method, and nearly all of modern gradient based optimization algorithms emerge as extensions from this core idea. Newton's Method is summarized in the following algorithm:

Algorithm for Newton's Method	
Step 0	Given $\mathbf{x}_0$, set $k = 0$
Step 1	$\mathbf{d}_k = -H(\mathbf{x}_k)^{-1}\nabla f(\mathbf{x}_k)$
	If $\|\|\mathbf{d}_k\|\| \leq \epsilon$ then terminate
Step 2	$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{d}_k$
	$k = k + 1$
	Go to Step 1

Critical to the work here is that Newton's Method requires no knowledge of the actual function $f(\mathbf{x})$. Iterations in Newton's Method rely only on knowledge of the inverse of the Hessian matrix $H(\mathbf{x}_k)^{-1}$ and the gradient vector $\nabla f(\mathbf{x}_k)$ evaluated at each $\mathbf{x}_k$. Furthermore, the only assumptions made of the function $f(\mathbf{x})$ are that the function is continuous and twice differentiable. No assumptions have been made about the *convexity* of $f(\mathbf{x})$, nor the accuracy of the quadratic approximation.

3.2.2 Quadratic Programming

In the case of constrained optimization:

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & \mathbf{g}_i(\mathbf{x}) \leq 0,\ i = 1, \ldots, N \\ & \mathbf{h}_j(\mathbf{x}) = 0,\ j = 1, \ldots, M \end{aligned} \tag{3.7}$$

the problem once again has no general solution method. But armed with knowledge of Newton's method, one approach is to begin with Taylor Series expansions of the objective and constraints (again to second order):

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T (\mathbf{x} - \mathbf{x}_k) + \frac{1}{2} (\mathbf{x} - \mathbf{x}_k)^T \nabla^2 f(\mathbf{x}_k) (\mathbf{x} - \mathbf{x}_k) \\ \text{subject to} \quad & g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k)^T (\mathbf{x} - \mathbf{x}_k) + \frac{1}{2} (\mathbf{x} - \mathbf{x}_k)^T \nabla^2 g_i(\mathbf{x}_k) (\mathbf{x} - \mathbf{x}_k) \leq 0,\ i = 1, \ldots, N \\ & h_j(\mathbf{x}_k) + \nabla h_j(\mathbf{x}_k)^T (\mathbf{x} - \mathbf{x}_k) + \frac{1}{2} (\mathbf{x} - \mathbf{x}_k)^T \nabla^2 h_j(\mathbf{x}_k) (\mathbf{x} - \mathbf{x}_k) = 0,\ j = 1, \ldots, M \end{aligned} \tag{3.8}$$

In the case of Newton's Method, taking these Taylor series expansions resulted in a problem that was easy to solve. However, this time no clear analytical presents itself. As was stated in Section 2.3, the key to analytic solutions in constrained optimization is *convexity*. Equation 3.8 is a Quadratically Constrained Quadratic Program (QCQP) which are generally non-linear, non-convex, and do not have an exploitable structure. Thus, there is no established method for solving these problems, though this has recently become an active area of research [50]. But now consider expanding the constraint functions to only first order:

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T (\mathbf{x} - \mathbf{x}_k) + \frac{1}{2} (\mathbf{x} - \mathbf{x}_k)^T \nabla^2 f(\mathbf{x}_k) (\mathbf{x} - \mathbf{x}_k) \\ \text{subject to} \quad & g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k)^T (\mathbf{x} - \mathbf{x}_k) \leq 0,\ i = 1, \ldots, N \\ & h_j(\mathbf{x}_k) + \nabla h_j(\mathbf{x}_k)^T (\mathbf{x} - \mathbf{x}_k) = 0,\ j = 1, \ldots, M \end{aligned} \tag{3.9}$$

This type of problem with quadratic objective and linear constraints is a Quadratic Program

(QP), a form which is well known to be convex[2] [31]. In standard notation, the QP form is:

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}\mathbf{x}^T\mathbf{P}\mathbf{x} + \mathbf{q}^T\mathbf{x} + r \\ \text{subject to} \quad & \mathbf{a}_i^T\mathbf{x} = b_i,\ i = 1, \ldots, N \\ & \mathbf{g}_j^T\mathbf{x} \leq h_j,\ j = 1, \ldots, M \end{aligned} \tag{3.10}$$

As convex programs, QPs can be solved quickly and efficiently, with guaranteed convergence to a global optimum. In this way, they are the intellectual sibling of the quadratic optimization problem presented in Equation 3.3.

3.2.3 The Mathematical Definition of Sequential Quadratic Programming

Sequential Quadratic Programming solves the the general non-linear program, restated here:

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & \mathbf{g}_i(\mathbf{x}) \leq 0,\ i = 1, \ldots, N \\ & \mathbf{h}_j(\mathbf{x}) = 0,\ j = 1, \ldots, M \end{aligned} \tag{3.11}$$

using a similar approach to that of Newton's Method for unconstrained optimization. A simpler quadratic programming optimization problem is constructed, often referred to as the Quadratic Programming Sub-Problem[3] [1]:

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T\mathbf{d} + \frac{1}{2}\mathbf{d}^T\nabla^2\mathcal{L}(\mathbf{x}_k)\mathbf{d} \\ \text{subject to} \quad & g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k)^T\mathbf{d} \leq 0,\ i = 1, \ldots, N \\ & h_j(\mathbf{x}_k) + \nabla h_j(\mathbf{x}_k)^T\mathbf{d} = 0,\ j = 1, \ldots, M \\ & \mathbf{d} = \mathbf{x} - \mathbf{x}_k \end{aligned} \tag{3.12}$$

[2]Subject to a positive semi-definite Hessian matrix, which will be discussed later

[3]Note that the Hessian of the Lagrangian appears in this sub-problem, but will be discussed in a later section

and solved for variable $\mathbf{d}$. This new variable $\mathbf{d}$ is the suggested vector that should be added to the current point $\mathbf{x}_k$ to yield the next $\mathbf{x}_{k+1}$. Once this vector $\mathbf{d}$ has been obtained, some variation of a line search is used to find the best step size to move to point $\mathbf{x}_{k+1}$[4].

Sequential Quadratic Programming (SQP) has become one of the most successful optimization algorithms used in aircraft design [6, 5], and along with interior point methods represents the state of the art algorithm for solving aircraft design optimization problems [2]. Beyond the realms of aerospace, the SQP algorithm has become a staple algorithm in most modern languages used to solve optimization problems including MATLAB, Python, R, and Julia.

A non-linear program which is exactly a QP will reach the optimal solution in exactly one iteration of Equation 3.12 (assuming an exact representation of the Hessian). And when a non-linear program is not a QP, the error between the original non-linear program and the QP approximation will cause some error between the new vector $\mathbf{x}$ and the true solution to the non-linear program. Sequential iterations of Equation 3.12 drive this error to an acceptably small tolerance in a reasonable number of iterations.

As with Newton's Method, the actual mathematical form of the functions $f(\mathbf{x})$, $g_i(\mathbf{x})$, and $h_j(\mathbf{x})$ do not matter to the SQP algorithm, only the function evaluations and gradients at each $\mathbf{x}_k$. Therefore, if an existing black box analysis model returns function evaluations and gradients, then it can be naturally integrated into the SQP algorithm as if it were any other function. Even if gradients are not available, methods like finite difference provide a means to incorporate any existing black box analysis tool.

[4]Other variations of SQP exist, but this is the easiest to understand [1, 2]

3.3 Exploiting Log-Log Convexity When Black Box Analysis Tools Are Present

The optimization first approach demonstrated a practical method of framing aircraft design as a numerical optimization problem, and Sequential Quadratic Programming provides a method of solving numerical optimization problems when black box analysis models are present. It should then follow that to enable the use black box analyses in the optimization first design approach, one need only swap from solving the convex GP to solving the non-linear program with the SQP algorithm. Unfortunately, the SQP algorithm struggles to solve even simple geometric programs for aircraft design [51, 52]. The reasons for this are somewhat unclear, but the struggles of SQP in solving GPs are not entirely unsurprising. Geometric programs are convex in log-log space, but the core Taylor Series approximations used in SQP assume a quadratic objective and linear constraints, and so the fundamental assumptions of underlying mathematical structure are not in agreement.

So to fully utilize black box analysis tools in the optimization first approach to aircraft design, the best elements of both the GP approach and the SQP algorithm must be combined. Geometric Programming works well for aircraft design because the aircraft design space has a large degree of underlying log-log convexity, and SQP works well for traditional MDAO due to its natural integration with the black box analysis tools. Is it possible to combine these two elements in a unified numerical optimization approach?

The remainder of this dissertation will answer this question. First, new algorithms will be developed that combine the log-log transformation of GP with the solution method used in SQP. Next, these algorithms will be benchmarked on a number of test problems and compared to the original SQP algorithm. Then, the new algorithms will be placed in a design context before finally being used in two representative design examples.

Chapter 4

Optimization Algorithms for the Extended Optimization First Approach

4.1 Logspace Sequential Quadratic Programming

4.1.1 Motivation for the LSQP Algorithm

Newton's Method and SQP are related by the common assumption of underlying structure that general functions $f(\mathbf{x})$ can be accurately represented by a second order (quadratic) Taylor Series approximation. But the success of Geometric Programming in the aircraft design community [10, 33, 35, 37, 36, 38, 39, 40, 41] reveals a different kind of structure: log-log convexity.

So, rather than starting with a quadratic Taylor Series structure, what happens if an iterative algorithm is constructed assuming log-log convexity instead?

4.1.2 The Transformed Problems of LSQP

Consider the log transformation utilized to solve geometric programs [11]. The design variables $\mathbf{x}_i$ are transformed according to:

$$x_i = e^{y_i} \tag{4.1}$$

or alternatively:

$$y_i = \log x_i \tag{4.2}$$

Due to this transformation, the general non-linear program must be slightly modified:

$$\begin{aligned} &\text{minimize} && f(\mathbf{x}) \\ &\text{subject to} && \mathbf{g}_i(\mathbf{x}) \leq 1,\ i = 1, \ldots, N \\ &&& \mathbf{h}_j(\mathbf{x}) = 1,\ j = 1, \ldots, M \end{aligned} \tag{4.3}$$

putting a one on the right hand side of the constraints. Equation 4.3 will be referred to as standard form for the purposes of this discussion, as opposed to the more standard definition described by Equation 3.7. Constructing either standard form has difficulties, but in general the same rules should be followed as those used for constructing the standard form for signomial programs (see Section 2.5). In particular, division and multiplication should be used to construct the constraints as opposed to addition and subtraction. For example, the constraint $y \geq x$ should become $x/y \leq 1$ and not $x - y + 1 \leq 1$.

Equation 4.3, can then be transformed in a similar fashion to that of a geometric program [11]:

$$\begin{aligned} &\text{minimize} && \log f(e^{\mathbf{y}}) \\ &\text{subject to} && \log \mathbf{g}_i(e^{\mathbf{y}}) \leq 0,\ i = 1, \ldots, N \\ &&& \log \mathbf{h}_j(e^{\mathbf{y}}) = 0,\ j = 1, \ldots, M \end{aligned} \tag{4.4}$$

Rather than implementing the SQP algorithm on the original problem (Equation 3.7), instead

consider implementing the traditional SQP algorithm on this transformed problem. This 'new' algorithm (which consists merely of a well known transformation and a well established algorithm) is Logspace Sequential Quadratic Programming (LSQP).

This rooting in existing methods is actually one advantage of LSQP over the second algorithm that will be developed later in this chapter, since LSQP can take advantage of existing SQP solvers, with the log-log transformation being handled as pre-processing and post-processing step. However, the case will be made in the next chapter that a dedicated LSQP algorithm is of value for practical reasons.

4.1.3 Expressions for the Gradients and Hessian of the LSQP Functions

Deriving the QP sub-problem for the LSQP algorithm requires deriving the expressions for the gradients of the objective and constraint functions. Consider first the individual elements of the gradient of the functions [11] where F is a general placeholder for $F = \log f(e^{\mathbf{y}})$:

$$\frac{\partial F}{\partial y_i} = \frac{\partial F}{\partial f}\frac{\partial f}{\partial x_i}\frac{\partial x_i}{\partial y_i} \tag{4.5}$$

which becomes:

$$\frac{\partial F}{\partial y_i} = \frac{1}{f(e^{\mathbf{y}})}\frac{\partial f}{\partial x_i}e^{y_i} \tag{4.6}$$

or:

$$\frac{\partial F}{\partial y_i} = \frac{x_i}{f(\mathbf{x})}\frac{\partial f}{\partial x_i} \tag{4.7}$$

According to the above equation, the gradients of the original functions can be used *as is* after multiplying by the value of $\frac{x_i}{f(\mathbf{x})}$ (or similar for $\mathbf{g}_i(\mathbf{x})$ and $\mathbf{h}_j(\mathbf{x})$). If these functions represent black box analysis tools, then the tool can now be integrated into the LSQP algorithm with minimal modification.

The Hessian of the transformed functions LSQP functions is not necessary for the algorithm,

but is an interesting exercise that gives insight into the log-convex structure. Begin with the chain rule:

$$\begin{aligned}\frac{\partial F}{\partial y_i \partial y_j} &= \frac{\partial}{\partial y_j}\left(\frac{1}{f(e^{\mathbf{y}})}\frac{\partial f}{\partial x_i}e^{y_i}\right) \\ \frac{\partial F}{\partial y_i \partial y_j} &= \frac{\frac{\partial}{\partial y_j}\left(\frac{\partial f}{\partial x_i}e^{y_i}\right) f(e^{\mathbf{y}}) - \left(\frac{\partial f}{\partial x_i}e^{y_i}\right)\left(\frac{\partial f(e^{\mathbf{y}})}{\partial y_j}\right)}{\left(f(e^{\mathbf{y}})\right)^2}\end{aligned} \tag{4.8}$$

Now compute the following substitutions:

$$\begin{aligned}\frac{\partial}{\partial y_j}\left(\frac{\partial f}{\partial x_i}e^{y_i}\right) &= \frac{\partial^2 f}{\partial x_i \partial y_j}e^{y_i} + \frac{\partial f}{\partial x_i}\frac{\partial e^{y_i}}{\partial y_j} \\ \frac{\partial}{\partial y_j}\left(\frac{\partial f}{\partial x_i}e^{y_i}\right) &= \frac{\partial^2 f}{\partial x_i \partial x_j}e^{y_i}\frac{\partial x_j}{\partial y_j} + \frac{\partial f}{\partial x_i}\frac{\partial e^{y_i}}{\partial y_j} \\ \frac{\partial}{\partial y_j}\left(\frac{\partial f}{\partial x_i}e^{y_i}\right) &= \frac{\partial^2 f}{\partial x_i \partial x_j}e^{y_i}e^{y_j} + \frac{\partial f}{\partial x_i}\frac{\partial e^{y_i}}{\partial y_j}\end{aligned} \tag{4.9}$$

$$\frac{\partial f(e^{\mathbf{y}})}{\partial y_j} = \frac{\partial f}{\partial x_j}e^{y_j} \tag{4.10}$$

Note in the above equations that $\frac{\partial e^{y_i}}{\partial y_j} = e^{y_j} = x_j$ when $i = j$ and $\frac{\partial e^{y_i}}{\partial y_j} = 0$ otherwise. Thus, the second derivative expression becomes:

$$\begin{aligned}\frac{\partial F}{\partial y_i \partial y_j} &= \frac{\left(\frac{\partial^2 f}{\partial x_i \partial x_j}e^{y_i}e^{y_j} + \frac{\partial f}{\partial x_i}\frac{\partial e^{y_i}}{\partial y_j}\right) f(e^{\mathbf{y}}) - \left(\frac{\partial f}{\partial x_i}e^{y_i}\right)\left(\frac{\partial f}{\partial x_j}e^{y_j}\right)}{\left(f(e^{\mathbf{y}})\right)^2} \\ \frac{\partial F}{\partial y_i \partial y_j} &= \frac{\left(\frac{\partial^2 f}{\partial x_i \partial x_j}x_i x_j + \frac{\partial f}{\partial x_i}\frac{\partial x_i}{\partial y_j}\right) f(\mathbf{x}) - \frac{\partial f}{\partial x_i}\frac{\partial f}{\partial x_j}x_i x_j}{\left(f(\mathbf{x})\right)^2}\end{aligned} \tag{4.11}$$

which once again enables the evaluation of these derivatives without taking derivatives of the log-transformed functions. It should be noted that this exercise in second derivatives is largely academic to begin with, since no reasonable SQP algorithm actually utilizes the exact computation of the Hessian matrices due to the abundance of approximate methods like BFGS, which are widely deemed superior.

4.1.4 Formal Definition of LSQP

Armed with derivatives, the LSQP algorithm can be defined as solving the general non-linear constrained optimization problem:

$$\begin{aligned} &\text{minimize} && f(\mathbf{x}) \\ &\text{subject to} && \mathbf{g}_i(\mathbf{x}) \leq 1,\ i = 1, \ldots, N \\ & && \mathbf{h}_j(\mathbf{x}) = 1,\ j = 1, \ldots, M \end{aligned} \tag{4.12}$$

by solving a series of iterative sub-problems:

$$\begin{aligned} \underset{\mathbf{d}}{\text{minimize}} \quad & \log f(\mathbf{x}_k) + \frac{1}{f(\mathbf{x}_k)} \left(\mathbf{x}_k \odot \nabla f(\mathbf{x}_k)\right)^T \mathbf{d} + \frac{1}{2}\mathbf{d}^T \nabla^2 \mathcal{L}(\mathbf{y}_k)\mathbf{d} \\ \text{subject to} \quad & \log g_i(\mathbf{x}_k) + \frac{1}{g_i(\mathbf{x}_k)} \left(\mathbf{x}_k \odot \nabla g_i(\mathbf{x}_k)\right)^T \mathbf{d} \leq 0,\ i = 1, \ldots, N \\ & \log h_j(\mathbf{x}_k) + \frac{1}{h_j(\mathbf{x}_k)} \left(\mathbf{x}_k \odot \nabla h_j(\mathbf{x}_k)\right)^T \mathbf{d} = 0,\ j = 1, \ldots, M \\ & \mathbf{d} = \mathbf{y} - \log \mathbf{x}_k \\ & \mathbf{y} = \log \mathbf{x} \end{aligned} \tag{4.13}$$

that are quadratic programs under a log-log transformation from the original space. Note that the term $\frac{1}{2}\mathbf{d}^T \nabla^2 \mathcal{L}(\mathbf{y}_k)\mathbf{d}$ has been left as is, since this matrix is rarely ever explicitly computed. But there is sufficient information in the previous section if a future reader wishes to explicitly construct this term for some as of yet unknown reason.

If foreknowledge of log convexity is assumed, Equation 4.4 is expected to be well approximated as a log convex optimization problem based on the similar properties of geometric and signomial programs. For example, in the case where the original non-linear program (NLP) is a geometric program, monomial constraints will be exactly represented as lines in log-log space, while posynomials are represented by their linear approximations. Thus, the quadratic programming approximation after log-log transformation (Equation 4.13) should be a more accurate representation of the original NLP than the quadratic programming

approximation with no transformation (Equation 3.12).

Though an incremental step forward, the superior representation of underlying structure should allow LSQP to operate better on the GP-like problems common in the new approach to aircraft design and enable the desired integration of black box analysis tools. Furthermore, the exploitation of log-log convexity should yield computational savings in cases where SQP also converges.

4.2 Sequential Log Convex Programming

4.2.1 Sequential Convex Programming

To this point the assumption has been that sub-problems must be quadratic programs in order to be effectively solved, but as has been discussed, the true structure that matters is *convexity*. The foundational principle of Sequential Convex Programming is that rather than require sub-problems to be strictly quadratic programs, instead the sub-problems are required only to be convex. The goal is to create sub-problems that are more accurate, resulting in fewer iterations being necessary for convergence. Sequential Convex Programming has not received much attention in the literature, but is mentioned in a few course notes from Stephen Boyd and others [53, 54]. One specific method, Sequential Quadratically Constrained Quadratic Programming has received some additional attention in the literature [55, 56, 57, 58] but this algorithm remains impractical for widespread application.

4.2.2 Breaking Down Problem Structure

The SQP and LSQP algorithms both treat the constraints as a monolithic block, differentiating only between equality and inequality constraints. In reality, optimization problems

generally exhibit far more structure. Consider the following differentiation:

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimize}} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & \mathbf{p}(\mathbf{x}) \leq 1 \\ & \mathbf{m}(\mathbf{x}) = 1 \\ & \mathbf{s}_i(\mathbf{x}) \leq 0 \\ & \mathbf{s}_e(\mathbf{x}) = 0 \\ & \mathbf{g}(\mathbf{x}) \leq 1 \\ & \mathbf{h}(\mathbf{x}) = 1 \\ & \mathcal{B}_i(\mathbf{x}) \leq 1 \\ & \mathcal{B}_e(\mathbf{x}) = 1 \end{aligned} \tag{4.14}$$

Where functions $\mathbf{p}(\mathbf{x})$ are posynomial functions, functions $\mathbf{m}(\mathbf{x})$ are monomial functions, functions $\mathbf{s}(\mathbf{x})$ are signomials, functions $\mathbf{g}(\mathbf{x})$ and $\mathbf{h}(\mathbf{x})$ are general explicitly known functions with no specified structure (for example, sin, cot, or perhaps somewhat ironically the log function), and functions $\mathcal{B}(\mathbf{x})$ are black boxes not explicitly known. Recall that signomials can be represented by a fraction of convex posynomials $\mathbf{p}(\mathbf{x})$ and concave neginomials $\mathbf{n}(\mathbf{x})$, and thus the structure can be rewritten as:

$$
\begin{aligned}
\underset{\mathbf{x}}{\text{minimize}} \quad & f(\mathbf{x}) \\
\text{subject to} \quad & \mathbf{p}(\mathbf{x}) \leq 1 \\
& \mathbf{m}(\mathbf{x}) = 1 \\
& \frac{\mathbf{p}_i(\mathbf{x})}{\mathbf{n}_i(\mathbf{x})} \leq 1 \\
& \frac{\mathbf{p}_e(\mathbf{x})}{\mathbf{n}_e(\mathbf{x})} = 1 \\
& \mathbf{g}(\mathbf{x}) \leq 1 \\
& \mathbf{h}(\mathbf{x}) = 1 \\
& \mathcal{B}_i(\mathbf{x}) \leq 1 \\
& \mathcal{B}_e(\mathbf{x}) = 1
\end{aligned}
\tag{4.15}
$$

For the purposes of this discussion, the signomial functions will be treated the same as the generally explicit non-linear functions, so these will be collapsed together:

$$
\begin{aligned}
\underset{\mathbf{x}}{\text{minimize}} \quad & f(\mathbf{x}) \\
\text{subject to} \quad & \mathbf{p}(\mathbf{x}) \leq 1 \\
& \mathbf{m}(\mathbf{x}) = 1 \\
& \mathbf{g}(\mathbf{x}) \leq 1 \\
& \mathbf{h}(\mathbf{x}) = 1 \\
& \mathcal{B}_i(\mathbf{x}) \leq 1 \\
& \mathcal{B}_e(\mathbf{x}) = 1
\end{aligned}
\tag{4.16}
$$

Future work may examine exploiting signomials independently, possibly via the relaxation used in the DCA solution algorithm for signomial programs (Section 2.5), but efforts attempting such an implementation have been unsuccessful thus far, and so there is no value in giving them special treatment.

The key difference in Equation 4.16 is that the posynomial constraints $\mathbf{p}(\mathbf{x})$ have been separated from the other constraints. These posynomials are unique in that they are known *a priori* to be convex constraints in logspace, and thus can be represented exactly in a convex sub-problem that is not a quadratic program.

4.2.3 Formal Definition of SLCP

Representing posynomials exactly in the sub-problem yields the following:

$$
\begin{aligned}
\underset{\mathbf{d}}{\text{minimize}} \quad & \log f(\mathbf{x}_k) + \frac{1}{f(\mathbf{x}_k)} \left(\mathbf{x}_k \odot \nabla f(\mathbf{x}_k)\right)^T \mathbf{d} + \frac{1}{2}\mathbf{d}^T \nabla^2 \mathcal{L}_R(\mathbf{y}_k)\mathbf{d} \\
\text{subject to} \quad & \log\left(\sum_j \exp\left(P_j(\mathbf{d} + \log \mathbf{x}_k) + q_j\right)\right) \leq 0 \\
& A_m(\mathbf{d} + \log \mathbf{x}_k) + b_m \leq 0 \\
& \log g(\mathbf{x}_k) + \frac{1}{g(\mathbf{x}_k)} \left(\mathbf{x}_k \odot \nabla g(\mathbf{x}_k)\right)^T \mathbf{d} \leq 0 \\
& \log h(\mathbf{x}_k) + \frac{1}{h(\mathbf{x}_k)} \left(\mathbf{x}_k \odot \nabla h(\mathbf{x}_k)\right)^T \mathbf{d} = 0 \\
& \log \mathcal{B}_i(\mathbf{x}_k) + \frac{1}{\mathcal{B}_i(\mathbf{x}_k)} \left(\mathbf{x}_k \odot \nabla \mathcal{B}_i(\mathbf{x}_k)\right)^T \mathbf{d} \leq 0 \\
& \log \mathcal{B}_e(\mathbf{x}_k) + \frac{1}{\mathcal{B}_e(\mathbf{x}_k)} \left(\mathbf{x}_k \odot \nabla \mathcal{B}_e(\mathbf{x}_k)\right)^T \mathbf{d} = 0 \\
& \mathbf{d} = \mathbf{y} - \log \mathbf{x}_k \\
& \mathbf{y} = \log \mathbf{x}
\end{aligned}
\tag{4.17}
$$

Equation 4.17 differs from Equation 4.13 in three ways. First, the posynomials have been exactly represented. Second, the monomial functions have also been pulled out as they are known to be linear, but this is not practically different from the LSQP method. Finally, the Reduced Lagrangian $\mathcal{L}_R$ has been used in place of the original Lagrangian $\mathcal{L}$.

The need for the Reduced Lagrangian is apparent when considering the Lagrangian as written

for Equation 4.17:

$$\mathcal{L}(\mathbf{y},\lambda) = \log f(\mathbf{x}) + \lambda \log \mathbf{p}(\mathbf{x}_k) + \lambda \log \mathbf{m}(\mathbf{x}_k) + \lambda \log \mathbf{g}(\mathbf{x}_k) + \lambda \log \mathbf{h}(\mathbf{x}_k) + \lambda \log \mathcal{B}_i(\mathbf{x}_k) + \lambda \log \mathcal{B}_e(\mathbf{x}_k) \tag{4.18}$$

The purpose of using the term $\frac{1}{2}\mathbf{d}^T\nabla^2\mathcal{L}(\mathbf{y}_k)\mathbf{d}$ in the objective function as opposed to the term $\frac{1}{2}\mathbf{d}^T\nabla^2 f(\mathbf{x}_k)\mathbf{d}$ is to bring in second order information from the constraints into the sub-problem [59]. However, the posynomial constraints are now being represented exactly in the sub-problem, and therefore include all the curvature information. Including curvature information from these constraints in the objective yields an inconsistency between the approximation in the sub-problem objective and the true curvature captured by the constraint. This inconsistency causes the sub-problems not to converge to an optimum solution for the general NLP. As such, the curvature approximation must be removed from the Lagrangian, resulting in a reduction from the original Lagrangian:

$$\mathcal{L}_R(\mathbf{y},\lambda) = \log f(\mathbf{x}) + \lambda \log \mathbf{g}(\mathbf{x}_k) + \lambda \log \mathbf{h}(\mathbf{x}_k) + \lambda \log \mathcal{B}_i(\mathbf{x}_k) + \lambda \log \mathcal{B}_e(\mathbf{x}_k) \tag{4.19}$$

The use of this reduced Lagrangian is critical for the success of SLCP. Using the original Lagrangian does result in convergence, but significant computational expense is wasted in resolving the inconsistent curvatures.

The SLCP algorithm goes one step further than LSQP, allowing for maximum exploitation of the known log-log convex structure while still allowing for black box analysis tools to be integrated into the optimization formulation. However, it is important to note that this is only *one* method for Sequential Log Convex Programming, with LSQP being another example, but many more possible extensions exist. Work by Agrawal [12] has outlined methods for constructing optimization problems that are convex in log space using a disciplined approach. Using such methods could allow for generalizations far beyond the SLCP algorithm presented here. The key however seems to be in the objective function. Many attempts

were made to develop a more general Sequential Geometric Programming algorithm, but the quadratic objective function constructed using the BFGS approximation of the Hessian seems to be a remarkably strong approach. Future work may continue down this path.

Further discussion of SLCP can be found in the paper [60].

4.3 Graphical Intuition

The mathematics behind SQP, LSQP, and SLCP is important, but perhaps not intuitive to those without extensive experience in optimization. Understanding can also be achieved using a simple graphical example. Consider the following geometric program:

$$\begin{aligned} &\text{minimize} && x^{-0.1} + 15x^{0.01} + y^{-0.1} + 15y^{0.01} \\ &\text{subject to} && y + 0.01x^{-1.1} + x^{0.1} \leq 1 \end{aligned} \tag{4.20}$$

which in transformed space becomes:

$$\begin{aligned} &\underset{u,v}{\text{minimize}} && \log\left(e^{-0.1u} + 15e^{0.01u} + e^{-0.1v} + 15e^{0.01v}\right) \\ &\text{subject to} && \log\left(e^{v} + 0.01e^{-1.1u} + e^{0.1u}\right) \leq 0 \end{aligned} \tag{4.21}$$

The problem can be directly visualized in 2D:

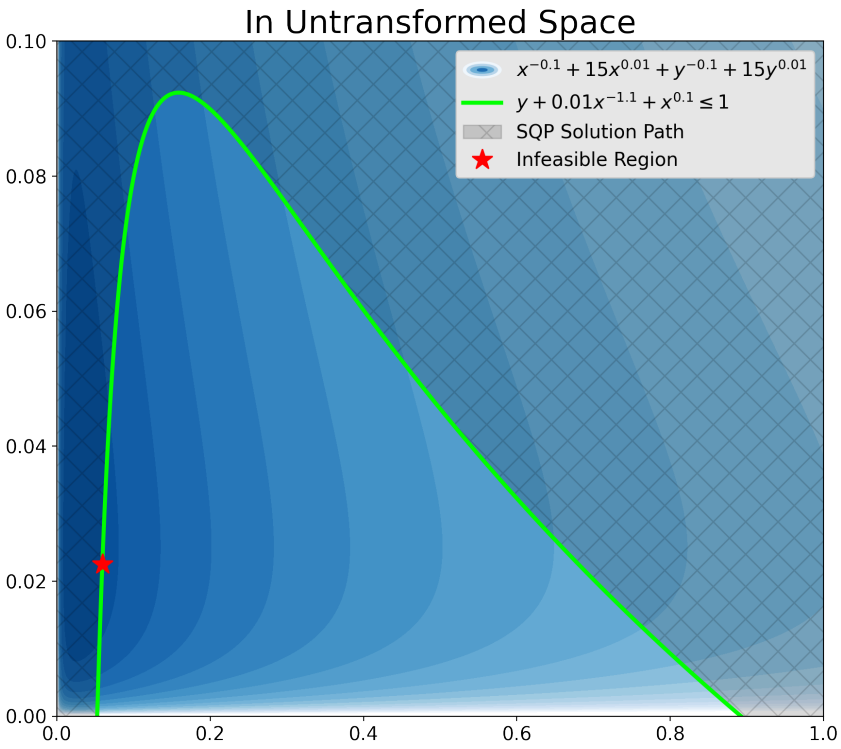
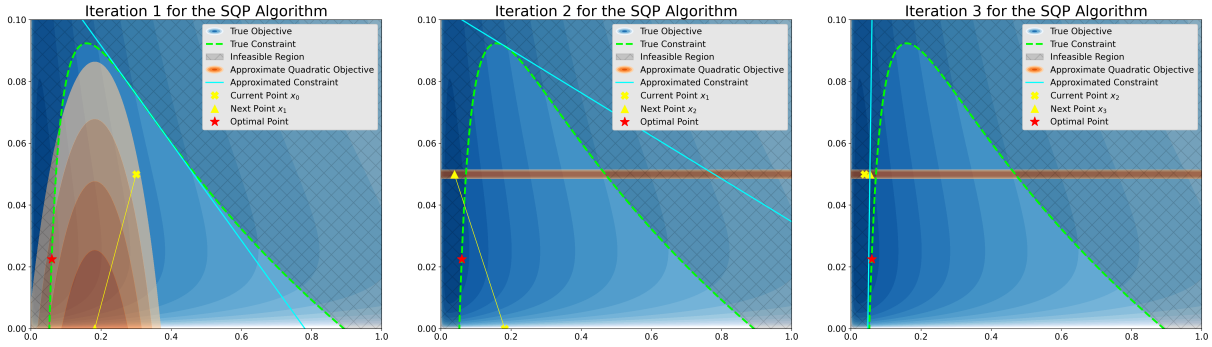


Figure 4-1: Visualization of the problem in Equation 4.20

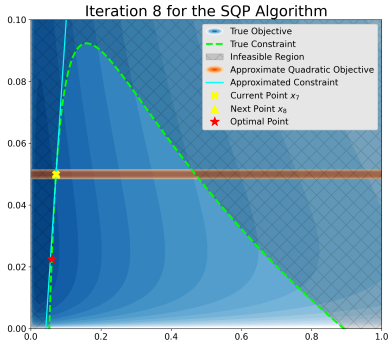
Sequential Quadratic Programming solves the problem in this space, and a nominal progression of the algorithm is shown in Figure 4-2.



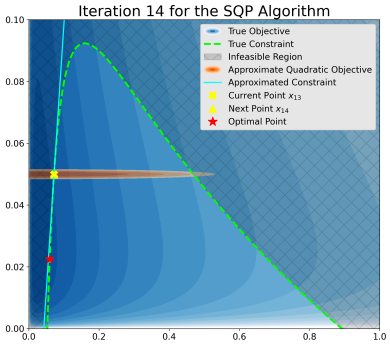
(a) Iteration 1

(b) Iteration 2

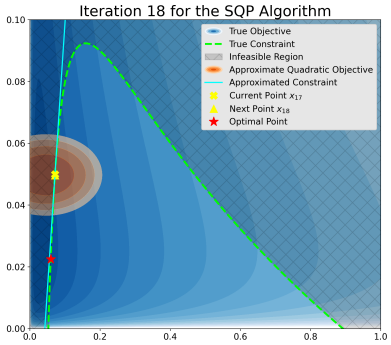
(c) Iteration 3



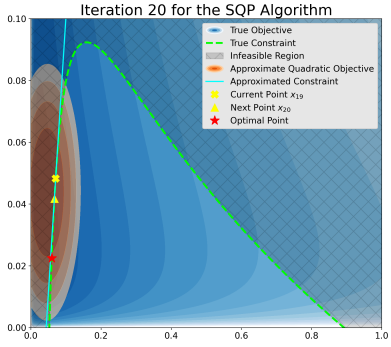
(d) Iteration 8



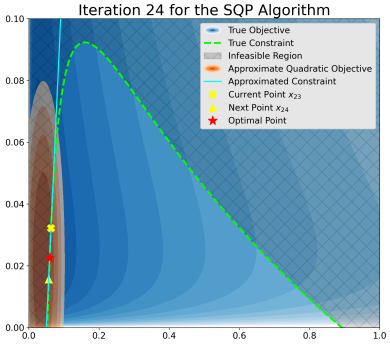
(e) Iteration 14



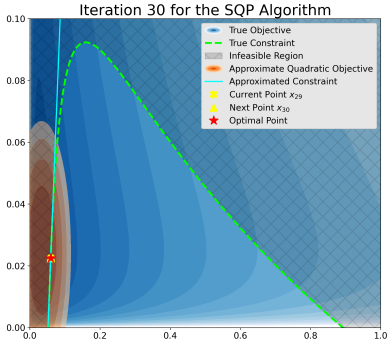
(f) Iteration 18



(g) Iteration 20



(h) Iteration 24



(i) Iteration 30

Figure 4-2: Visualization of the SQP algorithm solving a simple example problem

The first iterations send the candidate point to an infeasible region, at which point the algorithm spends approximately 10 iterations recovering from an exceptionally poor Hessian approximation. Eventually, the Hessian improves and the algorithm does find the optimal solution.

Logspace Sequential Quadratic Programming simply transforms this space to something better scaled and easier to iterate on, shown in Figure 4-3. An example of the algorithm in operation is shown in Figure 4-4.

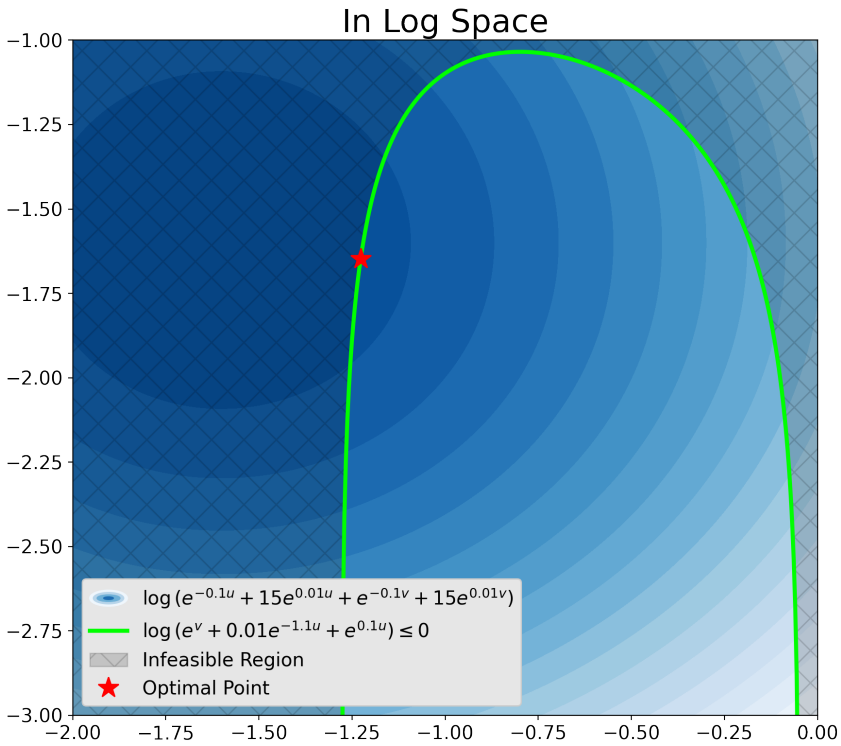
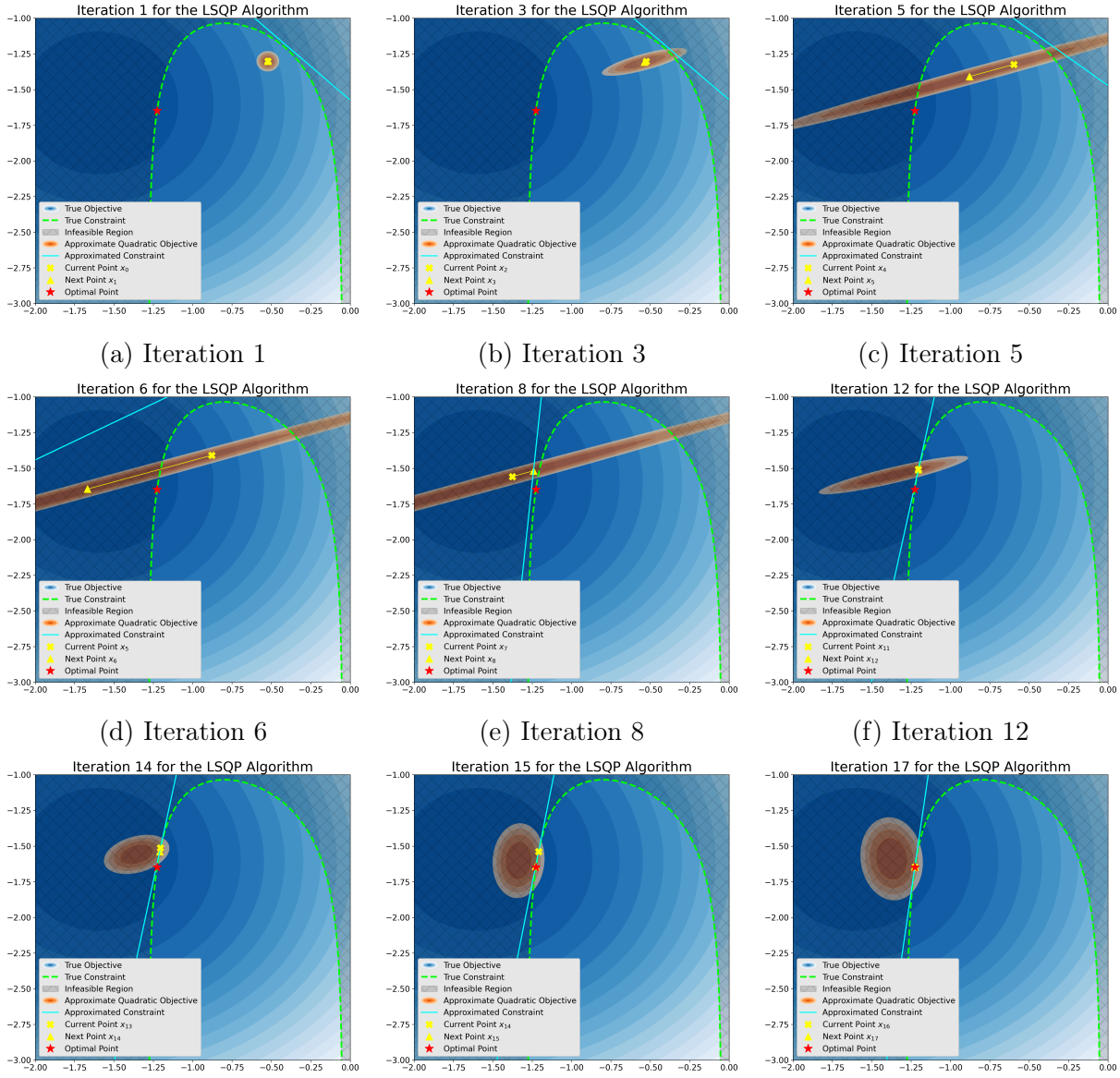


Figure 4-3: Visualization of the problem in Equation 4.20 in transformed space



(a) Iteration 1 (b) Iteration 3 (c) Iteration 5

(d) Iteration 6 (e) Iteration 8 (f) Iteration 12

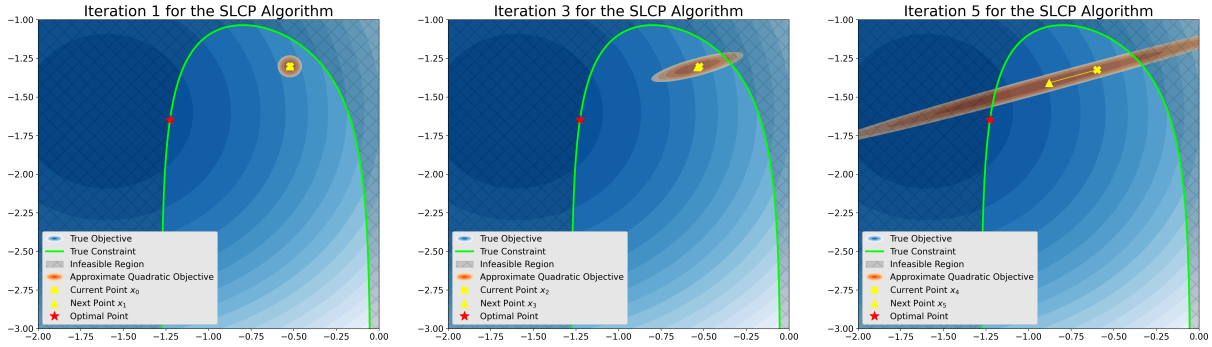
(g) Iteration 14 (h) Iteration 15 (i) Iteration 17

Figure 4-4: Visualization of the LSQP algorithm solving a simple example problem

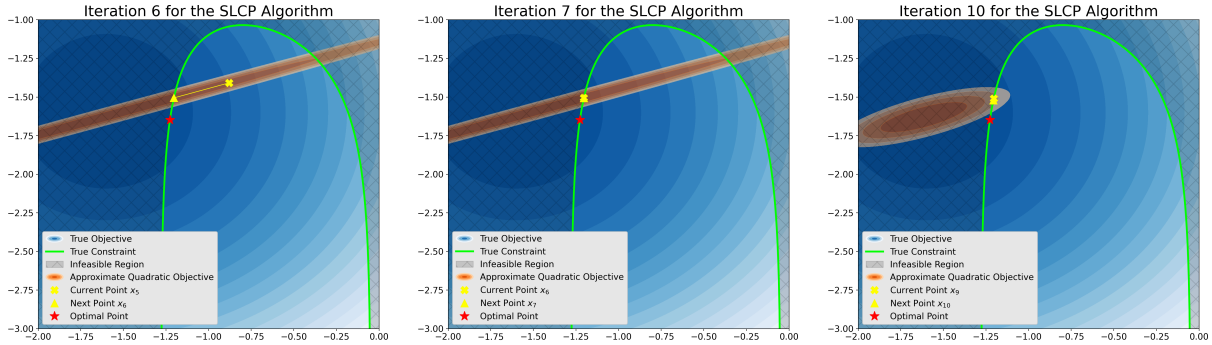
The benefits here are quite substantial. Level sets are much closer to elliptical, leading to a far better Hessian approximation, and the final convergence steps are much smoother.

Note however that the sixth iteration of LSQP steps well beyond the true feasible region since the posynomial constraint is not truly being enforced. In fact, the linear approximation is providing almost no useful information to the sub-problem, with the quadratic objective function driving this iterative step. But the posynomial constraint is known to be enforcing a convex feasible region, and so SLCP takes advantage of this additional knowledge (Figure 4-5).

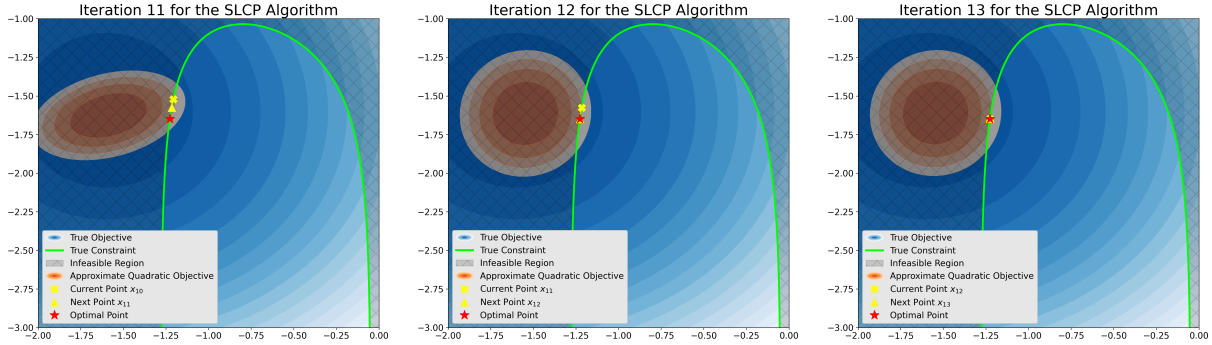
The direct enforcement of the posynomial constraint eliminates the overshoot at Iteration 6 and results in a reduction of 4 iterations between LSQP and SLCP. Additionally, the use of the Reduced Lagrangian allows the Hessian approximation to be a more faithful representation of the true underlying objective, since constraint curvature information is not being included.



(a) Iteration 1 (b) Iteration 3 (c) Iteration 5



(d) Iteration 6 (e) Iteration 7 (f) Iteration 10



(g) Iteration 11 (h) Iteration 12 (i) Iteration 13

Figure 4-5: Visualization of the SLCP algorithm solving a simple example problem

4.4 The Middle Ground Between GP and MDAO

The LSQP and SLCP algorithms use the underlying log-log convex structure discovered by the GP and SP approaches to aircraft design in order to more effectively solve the non-linear optimization problems that result form the presence of black box analysis tools. As a result, the LSQP and SLCP algorithms can therefore be viewed from one of two perspectives. First, LSQP and SLCP are the key enabling intellectual connections that allow the optimization first approach to design to be useful in practical aircraft design. Alternatively, LSQP and SLCP are tools for improving the efficiency of existing MDAO optimization methods that exploit new knowledge uncovered by the GP approach. But from either perspective, LSQP and SLCP are the key bridge between these two approaches that were previously fundamentally incompatible. This lowers the risk for committing to an optimization first approach to design, since a clear path now exists back to more traditional analysis first MDAO.

Chapter 5

Benchmarking the Performance of the LSQP and SLCP

5.1 Methodology

The primary objective in this chapter is to benchmark the LSQP and SLCP algorithms against a SQP baseline to validate their expected theoretical performance. To do this, 5 test problems were considered: Boyd [11], Rosenbrock, Floudas [61], Kirschen-Ozturk [51], and Hoburg [10]. These problems were selected using a number of criteria, including their visibility in the community as test problems, the extent to which the problems are GP compatible, if the problems were compatible with the log-log transformation, and the extent to which it was expected that LSQP and SLCP would provide benefit over SQP.

Test sets like CUTEst [62] were considered for benchmarking, but many of the problems in the CUTEst test set have negative variables and were not readily compatible with the log-transformation. The broader issue is that LSQP and SLCP open up a set of problems to solution that have previously been viewed as challenging, and so testing these algorithms against the existing test problems in CUTEst may not yield the desired understanding.

However, future work may look more seriously into this as LSQP and SLCP become more mature algorithms.

Each of the 5 test problems was subjected to some subset of 5 different solvers:

- A custom implementation of SQP (as described in Appendix B)
- A Matlab implementation of SQP (fmincon with the 'SQP' flag)
- A custom implementation of LSQP (as described in Appendix B)
- A log transformation followed by application of Matlab SQP (abbreviated as LT+SQP)
- A custom implementation of SLCP (as described in Appendix B)

The Matlab implementations are included primarily as a check to ensure that the custom algorithms were in line with an existing top of the line solver. The SQP algorithm in Python's Scipy package was also considered for comparison (an implementation of the algorithm from Kraft [63]), but lack of control on the termination condition meant a meaningful comparison was not possible.

For each test problem and algorithm pair, 3000 random initial guesses were generated. For 1000 of these guesses, the values of the optimization variables were bounded to be within +/- 10% of the known optimum. These 1000 guesses are referred to as being 'Good' initial guesses for this work. The second 1000 guesses were bounded to be within +/- 50% of the known optimum and are referred to as 'Reasonable' guesses. The final 1000 guesses were bounded to within +/- 80% and are referred to as 'Poor' guesses.

The desired algorithms were then run on the test problem from all 3000 initial guesses, leading to a total of 75000 cases considered in this benchmarking study. Due to the computational expense of these trials, the computational resources of the MIT SuperCloud were utilized [64] and a limit of 500 iterations was placed on each algorithm.

The results were then aggregated to obtain mean values for the optimal solution, the optimal objective value, and the number of iterations required to achieve solution. The number of cases (1000 for each combination of problem, algorithm, and guess quality) was selected such that removal of any single run from this aggregated set would result in a sufficiently small change in the aggregated mean values.

For all of the problems considered here, the objectives and constraints are all known explicitly and the solution is known from either an exact solve or from the literature. For all custom algorithms, analytical derivatives are available to the solver, with the exception of the Hoburg problem which will be discussed later. Both cases using the Matlab SQP algorithm were implemented with the default gradient computation method, as these cases were not the focus of this work.

Data will be presented in graphical format and via tables (see Appendix C). The tables are easy to interpret, but the graphs require some explanation. Consider Figure 5-1), which shows the probability of a randomly chosen initial condition in the set of 1000 converging to the known optimum in the number of iterations along the x-axis.

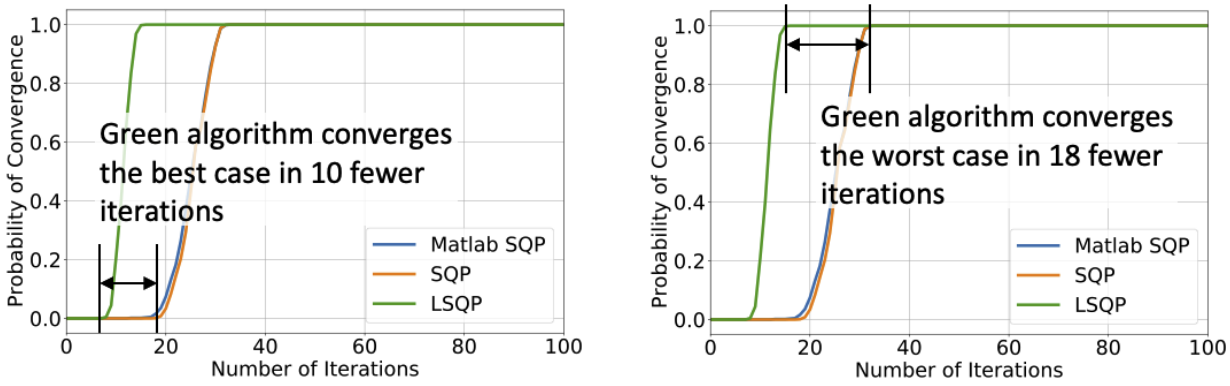


Figure 5-1: Plot of success probability vs. number of iterations. The green algorithm (LSQP) converges in 10 fewer iterations from the best of the 1000 starting guesses (8 vs. 18), and in 18 fewer iterations in the worst of the 1000 starting guesses (17 vs. 35)

The ideal algorithm would have a 'Γ' shape, converging 100% of the time with a single iteration, and so the area between curves is the computational savings of one algorithm over

the other. Additionally, the final success rate at the far right of the graphs indicates which algorithm has a higher overall success rate, as shown in Figure 5-2.

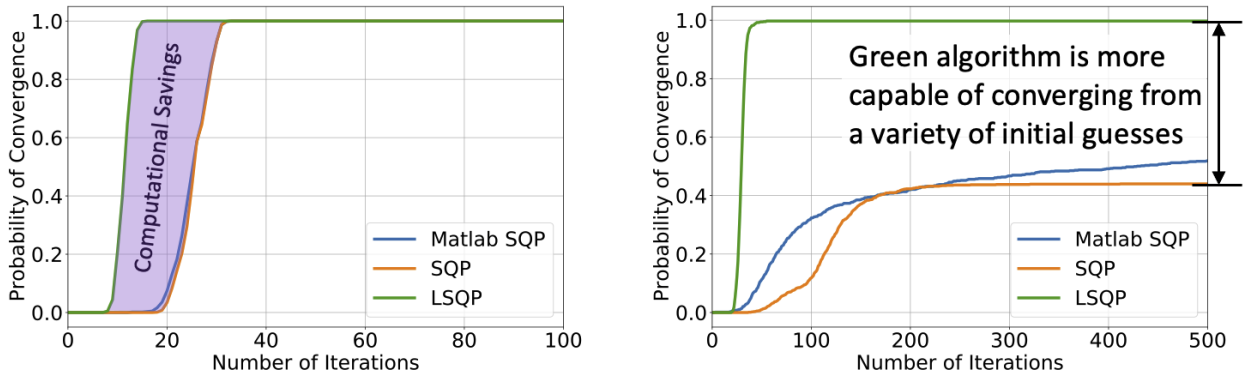


Figure 5-2: The two key criteria for interpreting the graphs. Area between curves is computational savings, and the displacement at right edge indicates the overall success rate of the algorithms.

5.2 The Boyd Problem

The first test case is a toy problem proposed by Boyd as an example of a simple Geometric Program [11]:

$$\begin{aligned} \underset{h,w,d}{\text{minimize}} \quad & 1/(hwd) \\ \text{subject to} \quad & 2\frac{hw}{A_{wall}} + 2\frac{hd}{A_{wall}} \leq 1 \\ & \frac{wd}{A_{floor}} \leq 1 \\ & \frac{\alpha w}{h} \leq 1 \\ & \frac{h}{\beta w} \leq 1 \\ & \frac{\gamma w}{d} \leq 1 \\ & \frac{d}{\delta w} \leq 1 \end{aligned} \tag{5.1}$$

This problem was selected to be a quick hit on the power of the log transformation, and is therefore only considered by the custom SQP, Matlab SQP, Matlab LT+SQP, and custom

LSQP. Since the problem consists primarily monomials, it was not expected to produce interesting comparisons with SLCP.

Solving this problem with LSQP requires significantly fewer iterations than traditional SQP. Figures 5-3 through 5-5 show that all of the LSQP trials converged within 10 iterations, while the best SQP case converges in the same number of iterations less than 80% of the time for a good initial guess, and less than 20% of the time for a poor initial guess. Appendix C includes tables that report a more detailed breakdown of the data runs.

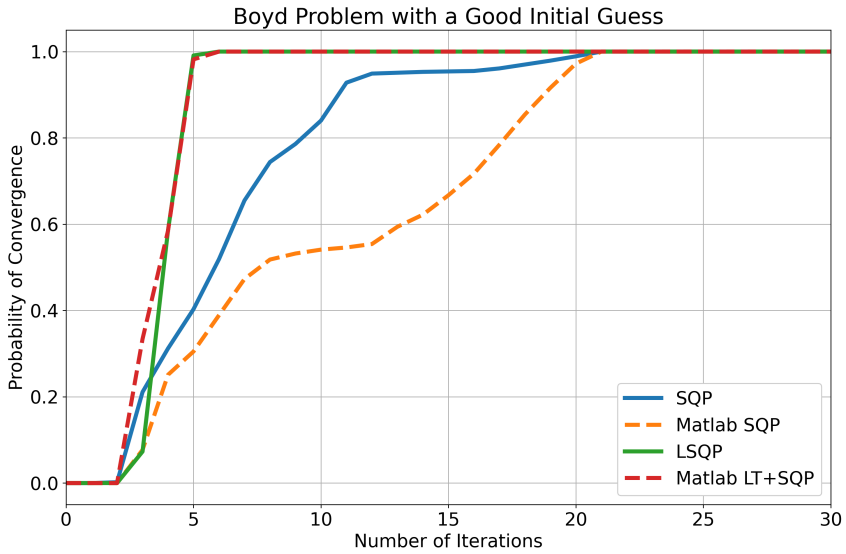


Figure 5-3: The probability of convergence vs. iteration count for initial guesses within +/- 10% of the known optimum

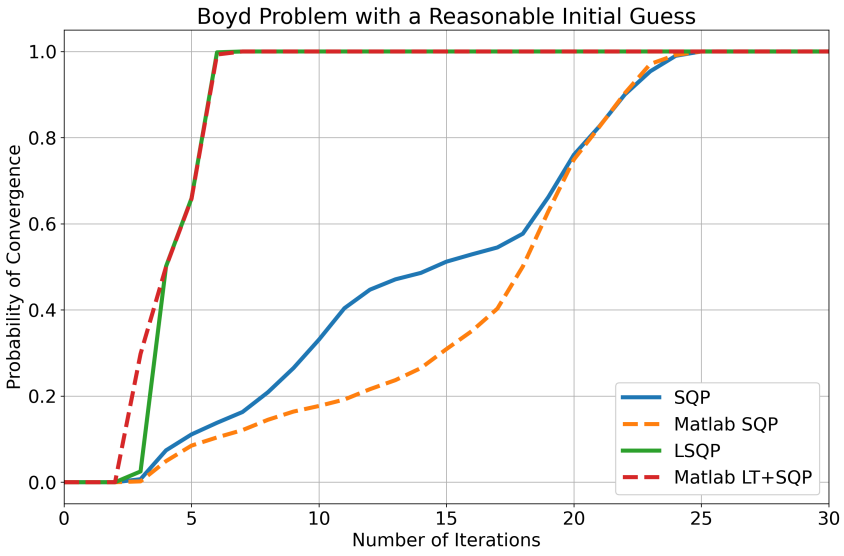


Figure 5-4: The probability of convergence vs. iteration count for initial guesses within +/- 50% of the known optimum

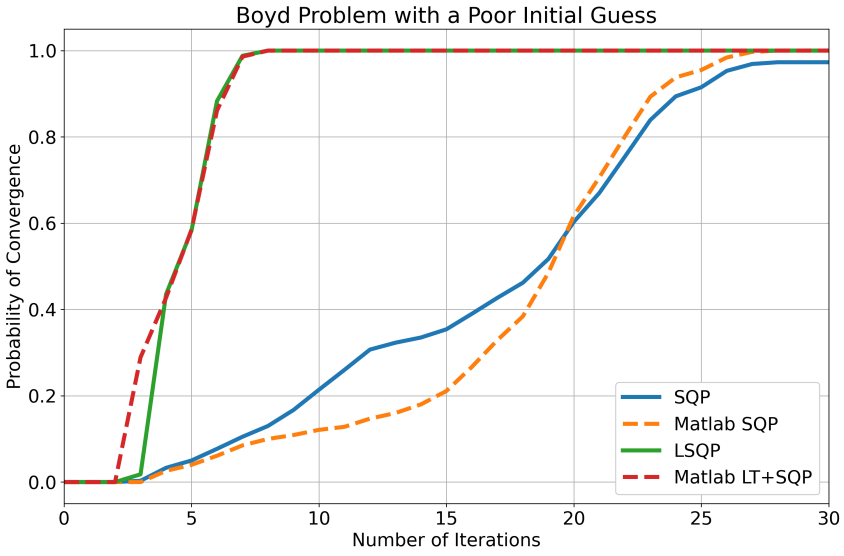


Figure 5-5: The probability of convergence vs. iteration count for initial guesses within +/- 80% of the known optimum

The Boyd Geometric Program demonstrates a clear win for LSQP and for the log transformation more generally: the number of required iterations is decreased by approximately 70% at no additional computational cost or sacrifice to solution quality. This benefit is unsurprising due to the large number of monomials which become exactly represented as affine under the transformation to logspace, and the perfect log-convexity exhibited by all geometric programs.

5.3 The Constrained Rosenbrock Problem

The Rosenbrock function is a common test case for optimization methods. It is unique in that the optimum resides in an extremely shallow local valley, making it an excellent test problem for gradient based methods. Consider the following constrained version of the problem:

$$\begin{aligned} \underset{x,y}{\text{minimize}} \quad & (1-x)^2 + 100(y-x^2)^2 + 1 \\ \text{subject to} \quad & (x-1)^3 - y + 2 \leq 1 \\ & x + y - 1 \leq 1 \\ & \frac{x}{1.5} \leq 1 \\ & \frac{y}{2.5} \leq 1 \end{aligned} \tag{5.2}$$

Note that a constant of 1 has been added to the objective to shift up the optimum and enable the log transformation of LSQP. The variables are also bound to be greater than a small positive constant to aid the construction of sub-problems.

Rosenbrock's problem is not a GP or SP as formulated, but can be reformulated by setting an intermediate variable z equal to the objective. However, an attempt to solve the problem in this modified form with the Difference of Convex Algorithm did not succeed. Fortunately, the optimum is known to be $f(1,1) = 1$. Only the SQP and LSQP cases for both custom and Matlab were considered for this problem due to its unique structure and the fact that it presents such a difficult challenge for the log transformed problems. Note that because

no posynomials exist in this problem as formulated, the SLCP algorithm will be exactly the LSQP algorithm in this case, hence a comparison is not necessary.

Unlike Boyd's Geometric Program, the Rosenbrock problem is a clear win for traditional SQP. Figures 5-6 through 5-8 along with tables in Appendix C show a 40-50% increase in iteration count for the LSQP modification.

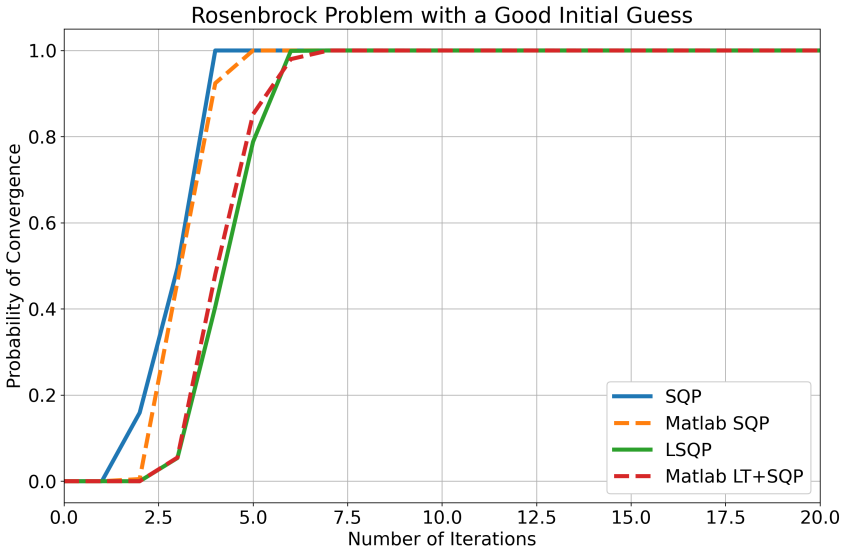


Figure 5-6: The probability of convergence vs. iteration count for initial guesses within +/- 10% of the known optimum

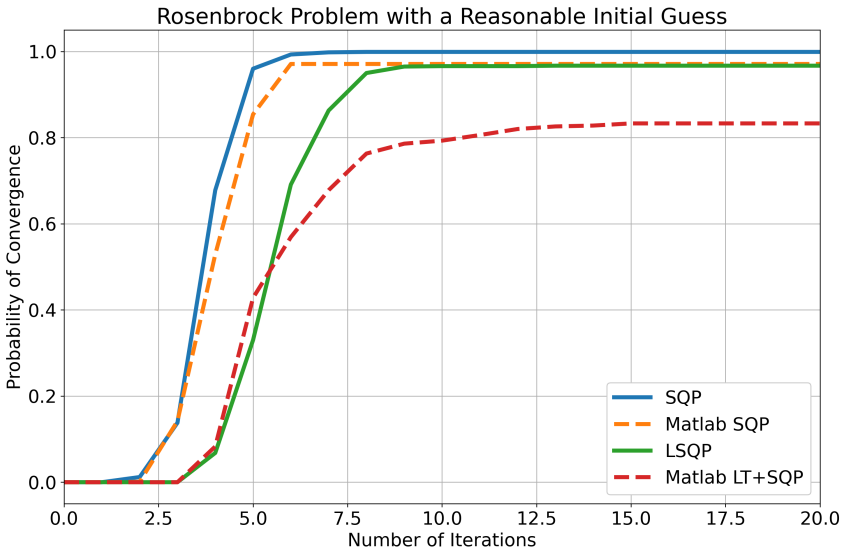


Figure 5-7: The probability of convergence vs. iteration count for initial guesses within +/- 50% of the known optimum

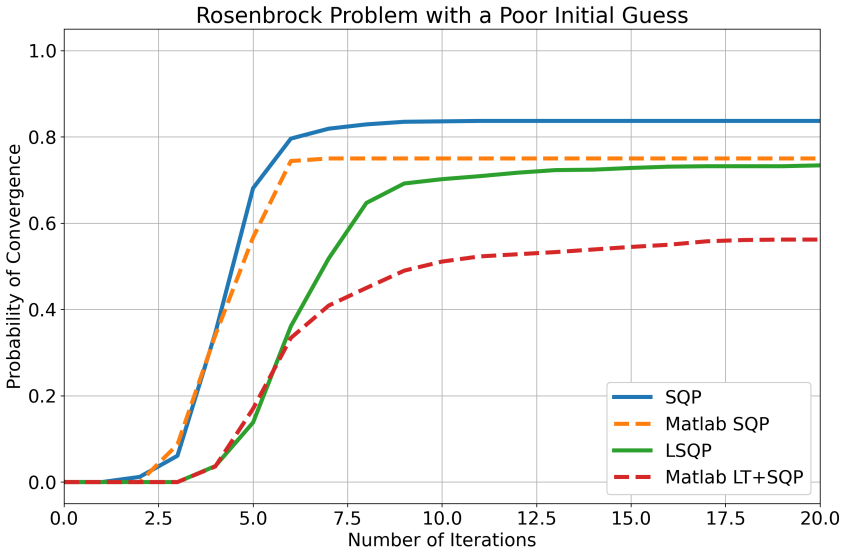
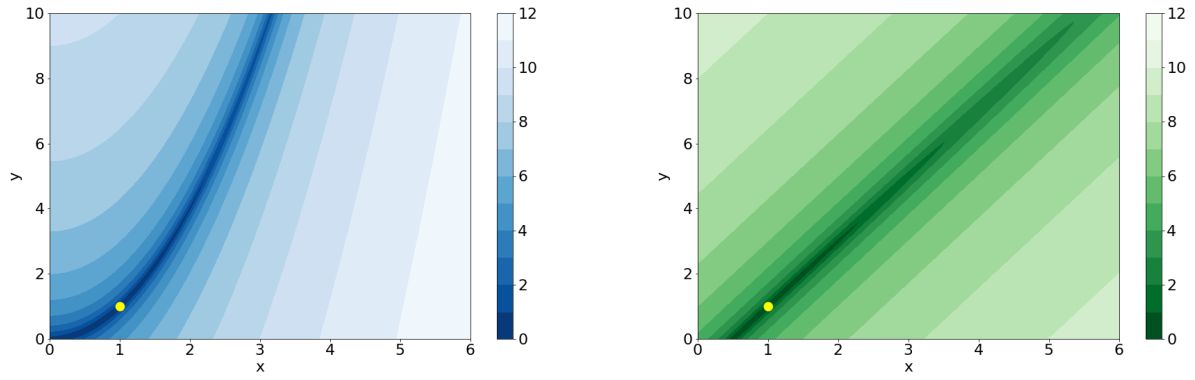


Figure 5-8: The probability of convergence vs. iteration count for initial guesses within +/- 80% of the known optimum

Note that in this case a trust region was implemented on the first three iterations, bounding $|\Delta x_i|/x_i \leq [0.2, 0.5, 1.0]$. In these early iterations, the Hessian approximation is quite poor and can send the candidate point off very far from the local region of interest. In addition, all three algorithms reach 100% success in the case of a good initial guess, but in Figure 5-8 between 15% and 45% of the trials converge to the local optimum $f(0, 0) = 2$, which is counted a failure of the algorithm since the global optimum is not reached.

So why does the traditional SQP outperform the LSQP modification? Signomial programming was presented in Chapter 2 as an extension of geometric programming, but just because the problem can be constructed as a Signomial Program does not imply underlying log-convexity, and the Rosenbrock Problem exhibits almost no underlying log-convexity.

This can be seen by visualizing the original Rosenbrock problem and the Rosenbrock problem under log transformation directly in Figures 5-9 and 5-10.

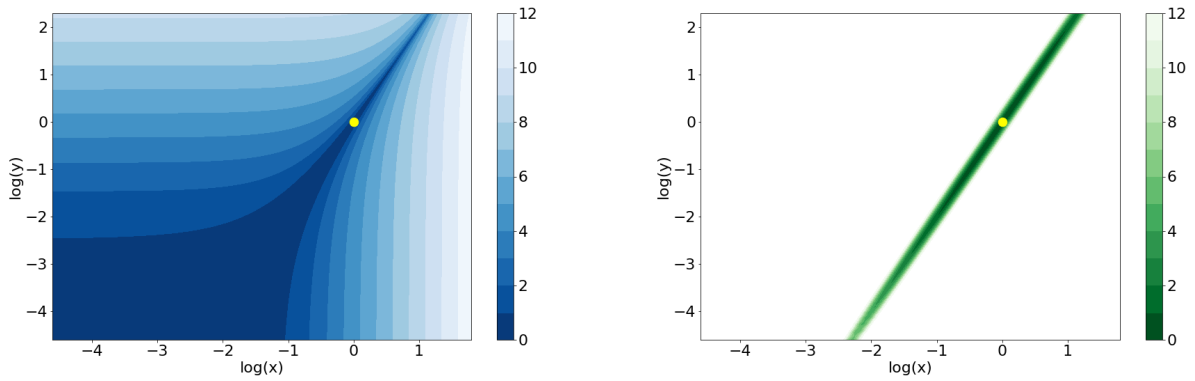


(a) The Untransformed Rosenbrock Problem

(b) Quadratic Approximation

Figure 5-9: A quadratic approximation of the Rosenbrock objective function in untransformed space. Approximation referenced about the global optimum indicated by a yellow dot.

The traditional SQP algorithm takes QP approximations in the space of Figure 5-9a, resulting in a QP objective function that appears like Figure 5-9b. Compare this visually to Figure 5-10, which represents the LSQP modification. In order to capture the local region around the reference point, the quadratic approximation in Figure 5-10b creates a deep narrow



(a) Rosenbrock Under Log Transformation

(b) Quadratic Approximation

Figure 5-10: A quadratic approximation of the Rosenbrock objective function in log transformed space. Approximation referenced about the global optimum indicated by a yellow dot. Note the large white area in (b) represents a region where the value of the function exceeds the contour range.

valley that clearly is a poor approximation of the objective function globally, indicated by the large white area where the contour range is exceeded. In other words, there is generally far more agreement between the original objective function Figure 5-9a and its quadratic approximations Figure 5-9b than there is between the transformed objective Figure 5-10a and its quadratic approximations Figure 5-10b. Since the traditional SQP algorithm has superior sub-problem representations, it is no surprise that it outperforms the LSQP modification in this case.

5.4 The Floudas Problem

Floudas [61] provides the following example for the design of a heat exchanger:

$$\begin{aligned}
\underset{x_1,\ldots,x_8}{\text{minimize}} \quad & x_1 + x_2 + x_3 \\
\text{subject to} \quad & \frac{833.33252x_4}{x_2x_6} + \frac{100}{x_6} - \frac{83333.333}{x_1x_6} \leq 1 \\
& \frac{1250x_5}{x_2x_7} + \frac{x_4}{x_7} - \frac{1250x_4}{x_2x_7} \leq 1 \\
& \frac{1250000}{x_3x_8} + \frac{x_5}{x_8} - \frac{2500x_5}{x_3x_8} \leq 1 \\
& 0.0025x_4 + 0.0025x_6 \leq 1 \\
& -0.0025x_4 + 0.0025x_5 + 0.0025x_7 \leq 1 \\
& -0.01x_5 + 0.01x_8 \leq 1
\end{aligned} \tag{5.3}$$

The Floudas problem is the first true design problem presented here, representing the design of a chemical reactor. The problem has 5 signomial constraints, and only a single GP-compatible posynomial (the 4th constraint). All 5 algorithms are considered, but the figures here are arranged for more direct comparison between algorithms, as a full reporting of results on a single plot would be excessively cluttered.

First, comparing the custom SQP and LSQP algorithms with their Matlab counterparts yields interesting results. Figure 5-11 clearly shows a win for LSQP over traditional SQP, but as guess quality decreases (Figures 5-12 and 5-13), the final convergence probability for custom LSQP begins to plateau at a value less than 100%.

Recall the discussion of the construction of standard form from Section 4.1. The Floudas problem presents in a compatible standard form, but there are signomials of the form $\mathbf{p}(\mathbf{x}) - \mathbf{n}(\mathbf{x}) \leq 1$. During the solution process, some of these constraints drop to ≤ 0, which invalidates the log transformation. In addition to the changes in underlying mathematics, the custom SLCP algorithm has been modified to reconstruct signomials into the form $\frac{\mathbf{p}(\mathbf{x})}{\mathbf{n}(\mathbf{x})} \leq 1$,

which negates the possibility of one of these constraints going negative during solve. LSQP can be modified in this way too, but leaving it as is led to this more interesting result for the purposes of discussion.

But why does the Matlab LT+SQP algorithm outperform SQP? One other solution to this problem is to bound the line search phase of the algorithm to keep all $g_i(\mathbf{x}_k) > 0$ and $h_j(\mathbf{x}_k) > 0$ at each step k. Though not possible to confirm, it is likely that Matlab's SQP algorithm is doing this bounding in keeping all candidate points x_k real, hence the higher success rate. However, enforcing these bounds comes at the expense of extra calls to the objective and constraint functions (the step size is decreased as if Armijo's rule was violated), and so it is not a preferred solution. A dedicated LSQP solver is already of value because it offloads the burden of the log transformation from the end user, but controlling these factors is perhaps the most substantive reason for a dedicated solver as opposed to implementing with an off the shelf SQP package.

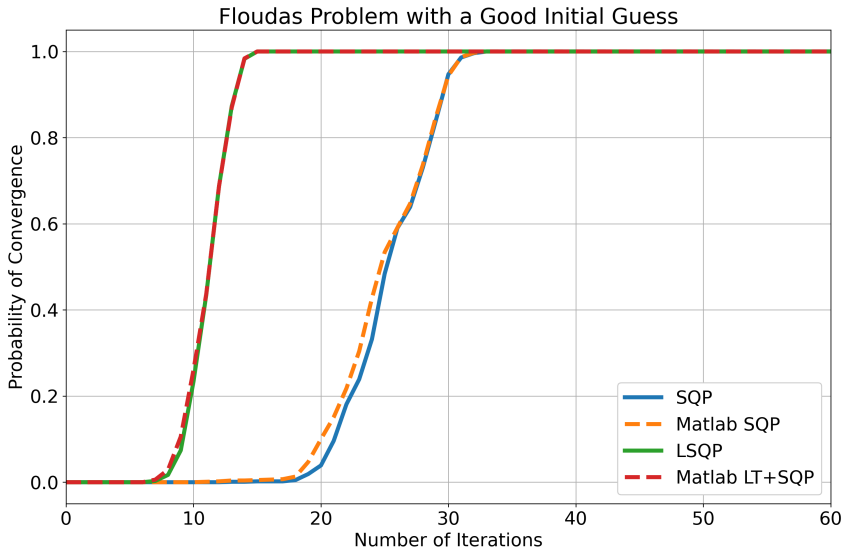


Figure 5-11: The probability of convergence vs. iteration count for initial guesses within +/- 10% of the known optimum

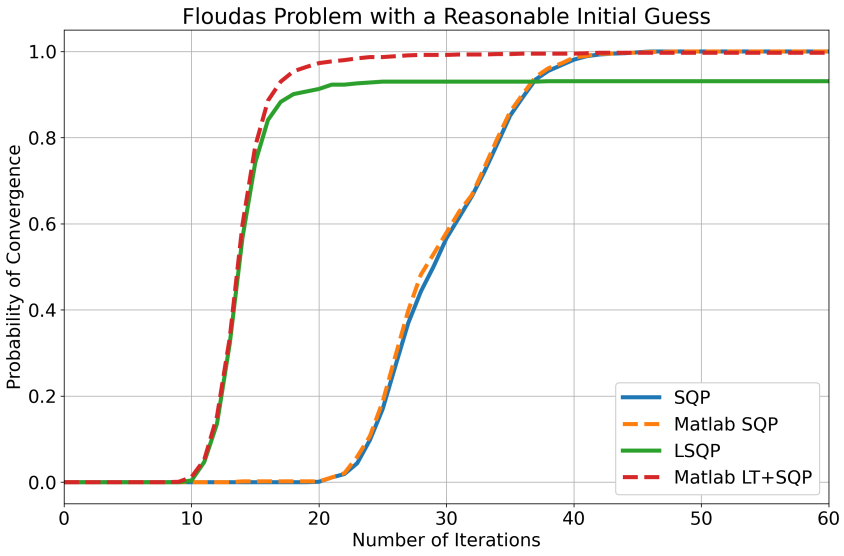


Figure 5-12: The probability of convergence vs. iteration count for initial guesses within +/- 50% of the known optimum

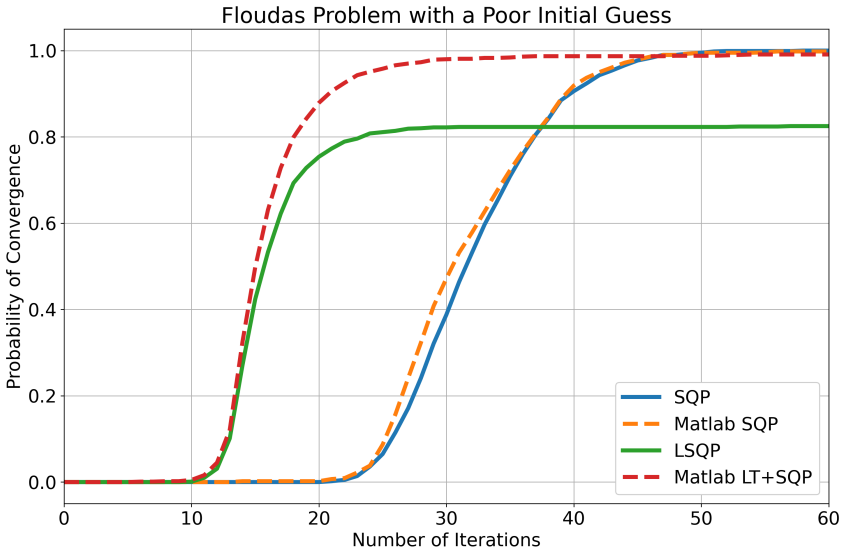


Figure 5-13: The probability of convergence vs. iteration count for initial guesses within +/- 80% of the known optimum

Figure 5-14 reports the probability of convergence curves for all three guess qualities for only the custom algorithms, now including SLCP. As expected, SLCP performs noticeably better than LSQP due to the more general sub-problem.

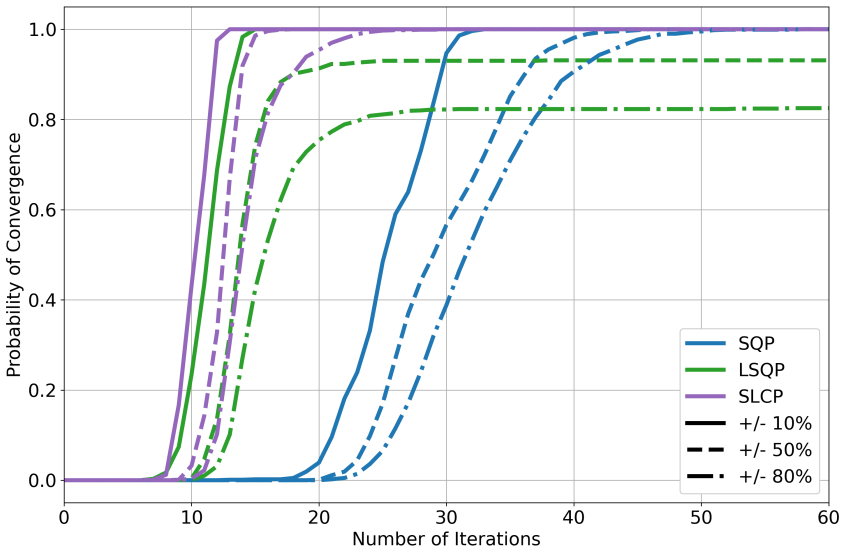


Figure 5-14: The probability of convergence vs. iteration count for all three custom algorithms

On the whole, the Floudas problem is another win for LSQP and SLCP, as significant computational savings is achieved by implementing the log transformation.

5.5 The Kirschen-Ozturk Problem

Kirschen [51] proposes a signomial program representing a low fidelity aircraft sizing, originally attributed to Ozturk. This problem is an extension of one of the more simple cases proposed by Hoburg [10]. It is GP compatible with the exception of a single fuel volume constraint, making it a non-convex signomial program:

$$\begin{aligned} \text{minimize} \quad & W_f \\ \text{subject to} \quad & W_f \geq c_T t D \\ & t \geq \frac{R}{V} \end{aligned}$$

$$
\begin{aligned}
&D \geq \frac{1}{2}\rho V^2 S C_D \\
&C_D \geq \frac{A_{C_{D_0}}}{S} + kC_f\frac{S_{wet}}{S} + \frac{C_L^2}{\pi Ae} \\
&C_f \geq 0.074 Re^{-0.02} \\
&Re \leq \frac{\rho V\sqrt{S/A}}{\mu} \\
&\frac{1}{2}\rho V^2 S C_L \geq W_0 + W_w + \frac{1}{2}W_f \\
&\frac{1}{2}\rho V_{min}^2 S C_{L_{max}} \geq W \\
&W \geq W_0 + W_w + W_f \\
&W_w \geq W_{w_{surf}} + W_{w_{strc}} \\
&W_{w_{surf}} \geq C_{W_{w,1}} S \\
&W_{w_{strc}} \geq C_{W_{w,2}}\frac{N_{ult}A^{\frac{3}{2}}\sqrt{(W_0 + V_{f_{fuse}}g\rho_f)WS}}{\tau} \\
&V_f \leq V_{f_{avail}} \\
&V_f = \frac{W_f}{g\rho_f} \\
&V_{f_{avail}} \leq V_{f_{wing}} + V_{f_{fuse}} \\
&V_{f_{wing}}^2 \leq 0.0009\frac{S^3}{A}\tau^2 \\
&V_{f_{fuse}} \leq A_{C_{D_0}} 10[m]
\end{aligned}
\tag{5.4}
$$

Unlike the previous problems which had known optima, this problem is non-convex and has no known solution *a priori*. Thus, the reference solution in this case is determined by using a signomial programming formulation and the Difference of Convex Algorithm (abbreviated as SP+DCA). This solution method is used in the original publication [51] and will therefore be treated as the optimum for the purposes of this section.

Kirschen showed an approximately 40% decrease in the number of iterations in comparing what was essentially LSQP vs SQP [51], but only considered a single initial guess that yielded an interpretable result. Figure 5-15 and tables in Appendix C confirm that result with a good

initial guess. However the trials with poor initial guesses demonstrate a far more significant win for LSQP and SLCP, as shown in Figure 5-17.

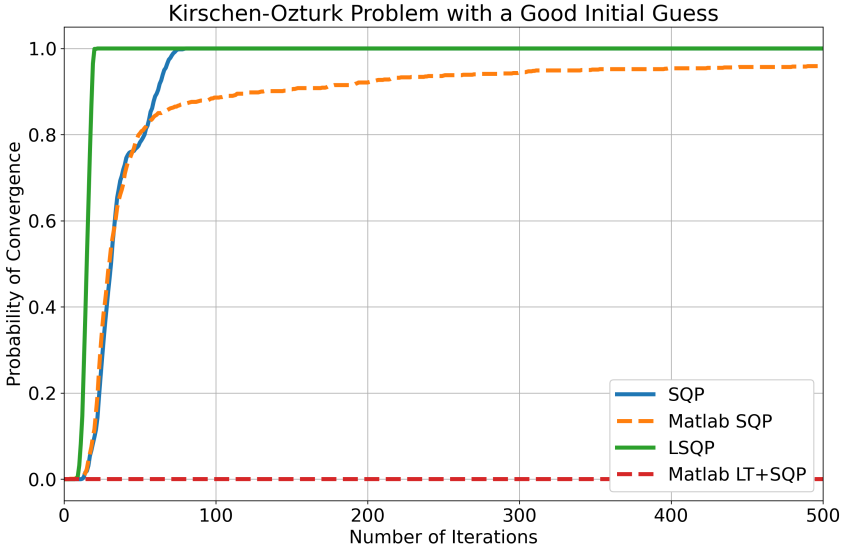


Figure 5-15: The probability of convergence vs. iteration count for initial guesses within +/- 10% of the known optimum

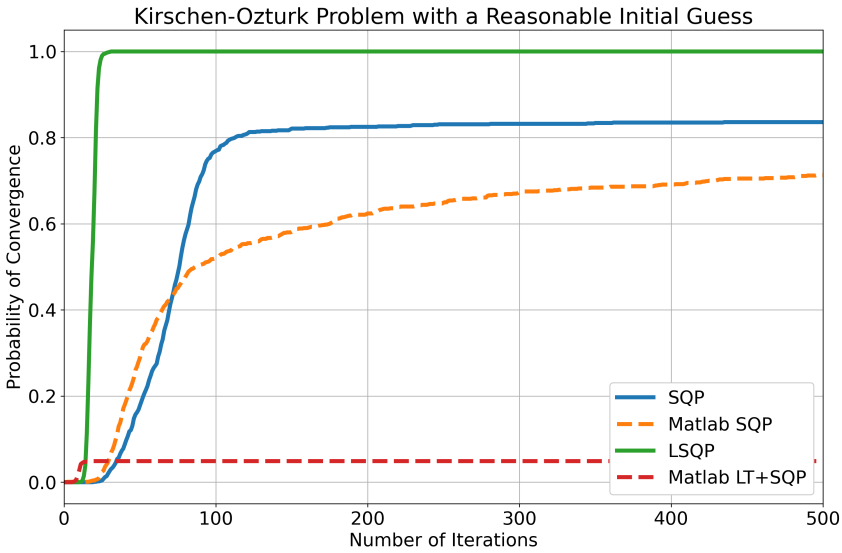


Figure 5-16: The probability of convergence vs. iteration count for initial guesses within +/- 50% of the known optimum

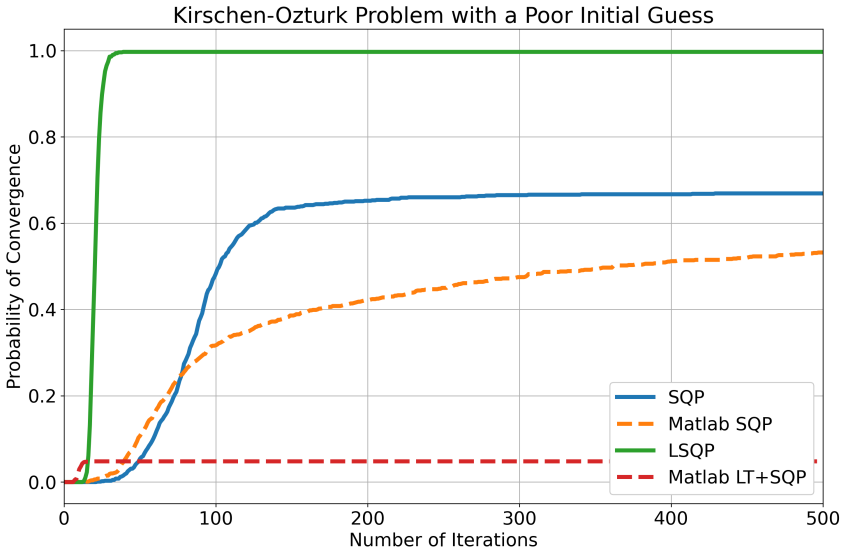


Figure 5-17: The probability of convergence vs. iteration count for initial guesses within +/- 80% of the known optimum

Figures 5-15 through 5-17 also show that almost none of the Matlab LT+SQP cases converged. Further analysis revealed that in these cases the algorithm terminated at an infeasible point due to a shrinking step size, but no further information is provided by the Matlab output. Those cases that did converge were quite far from the known optimum and are therefore counted as failed cases. It is difficult to draw a general conclusion from this single test problem, but the failure of the Matlab LT+SQP on so many cases further points to the value of a dedicated LSQP tool as opposed to working with an existing SQP software package.

Figure 5-18 also shows the superior performance of SLCP compared to LSQP as expected.

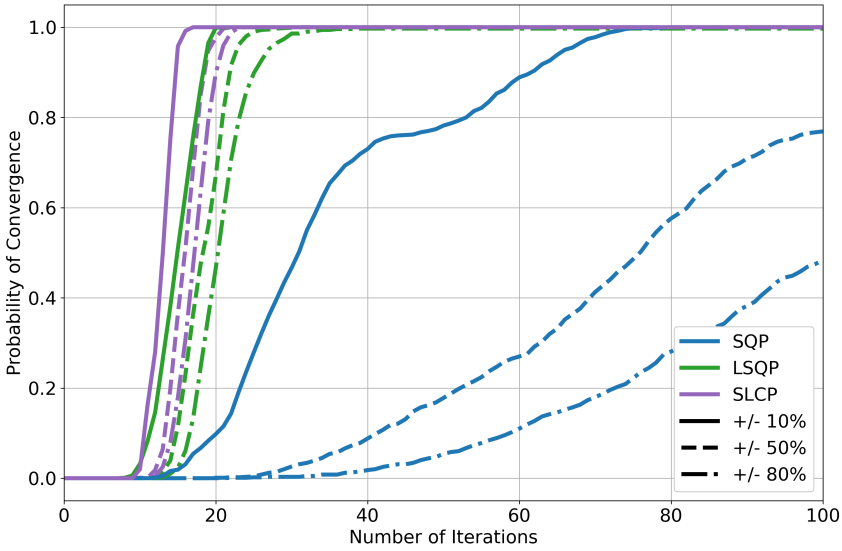


Figure 5-18: The probability of convergence vs. iteration count for all three custom algorithms

5.6 The Hoburg Problem

Hoburg [10] proposes a geometric program for conceptual aircraft design of a UAV. Note that this problem has already been written out in Chapter 1, but is reproduced here:

$$\text{minimize} \quad W_{\text{fuel,out}} + W_{\text{fuel,ret}}$$

subject to

$$R \geq 5000 \times 10^3$$

$$z_{\text{bre}} \geq \frac{gRT}{h_{fuel}\eta_0 W}$$

$$\frac{W_{fuel}}{W} \geq z_{\text{bre}} + \frac{z_{\text{bre}}^2}{2} + \frac{z_{\text{bre}}^3}{6} + \frac{z_{\text{bre}}^4}{24}$$

$$W = \frac{1}{2}\rho V^2 C_L S$$

$$T \geq \frac{1}{2}\rho V^2 C_D S$$

$$Re = \frac{\rho V S^{1/2}}{A^{1/2}\mu}$$

$$\eta_0 \leq \eta_{\text{eng}}\eta_{\text{prop}}$$

$$\eta_{\text{prop}} \leq \eta_i \eta_v$$

$$4\eta_i + \frac{T\eta_i^2}{\frac{1}{2}\rho V^2 A_{\text{prop}}} \leq 4$$

$$W_{\text{MTO}} \leq \frac{1}{2}\rho_{\text{sl}} V_{\text{stall}}^2 C_{L,\max} S$$

$$V_{\text{stall}} \leq 38$$

$$C_D \geq \frac{0.05}{S} + C_{Dp} + \frac{C_L^2}{\pi e A}$$

$$1 \geq 2.56\frac{C_L^{5.88}}{Re^{1.54}\tau^{3.32}C_{Dp}^{2.62}} + 3.8 \times 10^{-9}\frac{\tau^{6.23}}{C_L^{0.92}Re^{1.38}C_{Dp}^{9.57}} + 0.0022\frac{Re^{0.14}\tau^{0.033}}{C_L^{0.01}C_{Dp}^{0.73}}$$
$$+ ..... + 1.19 \times 10^4\frac{C_L^{9.78}\tau^{1.76}}{ReC_{Dp}^{0.91}} + 6.14 \times 10^{-6}\frac{C_L^{6.53}}{Re^{0.99}\tau^{0.52}C_{Dp}^{5.19}}$$

$$P_{\max} \geq \frac{T_{\text{sprint}} V_{\text{sprint}}}{\eta_{0,\text{sprint}}}$$

$$V_{\text{sprint}} \geq 150$$

$$W_{\text{pay}} \geq 0.500$$

$$\tilde{W} \geq W_{\text{fixed}} + W_{\text{pay}} + W_{\text{eng}}$$

$$W_{\text{zfw}} \geq \tilde{W} + W_{\text{wing}}$$

$$W_{\text{eng}} \geq 0.0372 P_{\max}^{0.803} \tag{5.5}$$

$$\frac{W_{\text{wing}}}{f_{\text{wadd}}} \geq W_{\text{web}} + W_{\text{cap}}$$

$$W_{\text{outbound}} \geq W_{\text{zfw}} + W_{\text{fuel,ret}}$$

$$W_{\text{MTO}} \geq W_{\text{outbound}} + W_{\text{fuel,out}}$$

$$W_{\text{sprint}} = W_{\text{outbound}}$$

$$2q \geq 1 + p$$

$$p \geq 1.9$$

$$\tau \leq 0.15$$

$$\bar{M}_r \geq \frac{\tilde{W} A p}{24}$$

$$0.92 \bar{w} \tau \bar{t}_{\text{cap}}^2 + \bar{I}_{\text{cap}} \leq \frac{0.92^2}{2} \bar{w} \tau \bar{t}_{\text{cap}}$$

$$8 \geq \frac{N_{\text{lift}} \bar{M}_r A q^2 \tau}{S \bar{I}_{\text{cap}} \sigma_{\max}}$$

$$12 \geq \frac{A \tilde{W} N_{\text{lift}} q^2}{\tau S \bar{t}_{\text{web}} \sigma_{\max,\text{shear}}}$$

$$\nu^{3.94} \geq 0.86 p^{-2.38} + 0.14 p^{0.56}$$

$$W_{\text{cap}} \geq \frac{8 \rho_{\text{cap}} g \bar{w} \bar{t}_{\text{cap}} S^{3/2} \nu}{3 A^{1/2}}$$

$$W_{\text{web}} \geq \frac{8 \rho_{\text{web}} g r_h \tau \bar{t}_{\text{web}} S^{3/2} \nu}{3 A^{1/2}}$$

This problem was not evaluated with SQP, as neither the Matlab algorithm nor the custom implementation successfully converged from any tested initial guess, but results from LSQP and SLCP are shown in Figure 5-19.

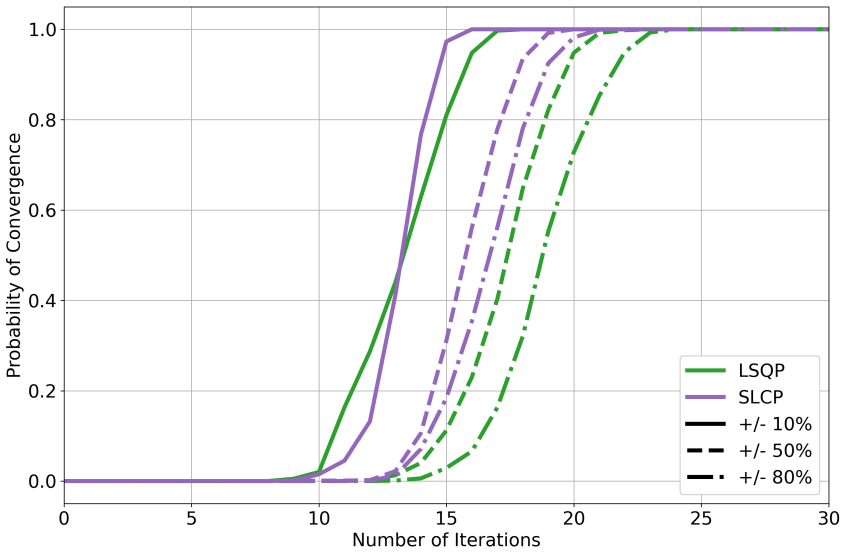


Figure 5-19: The probability of convergence vs. iteration count for the LSQP and SLCP algorithms

SLCP clearly outperforms LSQP from Reasonable and Poor initial guesses, however the Good initial guess graph shows no clear winner between the two algorithms. SLCP has some cases that converge faster than LSQP and some cases which converge slower, and so the two algorithms come out at somewhat of a wash. The lack of performance separation is because when the initial guess is close to the true optimum, the linear constraints of LSQP are good approximations of the local region. In contrast, SLCP extracts its benefit from mismatches between the true constraint and the linear approximation (see Section 4.3). And so, the further the initial guess is from the optimum, and the larger the discrepancy between the true constraint and the linear approximation, the larger the performance gap will be between LSQP and SLCP.

Though the LSQP and SLCP algorithms were developed with black box analysis in mind, none of the previous example problems actually operated on black box functions. But taking the Hoburg problem to be representative of an aircraft design problem, some of the constraints such as the weight relations and efficiency build ups are exact representations,

while other constraints are clearly not. Consider specifically the constraint:

$$1 \geq 2.56\frac{C_L^{5.88}}{Re^{1.54}\tau^{3.32}C_{Dp}^{2.62}} + 3.8\times10^{-9}\frac{\tau^{6.23}}{C_L^{0.92}Re^{1.38}C_{Dp}^{9.57}} + 0.0022\frac{Re^{0.14}\tau^{0.033}}{C_L^{0.01}C_{Dp}^{0.73}} + ... + 1.19\times10^{4}\frac{C_L^{9.78}\tau^{1.76}}{ReC_{Dp}^{0.91}} + 6.14\times10^{-6}\frac{C_L^{6.53}}{Re^{0.99}\tau^{0.52}C_{Dp}^{5.19}} \tag{5.6}$$

which is an expression for the profile drag of an airfoil section, C_{Dp}. It is highly likely that this constraint will eventually need to be replaced by a higher fidelity tool such as Xfoil [46], providing the perfect opportunity to study the effect of black box functions on the performance of LSQP vs SLCP.

Recall that SLCP gains advantage over LSQP by exactly representing posynomial constraints. Thus, if all the posynomials become black boxes, it is expected that LSQP and SLCP would exhibit exactly the same performance. So, given that the constraint in Equation 5.6 is actually three constraints (one for the outbound leg, one for the inbound leg, and one for the sprint leg), two other variants of this problem are used to simulate slowly integrating a higher fidelity aerodynamic model into the optimization problem.

First, the sprint constraint is black boxed to the solver, returning only function evaluations and gradients. The results show a minor reduction in the performance gap between LSQP and SLCP as expected, but the most important takeaway here is that with the introduction of a true black box (albeit a black box of known mathematics), both algorithms still perform well.

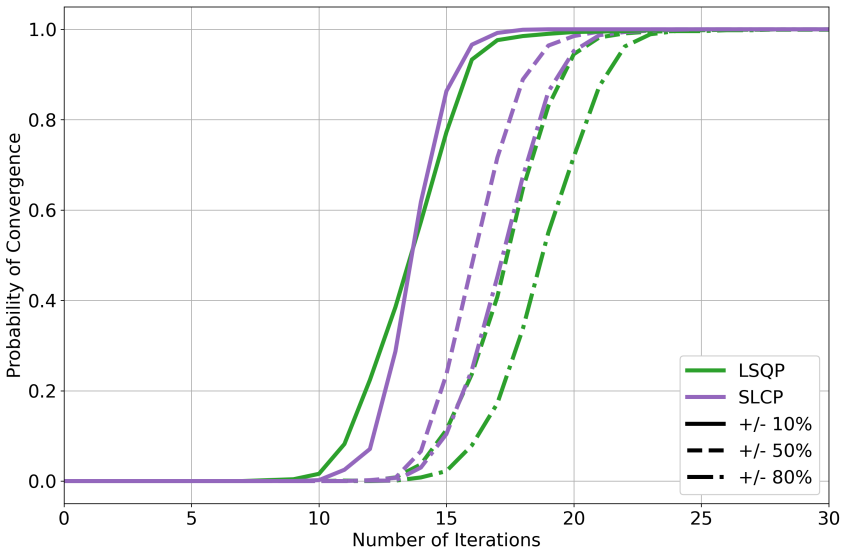


Figure 5-20: The probability of convergence vs. iteration count for the LSQP and SLCP algorithms

Black boxing all three constraints for C_{Dp} further reduces the performance gap between SLCP and LSQP, but once again the use of additional black box constraints does not pose problems for either LSQP or SLCP.

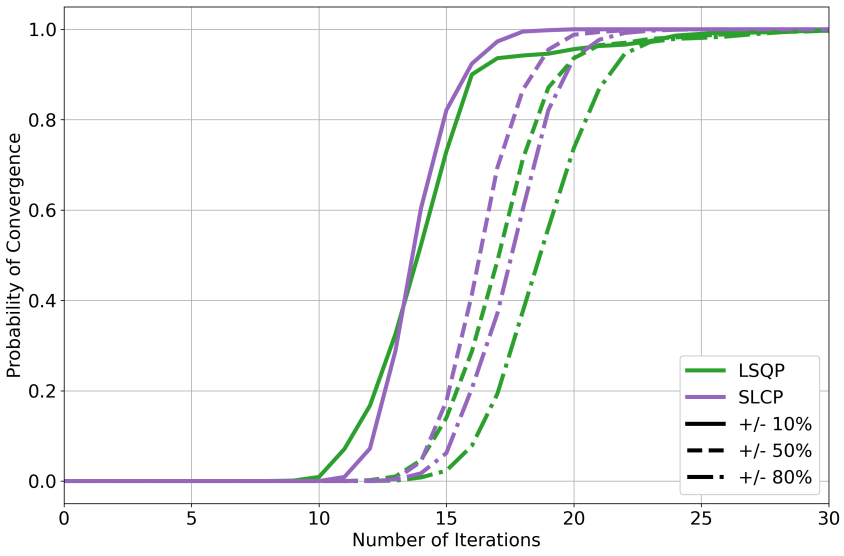


Figure 5-21: The probability of convergence vs. iteration count for the LSQP and SLCP algorithms

5.7 Discussion of Algorithm Performance

A full report of the aggregated data is available in Appendix C, but Tables 5.1 through highlight the benefits of LSQP and SLCP for the representative engineering design problems considered here.

Table 5.1: Aggregated iteration count and percentage improvement from the benchmarking test cases with good initial guesses (+/- 10%)

Problem	SQP	LSQP	SLCP
Floudas	25.87	11.70 (-55%)	10.74 (-58%)
Kirschen-Ozturk	33.93	15.36 (-57%)	13.34 (-63%)
Hoburg-0	-	13.70	13.65 (-0.32%)
Hoburg-1	-	14.06	14.17 (+0.78%)
Hoburg-3	-	14.65	14.32 (-2.27%)

Table 5.2: Aggregated iteration count and percentage improvement from the benchmarking test cases with reasonable initial guesses (+/- 50%)

Problem	SQP	LSQP	SLCP
Floudas	30.17	14.30 (-53%)	12.92 (-57%)
Kirschen-Ozturk	72.55	18.89 (-74%)	16.38 (-77%)
Hoburg-0	-	17.79	16.29 (-8.44%)
Hoburg-1	-	17.82	16.66 (-6.47%)
Hoburg-3	-	17.66	16.87 (-4.49%)

Table 5.3: Aggregated iteration count and percentage improvement from the benchmarking test cases with poor initial guesses (+/- 80%)

Problem	SQP	LSQP	SLCP
Floudas	32.79	16.29 (-50%)	14.92 (-54%)
Kirschen-Ozturk	91.18	21.06 (-77%)	17.66 (-81%)
Hoburg-0	-	19.33	17.13 (-11.40%)
Hoburg-1	-	19.30	17.70 (-8.30%)
Hoburg-3	-	19.33	18.02 (-6.77%)

On the whole, the SLCP algorithm demonstrates a 54-81% reduction in iteration count when compared to SQP, and enables some problems (such as the Hoburg problem considered here) to be solved where SQP comes up short. LSQP also does well, coming in between 5-11% short of SLCP.

These benchmarking cases go a long way to demonstrating the utility of LSQP and SLCP, and provide evidence that these algorithms will successfully bridge the gap between GP and SQP. But how should these algorithms be used in practical aircraft design?

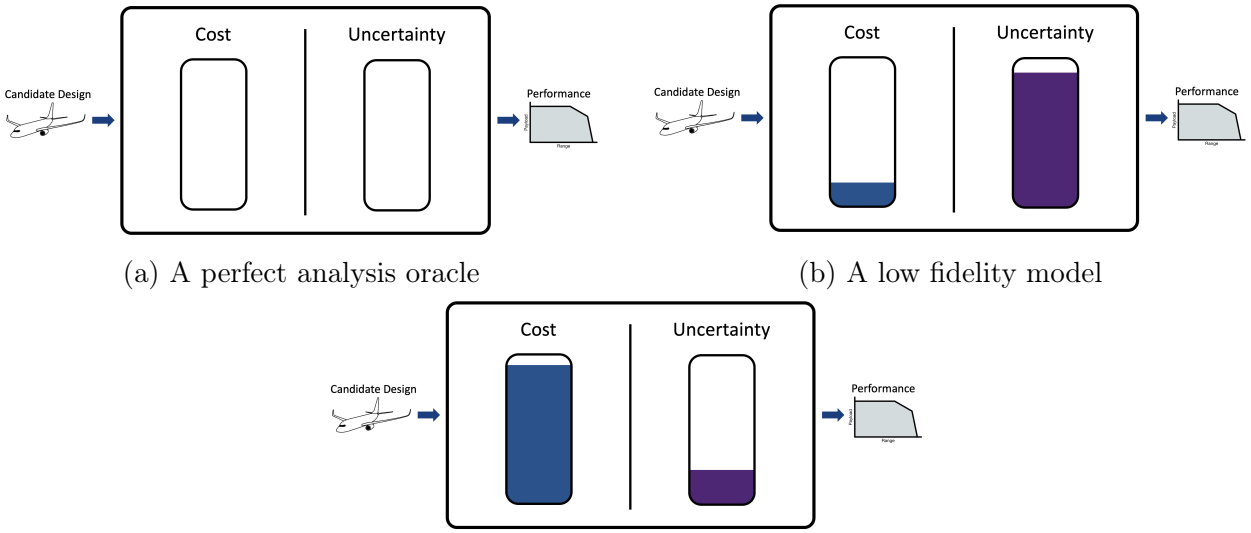
Chapter 6

An Optimization Centered Multifidelity Approach to Design

6.1 Multifidelity Modeling

Section 1.3 introduced the concepts of analysis models and model fidelity, and briefly discussed the perceived trend between the cost of evaluating a model versus the uncertainty of the result that comes out from that model (Figure 1-4). Based on that trend, each analysis model can qualitatively represented by bar gauges for cost and uncertainty (Figure 6-1).

In the traditional three phase process for aircraft design (Conceptual, Preliminary, Detailed), low fidelity models are used in the early Conceptual Phase, where a large space of possible designs is being considered. Then higher fidelity models are integrated in later design phases when reducing uncertainty becomes more of a premium. In this way, analysis models form something of a continuum from low to high fidelity, and a critical task for the human design team is selecting the appropriate analysis models to use at any given stage in the design process. This notion of using many different levels of model fidelity during the design process



(a) A perfect analysis oracle

(b) A low fidelity model

(c) A high fidelity model

Figure 6-1: Representation of various levels of fidelity using cost and uncertainty bars

is sometimes referred to as *mutifidelity* design or modeling.

There is extensive literature in the field of multifidelity modeling and uncertainty quantification (see [65] for an overview), but the focus here is not rigorous mathematical approaches. Instead, the purpose here is to discuss how LSQP and SLCP enable an optimization centered approach to multifidelity design and demonstrate this approach on two representative example problems, laying the foundation for future work that may connect to more rigorous approaches to multifidelity (see Chapter 9 for a brief discussion).

6.2 Multifidelity Analysis and Geometry

As fidelity is increased, the notion is that the core mathematical models become more and more accurate at modeling the true physics. But higher fidelity physics modeling is also coupled with an evolution in the geometry being used to represent the candidate design. For example, running 3D Navier-Stokes CFD on a wing that is defined by only span and chord does not make a lot of sense, and in fact would likely be counterproductive. Similarly,

requiring a fully defined 3D wing to use Lifting Line Theory is also a waste of resources. Thus, geometry fidelity must evolve to match analysis fidelity over the course of any effort that utilizes multifidelity modeling.

But for a fixed selection of analysis model, increasing geometry fidelity often yields improved performance. For example, an optimizer can only do so much with an airfoil defined by only 2 variables, but if that same analysis is used with an airfoil defined by 100 variables, then the optimizer is likely to find a superior design. However, there will be diminishing returns at some point due once the design becomes limited by physics and not geometry fidelity. And excessive numbers of design variables can yield unexpected and non-optimal solutions [66].

6.3 Swapping Constraints to Change Model Fidelity

Multifidelity modeling can be leveraged in the design process in few ways, but the simplest option is to select a model that has the appropriate balance of cost and uncertainty, and then increase the fidelity of that model whenever it is perceived that the uncertainty must be reduced. Other approaches exist (see one example in [67]), but nearly every method relies on expert information to drive the choice of analysis model.

The SLCP algorithm offers a path forward for integrating analysis models of varying fidelities into the optimization first approach to design. Conceptual design that begins as a geometric program can grow in preliminary design to include black box analyses that can range from low to mid fidelity all the way up through high fidelity tools used in traditional MDAO. All that is required is the exchanging of the relevant constraints that are used to represent that particular analysis model.

Consider the Hoburg Problem from Section 5.6. The constraint for C_{D_p} was isolated as being a model that could potentially be a target for increased fidelity modeling. The constraint

itself is merely a model for the function:

$$C_{D_p} = f(C_L, Re, \tau) \tag{6.1}$$

and in the final two cases, this model was black boxed away from the LSQP and SLCP algorithms and appeared only in this form. But the actual mathematics used to evaluate $C_{D_p} = f(C_L, Re, \tau)$ are irrelevant to the optimization algorithm. Work by Hoburg [47] has suggested this function is fairly close to log-log convex, and increasing fidelity should not substantially change the underlying shape of a drag polar. Thus, any model which captures the relationship $C_{D_p} = f(C_L, Re, \tau)$ should work just fine as a constraint, whether this model be the GP compatible fit, XFOIL, 2D DNS, or a run in a wind tunnel.

Should fidelity need to be increased, one only needs to simply change the mathematics of $C_{D_p} = f(C_L, Re, \tau)$ to be of higher fidelity and rerun the optimization. And should some mistake be discovered and a redesign cycle become necessary, the optimization formulation can simply be reverted back to the simpler mathematics, at which point the optimization formulation can be corrected and the solver rerun. To make this even more appealing, the results of the lower fidelity analysis can be fed in to the next level of fidelity as an initial guess, which minimizes the computational cost that comes from each new optimization run.

This capability is a unique feature that follows from the SAND approach to design. Since analysis models are available in the optimization formulation, making exchanges is relatively simple. But as problems become more complex and more black boxes are added, the architecture will increasingly appear as an IDF framework. Once that point is reached, algorithms like LSQP and SLCP may be of less benefit, as there is no longer any structure to exploit.

6.4 The Multifidelity Design Process

The multifidelity design process will be represented here by a flowchart with two main loops (Figure 6-2), a design loop and an analysis fidelity loop.

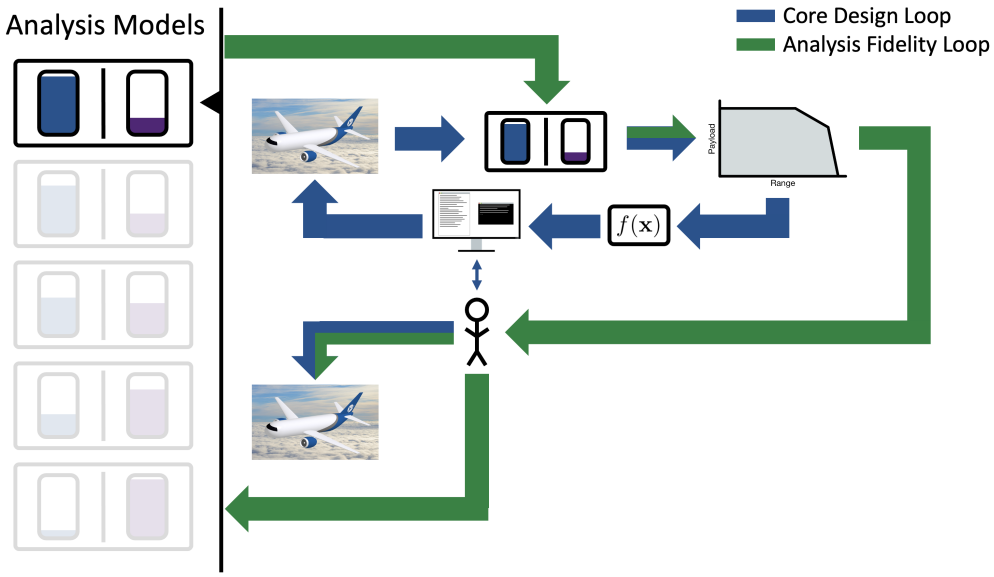


Figure 6-2: The accelerated multifidelity design process

The design loop represents one cycle of numerical optimization, in this case a GP, SQP, LSQP, or SLCP solve. The outer analysis fidelity loop is driven by a human designer or design team, and is responsible for choosing the appropriate analysis model or models to be used in the core design loop.

The design cycle therefore proceeds as follows:

1. A human chooses an analysis model
2. A human chooses a candidate aircraft design
3. The analysis model is queried to produce the performance data associated with the candidate design
4. The the optimization algorithm cycles the core design loop to completion

5. The human examines the final performance data and decides if the analysis model produces a result with satisfactory uncertainty

6. If the analysis model lacks sufficient fidelity, then swap the relevant constraints for ones that represent a higher fidelity model. If fidelity is sufficient, then proceed with the selected design.

The following Chapters will demonstrate this design process on two representative design cases.

Chapter 7

Application to the Hoburg Aircraft Design Problem

7.1 The Test Problem

One of the test cases used to benchmark LSQP and SLCP was a geometric program proposed by Hoburg [10] for aircraft conceptual design. As formulated, this problem contained a GP compatible fit to XFOIL data that was used to represent an analysis model for C_{D_p}:

$$\begin{aligned} 1 \geq & 2.56\frac{C_L^{5.88}}{\tau^{3.32}Re^{1.54}C_{D_p}^{2.26}} + 3.80\times10^{-9}\frac{\tau^{6.23}}{C_L^{0.92}Re^{1.38}C_{D_p}^{9.57}} + \\ & 2.20\times10^{-3}\frac{\tau^{0.03}Re^{0.14}}{C_L^{0.01}C_{D_p}^{0.73}} + 1.19\times10^{4}\frac{C_L^{9.78}\tau^{1.76}}{Re^{1.00}C_{D_p}^{0.91}} + \\ & 6.14\times10^{-6}\frac{C_L^{6.53}}{\tau^{0.52}Re^{0.99}C_{D_p}^{5.19}} \end{aligned} \tag{7.1}$$

In this chapter, this constraint will be swapped out for analysis models of progressively higher levels of fidelity. The flowchart that represents this example case is shown in Figure 7-1.

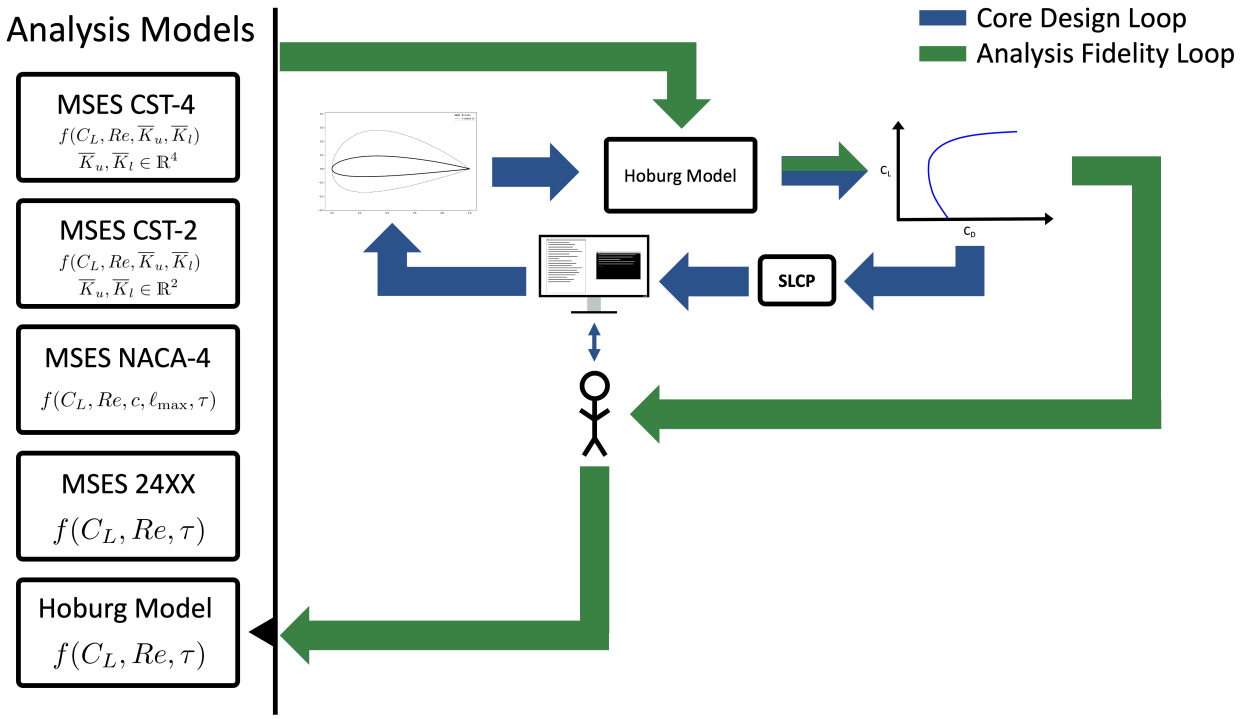


Figure 7-1: A sketch of the multifidelity design process for the Hoburg problem

7.2 Solving the Original Geometric Program

In the original geometric program, the model for drag coefficient assumes a NACA 24XX airfoil, operated at a Mach number $M = 0$. The only geometry variable that defines the airfoil geometry is thickness to chord ratio, τ. The wing structural model applies a strong upward pressure on τ, seeking to both improve structural efficiency and to drive the aspect ratio as high as possible to minimize induced drag, but the assumed structural cross section is invalid for thick airfoils, and so τ is constrained as $\tau \leq 0.15$. The drag model should also apply downward pressure on τ, as thick airfoils tend to have an increased drag, but this particular aerodynamic model does not sufficiently capture the aerodynamic penalty and therefore also benefits from the hard constraint on τ.

The most relevant results are reported in Table 7.1, with the full results available in Appendix D.

Table 7.1: Summary of the relevant optimal variables using the original GP formulation

Variable	Value	Units
W_{fuel}	5911.31	N
AR	20.39	-
$C_{D_i,\text{out}}$	0.005417	-
$C_{D_i,\text{ret}}$	0.005416	-
$C_{D_i,\text{sprint}}$	0.0002323	-
$C_{D_p,\text{out}}$	0.005455	-
$C_{D_p,\text{ret}}$	0.005470	-
$C_{D_p,\text{sprint}}$	0.005732	-
$C_{L,\text{out}}$	0.57411	-
$C_{L,\text{ret}}$	0.57408	-
$C_{L,\text{sprint}}$	0.11889	-
Re_{out}	4.66e6	-
Re_{ret}	4.47e6	-
Re_{sprint}	1.03e7	-
S	27.84	m^2
V_{out}	68.26	m/s
V_{ret}	65.35	m/s
V_{sprint}	150.00	m/s
W_{MTO}	36.94	kN
τ	0.150	-

The optimal airfoil design is a NACA 2415 airfoil, as the hard τ limit has been hit.

7.3 MSES: A Higher Fidelity Tool for Airfoil Aerodynamics

7.3.1 Description of the Software

The constraint in Equation 7.1 is a very low fidelity model for computing the C_{D_p} of an airfoil section, and progressing to the next phase of the design process will require that this model be upgraded. The sources of uncertainty in a fitted model like Equation 7.1 are outlined in Figure 7-2.

The two sources of uncertainty that come from the actual fitting process (labeled 1 and 2 in Figure 7-2) could be removed by replacing Equation 7.1 with a function that actually calls XFOIL, which was the source of the underlying data. However XFOIL does not return

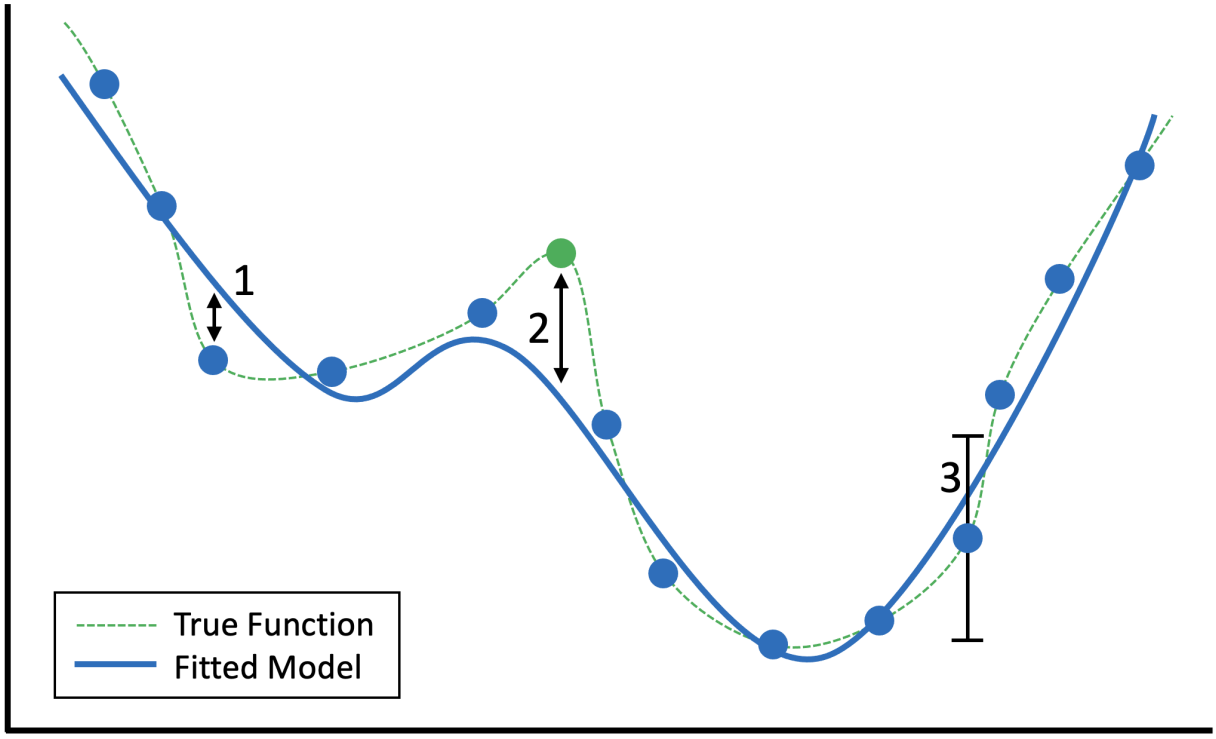


Figure 7-2: The sources of uncertainty in a fitted surrogate: 1) Discrepancies between the fitted surrogate and the data used in fitting caused by insufficient model order 2) Features in the true function that are of higher frequency than can be captured by the sampling interval 3) The uncertainty of each data point caused by lack of physical fidelity

gradients with respect to either Reynolds number or thickness ratio τ, and therefore is not something that can be easily integrated with any gradient based optimization scheme such as SQP, LSQP, and SLCP. But MSES, an analysis tool with an improved level of physics fidelity, does contain the desired derivatives.

MSES [68] is a 2D airfoil analysis tool that is similar to the more commonly known XFOIL [46], but that is significantly more powerful. Airfoil geometry is defined using a coordinate file, and flow parameters are set with a separate input file. While XFOIL uses an integral boundary layer (IBL) corrected potential flow model, MSES uses an IBL corrected 2D Euler model that allows for more accurate modeling of the flow physics, particularly at transonic mach numbers. This enhanced fidelity makes MSES incredibly powerful, but also slightly more fragile and noticeably more computationally expensive than XFOIL. MSES also re-

quires a great deal more effort and expertise to run on the part of the user and has limited history being run in any kind of automated aircraft design architecture.

7.3.2 The Supporting Architecture in CAPS

As shipped, MSES does not support being run in a largely automated design and optimization framework, and so it was integrated into the CAPS tool for multi-fidelity aircraft design [69]. An Analysis Interface Module (AIM) was constructed that handles the various different software routines that must be pieced together in a single MSES run, the details of which are not relevant for this work.

However, obtaining derivatives of the computed flow parameters (C_L, C_D, C_M, etc.) with respect to the airfoil geometry variables did require some effort. By setting up the input file correctly, MSES returns derivatives of the computed flow parameters with respect to perturbations off of the baseline airfoil using a form of Chebyshev Polynomials, but other geometry parameterizations are not supported. The CAPS AIM therefore was developed to convert the geometry parameterizations available in CAPS to the Chebyshev modes used by MSES, and to close the chain rules on the derivatives appropriately.

7.3.3 Challenges of Using MSES

Using MSES in an automated fashion posed two significant challenges during this work. First, when the MSES run is automated, the solver is quite fragile and often fails to converge, even when the flow conditions and geometry seem reasonable. When run by a human in the loop, MSES failures can be handled in many intelligent ways, like exploring the local region for cases that can be converged, warm starting from a known converged case, and adjusting grid parameters to better capture the flow physics. No systematic repeatable method for this human in the loop debugging could be developed, and so cases that fail do not have any kind of correction mechanism. Anecdotally, somewhere between one in three and one in ten cases seemed to fail, with no real discernible pattern. However, some regions of the design

space clearly have more difficulty converging than others, though again no pattern to this could be ascertained.

When a black box analysis tool is being used as a constraint in SQP, LSQP, or SLCP, and that analysis tool fails to converge, it is assumed that the step size suggested by the sub-problem is so large that the candidate design has been pushed into an infeasible region. Thus, the step size is reduced in the inexact line search until the case does converge (see Appendix B). At times in this work, repeated MSES failures led to multiple shrinkings of the step size until the actual step taken was so small that it significantly hindered the algorithm's ability to make progress. And the problem becomes worse if multiple MSES cases are run at a given iteration (ie, if the wing has multiple sections that are each being analyzed by MSES), as all of those cases must converge for a successful step to be taken.

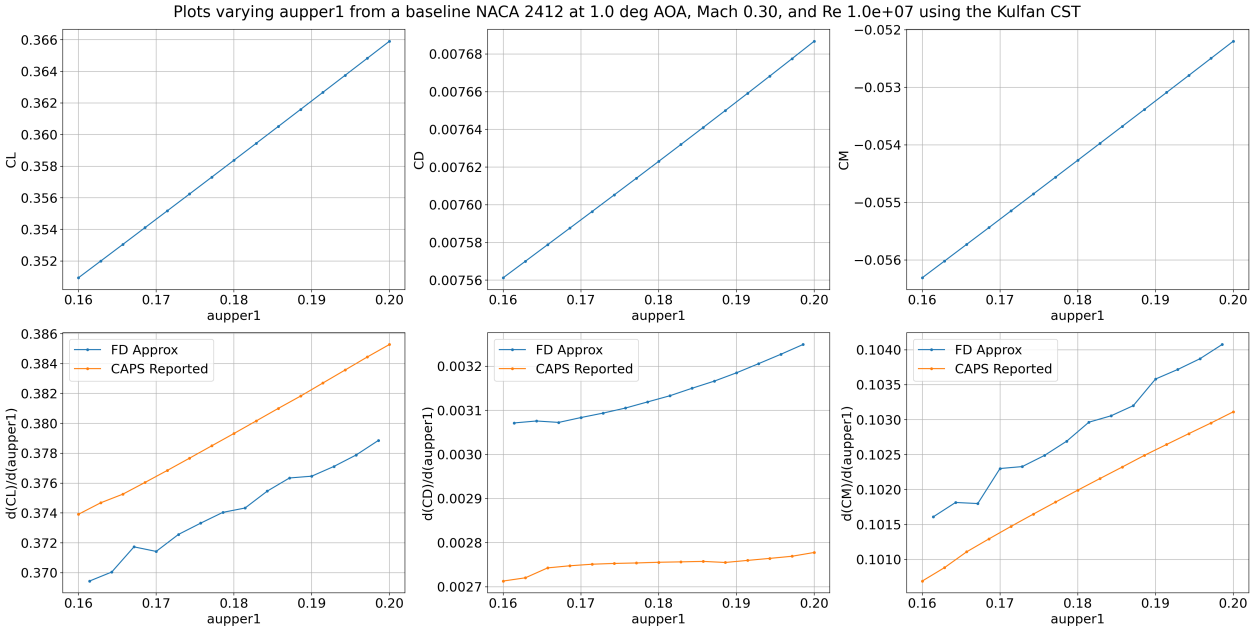
The relatively high failure rate of MSES when it is being used in this context means that 1) the number of cases considered at each iteration must be kept to a minimum in the hope that the step size will not be reduced excessively and 2) even under the best conditions, sometimes the step size at a given iteration shrinks to a very small value and limits the ability of the algorithm to quickly converge to an optimal solution. Though frustrating, no obvious or simple solution exists, but future work could look at providing additional runtime logic to MSES so that it can handle failures internally as if a human were running the cases. Alternatively, human input could be solicited at each optimization step [68], though this limits the optimization cycle time to the bandwidth of the human in the loop.

The second challenge is that MSES returns derivatives that are noisy, particularly when laminar to turbulent transition is being modeled. Figures 7-3 and 7-4 show sweeps of one of the Kulfan CST-4 coefficients, along with the reported values from the function, the finite difference approximation of these reported values, and the reported derivatives from MSES/CAPS.

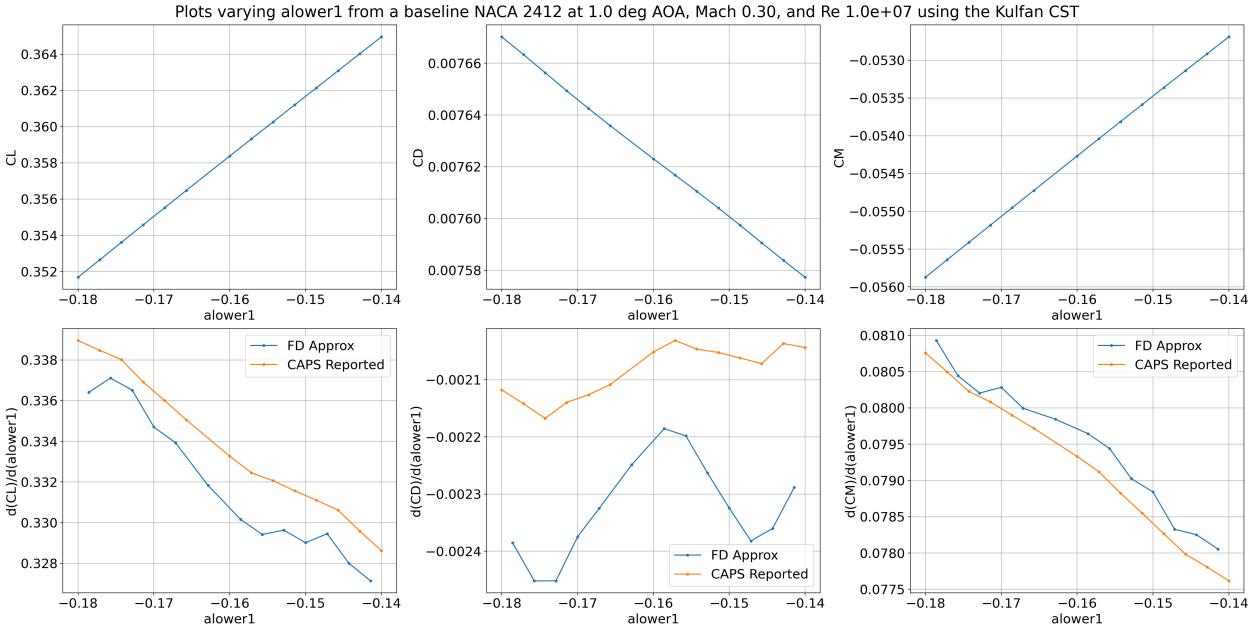
In Figure 7-3, transition is being forced at 1% chord on both upper and lower surfaces,

and the derivatives from MSES are in reasonably good agreement with the finite difference approximations. There is a small offset in most of the plots, but the vertical axis scale reveals this to be small. However, when free transition is allowed (Figure 7-4), the derivatives from MSES are noticeably different from the finite difference approximations. The best working theory here is that the discrepancy may be due to transition moving across discrete cells in the 2D mesh, though this behavior is poorly understood.

It is not uncommon for derivatives returned from engineering analysis software to have noise, particularly when residual equations are being solved. Unfortunately, the inability to trust derivatives has significant consequences. First order termination conditions that rely on accurate derivative information must be thrown out and replaced by relative convergence metrics that lack mathematical rigor. In addition, a step schedule must be introduced that slowly decreases the maximum step size with increasing iteration count to prevent cycling and random walking near the optimal solution. Put simply, the pure mathematical principles that have been developed to this point become much far murkier in the presence of real engineering problems, but best practices from SQP can be carried forward the the problems considered here.

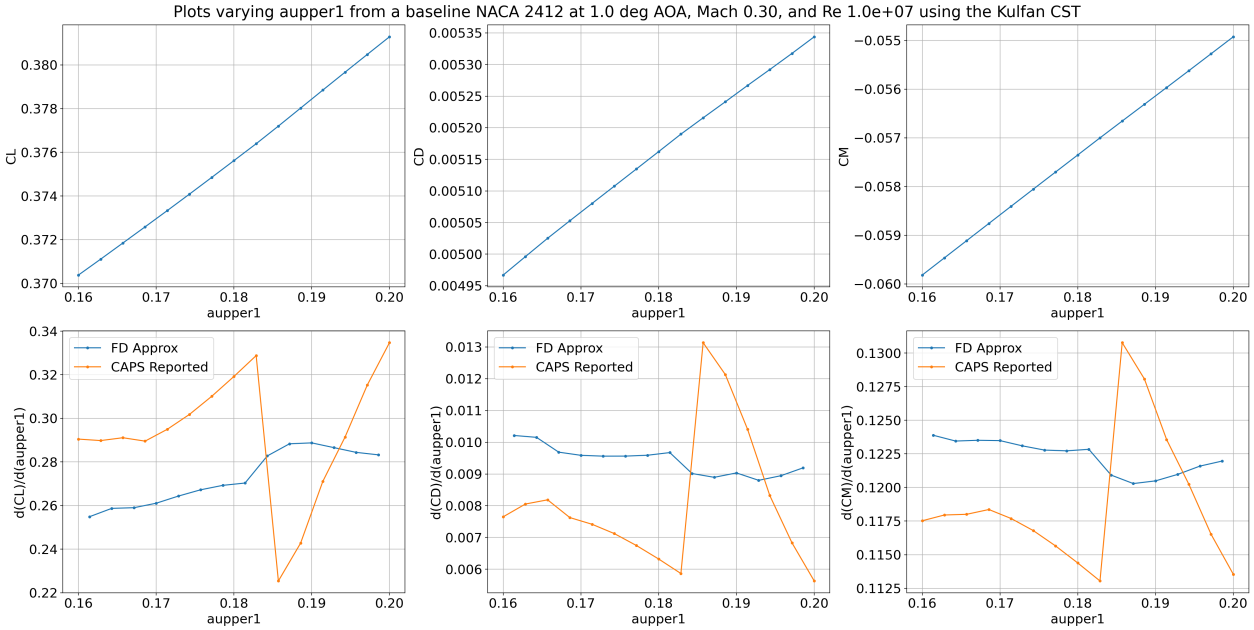


(a) Upper Surface

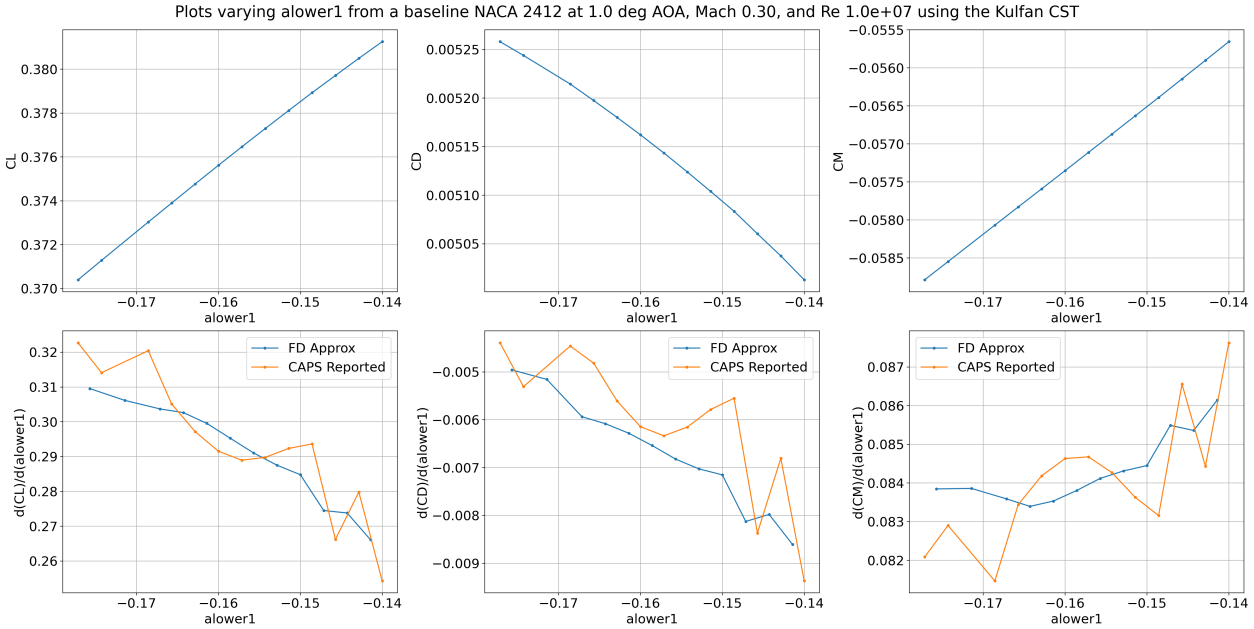


(b) Lower Surface

Figure 7-3: Comparing MSES derivatives with a forced transition at 1% chord, leading edge mode of a Kulfan CST-4 with a NACA 2412 baseline airfoil



(a) Upper Surface



(b) Lower Surface

Figure 7-4: Comparing MSES derivatives with free transition, leading edge mode of a Kulfan CST-4 with a NACA 2412 baseline airfoil

7.4 Increasing Model Fidelity with MSES

Integrating MSES into the design optimization problem both removes the uncertainty that comes from data fitting, and increases the level of fidelity of the underlying physics. As has been discussed, MSES will simply be swapped in for the function:

$$C_{D_p} \geq f(C_L, Re, \tau) \tag{7.2}$$

However, MSES is only capable of returning derivatives when run for a fixed angle of attack, and not a fixed C_L, and so the problem formulation must be tweaked slightly. Rather than a direct swap, the variable α is introduced along with $C_{L_{\text{MSES}}}$, and the following constraints are imposed:

$$\begin{aligned} C_{D_p} &\geq f(\alpha, Re, \tau) \\ C_{L_{\text{MSES}}} &= f(\alpha, Re, \tau) \\ C_L &= C_{L_{\text{MSES}}} \end{aligned} \tag{7.3}$$

The fact that the required analysis inputs do not match the variables used in the lower fidelity GP is not surprising, but does introduce new challenges. Variable α may well be zero or negative under certain conditions, so an additive shift must be implemented to ensure it remains positive in all reasonable cases. Also, the constraint $C_{D_p} \geq f(C_L, Re, \tau)$ is known to be very nearly convex under log-log transformation, but the constraint set in Equation 7.3 is unlikely to be due to the need for an equality constraint in the $C_{L_{\text{MSES}}} = f(\alpha, Re, \tau)$ constraint.

Unlike the geometric program, the SLCP optimization algorithm now requires an initial guess due to the loss of convex structure. Fortunately, the result from the GP provides an excellent initial guess to this new problem, though variables α and $C_{L_{\text{MSES}}}$ will need initial values that are easy to obtain. This ability to propagate guesses forward as higher fidelity models are added is a key advantage to this design approach, as ensures a high quality initial

guess is always available to the optimizer.

Although MSES has been brought in to the problem, the fundamental geometry parameterization of the airfoil has not been changed, and there remains a constraint bounding $\tau \leq 0.15$, and so it is expected that the same airfoil geometry will return as optimal. Table 7.2 reports relevant results, with more details available in Appendix D.

Table 7.2: Summary of the relevant optimal variables using the original formulation with MSES

Variable	Value	Units
W_{fuel}	5968.04	N
AR	21.26	-
$C_{D_i,\text{out}}$	0.005230	-
$C_{D_i,\text{ret}}$	0.004945	-
$C_{D_i,\text{sprint}}$	0.002214	-
$C_{D_p,\text{out}}$	0.006077	-
$C_{D_p,\text{ret}}$	0.006054	-
$C_{D_p,\text{sprint}}$	0.005638	-
$C_{L,\text{out}}$	0.57606	-
$C_{L,\text{ret}}$	0.56013	-
$C_{L,\text{sprint}}$	0.11852	-
Re_{out}	4.50e6	-
Re_{ret}	4.36e6	-
Re_{sprint}	9.93e6	-
S	27.22	m^2
V_{out}	68.04	m/s
V_{ret}	65.95	m/s
V_{sprint}	150.00	m/s
W_{MTO}	36.11	kN
τ	0.172	-

Note that the fuel burn has actually *increased* slightly due to the fact that the previous constraint was not modeling the modest effect on drag of cruising at $M \approx 0.2$. In essence, the uncertainty of the GP result included this higher objective value, but running with MSES provides a worse performing result of higher certainty. Figure 7-5 shows the airfoil performance in MSES as the aircraft flies its outbound cruise leg, and Figure 7-6 compares the drag polars of the first two phases.

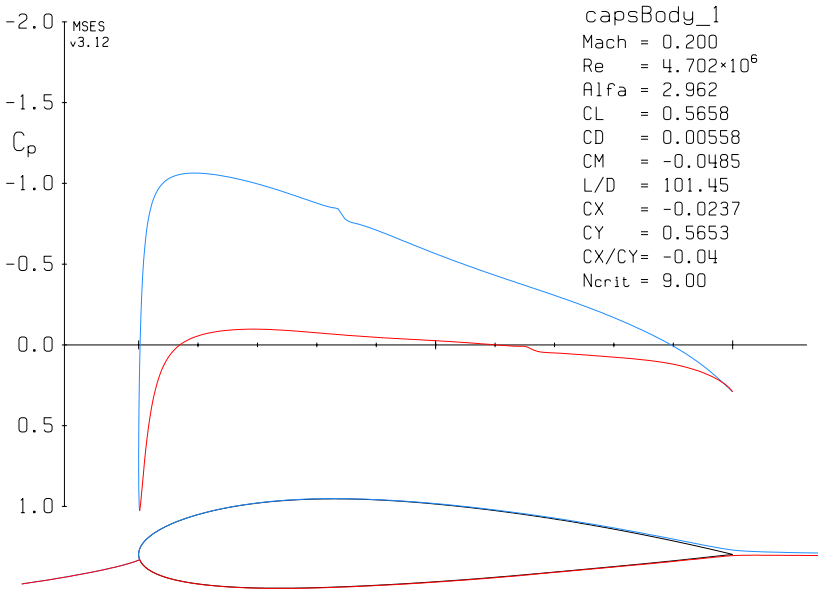


Figure 7-5: The result from MSES during the outbound cruise leg for Design Phase 2

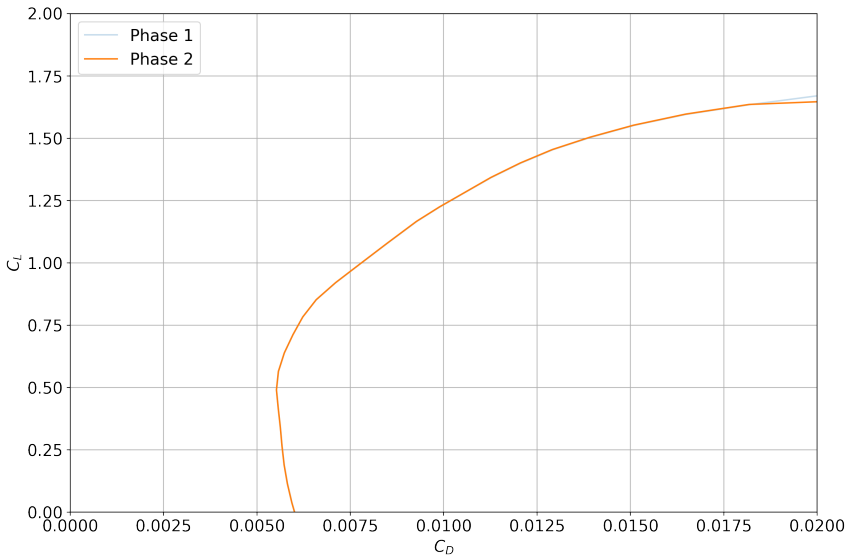


Figure 7-6: Drag polars for the first two design phases

7.5 Adding New Geometry Variables

MSES is a highly trusted tool for 2D airfoil analysis, and given the level of fidelity of other analysis models used the design problem, further increasing the analysis fidelity would be expected to produce diminishing returns. A 2D CFD tool like SU2 could be of some value, but this tool could be integrated using exactly the same method as has been used for MSES. Once again, the mathematics occurring inside the black boxed functions of Equation 7.3 are of no consequence to the optimization first approach being discussed here, nor to the algorithms being used to solve the optimization problem. The only consequence of such a change would be an increase in solve time, and to provide some unquantifiable reduction in perceived uncertainty. Thus, remaining cases will consider increasing the level of geometry fidelity.

Geometry fidelity can be increased now because while all airfoils of thickness $\tau = 0.12$ have the same performance in Equation 7.1, MSES is sensitive to the entire geometry of the 2D airfoil. The airfoil can then be parameterized according to some new set of design variables, and these new variables can be exposed to the optimization problem, allowing the optimizer to drive these variables towards a higher performing design.

Keeping things simple, first consider using the full NACA-4 series definition of airfoil geometry, which introduces two new variables to the optimization problem: maximum camber per unit chord $c_{\max}$, and the location of the maximum camber as a fraction of chord length $d_{c\max}$.

These new design variables change the relevant optimization constraints as follows:

$$\begin{aligned} C_{D_p} &\geq f(\alpha, Re, c_{\max}, d_{c\max}, \tau) \\ C_{L_{\text{MSES}}} &= f(\alpha, Re, c_{\max}, d_{c\max}, \tau) \\ C_L &= C_{L_{\text{MSES}}} \end{aligned} \tag{7.4}$$

Some additional constraints are imposed to ensure the solution is sensible, such as limitations on angle of attack and bounds on the location of the max camber, but these are not expected to be active at the optimal solution and are intended to assist in the sub-problem construction during the optimization run. But solving this problem yields a highly unusual result, as seen in Figure 7-7. The airfoil is so highly cambered at the trailing edge, that the pressure differential between upper and lower surfaces is likely to overload any real structure that can be placed on the aft section of the airfoil. Furthermore, the airfoil polar in Figure 7-8 shows poor performance outside of a single sharp cusp, and an increase in turbulence in the incoming flow would likely cause such a design to be infeasible from an aerodynamic perspective as well. So although Table 7.3 reveals significant fuel savings, this design is not achievable in practice.

Table 7.3: Summary of the relevant optimal variables using the full NACA-4 geometry with MSES

Variable	Value	Units
W_{fuel}	5051.15	N
AR	22.92	-
$C_{D_i,\text{out}}$	0.005257	-
$C_{D_i,\text{ret}}$	0.005299	-
$C_{D_i,\text{sprint}}$	0.002120	-
$C_{D_p,\text{out}}$	0.004301	-
$C_{D_p,\text{ret}}$	0.004310	-
$C_{D_p,\text{sprint}}$	0.004829	-
$C_{L,\text{out}}$	0.59972	-
$C_{L,\text{ret}}$	0.60211	-
$C_{L,\text{sprint}}$	0.12044	-
Re_{out}	4.31e6	-
Re_{ret}	4.15e6	-
Re_{sprint}	9.63e6	-
S	27.61	m^2
V_{out}	67.22	m/s
V_{ret}	64.64	m/s
V_{sprint}	150.00	m/s
W_{MTO}	36.62	kN
$c_{\max}$	0.0458	-
$d_{c\max}$	0.7656	-
τ	0.161	-

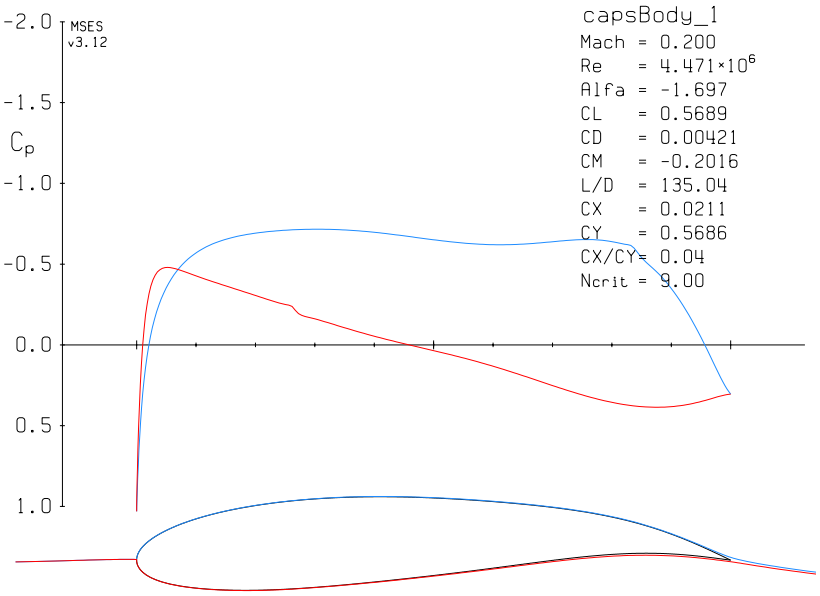


Figure 7-7: The result from MSES during the outbound cruise leg for Design Phase 3

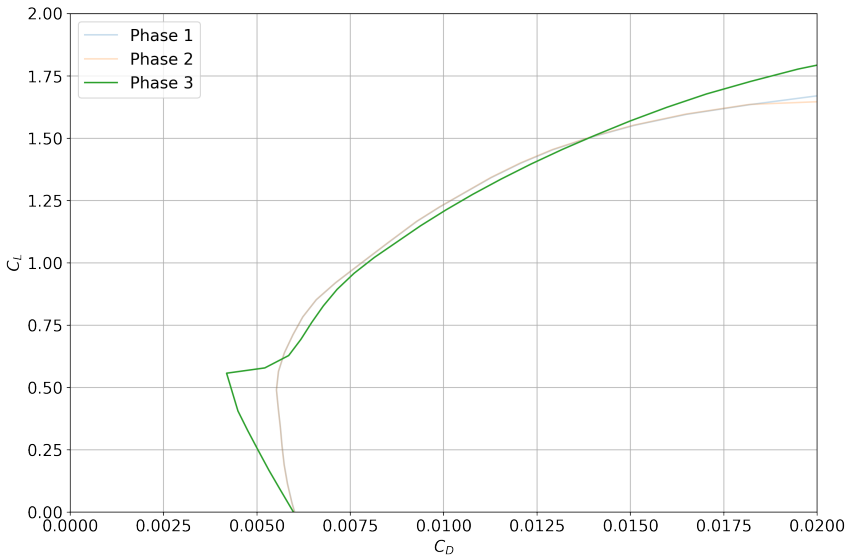


Figure 7-8: Drag polars for the first three design phases

In this case, the optimizer has exploited a poor design formulation. In an effort to reduce drag, the top surface of the airfoil has been designed to delay transition to 85% chord, and then the lower surface has been designed to produce the required lift coefficient. None of the analysis models, enforced as constraints, have accounted for the structural consequences of this design, and so the optimizer is happy to find this optimum.

As is often the case in design, this design phase reveals previously unknown information that now must be included in the design formulation. Fortunately, re-design in this design approach is simply a matter of modifying a few constraints and re-running the optimizer.

7.6 Imposing a Constraint on Moment Coefficient

The previous design phase made it clear that a structural constraint needs to be added to limit the percentage of lift generation that can occur on the aft region of the airfoil. Rather than write a detailed model analyzing the stress in the airfoil section, consider imposing a bound on the moment coefficient of the airfoil for all three different flight segments. The relevant constraints become:

$$\begin{aligned} C_{D_p} &\geq f(\alpha, Re, c_{\max}, d_{c\max}, \tau) \\ C_{L_{\text{MSES}}} &= f(\alpha, Re, c_{\max}, d_{c\max}, \tau) \\ C_M &= f(\alpha, Re, c_{\max}, d_{c\max}, \tau) \\ C_L &= C_{L_{\text{MSES}}} \\ C_M &\geq -0.06 \end{aligned} \tag{7.5}$$

Since moment coefficient cannot be driven to a large negative magnitude, the optimizer will be forced to generate lift on the front half of the airfoil, sacrificing the delay of transition on the top surface. Indeed, Figure 7-9 reveals this to be the case, with an airfoil that is very similar to the original NACA 2415. Maximum camber has been slightly decreased and shifted forward, resulting in an improved drag polar in the region of the two cruise legs

(Figure 7-10). Table 7.4 confirms that this case results in less fuel burned over the course of the full mission when compared to the baseline NACA 2415 airfoil. Thus, by allowing the optimizer more control over the airfoil geometry, it is able to find a more efficient aircraft design.

Note that since C_M is by convention defined with negative values being nose down, an additive shift was implemented to ensure the value of C_M could be handled by the optimizer as a positive number.

Table 7.4: Summary of the relevant optimal variables using the full NACA-4 geometry and a constraint on moment coefficient

Variable	Value	Units
W_{fuel}	5901.98	N
AR	22.44	-
$C_{D_i,\text{out}}$	0.006087	-
$C_{D_i,\text{ret}}$	0.005747	-
$C_{D_i,\text{sprint}}$	0.002104	-
$C_{D_p,\text{out}}$	0.006302	-
$C_{D_p,\text{ret}}$	0.006290	-
$C_{D_p,\text{sprint}}$	0.005962	-
$C_{L,\text{out}}$	0.63843	-
$C_{L,\text{ret}}$	0.62034	-
$C_{L,\text{sprint}}$	0.11870	-
Re_{out}	4.18e6	-
Re_{ret}	4.05e6	-
Re_{sprint}	9.68e6	-
S	27.35	m^2
V_{out}	64.68	m/s
V_{ret}	62.77	m/s
V_{sprint}	150.00	m/s
W_{MTO}	36.28	kN
$c_{\max}$	0.0241	-
$d_{c_{\max}}$	0.3678	-
τ	0.185	-

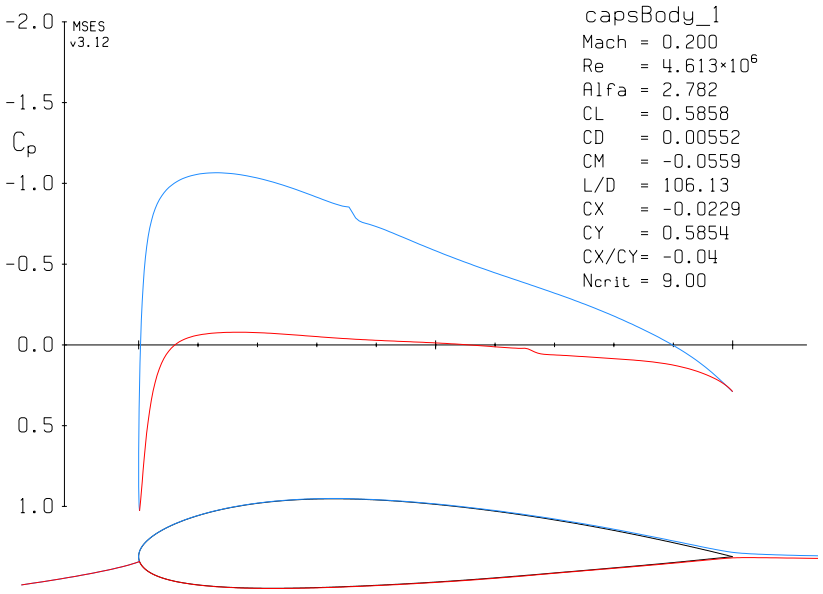


Figure 7-9: The result from MSES during the outbound cruise leg for Design Phase 4

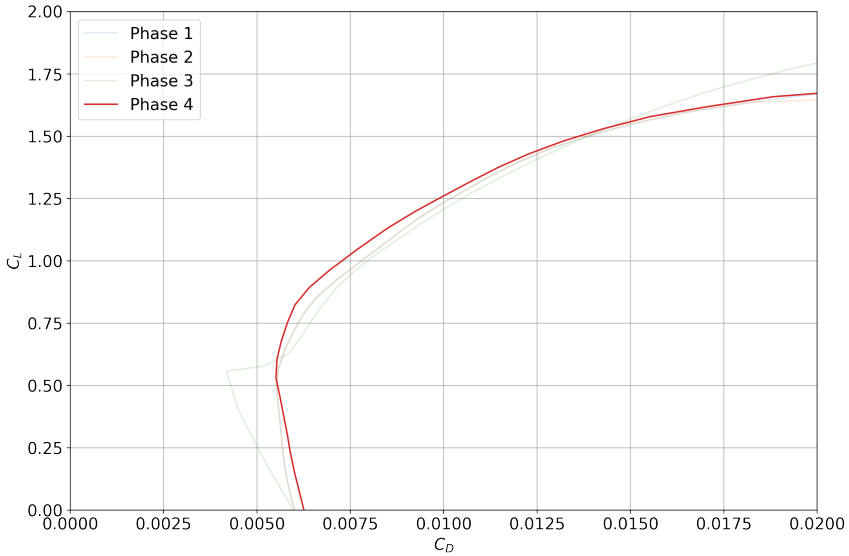


Figure 7-10: Drag polars for the first four design phases

7.7 Using a Kulfan CST-2 Representation of the Airfoil Geometry

The NACA-4 series of airfoils are easy to work with and have an intuitive geometry definition, but are not sufficient for the needs of modern aircraft design. The Kulfan representation is widely utilized for higher fidelity geometry representations [70], and is highly generalizable. Using 16 variables (8 per upper and lower surface) allows the parameterization to capture very near the entire design space of reasonable airfoil geometries [70].

To begin, consider only 2 Kulfan coefficients on both the upper and lower surfaces, which at a total of 4 design variables is only one more than the NACA-4 geometry parameterization. Given the introduction of the Kulfan coefficients to the new design formulation, previous variables $c_{\max}$, $d_{c_{\max}}$, and τ are no longer necessary. But the structural model still relies on τ for sizing the wing structure, and so one additional constraint must be included. Packing the relevant Kulfan coefficients into vector $\mathbf{A}$, the new constraint set is as follows:

$$\begin{aligned} C_{D_p} &\geq f(\alpha, Re, \mathbf{A}) \\ C_{L_{\text{MSES}}} &= f(\alpha, Re, \mathbf{A}) \\ C_M &= f(\alpha, Re, \mathbf{A}) \\ C_L &= C_{L_{\text{MSES}}} \\ \tau &= f(\mathbf{A}) \end{aligned} \tag{7.6}$$

The moment coefficient constraint has been removed because this parameterization does not require it.

The optimal airfoil design here is fascinating, shown in Figure 7-11. Both upper and lower surfaces are shaped to drive transition as far aft as possible, with the camber of the upper surface being used to generate the required lift. Though the polars confirm a highly efficient airfoil (Figure 7-12), this is clearly not a practical design. The tight leading edge radius is

highly susceptible to stalling at off-design angles of attack, and maintaining laminar flow for 85% of the chord is not achievable in reality.

Furthermore, examining Figure 7-13 reveals that an increase in turbulence in the incoming flow significantly narrows the operating window for this airfoil, in contrast to more realistic designs that simply produce higher drag.

Table 7.5: Summary of the relevant optimal variables using the Kulfan CST-2 representation for airfoil geometry

Variable	Value	Units
W_{fuel}	4816.57	N
AR	19.42	-
$C_{D_i,\text{out}}$	0.002303	-
$C_{D_i,\text{ret}}$	0.002262	-
$C_{D_i,\text{sprint}}$	0.000232	-
$C_{D_p,\text{out}}$	0.003298	-
$C_{D_p,\text{ret}}$	0.003278	-
$C_{D_p,\text{sprint}}$	0.003708	-
$C_{L,\text{out}}$	0.36544	-
$C_{L,\text{ret}}$	0.36211	-
$C_{L,\text{sprint}}$	0.11606	-
Re_{out}	5.71e6	-
Re_{ret}	5.52e6	-
Re_{sprint}	1.01e7	-
S	25.91	m^2
V_{out}	84.53	m/s
V_{ret}	81.66	m/s
V_{sprint}	150.00	m/s
W_{MTO}	34.00	kN
τ	0.164	-
A_{u_1}	0.146	-
A_{u_2}	0.416	-
A_{l_1}	-0.111	-
A_{l_2}	-0.259	-

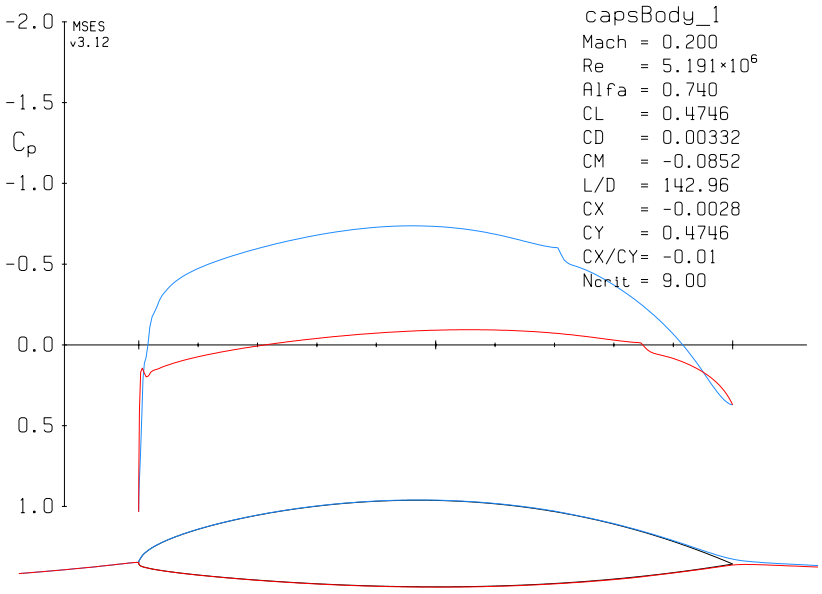


Figure 7-11: The result from MSES during the outbound cruise leg for Design Phase 5

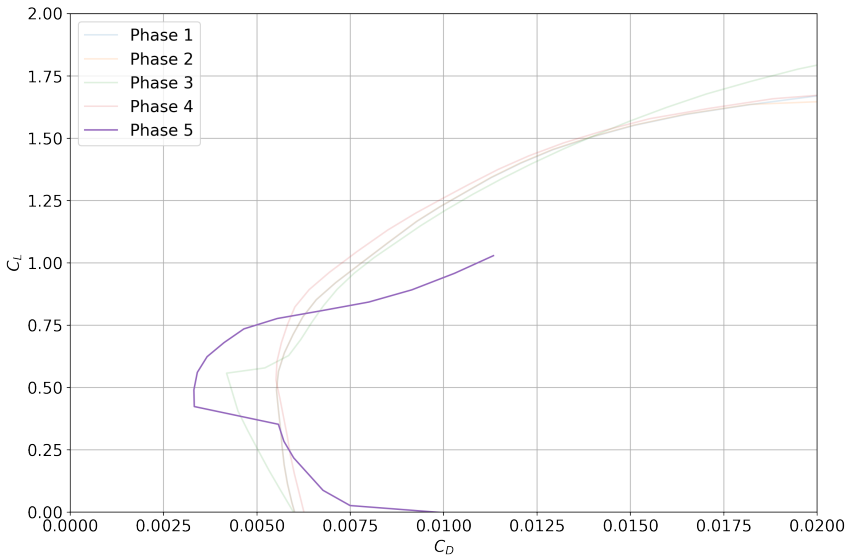


Figure 7-12: Drag polars for the first five design phases

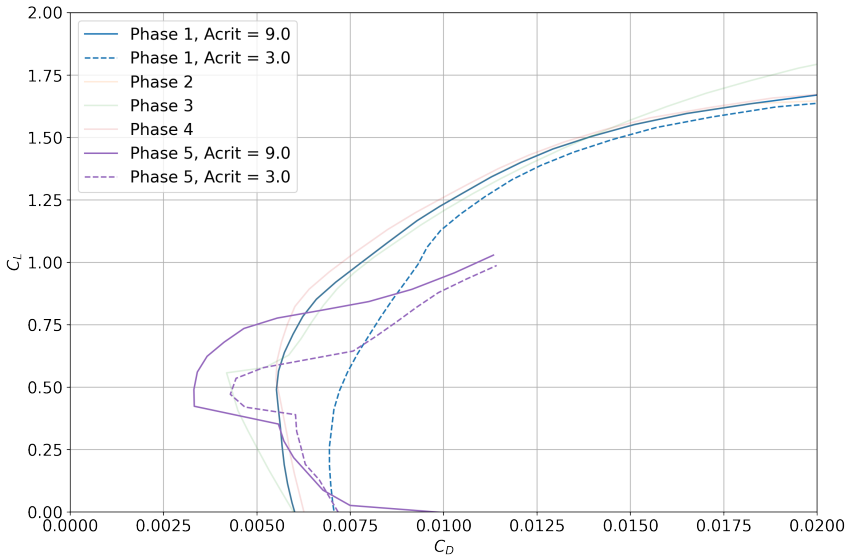


Figure 7-13: Drag polars for the first five design phases comparing decreased values of N_{crit}

Despite these concerns, this design is interesting in that the impracticality here is not driven so much by physics (ie, an inability to construct realizable structure), but by operational limits that are difficult to quantify. It is possible to develop an aircraft using this airfoil that implements a control limiter to keep the aircraft operating in the narrow drag polar at all times, and regular wing cleanings and vibration isolation may enable a substantially laminar airfoil. Unlike the previous failure, which was the result of insufficient physics fidelity, this design is limited only by human will. Or if you prefer a more concrete reason, by financial cost of operation. So once again, although Table 7.5 shows substantial fuel savings, a re-design cycle must take place.

7.8 Tripping the flow with Kulfan CST-2

In order to address issues with laminar flow, an MSES setting was adjusted to force transition to occur before or at 40% of the chord length along the upper surface of the airfoil, and before 65% on the lower surface. No changes are necessary to the optimization formulation since this is an internal MSES setting, but conceptually this could be thought of as adding one additional constraint on transition location.

The result in Figure 7-14 is an unusual airfoil with a tight leading edge radius. This feature will likely hurt off-design performance, but on the whole the airfoil is reasonable. The increase in the number of variables defining the airfoil geometry does indeed allow for more refinement, with the overall fuel weight decreasing from the last case (Table 7.6).

Table 7.6: Summary of the relevant optimal variables using the Kulfan CST-2 representation for airfoil geometry and tripping the flow

Variable	Value	Units
W_{fuel}	5774.12	N
AR	22.46	-
$C_{D_i,\text{out}}$	0.005803	-
$C_{D_i,\text{ret}}$	0.005660	-
$C_{D_i,\text{sprint}}$	0.002110	-
$C_{D_p,\text{out}}$	0.005999	-
$C_{D_p,\text{ret}}$	0.005977	-
$C_{D_p,\text{sprint}}$	0.006717	-
$C_{L,\text{out}}$	0.62371	-
$C_{L,\text{ret}}$	0.61601	-
$C_{L,\text{sprint}}$	0.11894	-
Re_{out}	4.26e6	-
Re_{ret}	4.07e6	-
Re_{sprint}	9.68e6	-
S	27.34	m^2
V_{out}	65.50	m/s
V_{ret}	63.12	m/s
V_{sprint}	150.00	m/s
W_{MTO}	36.27	kN
τ	0.190	-
A_{u_1}	0.346	-
A_{u_2}	0.325	-
A_{l_1}	-0.144	-
A_{l_2}	-0.173	-

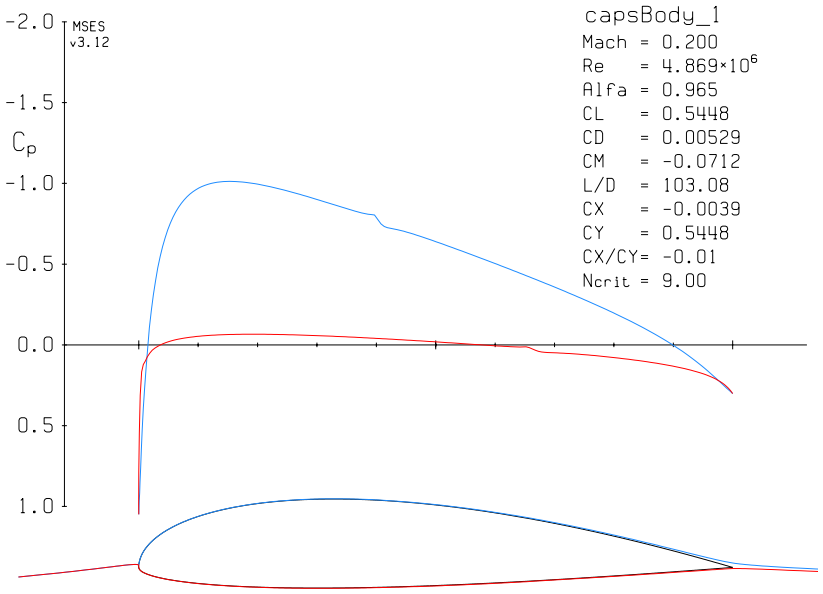


Figure 7-14: The result from MSES during the outbound cruise leg for Design Phase 6

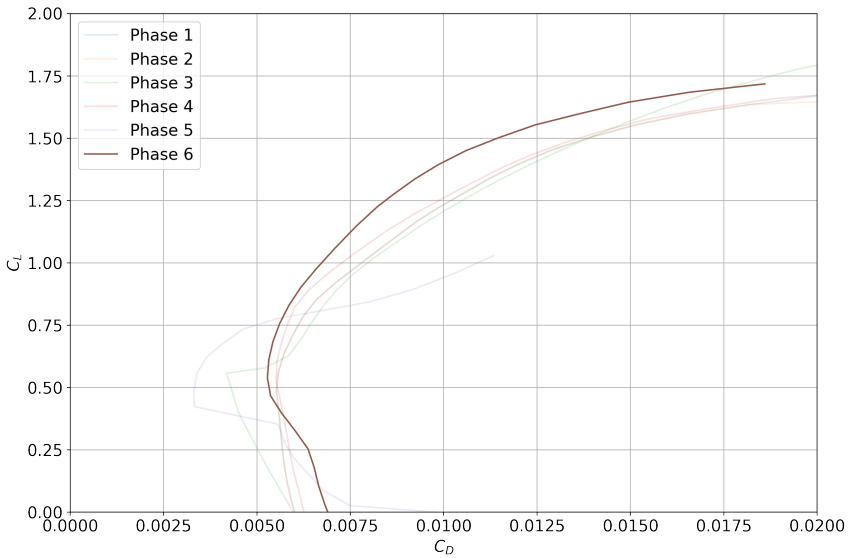


Figure 7-15: Drag polars for the first six design phases

A deep dive into the data in Appendix D reveals why this airfoil geometry seems substantially different from previous designs. Cruise efficiency is very high, which is seen in Figure 7-14 since the airfoil has less drag than previous airfoils at the cruise lift coefficient. However, the airfoil performs far worse than previous designs at the low lift coefficient used in the sprint constraint that sizes the powerplant, forcing this design to carry a much heavier engine than previous cases.

7.9 Increasing Geometry Fidelity to Kulfan CST-4

Introducing more design variables that define the airfoil geometry should allow the optimizer to select an airfoil with reasonable performance in both cruise and sprint conditions. To that end, consider the use of a Kulfan CST-4 parameterization for a total of 8 variables that define the airfoil. As before, flow on the upper surface is tripped at 40% chord and at 65% chord on the lower surface. The optimization formulation once again does not change, though the length of vector $\mathbf{A}$ is now 8 instead of 4.

Figure 7-16 shows an airfoil notably different from previous designs, but appears to be a reasonable geometry that could be physically realizable. The drag polar (Figure 7-17) shows good performance across a large range of lift coefficients, before finally dropping off around a $C_L \approx 0.9$.

Table 7.7: Summary of the relevant optimal variables using the Kulfan CST-4 representation for airfoil geometry

Variable	Value	Units
W_{fuel}	5366.07	N
AR	23.36	-
$C_{D_i,\text{out}}$	0.004877	-
$C_{D_i,\text{ret}}$	0.004529	-
$C_{D_i,\text{sprint}}$	0.000205	-
$C_{D_p,\text{out}}$	0.005482	-
$C_{D_p,\text{ret}}$	0.005473	-
$C_{D_p,\text{sprint}}$	0.005998	-
$C_{L,\text{out}}$	0.58295	-
$C_{L,\text{ret}}$	0.56172	-
$C_{L,\text{sprint}}$	0.11945	-
Re_{out}	4.24e6	-
Re_{ret}	4.14e6	-
Re_{sprint}	9.35e6	-
S	26.55	m^2
V_{out}	67.94	m/s
V_{ret}	66.43	m/s
V_{sprint}	150.00	m/s
W_{MTO}	35.23	kN
τ	0.205	-
A_{u_1}	0.369	-
A_{u_2}	0.543	-
A_{u_3}	0.051	-
A_{u_4}	0.158	-
A_{l_1}	-0.128	-
A_{l_2}	-0.202	-
A_{l_3}	-0.068	-
A_{l_4}	-0.257	-

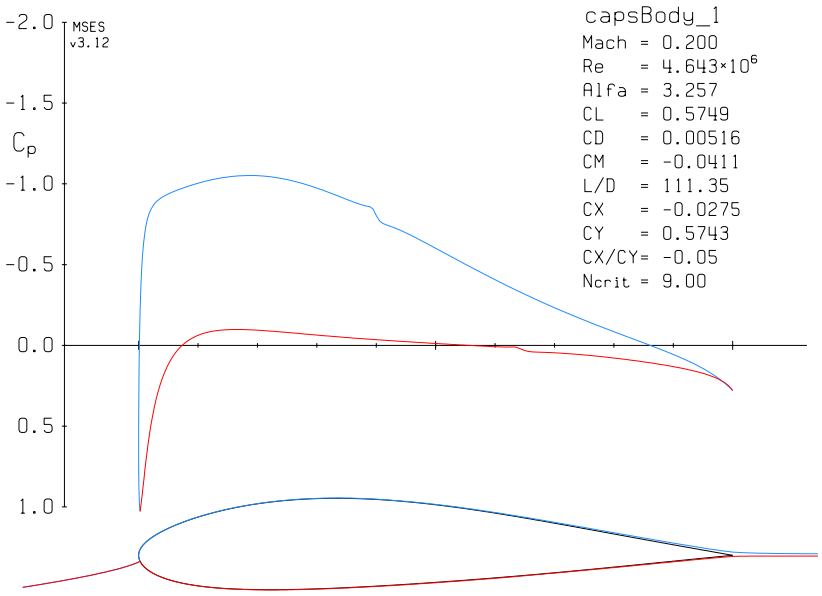


Figure 7-16: The result from MSES during the outbound cruise leg for Design Phase 7

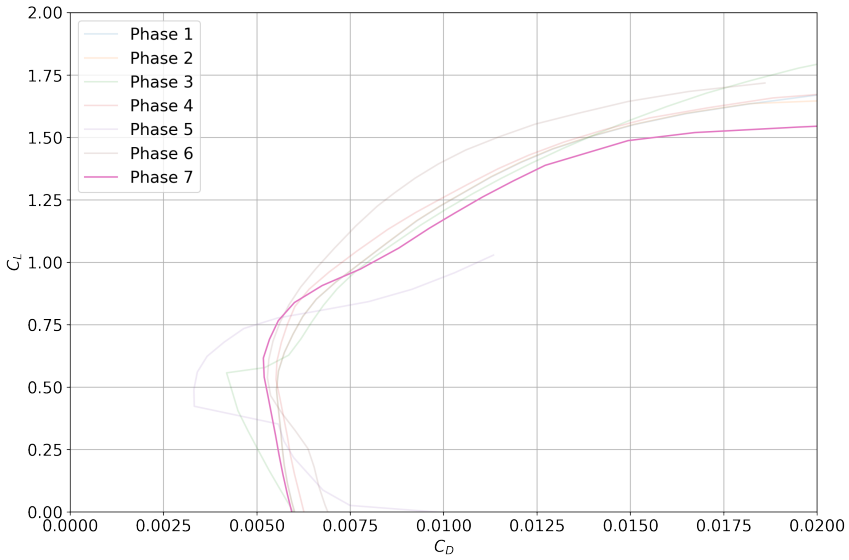


Figure 7-17: Drag polars for the all 7 design phases

7.10 Higher Order Geometry Refinement

Despite the gains realized by the CST-4 representation, there is a fundamental limit to the performance benefit that can be extracted from higher fidelity geometry prarameterizations. Figure 7-18 plots the objective function as a function of increasing Kulfan order, and it is apparent that the optimizer is unable to refine the design much further beyond 12 variables on each surface[1]. Adding more geometry variables at this point simply increased the run time for the optimizer.

[1]Note that for practical reasons, this design problem was slightly modified from the original problem with regards to the transition locations and the specific constraints on the various Kulfan coefficients

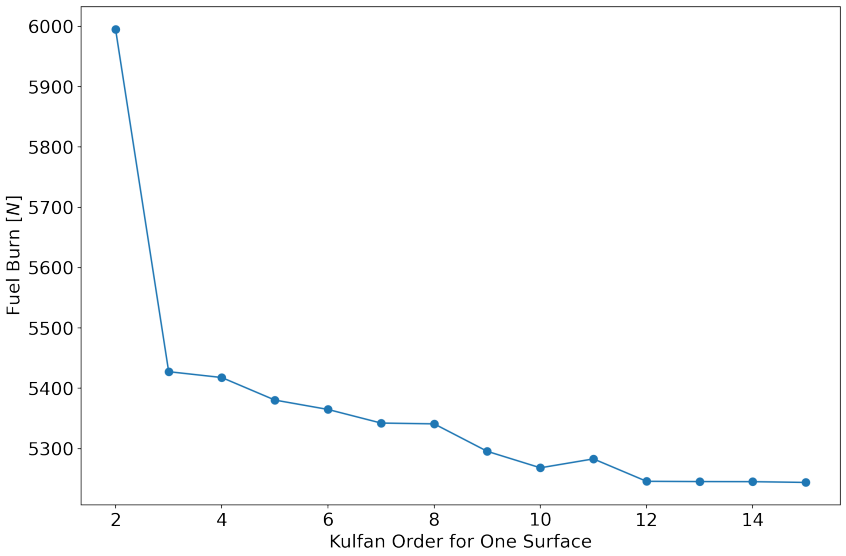


Figure 7-18: The objective function as the order of the Kulfan parameterization is increased. The inferior performance with 11 variables is due to the noise in the MSES gradients preventing smooth optimization termination

Thus, using MSES with a CST-12 parameterization is the logical end point for this design effort. Further steps would involve either an increase in the fidelity of the analysis physics, which is of trivial value given the level of fidelity in the analysis models that constitute the rest of the problem, or a replacement of the entire aerodynamic model, which will be discussed in the next chapter.

7.11 Post Design Discussion

At the completion of this multifidelity design process, a total fuel savings of 8.3% has been achieved from refining the airfoil geometry. The history of the fuel burn in each phase shown in Figure 7-19.

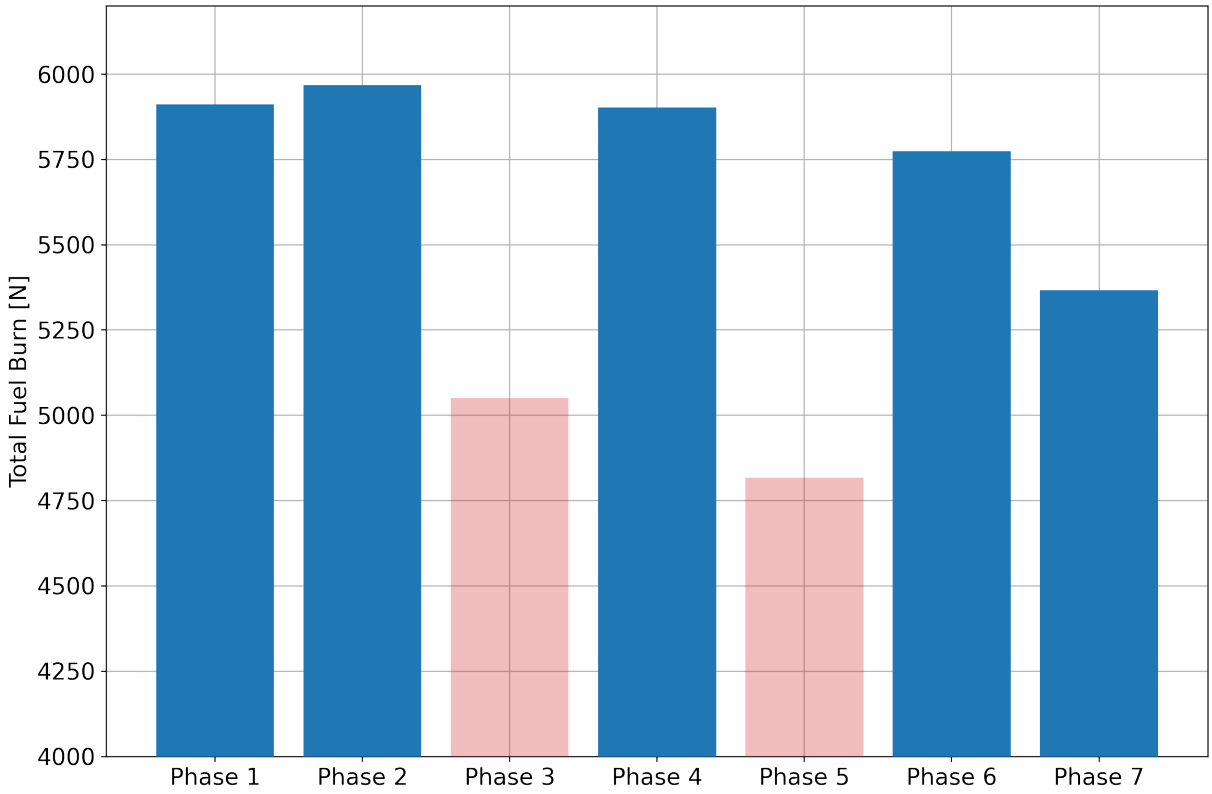


Figure 7-19: Total fuel burn in each phase

This design case study reveals that optimization centered, mutlifidelity approach to design is capable of handling changes in analysis and geometry fidelity by only swapping a few constraints in the optimization formulation. Furthermore, when redesign was necessary, constraints could be added or changed to reflect the newly obtained design knowledge, and the optimization could be run again in short order. This ability to rapidly adapt to changes in design knowledge was one of the most valuable elements of the GP approach to design, and it has been preserved here, but the added flexibility of SLCP allows for consideration of much higher fidelity tools than was possible in GP.

This effort also severs as a significant validation for SLCP, as this is the first time that a practical engineering black box analysis tool has been used with the algorithm. Despite the integration challenges, SLCP performs as expected and successfully improves designs.

Chapter 8

Application to a Subsonic Transport Design Problem

8.1 Problem Formulation

The previous design demonstration focused on improving the fidelity of only a single constraint in an existing Geometric Program, but did little to lay down a path towards integrating higher fidelity analysis models like CFD and FEA. And so two key questions remain: Can SLCP continue to converge to optimal solutions when the underlying design space changes to support high fidelity analysis? And how does the multifidelity process work when a model is made up of many constraints rather than one single constraint?

8.1.1 Kirschen-York Signomial Program

Since SLCP exploits underlying log-log convexity, there is a strong incentive to begin with a log-log convex optimization formulation. Fortunately, previous work by Kirschen et. al. [40] and York et. al. [41, 36] has outlined a Signomial Program that represents the design of a subsonic transport aircraft. For convenience, this work will collectively be referred to as the

Kirschen-York Signomial Program (KYSP).

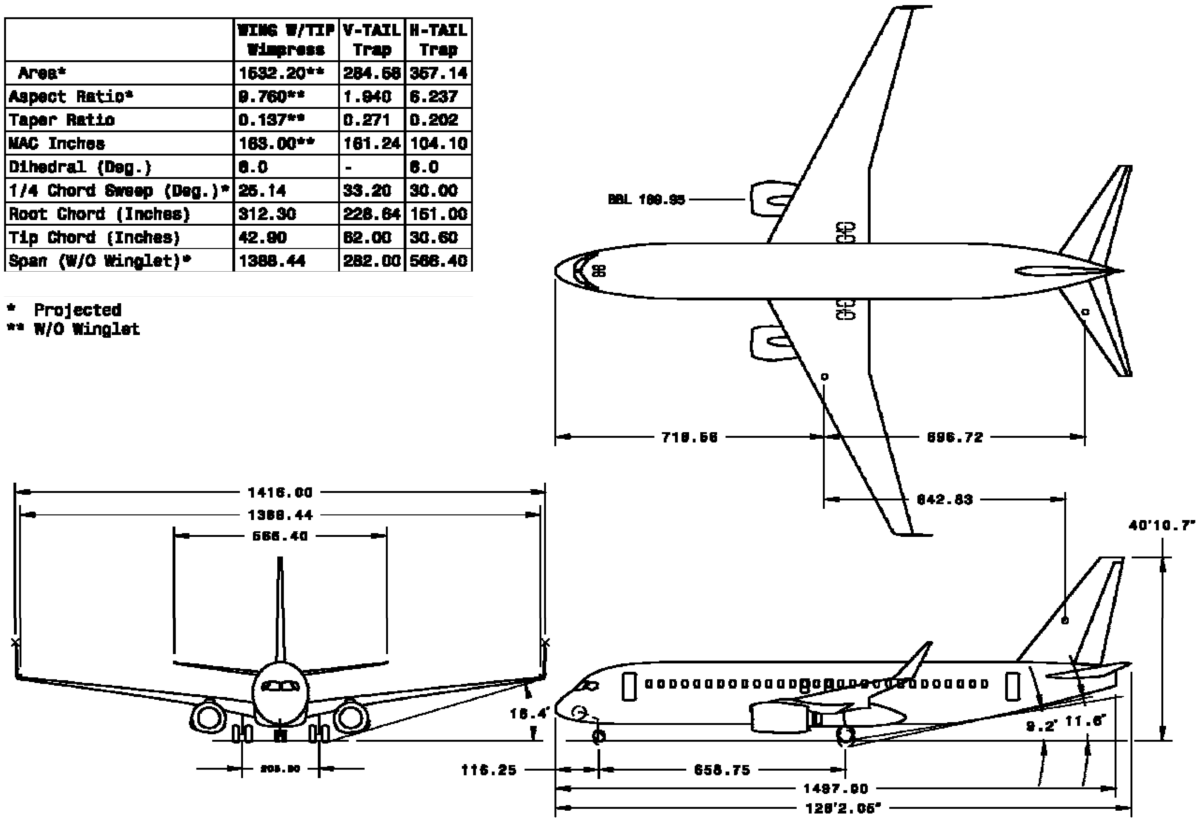
Though beginning with the KYSP is desirable, two challenges were encountered in attempting to use this as a starting point. First, the KYSP is an extremely large design optimization problem, varying in size from roughly 700 design variables to nearly 6000 design variables depending on a few different model options that can be selected, with a comparable number of constraints. Rebuilding the model from scratch was deemed a task too large to be in scope here. Additionally, a problem of this size was too large for practical implementation in the existing computational architecture, which was primarily designed for algorithm prototyping. Second, though the full KYSP was at one point implemented in GPkit, it has since been separated into a web of sub-models that proved difficult to untangle. Since GPkit does not support SLCP, the models would have to be migrated into the current computational architecture, and lack of sufficient institutional knowledge with the GPkit sub-models meant this was not a realistic approach.

Thus, rather than rebuild the entire KYSP from scratch, the focus here is on the signomial programming compatible wing model, and a simpler model was constructed that could represent the remainder of the aircraft design problem in a way that would be representative of the design space. This approach also made sense after a deep dive into the KYSP, which revealed many empirical factors and baked in assumptions that heavily biased the optimal solution towards the Boeing 737.

8.1.2 Boeing SUGAR Study

The aircraft system is modeled using data from the Subsonic Ultra Green Aircraft Research (SUGAR) final report [71]. The SUGAR study was a NASA funded project to have Boeing perform conceptual design studies on a variety of possible future subsonic aircraft configurations. Among the studied configurations are the conventional tube-tail-wing, the blended wing body, and the transonic truss braced wing. As a part of this study, a 737 sized aircraft is designed as a baseline for comparison to all other configurations. Named SUGAR

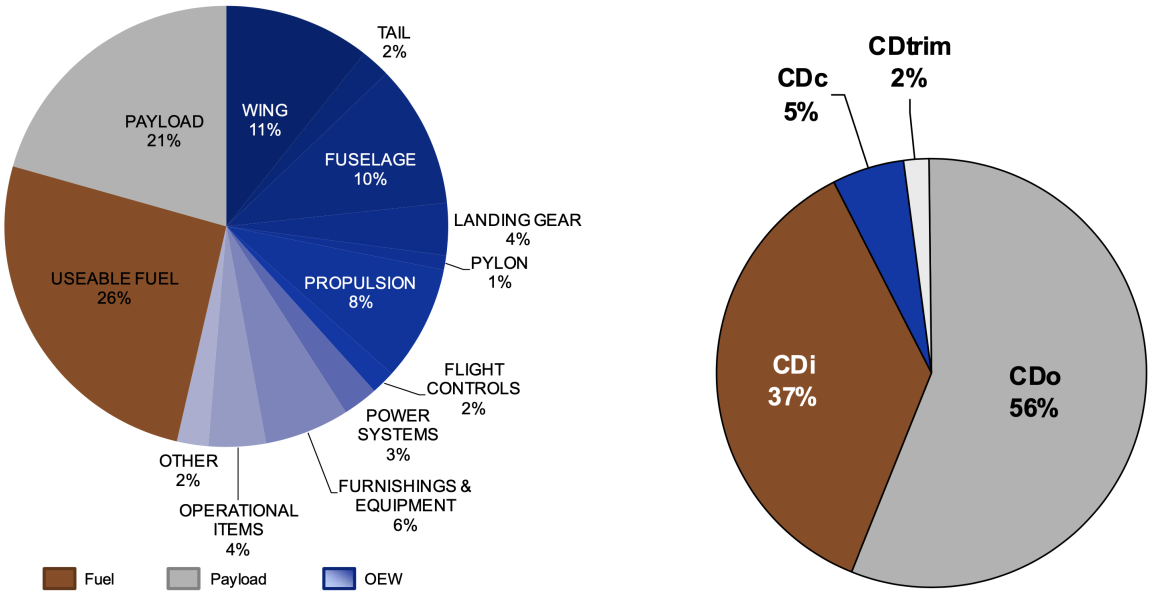
Free (Figure 8-1), this baseline study provides detailed breakdowns of the aircraft weight and aerodynamic performance that were obtained using industry design and analysis tools, making it the perfect data source for constructing the representing the aircraft system in this test case (Figures 8-2 and 8-3).



	WING W/TIP Winpress	V-TAIL Trap	H-TAIL Trap
Area*	1532.20**	284.58	357.14
Aspect Ratio*	9.760**	1.940	6.237
Taper Ratio	0.137**	0.271	0.202
MAC Inches	183.00**	161.24	104.10
Dihedral (Deg.)	6.0	-	6.0
1/4 Chord Sweep (Deg.)*	25.14	33.20	30.00
Root Chord (Inches)	312.30	228.64	151.00
Tip Chord (Inches)	42.90	62.00	30.60
Span (W/O Winglet)*	1388.44	282.00	566.40

* Projected
** W/O Winglet

Figure 8-1: Three view drawing of the SUGAR Free aircraft (from [71])



(a) Weight breakdown

(b) Drag breakdown at cruise

Figure 8-2: The weight and drag breakdowns of the SUGAR Free aircraft (from [71])

GROUP	WEIGHT (LB)	% TOGW
WING	**18,728**	**10.7**
BENDING MATERIAL	9,621	5.5
SPAR WEBS	1,290	0.7
RIBS AND BULKHEADS	1,226	0.7
AERODYNAMIC SURFACES	3,351	1.9
SECONDARY STRUCTURE	3,240	1.8
TAIL	**3,779**	**2.2**
FUSELAGE	**18,392**	**10.5**
LANDING GEAR	**6,712**	**3.8**
PYLON	**1,858**	**1.1**
PROPULSION	**14,874**	**8.5**
ENGINE	10,404	5.9
ENGINE SYSTEMS	263	0.1
EXHAUST SYSTEM	3,688	2.1
FUEL SYSTEM	520	0.3
FLIGHT CONTROLS	**3,084**	**1.8**
COCKPIT CONTROLS	252	0.1
SYSTEM CONTROLS	2,832	1.6
POWER SYSTEMS	**4,483**	**2.6**
AUXILIARY POWER PLANT	1,032	0.6
HYDRAULICS	894	0.5
ELECTRICAL	2,557	1.5
INSTRUMENTS	**686**	**0.4**
AVIONICS & AUTOPILOT	**1,533**	**0.9**
FURNISHINGS & EQUIPMENT	**10,866**	**6.2**
AIR CONDITIONING	**1,678**	**1.0**
ANTI-ICING	**118**	**0.1**
MANUFACTURER'S EMPTY WEIGHT (MEW)	**86,790**	**49.4**
OPERATIONAL ITEMS	7,342	4.2
OPERATING EMPTY WEIGHT (OEW)	**94,132**	**53.6**
USEABLE FUEL	45,313	25.8
PAYLOAD	36,190	20.6
TAKEOFF GROSS WEIGHT (TOGW)	**175,635**	**100.0**

Figure 8-3: Detailed weight breakdown of the SUGAR Free aircraft (from [71])

8.1.3 The Resulting Formulation

The resulting optimization formulation is a signomial program, consisting of the performance model from the KYSP, the wing model from the KYSP, and empirical factors that match the weight and drag breakdowns to the data from the SUGAR Free design. Figure 8-4 shows a graphical representation of how the KYSP wing model is 'grafted' into the SUGAR design problem.

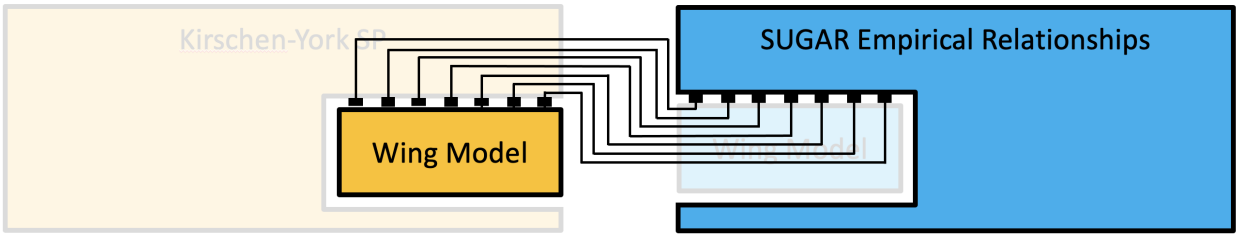


Figure 8-4: Visual representation of how the hybrid design problem was constructed. The KYSP wing model is grafted into a larger subsonic design framework constructed using the SUGAR Free empirical relationships.

The problem seeks to minimize the objective function:

$$W_{\text{fuel,total}} \tag{8.1}$$

Subject to the following constraints, which are classified for readability. Note that unless otherwise specified, constraints apply to each individual flight segment.

Mission Performance:

$$
\begin{aligned}
R_i &= \frac{R_{\text{req}}}{N_{\text{seg}}} \\
z_{\text{bre}} &\geq c_{\text{TSFC}} t D W_{\text{avg}} \\
W_{\text{fuel}} &\geq W_{\text{end}} \left(z_{\text{bre}} + \frac{z_{\text{bre}}^2}{2} + \frac{z_{\text{bre}}^3}{6} \right) \\
W_{\text{avg}} &= \sqrt{W_{\text{start}} W_{\text{end}}} \\
W_{\text{start},0} &= W_{\text{max}} \\
W_{\text{start}} &\geq W_{\text{end}} + W_{\text{fuel}} \\
W_{\text{start},i+1} &= W_{\text{end},i} \\
W_{\text{fuel,primary}} &\geq \sum_{i=1}^{N_{\text{seg}}} W_{\text{fuel},i} \\
W_{\text{fuel,reserve}} &= 0.02 W_{\text{fuel,primary}} \\
W_{\text{fuel,total}} &\geq W_{\text{fuel,primary}} + W_{\text{fuel,reserve}} \\
W_{\text{end},N_{\text{seg}}} &\geq W_{\text{dry}} + W_{\text{payload}} + W_{\text{fuel,reserve}} \\
M &= \frac{V_\infty}{a} \\
M &= 0.78 \\
W_{\text{payload}} &= 36190 \text{ lbf}
\end{aligned}
\tag{8.2}
$$

System Level Relationships

$$
\begin{aligned}
W_{\text{max}} &\geq W_{\text{dry}} + W_{\text{payload}} + W_{\text{fuel,total}} \\
W_{\text{dry}} &\geq W_{\text{wing}} + W_{\text{OEW,SUGAR-Wing}} \\
D &>= D_{\text{wing}} + D_{\text{fuse}} + D_{\text{vtail}} + D_{\text{htail}} \\
D_{\text{fuse}} &= 0.15 W_{\text{start}} \\
D_{\text{vtail}} &= 0.005 W_{\text{start}} \\
D_{\text{htail}} &= 0.005 W_{\text{start}}
\end{aligned}
\tag{8.3}
$$

Wing Geometry Relationships:

$$
\begin{aligned}
S &= b\left(\frac{c_{\text{root}} + c_{\text{tip}}}{2}\right) \\
\bar{c} &\geq \frac{2}{3} c_{\text{root}} \frac{1+\lambda+\lambda^2}{q} \\
y_{\bar{c}} &= \frac{bq}{3p} \\
\lambda &= \frac{c_{\text{tip}}}{c_{\text{root}}} \\
\lambda &= 0.4 \\
\text{Æ} &= \frac{b^2}{S}
\end{aligned}
\tag{8.4}
$$

Wing Weight Model:

$$
\begin{aligned}
2q &\geq 1 + p \\
p &\geq 1.9 \\
\tau &\leq 0.125 \\
\bar{M_r} &\geq \frac{\tilde{W}\text{Æ}p}{24} \\
0.92\bar{w}\tau\bar{t}_{\text{cap}}^2 + \bar{I}_{\text{cap}} &\leq \frac{0.92^2}{2}\bar{w}\tau^2\bar{t}_{\text{cap}} \\
8 &\geq \frac{N_{\text{lift}}\bar{M}_r\text{Æ}q^2\tau}{S\bar{I}_{\text{cap}}\sigma_{\max}} \\
12 &\geq \frac{\text{Æ}\tilde{W}N_{\text{lift}}q^2}{\tau S\bar{t}_{\text{web}}\sigma_{\max,\text{shear}}} \\
\nu^{3.94} &\geq 0.86p^{-2.38} + 0.14p^{0.56} \\
W_{\text{cap}} &\geq \frac{8\rho_{\text{cap}}g\tilde{w}\bar{t}_{\text{cap}}S^{3/2}\nu}{3\text{Æ}^{1/2}} \\
W_{\text{web}} &\geq \frac{8\rho_{\text{web}}gr_h\tau\bar{t}_{\text{web}}S^{3/2}\nu}{3\text{Æ}^{1/2}} \\
W_{\text{struct}} &\geq W_{\text{cap}} + W_{\text{web}} \\
W_{\text{wing}} &\geq 1.64W_{\text{struct}}
\end{aligned}
\tag{8.5}
$$

Wing Aerodynamic Model:

$$
\begin{aligned}
L_{\text{total}} &\geq W_{\text{avg}} \\
L_{\text{total}} &= 1.05 L_{\text{wing}} \\
L_{\text{wing}} &= \eta_{\text{wing}} \frac{1}{2} \rho_\infty V_\infty^2 S C_L \\
L_{\text{wing,max}} &\geq 2.5 W_{\max} \\
C_L &= C_{L_\alpha} \alpha \\
\alpha &\leq 14.3^\circ \\
t_{\Lambda^2} &= \frac{4}{225}\Lambda^{10} + \frac{4}{45}\Lambda^8 + \frac{17}{45}\Lambda^6 + \frac{2}{3}\Lambda^4 + \Lambda^2 \\
4\pi^2 &= \frac{C_{L_\alpha}^2}{\eta_{\text{wing}}^2}(1 + t_{\Lambda^2} - M^2) + \frac{8\pi C_{L_\alpha}}{\text{Æ}R} \\
D &= \frac{1}{2} \rho_\infty V_\infty^2 S C_D \\
C_D &\geq C_{D_p} + C_{D_i} \\
c_\Lambda &= 1 - \frac{\Lambda^2}{2!} + \frac{\Lambda^4}{4!} - \frac{\Lambda^6}{6!} + \frac{\Lambda^8}{8!} \\
C_{D_p}^{1.6515} &\geq 1.61418 (Re/1000)^{-0.550434} \tau^{1.29151} (M c_\Lambda)^{3.03609} C_L^{1.77743} \\
&\quad + 0.0466407 (Re/1000)^{-0.389048} \tau^{0.784123} (M c_\Lambda)^{-0.340157} C_L^{0.950763} \\
&\quad + 190.811 (Re/1000)^{-0.218621} \tau^{3.94654} (M c_\Lambda)^{19.2524} C_L^{1.15233} \\
&\quad + 2.82283 \times 10^{-12} (Re/1000)^{1.18147} \tau^{-1.75664} (M c_\Lambda)^{0.10563} C_L^{-1.44114} \\
Re &\geq \frac{\rho_\infty V_\infty \bar{c}}{\mu_\infty} \\
C_{D_i} &\geq \frac{C_L^2}{\pi e \text{Æ}R} \\
1 &\geq e(1 + \text{Æ}R(0.0524\lambda^4 - 0.15\lambda^3 + 0.1659\lambda^2 - 0.0706\lambda + 0.0119)) \\
e &\leq 1.0
\end{aligned}
\tag{8.6}
$$

8.1.4 The Multifidelity and Multidisciplinary Process for the Hybrid KYSP-SUGAR Problem

As before, it is possible to represent the multifidelity design process for this example problem using a simple flowchart (Figure 8-5).

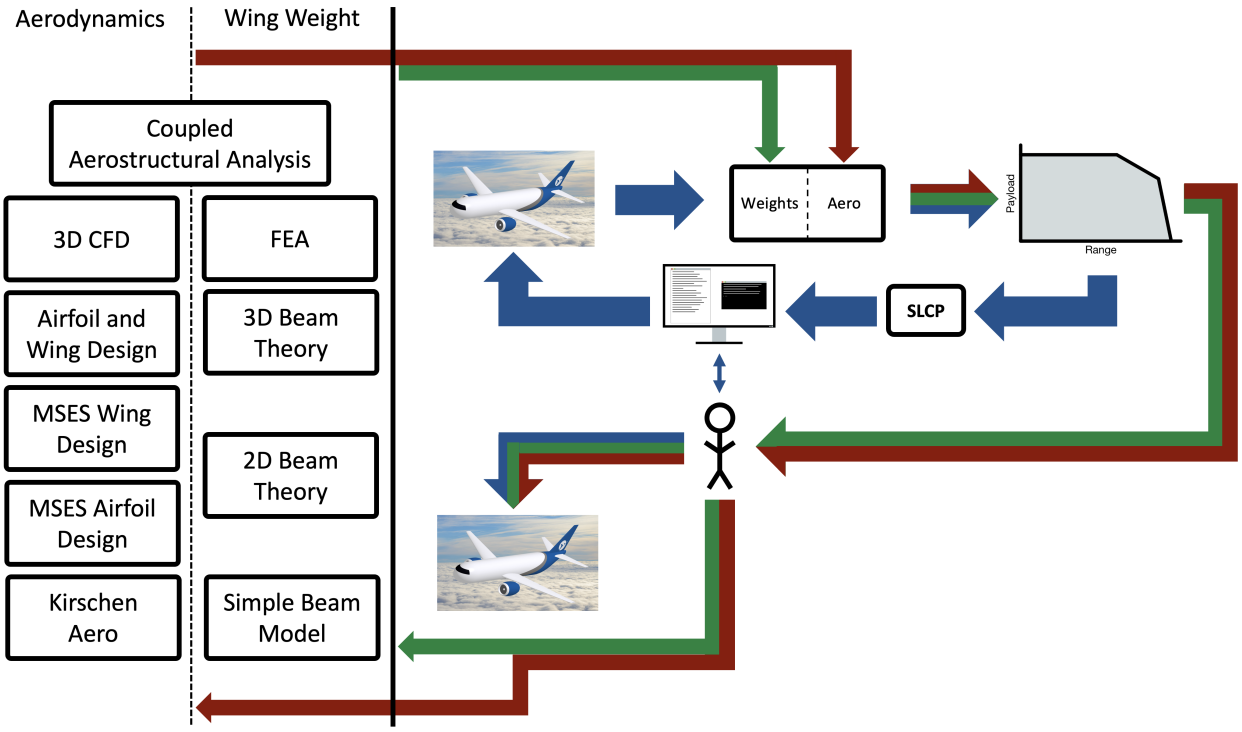


Figure 8-5: A sketch of the multifidelity design process for the KYSP-SUGAR hybrid problem. Note that unlike the previous example, there are now two multifidelity analysis trees in the problem.

Unlike the previous example problem, there are now two two multifidelity analysis trees (Wing Aerodynamics and Wing Weight) that ultimately combine at the highest levels of fidelity into a single aero-structural model. The goal here is not to climb the entire fidelity tree, rather, it is to demonstrate that the SLCP algorithm and optimization first design approach continues to be successful even under a significant change to the underlying structure of the design problem. Therefore, this example will focus on the lowest three levels of wing aerodynamic fidelity in Figure 8-5 and the lowest two levels of wing weight fidelity.

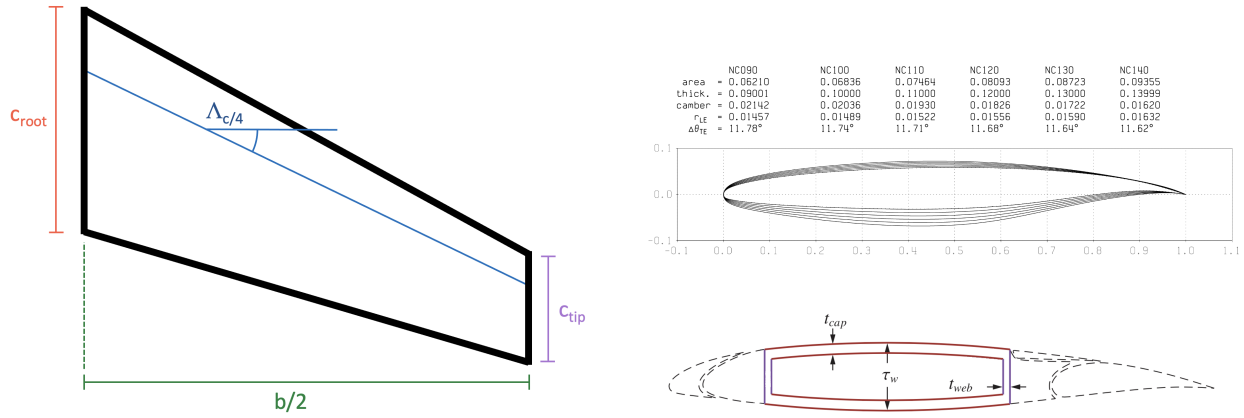
The two fidelity trees are not entirely decoupled, as evidenced by the fact that the two trees combine at the highest level at fidelity. In the previous example problem, an airfoil defined by the Kulfan CST still required a definition of airfoil thickness, τ. The aerodynamic model at that point had outpaced the structural model, and so an artificial reduced parameter had to be computed for compatibility across the two fidelity trees. The human decision maker or design team is responsible for ensuring that models 'match'. Current state of the art in this regard is to use the experience of a practiced aircraft design expert, though in general matching analysis models will operate on roughly the same geometry parameterization and be sensitive to changes in all of these parameters. In Chapter 9, a more quantitative approach is speculated on, but for the foreseeable future it is unlikely that any quantitative decision metric will become readily available.

8.2 The Original Signomial Program

The design variables that define the wing in the signomial program described in Section 8.1.3 are relatively simple: 4 variables define a swept trapezoidal wing planform, a single variable defines airfoil thickness, and two variables define the cross sectional thickness of the main wing spar (these are scaled based on local chord).

Unlike the previous NACA24XX family of airfoils, the TASOPT airfoil family used here does not have a consistent shape function as thickness is varied. But with the original GP compatible data fit[35] (Equation 8.6) the entirely family is being represented in the surrogate model where the only design variable is thickness, thus grouping the various shape changes into a single variable.

For this first design phase, the signomial program is implemented directly without modification. Design range is set to 3500 nautical miles at a flight altitude of 34000 ft, and two mission segments are used. These parameters will be the same throughout all of the design phases. Since this problem is known to be a signomial program, it can be solved with the basic SLCP. No step scheduling is required as all of the constraints have readily available



(a) Planform geometry

(b) Section geometry

Figure 8-6: The wing geometry definition for the first design phase. Airfoil image taken from [72]

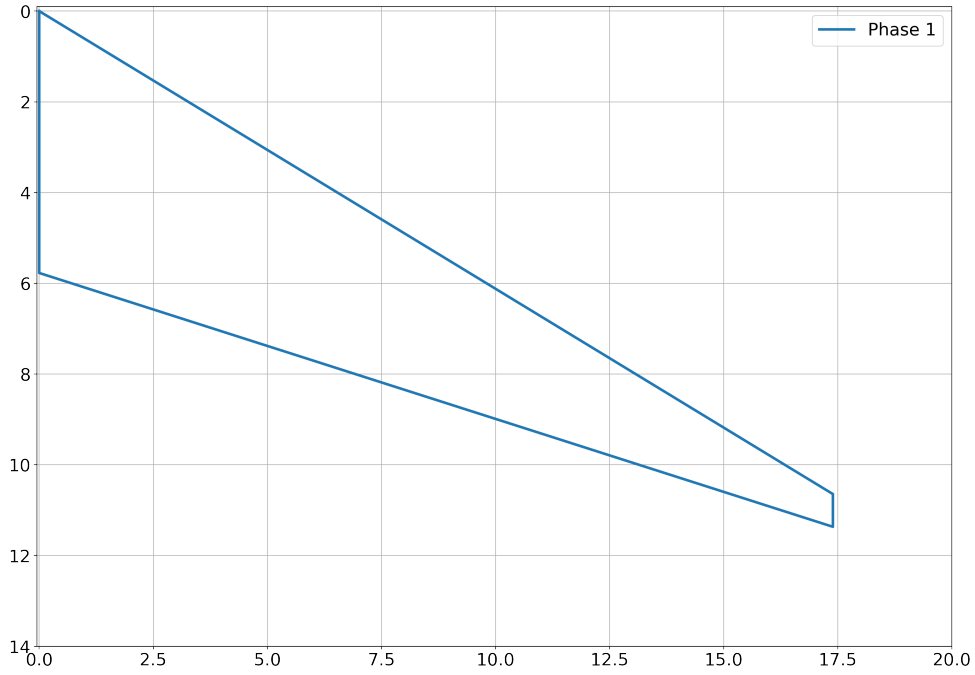
analytic derivatives.

The relevant results are summarized in Table 8.1, and a full reporting of results is available in Appendix E. Wing geometry is reported graphically in Figure 8-7.

Wing weight is significantly over predicted at 24385 pounds compared to the 18728 pounds reported by the SUGAR study [71], which casts some doubts on the structural model, however the success of the model in existing publications [10, 35, 40, 36] gives confidence to proceed to the next design phase.

Table 8.1: Relevant results from the KYSP-SUGAR hybrid design problem

Design Variable	Value	Units	Description
$\mathcal{AR}$	10.71392	[-]	Wing Aspect Ratio
$C_{D_{i,0}}$	0.01115	[-]	Wing Induced Drag Coefficient, segment 0
$C_{D_{i,1}}$	0.00897	[-]	Wing Induced Drag Coefficient, segment 1
$C_{D_{p,0}}$	0.00607	[-]	Wing Pressure Drag Coefficient, segment 0
$C_{D_{p,1}}$	0.00613	[-]	Wing Pressure Drag Coefficient, segment 1
C_{D_0}	0.01722	[-]	Wing Drag Coefficient, segment 0
C_{D_0}	0.01510	[-]	Wing Drag Coefficient, segment 1
C_{L_0}	0.59564	[-]	Wing Lift Coefficient, segment 0
C_{L_1}	0.53424	[-]	Wing Lift Coefficient, segment 1
Λ	32.64714	deg	Wing Sweep
S	112.93343	m^2	Vehicle reference area (wing area)
W_{dry}	99790.11788	lbf	Aircraft dry weight
$W_{\text{fuel,total}}$	33877.36229	lbf	Total fuel carried
W_{max}	169857.48017	lbf	Maximum aircraft weight
W_{wing}	24385.11788	lbf	Wing weight
α_0	6.22862	deg	Wing Angle of Attack, segment 0
α_1	5.58691	deg	Wing Angle of Attack, segment 1
b	34.78447	m	Wing Span
c_{root}	5.77184	m	Wing Root Chord
c_{tip}	0.72148	m	Wing Tip Chord
t_{cap}	0.00368	[-]	Spar cap thickness per unit chord
t_{web}	0.00244	[-]	Spar web thickness per unit chord
τ	0.12500	[-]	Wing Thickness Ratio



(a) Planform geometry

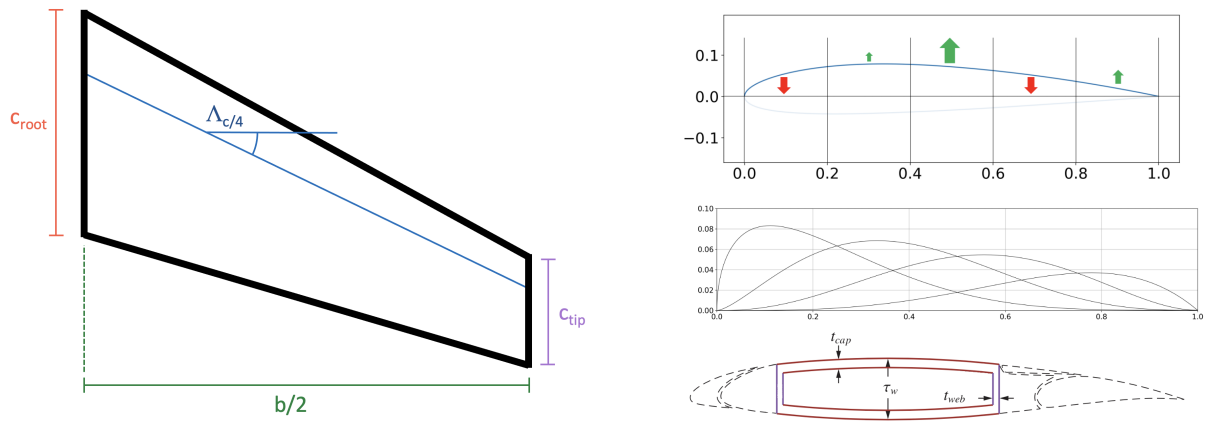
(b) Section geometry

Figure 8-7: The wing geometry results for the first design phase

8.3 Using MSES and CST-4 for Airfoil Design

As with the Hoburg example, one of the biggest areas for fidelity improvement is the fit to XFOIL data that is used to predict wing profile drag (Equation 8.6). As before, MSES can be used in the loop to significantly increase the aerodynamic fidelity. Based on the results of the Hoburg case study, the Kulfan CST-4 parameterization is used on both top and bottom surfaces of the airfoil for a total of 8 design variables that define the wing cross section. Figure 8-8 summarizes the geometry used.

The modifications to the optimization formulation are fairly minor. The Mission Performance



(a) Planform geometry

(b) Section geometry

Figure 8-8: The wing geometry definition for the second design phase

Constraints (Equation 8.2), System Level Relationships (Equation 8.3), Wing Geometry Relationships (Equation 8.4), and Weight Model (Equation 8.5) are used again with no modification. Though the wing section is now being varied, it is still assumed that the section is the same over the entire wing length and that the wing is untwisted.

The Wing Aerodynamic Model has four key changes. First, the aerodynamic black box which calls MSES is structured to return total wing drag, rather than simply the profile drag of a section. This modification is primarily to set up for higher fidelity as will be discussed later. Second, MSES cannot take C_L as an input, an intermediate $C_{L_{\text{MSES}}}$ variable must be introduced specifically for MSES, and then angle of attack is driven to ensure $C_L = C_{L_{\text{MSES}}}$. Third, constraints are now structured to utilize the Kulfan CST coefficients. Finally, wing thickness is computed using the Kulfan CST coefficients, and this constraint is implemented as a second black box in the optimization problem. The resulting wing model takes the

following form:

$$
\begin{aligned}
L_{\text{total}} &\geq W_{\text{avg}} \\
L_{\text{total}} &= 1.05 L_{\text{wing}} \\
L_{\text{wing}} &= \eta_{\text{wing}} \frac{1}{2} \rho_{\infty} V_{\infty}^2 S C_L \\
L_{\text{wing,max}} &\geq 2.5 W_{\text{max}} \\
D &= f(\bar{K}_u, \bar{K}_l, \alpha, b, c_{\text{root}}, c_{\text{tip}}, \Lambda, V_{\infty}) \\
C_{L_{\text{MSES}}} &= f(\bar{K}_u, \bar{K}_l, \alpha, b, c_{\text{root}}, c_{\text{tip}}, \Lambda, V_{\infty}) \\
C_L &= C_{L_{\text{MSES}}} \\
\tau &= f(\bar{K}_u, \bar{K}_l) \\
\bar{K}_u &\geq 0.0 \\
\bar{K}_l &\leq 0.0 \\
\bar{K}_{l,1} &\leq -0.09
\end{aligned}
\tag{8.7}
$$

The drag constraint implements the following analysis:

$$
\begin{aligned}
S &= b\left(\frac{c_{\text{root}} + c_{\text{tip}}}{2}\right) \\
\lambda &= \frac{c_{\text{tip}}}{c_{\text{root}}} \\
\bar{c} &= \frac{2}{3}c_{\text{root}}\frac{1+\lambda+\lambda^2}{q} \\
Re &= \frac{\rho_\infty V_\infty \bar{c}}{\mu_\infty} \\
M &= \frac{V_\infty}{a} \\
M_\perp &= M\cos\Lambda \\
C_{D_p} &= f_{\text{MSES}}(\bar{K}_u, \bar{K}_l, \alpha, Re, M_\perp) \\
C_L &= f_{\text{MSES}}(\bar{K}_u, \bar{K}_l, \alpha, Re, M_\perp) \\
\text{Æ} &= \frac{b}{(c_{\text{root}} + c_{\text{tip}})/2} \\
e &= 1/(1 + \text{Æ}(0.0524\lambda^4 - 0.15\lambda^3 + 0.1659\lambda^2 - 0.0706\lambda + 0.0119)) \\
C_{D_i} &\geq \frac{C_L^2}{\pi e \text{Æ}} \\
C_D &= C_{D_p} + C_{D_i} \\
D &= \frac{1}{2}\rho_\infty V_\infty^2 S C_D
\end{aligned}
\tag{8.8}
$$

which still maintains a substantial portion of the previous aerodynamic model, but has been contained within the black box model for this design phase. Derivatives are computed analytically or using automatic differentiation when possible, but the calls to MSES mean that the chain rule must be completed manually at various phases in the analysis run. The model for τ comes from the CST definition in [70].

Since the MSES black box is being used in this optimization problem, a step schedule must once again be utilized to damp the gradient noise coming from the black box. Failures of MSES are again handled by reducing step size according to the τ schedule[1] discussed in the

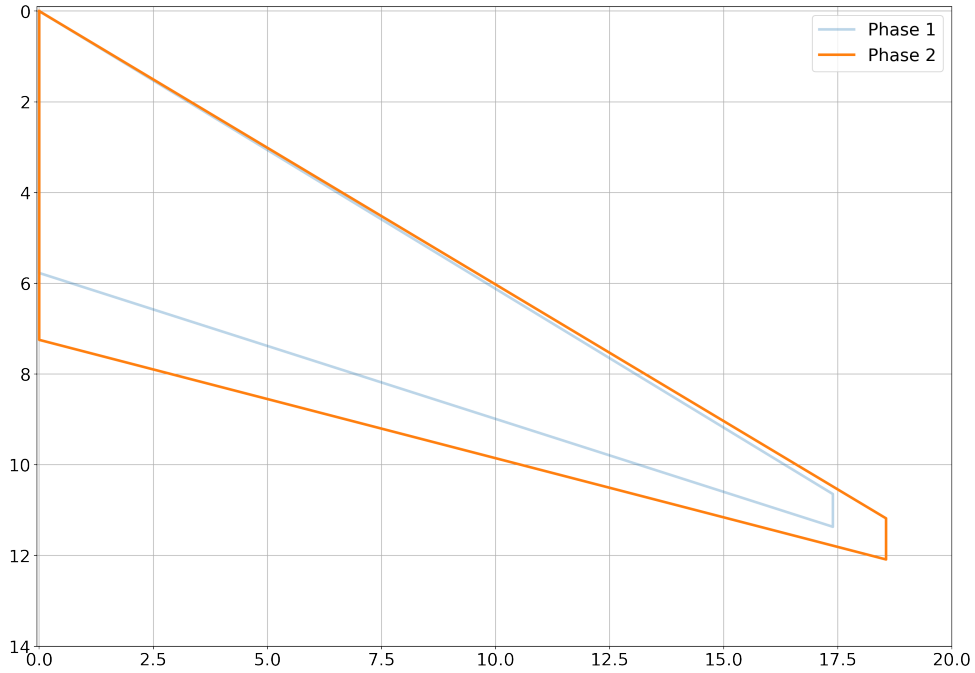
[1]This should not be confused with wing thickness ratio

SLCP algorithm section (see Appendix B).

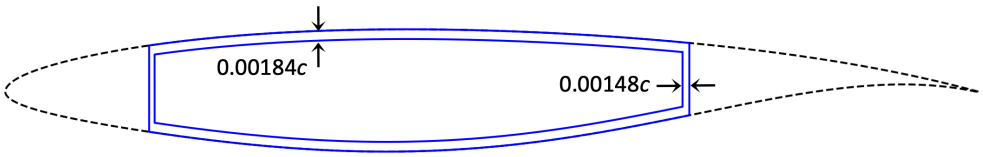
The relevant results are summarized in Table 8.2, and a full reporting of results is available in Appendix E. Wing geometry is reported graphically in Figure 8-9.

Table 8.2: Relevant results from the KYSP-SUGAR hybrid design problem

Design Variable	Value	Units	Description
Æ	9.10577	[-]	Wing Aspect Ratio
$K_{l,1}$	-0.10249	[-]	Kulfan coefficient, lower surface mode 1
$K_{l,2}$	-0.13671	[-]	Kulfan coefficient, lower surface mode 2
$K_{l,3}$	-0.32729	[-]	Kulfan coefficient, lower surface mode 3
$K_{l,4}$	0.22045	[-]	Kulfan coefficient, lower surface mode 4
$K_{u,1}$	0.12334	[-]	Kulfan coefficient, upper surface mode 1
$K_{u,2}$	0.18417	[-]	Kulfan coefficient, upper surface mode 2
$K_{u,3}$	0.15806	[-]	Kulfan coefficient, upper surface mode 3
$K_{u,4}$	0.27029	[-]	Kulfan coefficient, upper surface mode 4
C_{L_0}	0.42670	[-]	Wing Lift Coefficient, segment 0
C_{L_1}	0.39154	[-]	Wing Lift Coefficient, segment 1
D_0	34589.22841	N	Wing drag, segment 0
D_1	30102.39882	N	Wing drag, segment 1
Λ	31.13625	deg	Wing Sweep
S	151.27738	m^2	Vehicle reference area (wing area)
W_{dry}	96306.42165	lbf	Aircraft dry weight
$W_{\text{fuel,total}}$	29325.85797	lbf	Total fuel carried
W_{max}	161822.27962	lbf	Maximum aircraft weight
W_{wing}	20901.42165	lbf	Wing weight
α_0	-0.55027	deg	Angle of Attack, segment 0
α_1	-0.84434	deg	Angle of Attack, segment 1
b	37.10101	m	Wing Span
c_{root}	7.24450	m	Wing Root Chord
c_{tip}	0.90556	m	Wing Tip Chord
t_{cap}	0.00184	[-]	Spar cap thickness per unit chord
t_{web}	0.00148	[-]	Spar web thickness per unit chord
τ	0.12465	[-]	Wing Thickness Ratio



(a) Planform geometry



(b) Section geometry

Figure 8-9: The wing geometry results for the first design phase

Figure 8-10 compares the optimized airfoil to the initial guess airfoil taken from the TASOPT family.

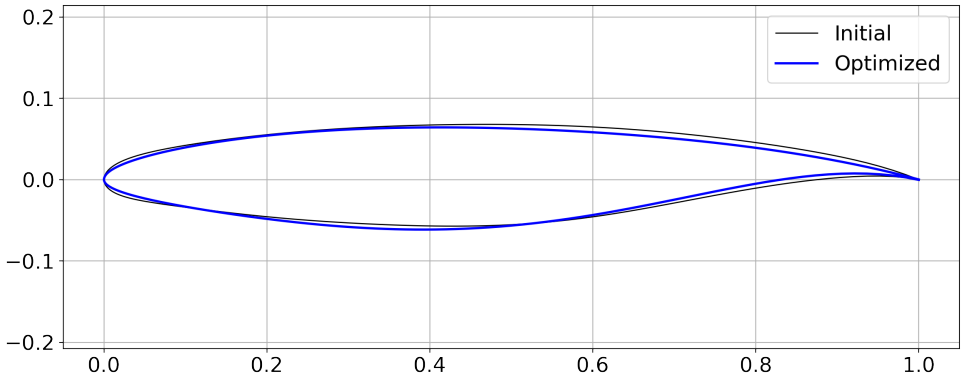


Figure 8-10: Comparison of the optimized airfoil to the initial guess from the TASOPT family of airfoils

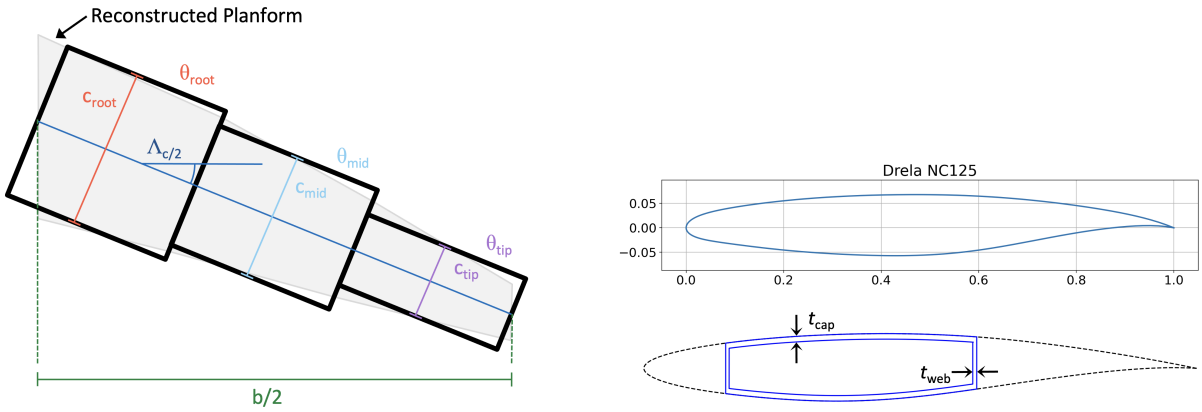
The optimized airfoil has a smaller leading edge radius, a thinner trailing edge, and has shifted the point of maximum thickness on the lower surface forward by approximately 5% chord. The result here is reasonable, but perhaps slightly optimistic from the aerodynamic perspective. The lack of structural constraints for the trailing edge allows the aft section of the airfoil to become thinner, and the consideration of a small number of aerodynamic operating conditions allows for the smaller leading edge radius.

8.4 Wing Planform Design

Despite improvements to the aerodynamic model, the previous design phase still slightly over-predicted the wing weight compared to the SUGAR baseline (20901 pounds compared to 18728 pounds), and so the fidelity of the wing weight model must be improved. The wing planform is divided into three 'panels' of equal span and equal sweep, but the chord length and twist of each panel is allowed to vary (Figure 8-11). This geometry definition yields a total of eight variables that define the planform geometry.

Given the somewhat optimistic nature of the airfoil obtained in the previous design phase, the TASOPT NC125 airfoil was used for all three panels.

Two structural variables are defined for each panel (for a total of six): spar cap thickness



(a) Planform geometry (b) Section geometry

Figure 8-11: The wing geometry definition for the third design phase

and spar web thickness, each normalized by local chord (Figure 8-11). Unlike the previous structural model, this allows the spar mass to be distributed along the span in the targeted areas it is most critical, in theory allowing for significant weight savings. As a result of these changes, the underlying structure of the design space changes significantly with the addition of the new planform variables, but the constraint implementation is similar to the last design phase. The Mission Performance Constraints (Equation 8.2) and System Level Relationships (Equation 8.3) are used with no modification. However, most of the rest of the optimization problem is now modified:

$$
\begin{aligned}
L_{\text{total}} &\geq W_{\text{avg}} \\
L_{\text{total}} &= 1.05 L_{\text{wing}} \\
L_{\text{wing,max}} &\geq 2.5 W_{\text{max}} \\
W_{\text{wing}} &\geq 1.64 W_{\text{struct}} \\
\alpha_{\text{out}} &= \alpha_{\text{wing}} + \theta_{\text{out}} \\
\alpha_{\text{mid}} &= \alpha_{\text{wing}} + \theta_{\text{mid}} \\
\alpha_{\text{in}} &= \alpha_{\text{wing}} + \theta_{\text{in}} \\
D &\geq f(b, c_{\text{out}}, c_{\text{mid}}, c_{\text{in}}, \Lambda, \alpha_{\text{out}}, \alpha_{\text{mid}}, \alpha_{\text{in}}, V_{\infty})
\end{aligned}
$$

$$
\begin{aligned}
L &= f(b, c_{\text{out}}, c_{\text{mid}}, c_{\text{in}}, \Lambda, \alpha_{\text{out}}, \alpha_{\text{mid}}, \alpha_{\text{in}}, V_\infty) \\
W_{\text{struct}} &\geq f(b, c_{\text{out}}, c_{\text{mid}}, c_{\text{in}}, \Lambda, t_{\text{cap,out}}, t_{\text{cap,mid}}, t_{\text{cap,in}}, t_{\text{web,out}}, t_{\text{web,mid}}, t_{\text{web,in}}, L_{\text{wing,max}}) \\
\sigma_{\text{out}} &= f(b, c_{\text{out}}, c_{\text{mid}}, c_{\text{in}}, \Lambda, t_{\text{cap,out}}, t_{\text{cap,mid}}, t_{\text{cap,in}}, t_{\text{web,out}}, t_{\text{web,mid}}, t_{\text{web,in}}, L_{\text{wing,max}}) \\
\sigma_{\text{mid}} &= f(b, c_{\text{out}}, c_{\text{mid}}, c_{\text{in}}, \Lambda, t_{\text{cap,out}}, t_{\text{cap,mid}}, t_{\text{cap,in}}, t_{\text{web,out}}, t_{\text{web,mid}}, t_{\text{web,in}}, L_{\text{wing,max}}) \\
\sigma_{\text{in}} &= f(b, c_{\text{out}}, c_{\text{mid}}, c_{\text{in}}, \Lambda, t_{\text{cap,out}}, t_{\text{cap,mid}}, t_{\text{cap,in}}, t_{\text{web,out}}, t_{\text{web,mid}}, t_{\text{web,in}}, L_{\text{wing,max}}) \\
\nu_{\text{out}} &= f(b, c_{\text{out}}, c_{\text{mid}}, c_{\text{in}}, \Lambda, t_{\text{cap,out}}, t_{\text{cap,mid}}, t_{\text{cap,in}}, t_{\text{web,out}}, t_{\text{web,mid}}, t_{\text{web,in}}, L_{\text{wing,max}}) \\
\nu_{\text{mid}} &= f(b, c_{\text{out}}, c_{\text{mid}}, c_{\text{in}}, \Lambda, t_{\text{cap,out}}, t_{\text{cap,mid}}, t_{\text{cap,in}}, t_{\text{web,out}}, t_{\text{web,mid}}, t_{\text{web,in}}, L_{\text{wing,max}}) \\
\nu_{\text{in}} &= f(b, c_{\text{out}}, c_{\text{mid}}, c_{\text{in}}, \Lambda, t_{\text{cap,out}}, t_{\text{cap,mid}}, t_{\text{cap,in}}, t_{\text{web,out}}, t_{\text{web,mid}}, t_{\text{web,in}}, L_{\text{wing,max}}) \\
\sigma_{\text{out}} &\leq \sigma_{\max} \\
\sigma_{\text{mid}} &\leq \sigma_{\max} \\
\sigma_{\text{in}} &\leq \sigma_{\max} \\
\nu_{\text{out}} &\leq \nu_{\max} \\
\nu_{\text{mid}} &\leq \nu_{\max} \\
\nu_{\text{in}} &\leq \nu_{\max}
\end{aligned}
\tag{8.9}
$$

It should be noted that at this point, nearly all of the critical analysis has been black boxed from the optimizer, with the exception of a simple performance model and a few simple weight relationships.

The wing aerodynamic model computes the sub-area of each panel as $S_{sub} = (b/3)c_{sub}\cos\Lambda$. These three sub areas are summed to give the full planform area. Mach number is computed as $M = V_\infty/a$, and $M_\perp = M\cos\Lambda$. Reynolds Number for each section is computed. MSES is then used to compute $C_L, C_{D_p} = f_{\text{MSES}}(\alpha, Re, M_\perp)$ for each section. Induced drag coefficient is computed via Treftz Plane Analysis from Drela [73], though the computed induced drag is reduced by 15% to correct for an overly pessimistic Oswald efficiency that comes from only using 3 wing panels. Wing drag coefficient is computed as $C_D = C_{D_p} + C_{D_i}$, then lift and

drag are both computed by multiplying the appropriate coefficient by $\frac{1}{2}\rho_\infty V_\infty^2 S$.

The wing weight model takes a fixed airfoil (in this case the RAE 2822) and sets the shear web locations at 15% and 60% of the chord. The wing is modeled as a stepped beam with length $\ell = b/\cos\Lambda$ divided into 3 equal segments. The bending moment of inertia of each panel is computed, and the sub-root of each stepped section (ie, the critical section) is analyzed to determine the bending and shear stresses in each of the three critical sections. An elliptical load distribution is assumed. Bending is assumed to be carried entirely by the caps, and shear entirely by the webs. The volume of the spar in each distinct panel is computed by multiplying the area of spar in the cross section by the length of the panel. These three values are summed and multiplied by the weight of aluminum to obtain the wing structural weight.

Once again the use of MSES requires the introduction of a step schedule that damps out the gradient noise near convergence, but one more 'conditioning' action is required here. As has been mentioned, utilizing MSES in this fully automatic framework causes MSES to fail roughly 30% of the time, even on reasonable airfoils at reasonable flow conditions. Therefore, using only three panels, the probability of all three MSES cases converging is only around 35% even under good conditions. As a result, the step size reduction routine regularly triggers in SLCP, significantly handicapping the ability of the algorithm to make meaningful progress. If the step size is too small, the relative convergence metric may trigger prematurely, and so this test case was set up to terminate manually.

Inspecting Figures 8-12 through 8-14 shows that the solution has converged by the 25th iteration, and the data at the design point indicated that the relevant constraints were tight, and so the algorithm was terminated manually at this point. The relevant results are summarized in Table 8.3, and a full reporting of results is available in Appendix E. Wing geometry is reported graphically in Figure 8-15.

The increase in design variables allows the optimizer to push to a design with a slightly

Table 8.3: Relevant results from the KYSP-SUGAR hybrid design problem Phase 3

Design Variable	Value	Units	Description
α_0	0.40409	deg	Angle of Attack, segment 0
α_1	0.11633	deg	Angle of Attack, segment 1
D_0	29437.22895	N	Wing drag, segment 0
D_1	26802.73540	N	Wing drag, segment 1
Λ	24.64614	deg	Wing Sweep
L_0	640545.41372	N	Wing lift, segment 0
L_1	585018.44472	N	Wing lift, segment 1
W_{dry}	89006.66780	lbf	Aircraft dry weight
W_{fuel}	25493.18388	lbf	Total fuel carried
W_{max}	150689.85168	lbf	Maximum aircraft weight
W_{wing}	13601.66780	lbf	Wing weight
b	34.59783	m	Wing Span
c_{out}	1.70296	m	Wing chord, outboard panel
c_{mid}	2.87933	m	Wing chord, middle panel
c_{in}	4.31439	m	Wing chord, inboard panel
$t_{\text{cap,out}}$	0.00768	[-]	Normalized spar cap thickness, outboard panel
$t_{\text{cap,mid}}$	0.00881	[-]	Normalized spar cap thickness, middle panel
$t_{\text{cap,in}}$	0.00662	[-]	Normalized spar cap thickness, inboard panel
$t_{\text{web,out}}$	0.00518	[-]	Normalized spar web thickness, outboard panel
$t_{\text{web,mid}}$	0.00449	[-]	Normalized spar web thickness, middle panel
$t_{\text{web,in}}$	0.00353	[-]	Normalized spar web thickness, inboard panel
θ_{out}	1.44305	deg	Wing twist, outboard panel
θ_{mid}	0.55445	deg	Wing twist, middle panel
θ_{in}	-0.11632	deg	Wing twist, inboard panel

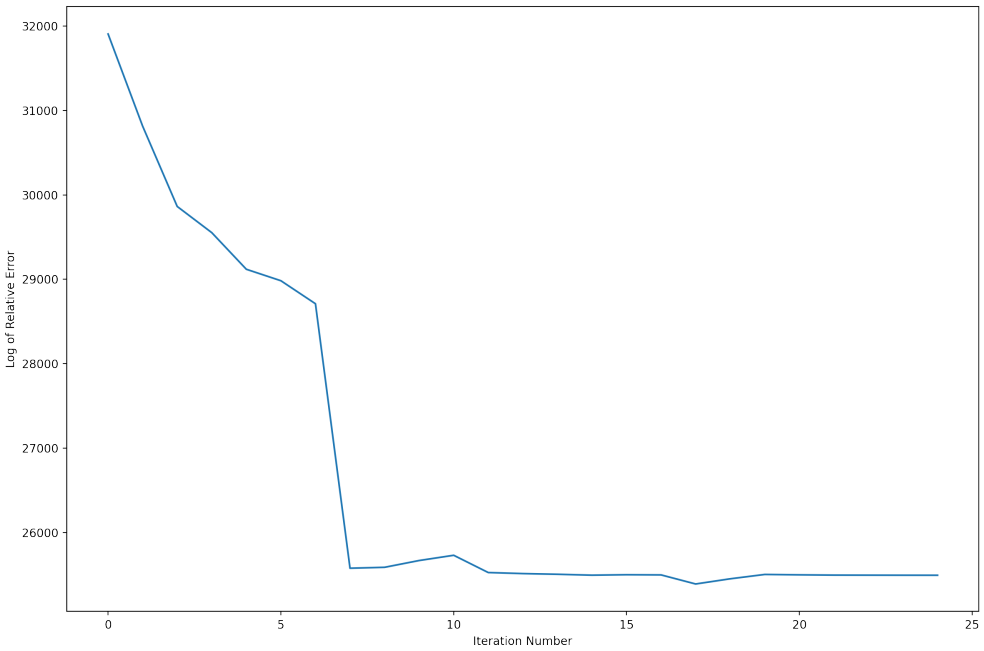


Figure 8-12: The history of the objective function (fuel weight) with iteration count for the third design phase

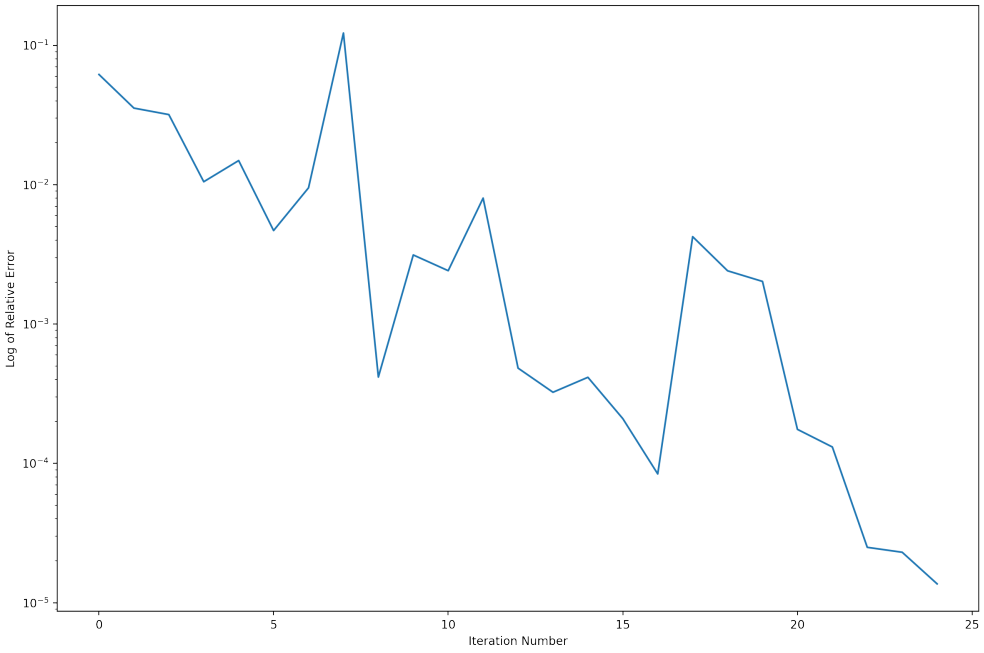


Figure 8-13: The history of the relative change in the objective function (fuel weight) with iteration count for the third design phase

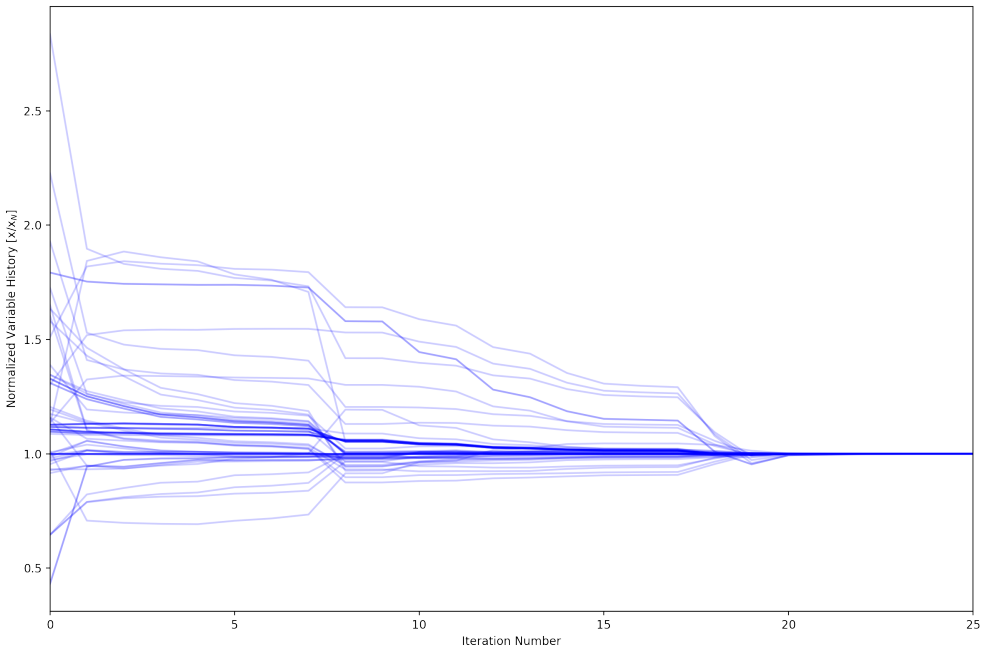
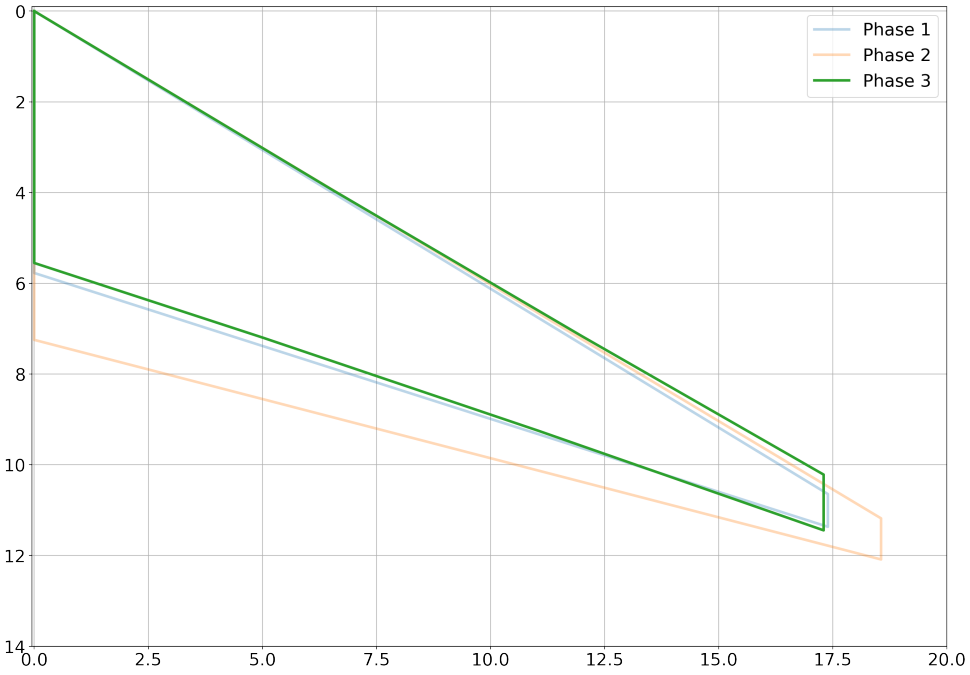
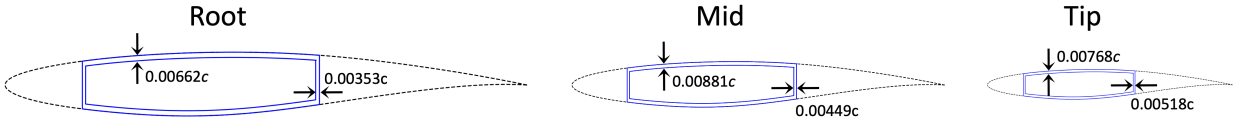


Figure 8-14: The history of the various design variables with iteration count for the third design phase

lower taper ratio and sweep. The thickness of the wing structures are sized at each panel to take maximum stress without failure, allowing for a lower overall wing weight. Though the wing weight here (13601 pounds) is somewhat lower than the SUGAR baseline (18728 pounds), the overall bending masses are more similar (8293 pounds vs 9621 pounds), and so though there are some model refinements that remain, assumptions for the empirical weight fractions from flaps, slats, skin, and others are likely another major source of error.



(a) Planform geometry



(b) Section geometry

Figure 8-15: The wing geometry results for the first design phase

8.5 Full Wing Optimization

Evolving to a combined airfoil and planform optimization was considered, but the convergence issues encountered in the planform design phase that stemmed from MSES failures, combined with the known concerns surrounding the noise of the MSES derivatives made it difficult to construct a meaningful test case. Future work could look at improving the reliability of MSES in this automated framework via warm starts and/or local surrogate models, but this is a significant undertaking that was deemed out of scope here. Alternatively, other tools exist that can be swapped in for MSES that may have the necessary reliability in an automated framework to successfully run this case.

8.6 Discussion of Final Results

Over the course of three design phases in this case study, new design variables are introduced by increasing analysis fidelity. The increased flexibility allows the SLCP algorithm to identify higher performing designs with each design phase. Airfoil optimization yielded a 13.4% reduction in fuel burn, and airfoil planform optimization yielded a 24.7% reduction in fuel burn (Figure 8-16).

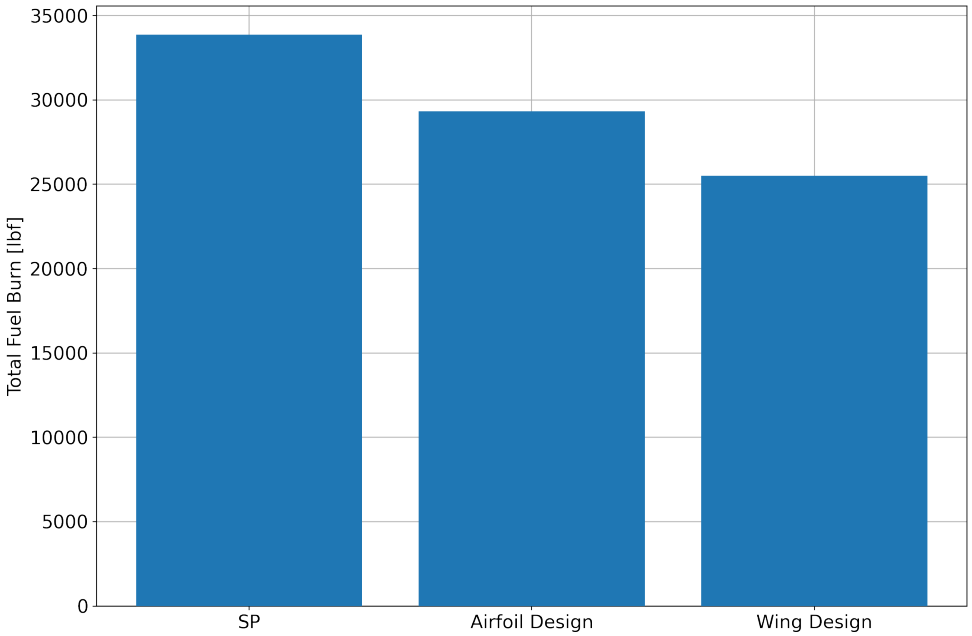


Figure 8-16: Evolution of the objective value during the three design phases

To be clear, this fuel burn reduction is not the result of some novel exploitation of physics, but only of expected design refinement. The result from the original SP is a poor design, and design evolution merely brings the design up to expectation.

8.7 Path Towards Higher Fidelity

From the results of each phase, it is clear that this test problem pushed the limits of the current implementation of SLCP, MSES, and CAPS, but it is worth noting that the optimization centered multifidelity design framework has operated as intended. The planform design

phase represents a significant increase in both geometry and analysis fidelity compared to the original signomial program. Though future work clearly remains in the implementation, this design example lays out a clear path towards integration of higher fidelity analysis tools like CFD and FEA.

Consider the final geometry used in the planform design phase (Figure 8-11). This geometry definition is essentially a simple airfoil stack, a commonly used geometry representation. The ESP and CAPS geometry framework [74] is capable of representing such airfoil stacks with arbitrary complexity (Figure 8-17). From these airfoil stacks, ESP/CAPS is capable of producing full 3D wing geometries, that can then be meshed for input to CFD and FEA tools (Figure 8-17).

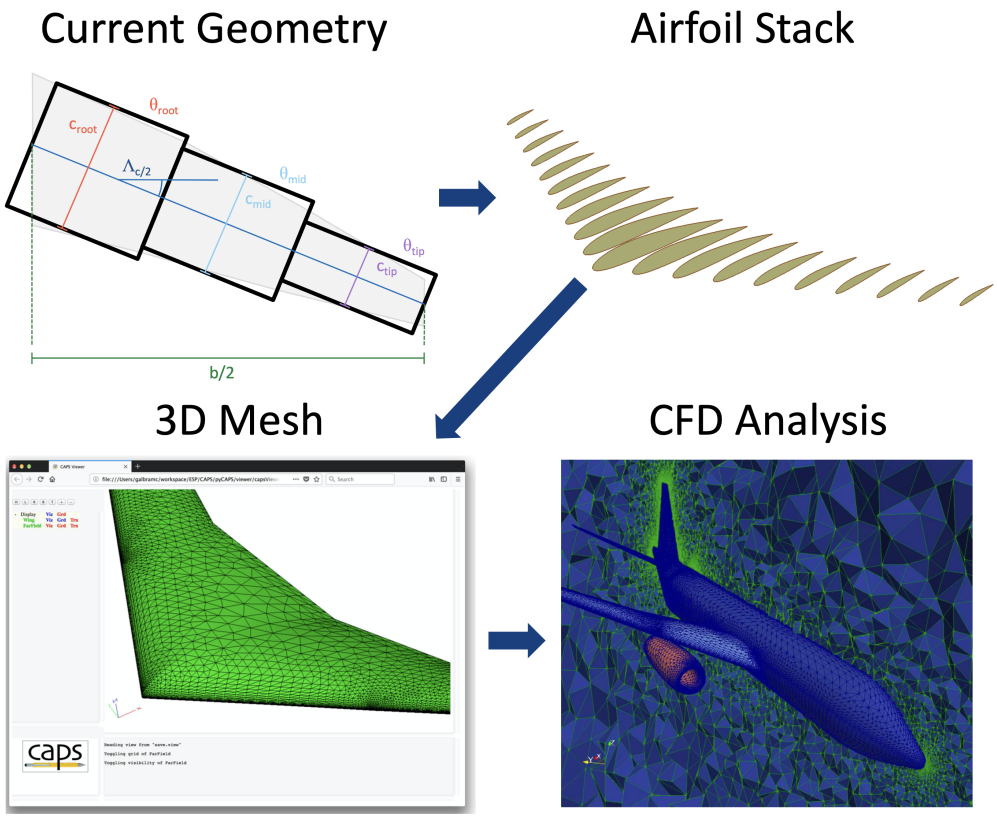


Figure 8-17: Path from Phase 3 geometry to high fidelity analysis. Images come from the CAPS Training Course, Summer 2020

Critically, the evolution from the current geometry definition to the geometry used in high

fidelity analysis does not require any change the topology of the underlying optimization problem. Put another way, the changes from the final design phase to a high fidelity CFD and FEA model are entirely contained within the black boxes that have already been integrated with the optimization framework. Though increasing the number of sections is advisable, as this would allow for continued wing refinement, this change would result in no meaningful change to the optimization problem beyond increased dimensionality.

But before this approach is seriously pursued, significant work remains in the area of analysis tool development. In general, two limitations in existing analysis tools prevent the kind of optimization integration demonstrated in this work. First, legacy tools often do not return derivatives with respect to the input geometry. Though recent decades have seen marked improvement thanks to a variety of computational tools, some of the most entrenched and trusted analysis tools simply cannot be brought up to this modern standard. Second, those tools that do return derivatives often rely solely on their own internal definition of geometry, which is rarely ever consistent with some kind of universally standard representation, let alone one that is useful for multifidelity design. Unlike the total absence of derivatives, this second deficiency can be overcome, as was done with the MSES implementation. But this approach is high-workload and in the end remains a band-aid solution even under the best circumstances.

Chapter 9

Future Work and Final Thoughts

9.1 Future Work

9.1.1 Improvements in Algorithm Implementation

One of the clearest areas for improvement on the work presented here is the implementation of the SLCP algorithm. Though not discussed in depth here, the SLCP algorithm used in this work represents a working prototype rather than true production quality code. Parsing of inputs is slow, any notion of sparsity is ignored, and derivatives to analytic functions are expensive to compute. Furthermore, the SLCP algorithm relies on a general convex solver (not a QP solver). While CVXOPT [30] did have this capability (some other solvers like Mosek [29] did not), extensive modifications had to be made to the software package to get it to work as desired. Ultimately, a dedicated SLCP algorithm similar to SNOPT [75], written in a fast compiled language with coupled inner and outer loop solves, must be developed if SLCP is to be used for any practical purpose.

9.1.2 Improvements in SLCP Theory

The SLCP theory also has a two key opportunities for improvement. First is the generalization of the SLCP sub-problem objective function. The current sub-problem objective function is quadratic, but any log-convex function can theoretically be used. But the path forward here is somewhat unclear, as the BFGS updates in SQP, LSQP, and SLCP serves both to improve the approximation of the Hessian and to ensure the objective function stays convex. Reproducing this method for a posynomial function, or a more general convex function, would be challenging.

Second is the increased ability to recognize general log-log convex constraints. Work by Agrawal [12] shows that the family of log-log convex functions is much larger than just monomial and posynomial functions. Significant gains could be made if the software implementation of SLCP could identify all of these log-log convex constraints and pass these directly into the convex sub-problem. Though not challenging from the theoretical perspective, implementation of this would require a more extensive software package than currently exists.

9.1.3 Considerations for Future Analysis Models

Working in this highly automated framework for design optimization has uncovered some a few key principles when thinking about the analysis models of the future. First, the analysis models of the future must be designed from the beginning to return high quality gradients. Though substantial progress has been made in this area, noisy gradients and finite difference approaches are simply not good enough for the future of aircraft design.

Second, analysis models of the future would ideally be designed to fail gracefully, or else have logic that can handle failures in some meaningful way. Numerical optimization is a fairly brutal way to test analysis models, the adage being that optimization is not so much good for finding optimal designs, but it is great at breaking your analysis tools. Even using MSES in this work, which on the whole is an excellent tool, even moderate failure rates had large

consequences for the ability to converge to optimal designs.

Third, analysis models of the future should have clear and quantifiable uncertainty functions. Consider Figure 9-1 which comes from Drela [73]. This chart shows qualitatively the regions of expected validity for a number of different aerodynamic models, ranging from the Laplace equations through Full Potential models. But in actuality, this plot is essentially just selecting a single contour of maximum acceptable uncertainty from a contour plot like Figure 9-2.

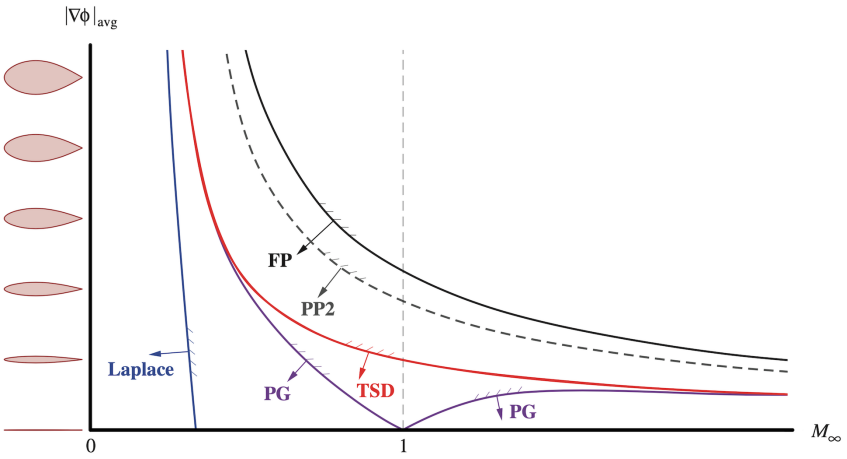


Figure 9-1: Qualitative plot of the validity of certain aerodynamic models given the airfoil thickness (geometry) and Mach number (flow parameter) [73]

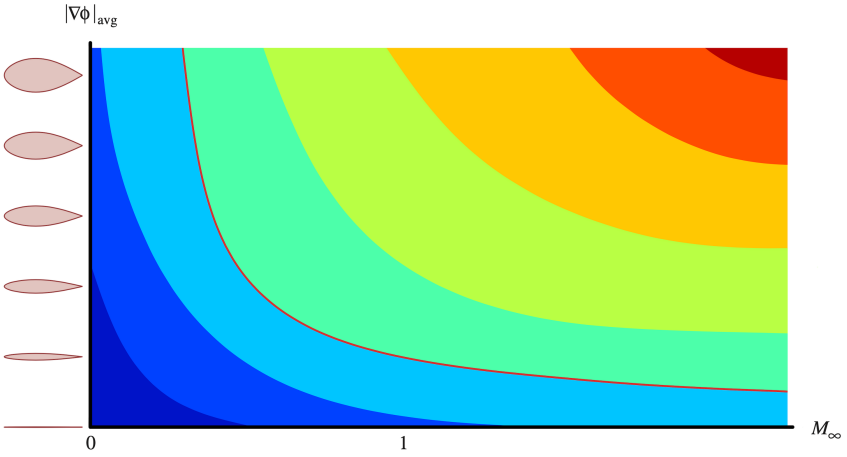


Figure 9-2: Qualitative contour plot of the uncertainty of the TSD model given the airfoil thickness (geometry) and Mach number (flow parameter)

All analysis models have an uncertainty function like this, though few if any have actually been quantified. At best, an experienced subject matter expert might be able to give vague notions of where they might begin to not trust the result, much like Figure 9-1. But as designs become more innovative and more integrated, the conventional wisdom here will start to break down. By investing research time and money in these uncertainty functions, it will be far easier to understand the risks of new and innovative designs earlier in the design process.

9.1.4 Uncertainty Propagation

If uncertainty functions like those mentioned above can be developed, either by further research efforts or by surveys of experts, then SLCP can be utilized to provide quantitative information on when model fidelity should be increased. Understanding when to increase fidelity is important, because for some applications like UAV design, the GP and SP frameworks can likely provide sufficient fidelity for the entire design cycle, a process demonstrated on Jungle Hawk Owl [49]. But for design of subsonic transport aircraft and other high capital ventures, high fidelity analysis will still be necessary.

The general agreement is that some combination of *uncertainty* and *risk* are traded off to decide if higher fidelity is necessary. The field of Uncertainty Quantification (UQ) is large, but it shrinks somewhat when considering application to aircraft design and multi-fidelity modeling [76, 77, 78] and a recent survey article [65] gives a good entry point to this field.

As it currently stands, state of the art is still Monte Carlo sampling of uncertain models to obtain distributions that are then propagated through, once again using Monte Carlo. This process is extremely computationally expensive, something mentioned by some of the literature [77].

But to give a general direction, consider the following very simple example optimization

problem:

$$\begin{aligned} \underset{x,y}{\text{minimize}} \quad & y \\ \text{subject to} \quad & y \geq x^2 + u_1(x,y) \\ & x \geq 1 + u_2(x,y) \end{aligned}$$

which has uncertainties associated with the two constraints or 'analysis models'. First, assign equal uncertainty distributions to both u_1 and u_2 and perform a Monte Carlo sampling. Next, set u_1 to zero (ie, assume perfect analysis) and then sample again. Finally, return the original distribution of u_1 and instead set u_2 to be zero. This will result in three distributions, seen here:

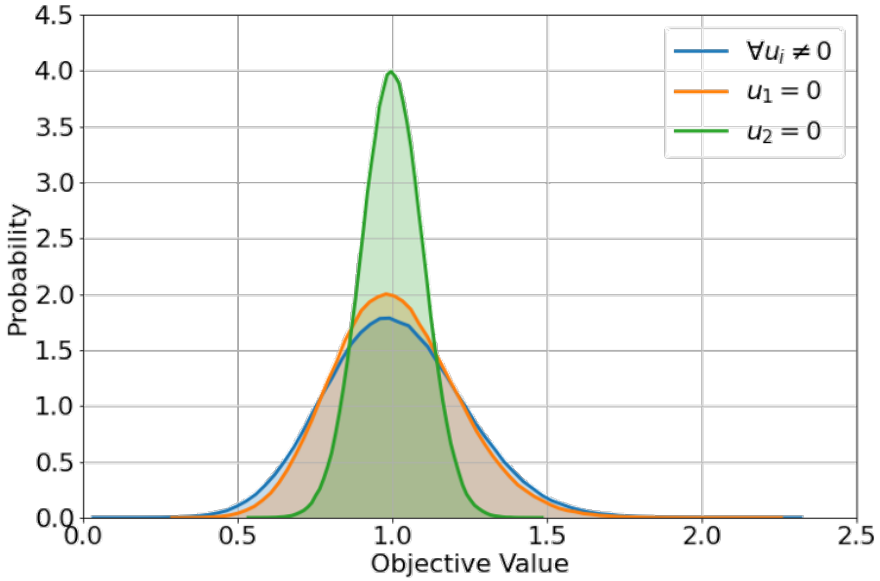


Figure 9-3: Varying uncertainty in a simple design problem, first with both uncertainties set to a standard deviation of $\sigma = 0.1$ (blue), then with u_1 set to 0 (orange), and finally u_2 set to 0 (green)

Setting u_2 to 0 has a dramatic affect on the uncertainty of the final objective, while setting $u_1 = 0$ does not, and so the fidelity of the constraint containing u_2 should be increased. Decreasing the standard deviation of u_2 to be $\sigma = 0.05$ (ie, increasing the fidelity of the constraint) and re-running the analysis results in the following:

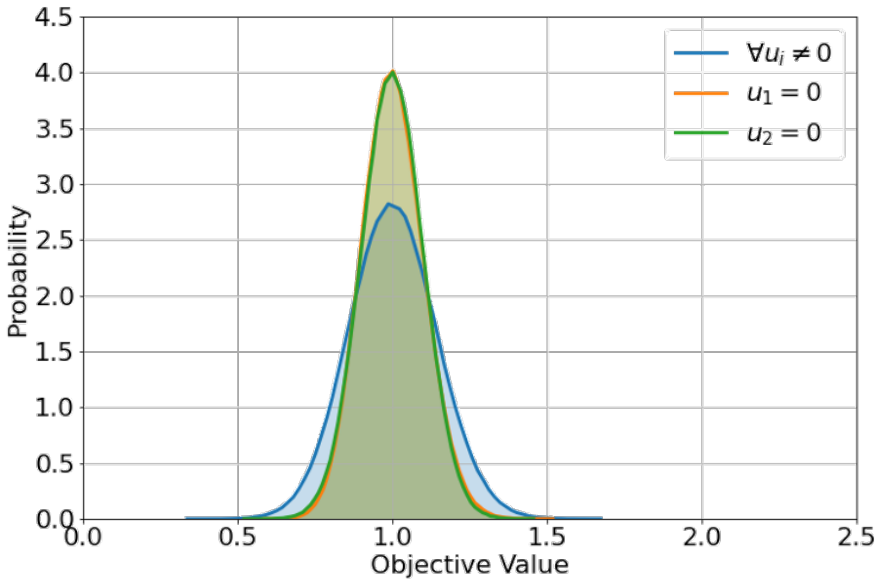


Figure 9-4: Varying uncertainty in a simple design problem, first with u_1 a standard deviation of $\sigma = 0.1$ and u_2 a standard deviation of $\sigma = 0.05$ (blue), then with u_1 set to 0 (orange), and finally u_2 set to 0 (green)

In this case, setting the uncertainties to zero had approximately the same effect, and so improvement on the final objective uncertainty must be achieved by improving fidelity of both models.

When using SLCP, the final sub-problem is a good local surrogate for the aircraft design space that due to its convexity, can be run repeatedly for minimal computational expense. This final convex sub-problem could be utilized in a similar way to this simple example to provide human decision makers with quantitative information on how much risk the optimal design has in meeting performance targets, and which models are the primary contributors to that risk.

9.2 Contributions

This work has resulted in two novel contributions to the state of the art in optimization and aircraft design.

Prior to this work, Geometric and Signomial Programming were novel and powerful tools for aircraft design, but the limited mathematical form capped the potential for these methods when it came to practical engineering design efforts. The LSQP and SLCP algorithms developed in this work remove this ceiling by enabling the integration of black box analysis tools necessary in MDAO, while still taking maximum advantage of the log-log convex structure revealed by GP and SP formulations.

Furthermore, the SLCP algorithm represents a novel step forward in optimization algorithms as the first known gradient based approach to solving constrained continuous non-linear optimization problems with generalized Sequential Convex Programming. Though preliminary, this work opens the door for substantial improvements over existing algorithms (ex. SQP) when combined with the significant body of work being done to identify convexity in engineering design problems.

These two contributions are the keystones to constructing the optimization first approach to multifidelity design that was described and demonstrated in Chapters 6 through 8. Though this framework is not the only viable approach to multifidelity design, the process demonstrated here is uniquely positioned to take maximum advantage of the GP and SP compatibility present in many aircraft design problems. Existing literature establishes a clear path to developing conceptual design tools in a GP or SP compatible form without meaningful sacrifices to analysis fidelity. As the design evolves and analysis fidelity must be increased, constraints in the GP or SP formulation can be swapped for higher fidelity analysis black boxes, and the SLCP algorithm can be utilized for the remainder of the design process. The general nature of SLCP enables this swapping across a wide range of disciplines and fidelities, thereby allowing the design process to be viewed as a multifidelity and multidisciplinary

continuum rather than as a series of major discrete phases. When design refinements are desired, new variables that are available in higher fidelity analysis tools can be exposed to the optimization problem to be traded in the design. And when new information is discovered that triggers a re-design, low fidelity models can be swapped back in to reduce the computational cost of the new cycle.

9.3 Final Thoughts

The beauty in the optimization first approach to design is not so much in that it is radically innovative, rather that it combines a decades old optimization algorithm, modern computational resources, and an idea for aircraft design previously rejected as not practical. The success of Geometric Programming forced the aircraft design community to grapple with this practical SAND approach to design and raise questions regarding the fundamental tenants of MDAO and of aircraft design more generally.

Section 1.1 began with a fundamental assumption of the presence of an analysis oracle. Everything else in Chapter 1 built on the foundation of this singular concept, from the notion of design optimization, to the nature of analysis models, and the entry point for the field of MDAO. A clear separation was made between the tasks of the automated numerical optimization loop, which merely suggested a steady stream of incrementally improving feasible designs, and the aircraft designer, who guided the higher level design optimization tasks.

In Section 2.1, the process instead begins by casting aircraft design as a numerical optimization problem without any notion of an oracle or an analysis model. In this approach, analysis models are a means to ensuring that the eventual set of optimal design decisions (variables) obey the laws of physics. Rather remarkably, this approach eliminated the boundary between design optimization and numerical optimization. In the optimization first approach, design optimization *is* numerical optimization.

In the modern era, computers are indisputably better than the human brain at brute force computation. Therefore, for well conditioned numerical optimization problems of non-trivial difficulty, a computer algorithm will always out-perform a human at finding the optimal solution. If the aircraft design problem can be written as a numerical optimization in its entirety, then there will come a day that computer programs are capable of producing higher performing aircraft designs in less time than a human design team.

But that day is not today. There are still too many factors in aircraft design that cannot be modeled in the form of a numerical optimization problem. Chief among them is that defining the requirements for an aircraft design program is often an iterative process that depend on a number of non-quantifiable factors like company fiscal performance, market opportunity, and prevailing societal attitudes. But the optimization first approach opens the gate to a path where the final destination is a fully automated design optimization tool, and in the meantime, research efforts should focus on developing methods that adapt the optimization first approach for immediate use in academia, industry, and government sectors. The Logspace Sequential Quadratic Programming (LSQP) and Sequential Log-Convex Programming (SLCP) algorithms are key steps forward in this pursuit, and will enable rigorously quantitative design tools that assist designers in creating the aircraft that will soar across the skies of the future.

Bibliography

[1] Nocedal, J. and Wright, S. J., *Numerical Optimization*, Springer New York, 2006.

[2] Martins, J. R. R. A. and Ning, A., *Engineering Design Optimization*, Cambridge University Press, November 2021.

[3] Martins, J. R. R. A. and Lambe, A. B., "Multidisciplinary Design Optimization: A Survey of Architectures," *AIAA Journal*, Vol. 51, No. 9, September 2013, pp. 2049–2075.

[4] Cramer, E. J., Dennis, Jr., J. E., Frank, P. D., Lewis, R. M., and Shubin, G. R., "Problem Formulation for Multidisciplinary Optimization," *SIAM Journal on Optimization*, Vol. 4, No. 4, November 1994, pp. 754–776.

[5] Kroo, I., "A Quasi-Procedural, Knowledge-Based System for Aircraft Design," *Aircraft Design, Systems and Operations Conference*, American Institute of Aeronautics and Astronautics, Reston, Virigina, September 1988, pp. 1–10.

[6] Wakayama, S., "Multidisciplinary Design Optimization of the Blended-Wing-Body," *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, American Institute of Aeronautics and Astronautics, Reston, Virigina, September 1998, pp. 1–9.

[7] Drela, M., "Development of the D8 Transport Configuration," *29th AIAA Applied Aerodynamics Conference*, American Institute of Aeronautics and Astronautics, Reston, Virigina, June 2011, pp. 1–14.

[8] Lukaczyk, T. W., Wendorff, A. D., Colonno, M., Economon, T. D., Alonso, J. J., Orra, T. H., and Ilario, C., "SUAVE: An Open-Source Environment for Multi-Fidelity Conceptual Vehicle Design," *16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, American Institute of Aeronautics and Astronautics, Reston, Virginia, June 2015, pp. 1–56.

[9] Anderson, J. D., *Aircraft Performance and Design*, McGraw Hill, Boston, Massachusetts, 1999.

[10] Hoburg, W. and Abbeel, P., "Geometric Programming for Aircraft Design Optimization," *AIAA Journal*, Vol. 52, No. 11, November 2014, pp. 2414–2426.

[11] Boyd, S., Kim, S.-J., Vandenberghe, L., and Hassibi, A., "A Tutorial on Geometric Programming," *Optimization and Engineering*, Vol. 8, No. 1, May 2007, pp. 67–127.

[12] Agrawal, A., Diamond, S., and Boyd, S., "Disciplined Geometric Programming," *Optimization Letters*, Vol. 13, No. 5, July 2019, pp. 961–976.

[13] Clasen, R. J., "The Solution of the Chemical Equilibrium Programming Problem with Generalized Benders Decomposition," *Operations Research*, Vol. 32, No. 1, February 1984, pp. 70–79.

[14] Greenberg, H. J., "Mathematical Programming Models for Environmental Quality Control," *Operations Research*, Vol. 43, No. 4, August 1995, pp. 578–622.

[15] Boyd, S. P., Kim, S.-J., Patil, D. D., and Horowitz, M. A., "Digital Circuit Optimization Via Geometric Programming," *Operations Research*, Vol. 53, No. 6, December 2005, pp. 899–932.

[16] del Mar Hershenson, M., Boyd, S. P., and Lee, T. H., "Optimal Design of a CMOS Op-Amp Via Geometric Programming," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 20, No. 1, 2001, pp. 1–21.

[17] Li, X., Gopalakrishnan, P., Xu, Y., and Pileggi, L. T., "Robust Analog/RF Circuit Design with Projection-Based Posynomial Modeling," *IEEE/ACM International Conference on Computer Aided Design*, IEEE, 2004, pp. 855–862.

[18] Xu, Y., Pileggi, L. T., and Boyd, S. P., "ORACLE: Optimization with Recourse of Analog Circuits Including Layout Extraction," *41st Annual Conference on Design Automation*, ACM Press, New York, New York, USA, June 2004, pp. 151–154.

[19] Jabr, R. A., "Application of Geometric Programming To Transformer Design," *IEEE Transactions on Magnetics*, Vol. 41, No. 11, November 2005, pp. 4261–4269.

[20] Chiang, M., *Geometric Programming for Communication Systems*, now Publishers Inc, 2005.

[21] Chiang, M., Tan, C. W., Palomar, D. P., O'neill, D., and Julian, D., "Power Control By Geometric Programming," *IEEE Transactions on Wireless Communications*, Vol. 6, No. 7, July 2007, pp. 2640–2651.

[22] Kandukuri, S. and Boyd, S., "Optimal Power Control in Interference-Limited Fading Wireless Channels with Outage-Probability Specifications," *IEEE Transactions on Wireless Communications*, Vol. 1, No. 1, 2002, pp. 46–55.

[23] Marin-Sanguino, A., Voit, E. O., Gonzalez-Alcon, C., and Torres, N. V., "Optimization of Biotechnological Systems Through Geometric Programming." *Theoretical Biology & Medical Modelling*, Vol. 4, September 2007, pp. 38.

[24] Vera, J., González-Alcón, C., Marín-Sanguino, A., and Torres, N., "Optimization of Biochemical Systems Through Mathematical Programming: Methods and Applications," *Computers & Operations Research*, Vol. 37, No. 8, August 2010, pp. 1427–1438.

[25] Preciado, V. M., Zargham, M., Enyioha, C., Jadbabaie, A., and Pappas, G. J., "Optimal Resource Allocation for Network Protection Against Spreading Processes," *IEEE Transactions on Control of Network Systems*, Vol. 1, No. 1, March 2014, pp. 99–108.

[26] Misra, S., Fisher, M. W., Backhaus, S., Bent, R., Chertkov, M., and Pan, F., "Optimal Compression in Natural Gas Networks: A Geometric Programming Approach," *IEEE Transactions on Control of Network Systems*, Vol. 2, No. 1, March 2015, pp. 47–56.

[27] Sela Perelman, L. and Amin, S., "Control of Tree Water Networks: A Geometric Programming Approach," *Water Resources Research*, Vol. 51, No. 10, October 2015, pp. 8409–8430.

[28] Torenbeek, E., *Advanced Aircraft Design: Conceptual Design, Analysis and Optimization of Subsonic Civil Airplanes*, John Wiley & Sons, Ltd, 2013.

[29] MOSEK ApS, *The MOSEK Optimization Toolbox for Python Manual. Version 9.3.13.*, 2022.

[30] Andersen, M., Dahl, J., Liu, Z., and Vandenberghe, L., "Interior-Point Methods for Large-Scale Cone Programming," *Optimization for Machine Learning*, edited by S. Sra, S. Nowozin, and S. J. Wright, The MIT Press, 2012, pp. 55–84.

[31] Boyd, S. P. and Vandenberghe, L., *Convex Optimization*, Cambridge University Press, 7th ed., 2009.

[32] Bertsimas, D., "15.093J/6.255J Optimization Methods," *Massachusetts Institute of Technology: MIT OpenCouseWare, https://ocw.mit.edu/. License: Creative Commons BY-NC-SA*, Fall 2009.

[33] Hoburg, W. and Abbeel, P., "Fast Wind Turbine Design Via Geometric Programming," *54th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, American Institute of Aeronautics and Astronautics, Reston, Virginia, April 2013, pp. 1–9.

[34] Maranas, C. D. and Floudas, C. A., "Global Optimization in Generalized Geometric Programming," *Computers & Chemical Engineering*, Vol. 21, No. 4, December 1997, pp. 351–369.

[35] Kirschen, P. G., Burnell, E., and Hoburg, W., "Signomial Programming Models for Aircraft Design," *54th AIAA Aerospace Sciences Meeting*, American Institute of Aeronautics and Astronautics, Reston, Virginia, January 2016, pp. 1–26.

[36] York, M. A., Öztürk, B., Burnell, E., and Hoburg, W. W., "Efficient Aircraft Multidisciplinary Design Optimization and Sensitivity Analysis Via Signomial Programming," *AIAA Journal*, Vol. 56, No. 11, November 2018, pp. 4546–4561.

[37] Brown, A. and Harris, W., "A Vehicle Design and Optimization Model for On-Demand Aviation," *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, American Institute of Aeronautics and Astronautics, Reston, Virginia, January 2018, pp. 1–46.

[38] Burton, M. and Hoburg, W., "Solar and Gas Powered Long-Endurance Unmanned Aircraft Sizing Via Geometric Programming," *Journal of Aircraft*, Vol. 55, No. 1, January 2018, pp. 212–225.

[39] Lin, B., Carpenter, M., and de Weck, O., "Simultaneous Vehicle and Trajectory Design Using Convex Optimization," *AIAA Scitech 2020 Forum*, American Institute of Aeronautics and Astronautics, Reston, Virginia, January 2020, pp. 1–18.

[40] Kirschen, P. G., York, M. A., Ozturk, B., and Hoburg, W. W., "Application of Signomial Programming To Aircraft Design," *Journal of Aircraft*, Vol. 55, No. 3, May 2018, pp. 965–987.

[41] York, M. A., Hoburg, W. W., and Drela, M., "Turbofan Engine Sizing and Tradeoff Analysis Via Signomial Programming," *Journal of Aircraft*, Vol. 55, No. 3, May 2018, pp. 988–1003.

[42] Saab, A., Burnell, E., and Hoburg, W. W., "Robust Designs Via Geometric Programming," *arXiv*, 2018.

[43] Hall, D. K., Dowdle, A., Gonzalez, J., Trollinger, L., and Thalheimer, W., "Assessment of a Boundary Layer Ingesting Turboelectric Aircraft Configuration Using Signomial Programming," *2018 Aviation Technology, Integration, and Operations Conference*, American Institute of Aeronautics and Astronautics, Reston, Virginia, June 2018, pp. 1–16.

[44] Opgenoord, M. M. J., Cohen, B. S., and Hoburg, W. W., "Comparison of Algorithms for Including Equality Constraints in Signomial Programming," Technical Report ACDL TR-2017-1, Massachusetts Institute of Technology, 2017.

[45] Burnell, E., Damen, N. B., and Hoburg, W., "GPkit: A Human-Centered Approach To Convex Optimization in Engineering Design," *Proceedings of the 2020 CHI Conference*

on Human Factors in Computing Systems, ACM, New York, NY, USA, April 2020, pp. 1–13.

[46] Drela, M., "XFOIL: An Analysis and Design System for Low Reynolds Number Airfoils," *Low Reynolds Number Aerodynamics*, edited by T. J. Mueller, Springer Berlin Heidelberg, 1989, pp. 1–12.

[47] Hoburg, W., Kirschen, P., and Abbeel, P., "Data Fitting with Geometric-Programming-Compatible Softmax Functions," *Optimization and Engineering*, Vol. 17, No. 4, December 2016, pp. 897–918.

[48] Karcher, C. J., "Data Fitting with Signomial Programming Compatible Difference of Convex Functions," *Optimization and Engineering*, April 2022.

[49] Ozturk, B., Burton, M. J., Zarse, O., Hoburg, W. W., Drela, M., and Hansman, J., "Design of an Unmanned Aerial Vehicle for Long-Endurance Communication Support," *18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, American Institute of Aeronautics and Astronautics, Reston, Virginia, June 2017, pp. 1–18.

[50] Park, J. and Boyd, S., "General Heuristics for Nonconvex Quadratically Constrained Quadratic Programming," *arXiv*, 2017, pp. 1–63.

[51] Kirschen, P. G. and Hoburg, W. W., "The Power of Log Transformation: A Comparison of Geometric and Signomial Programming with General Nonlinear Programming Techniques for Aircraft Design Optimization," *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, American Institute of Aeronautics and Astronautics, Reston, Virginia, January 2018, pp. 1–23.

[52] Karcher, C. J., "Logspace Sequential Quadratic Programming for Design Optimization," *AIAA Journal*, Vol. 60, No. 3, March 2022, pp. 1471–1481.

[53] Boyd, S. P., "Sequential Convex Programming," *Stanford EE364b Lecture Notes*, 2015.

[54] Duchi, J., Boyd, S., and Mattingley, J., "Sequential Convex Programming," *Lecture Notes EE364b*, 2018.

[55] Anitescu, M., "A Superlinearly Convergent Sequential Quadratically Constrained Quadratic Programming Algorithm for Degenerate Nonlinear Programming," *SIAM Journal on Optimization*, Vol. 12, No. 4, January 2002, pp. 949–978.

[56] Tang, C.-M. and Jian, J.-B., "A Sequential Quadratically Constrained Quadratic Programming Method with an Augmented Lagrangian Line Search Function," *Journal of Computational and Applied Mathematics*, Vol. 220, No. 1-2, October 2008, pp. 525–547.

[57] Liu, M. X., Jian, J. B., and Tang, C. M., "A Method Combining Norm-Relaxed QCQP Subproblems with Active Set Identification for Inequality Constrained Optimization," *Optimization*, September 2020, pp. 1–31.

[58] Jian, J., Liu, P., Yin, J., Zhang, C., and Chao, M., "A QCQP-based Splitting SQP Algorithm for Two-Block Nonconvex Constrained Optimization Problems with Application," *Journal of Computational and Applied Mathematics*, Vol. 390, July 2021, pp. 113368.

[59] Boggs, P. T. and Tolle, J. W., "Sequential Quadratic Programming," *Acta Numerica*, Vol. 4, January 1995, pp. 1–51.

[60] Karcher, C. and Haimes, R., "A Method of Sequential Log-Convex Programming for Engineering Design," *Optimization and Engineering*, July 2022.

[61] Floudas, C. A., Pardalos, P. M., Adjiman, C. S., Esposito, W. R., Gümüş, Z. H., Harding, S. T., Klepeis, J. L., Meyer, C. A., and Schweiger, C. A., *Handbook of Test Problems in Local and Global Optimization*, Springer US, Boston, MA, 1999.

[62] Gould, N. I. M., Orban, D., and Toint, P. L., "CUTEst: a Constrained and Unconstrained Testing Environment with Safe Threads for Mathematical Optimization," *Computational Optimization and Applications*, Vol. 60, No. 3, April 2015, pp. 545–557.

[63] Kraft, D., "A Software Package for Sequential Quadratic Programming," Technical Report DFVLR-FB 88-28, Institut für Dynamik der Flugsysteme, 1988.

[64] Reuther, A., Kepner, J., Byun, C., Samsi, S., Arcand, W., Bestor, D., Bergeron, B., Gadepally, V., Houle, M., Hubbell, M., Jones, M., Klein, A., Milechin, L., Mullen, J., Prout, A., Rosa, A., Yee, C., and Michaleas, P., "Interactive Supercomputing on 40,000 Cores for Machine Learning and Data Analysis," *2018 IEEE High Performance Extreme Computing Conference (HPEC)*, IEEE, September 2018, pp. 1–6.

[65] Peherstorfer, B., Willcox, K., and Gunzburger, M., "Survey of Multifidelity Methods in Uncertainty Propagation, Inference, and Optimization," *SIAM Review*, Vol. 60, No. 3, January 2018, pp. 550–591.

[66] Drela, M., "Pros & Cons of Airfoil Optimization," *Frontiers of Computational Fluid Dynamics 1998*, World Scientific, 1998, pp. 363–381.

[67] Lam, R., Allaire, D. L., and Willcox, K. E., "Multifidelity Optimization Using Statistical Surrogate Modeling for Non-Hierarchical Information Sources," *56th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, American Institute of Aeronautics and Astronautics, Reston, Virginia, January 2015, pp. 1–21.

[68] Drela, M., "A User's Guide To MSES 3.05," Technical Report, Massachussets Institute of Technology (MIT), 2007.

[69] Alyanak, E., Durscher, R., Haimes, R., Dannenhoffer, J., Bhagat, N. D., and Allison, D. L., "Multi-Fidelity Geometry-Centric Multi-Disciplinary Analysis for Design," *AIAA Modeling and Simulation Technologies Conference*, American Institute of Aeronautics and Astronautics, Reston, Virginia, June 2016, pp. 1–23.

[70] Kulfan, B. M., "Universal Parametric Geometry Representation Method," *Journal of Aircraft*, Vol. 45, No. 1, January 2008, pp. 142–158.

[71] Bradley, M. K. and Droney, C. K., "Subsonic Ultra Green Aircraft Research: Phase I Final Report," Technical Report NASA/CR-2011-216847, National Aeronautics and Space Administration (NASA), 2011.

[72] Greitzer, E. M., Bonnefoy, P., De la Rosa Blanco, E., Dorbian, C., Drela, M., Hall, D. K., Hansman, R. J., Hileman, J., Liebeck, R. H., Lovegren, J., Mody, P., Pertuze, J., Sato, S., Spakovszky, Z. S., Tan, C., Hollman, J., Duda, J., Fitzgerald, N., Houghton, J., Kerrebrock, J. L., Kiwada, G., Kordonowy, D., Parrish, J., Tylko, J., Wen, E., and Lord, W., "N+3 Aircraft Concept Designs and Trade Studies, Final Report: Volume II—Design Methodologies for Aerodynamics Structures Weights and Thermodynamic Cycles," Technical Report NASA/CR-2010-216794/VOL2, National Aeronautics and Space Administration (NASA), 2010.

[73] Drela, M., *Flight Vehicle Aerodynamics*, The MIT Press, 2014.

[74] Haimes, R. and Dannenhoffer, J., "The Engineering Sketch Pad: A Solid-Modeling, Feature-Based, Web-Enabled System for Building Parametric Geometry," *21st AIAA Computational Fluid Dynamics Conference*, American Institute of Aeronautics and Astronautics, Reston, Virginia, June 2013, pp. 1–21.

[75] Gill, P. E., Murray, W., and Saunders, M. A., "SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization," *SIAM Review*, Vol. 47, No. 1, January 2005, pp. 99–131.

[76] Chaudhuri, A., Lam, R., and Willcox, K., "Multifidelity Uncertainty Propagation Via Adaptive Surrogates in Coupled Multidisciplinary Systems," *AIAA Journal*, Vol. 56, No. 1, January 2018, pp. 235–249.

[77] Ng, L. W. T. and Willcox, K. E., "Monte Carlo Information-Reuse Approach To Aircraft Conceptual Design Optimization Under Uncertainty," *Journal of Aircraft*, Vol. 53, No. 2, March 2016, pp. 427–438.

[78] Opgenoord, M. M. J. and Willcox, K. E., "Sensitivity Analysis Methods for Uncertainty Budgeting in System Design," *AIAA Journal*, Vol. 54, No. 10, October 2016, pp. 3134–3148.

[79] Hartman, P., "On Functions Representable As a Difference of Convex Functions," *Pacific Journal of Mathematics*, Vol. 9, No. 3, September 1959, pp. 707–713.

[80] MathWorks, *Constrained Nonlinear Optimization Algorithms*, 2022.

[81] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mane, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viegas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems," *arXiv*, November 2015.

[82] Fletcher, R. and Powell, M. J. D., "A Rapidly Convergent Descent Method for Minimization," *The Computer Journal*, Vol. 6, No. 2, August 1963, pp. 163–168.

Appendix A

A Brief Summary of Signomial Programming Compatible Fitting Methods

A.1 Difference of Convex (DC) Functions

Most continuous functions[1] ($f(\mathbf{x})$) can be written as the difference of two convex functions ($g(\mathbf{x})$ and $h(\mathbf{x})$) [79]:

$$f(\mathbf{x}) = g(\mathbf{x}) - h(\mathbf{x}) \tag{A.1}$$

Functions of the form in Equation A.1 are said to be Difference of Convex (DC) functions.

By extension, it follows that most datasets that can be well approximated by a continuous function should also be well approximated by the difference between two convex functions. In other words, if there is some continuous function (or more precisely some function of

[1] The precise statement in Hartman [79] is that any function of bounded variation can be decomposed into the difference between two convex functions. For practical engineering purposes, this translates to most differentiable and non-differentiable continuous functions.

bounded variation) $f(\mathbf{x})$ that fits some data set sampled from a mapping from $\mathbb{R}^N \to \mathbb{R}$, then there also exist functions $g(\mathbf{x})$ and $h(\mathbf{x})$ such that $g(\mathbf{x})$ and $h(\mathbf{x})$ are both convex and $g(\mathbf{x}) - h(\mathbf{x})$ is an equally good fit to the mapping as the original function $f(\mathbf{x})$. Taking functions $g(\mathbf{x})$ and $h(\mathbf{x})$ to be log-log convex functions such as those proposed in Hoburg et. al. [47], it is possible to fit approximations for these functions $g(\mathbf{x})$ and $h(\mathbf{x})$ to data sets from mappings which are not log-log convex.

A.2 Notation

Consider a data set sampled from a black box mapping from $\mathbb{R}^N \to \mathbb{R}$. Consistent with the notation in Hoburg et. al. [47] let the vector $\mathbf{u}_j$ represent the independent variables in $\mathbb{R}^N$ for data point j and the scalar w_j represent the output in $\mathbb{R}$ for data point j. The log-log space variables are then represented as $\mathbf{x}_j = \log \mathbf{u}_j$ and $y_j = \log w_j$.

A.3 Difference of Max Affine (DMA) Functions

The first function proposed by Hoburg et. al. [47] is the Max Affine (MA) function:

$$f_{\text{MA}}(\mathbf{x}) = \max_{k=1\ldots K} \left[b_k + \mathbf{a}_k^{\text{T}} \mathbf{x} \right] \tag{A.2}$$

This function class is known to create a convex epigraph. In fact, any convex function can be reasonably approximated as a Max Affine function given a sufficient number of affine functions, K [32]. Upon transformation back to variables $\mathbf{u}_j$ and w_j, the Max Affine function becomes Max Monomial, which can be implemented as a set of monomial constraints in the Geometric Program.

Now consider the difference between two of these Max Affine functions (Equation A.2), which

henceforth will be called the Difference of Max Affine (DMA) function:

$$f_{\text{DMA}}(\mathbf{x}) = \max_{k=1\ldots K} \left[b_k + \mathbf{a}_k^{\text{T}}\mathbf{x}\right] - \max_{m=1\ldots M} \left[h_m + \mathbf{g}_m^{\text{T}}\mathbf{x}\right] \tag{A.3}$$

where subtracted term is represented by an entirely separable Max Affine function which is defined by fitting parameters M, h, and $\mathbf{g}$. Based on an understanding of DC functions and the fact that convex functions can be well approximated as Max Affine for large K or M, this DMA function should be highly versatile as a fitting function.

While the Max Affine function has a realizable transformation back from log-log space, the Difference of Max Affine function has no such transformation due to the inability to readily construct a meaningful epigraph or subgraph using a set of separable inequalities, making it somewhat impractical for use in optimization. Despite this, the DMA function is quite rapid to fit, and could be of application in other areas where a cheap surrogate is desired for non-convex fitting. Here, the DMA function is a used as an intellectual building block to the next function class, and as a seed function for the fitting process.

A.4 Difference of Softmax Affine (DSMA) Functions

The second function proposed by Hoburg et. al. [47] is the Softmax Affine (SMA) function:

$$f_{\text{SMA}}(\mathbf{x}) = \frac{1}{\alpha} \log \sum_{k=1}^{K} \exp\left(\alpha(b_k + \mathbf{a}_k^{\text{T}}\mathbf{x})\right) \tag{A.4}$$

The SMA function uses a global softening parameter (α) to 'smooth' the sharp corners of the Max Affine function and has the benefit of requiring far fewer affine terms K to capture smooth convex functions with reasonable accuracy [47]. However, the global softening parameter results in a poor representation in regions where the curvature deviates substantially from the global average [47].

Consider the following function which is the difference between two Softmax Affine functions

(Equation A.4):

$$f_{\text{DSMA}}(\mathbf{x}) = \frac{1}{\alpha}\log\sum_{k=1}^{K}\exp\left(\alpha(b_k + \mathbf{a}_k^{\mathrm{T}}\mathbf{x})\right) - \frac{1}{\beta}\log\sum_{m=1}^{M}\exp\left(\beta(h_m + \mathbf{g}_m^{\mathrm{T}}\mathbf{x})\right) \tag{A.5}$$

In the same way that an individual SMA function requires fewer terms K than the Max Affine function, the two SMA functions of DSMA require fewer terms K and M to fit smooth convex functions [47]. Thus, the DSMA function will generally require fewer terms $K + M$ than the DMA function to obtain an accurate fit to DC functions.

Transforming the DSMA function back to the optimization variables $\mathbf{u}_j$ and w_j proceeds as follows:

$$\begin{aligned}
y &= \frac{1}{\alpha}\log\sum_{k=1}^{K}\exp\left(\alpha(b_k + \mathbf{a}_k^{\mathrm{T}}\mathbf{x})\right) - \frac{1}{\beta}\log\sum_{m=1}^{M}\exp\left(\beta(h_m + \mathbf{g}_m^{\mathrm{T}}\mathbf{x})\right) \\
\exp(y) &= \exp\left(\frac{1}{\alpha}\log\sum_{k=1}^{K}\exp\left(\alpha(b_k + \mathbf{a}_k^{\mathrm{T}}\mathbf{x})\right) - \frac{1}{\beta}\log\sum_{m=1}^{M}\exp\left(\beta(h_m + \mathbf{g}_m^{\mathrm{T}}\mathbf{x})\right)\right) \\
w &= \frac{\exp\left(\frac{1}{\alpha}\log\sum_{k=1}^{K}\exp\left(\alpha(b_k + \mathbf{a}_k^{\mathrm{T}}\mathbf{x})\right)\right)}{\exp\left(\frac{1}{\beta}\log\sum_{m=1}^{M}\exp\left(\beta(h_m + \mathbf{g}_m^{\mathrm{T}}\mathbf{x})\right)\right)} \\
w &= \frac{\exp\left(\log\left(\left(\sum_{k=1}^{K}\exp\left(\alpha(b_k + \mathbf{a}_k^{\mathrm{T}}\mathbf{x})\right)\right)^{\frac{1}{\alpha}}\right)\right)}{\exp\left(\log\left(\left(\sum_{m=1}^{M}\exp\left(\beta(h_m + \mathbf{g}_m^{\mathrm{T}}\mathbf{x})\right)\right)^{\frac{1}{\beta}}\right)\right)}
\end{aligned} \tag{A.6}$$

At this point it is obvious from the definition of a posynomial function that the form will reduce to:

$$w = \frac{\left(\sum_{k=1}^{K} e^{\alpha b_k}\prod_{i=1}^{N} u_i^{\alpha a_{ik}}\right)^{1/\alpha}}{\left(\sum_{m=1}^{M} e^{\beta h_m}\prod_{i=1}^{N} u_i^{\beta g_{im}}\right)^{1/\beta}} \tag{A.7}$$

Though Equation A.7 is not compatible with the SP formulation, consider the following

substitutions:

$$p_{\text{convex}} = \sum_{k=1}^{K} e^{\alpha b_k} \prod_{i=1}^{N} u_i^{\alpha a_{ik}} \tag{A.8}$$

$$p_{\text{concave}} = \sum_{m=1}^{M} e^{\beta h_m} \prod_{i=1}^{N} u_i^{\beta g_{im}} \tag{A.9}$$

which then reduces Equation A.7 to:

$$w = \frac{(p_{\text{convex}})^{1/\alpha}}{(p_{\text{concave}})^{1/\beta}} \tag{A.10}$$

Thus, taking the three constraints Equations A.8, A.9, A.10 as a set does result in an SP compatible scheme. This method of substitution is consistent with other approaches to constructing GP and SP compatible constraints [11].

Derivations, additional discussion, and demonstrations on example data sets are contained within [48].

Appendix B

Implementation of SQP, LSQP, and SLCP

B.1 The SQP Algorithm

Though the basic principles of Sequential Quadratic Programming were introduced in Section 3.2, there are many implementation details that must be addressed. In this section, the full algorithm for an iterative Quasi-Newton method will be outlined and discussed in a level of depth that should be sufficient to reproduce.

The algorithm described here is a combination of methods proposed by Nocedal and Wright [1], Kraft [63], and the Matlab documentation [80].

Algorithm for Sequential Quadratic Programming (SQP)	
Step 0	Construct standard form
	Given $\mathbf{x}_0$, and $k = 0$
	Initialize Lagrange multipliers, $\mu_0 = 1$
	Initialize the matrix $\mathbf{B} = \mathbf{I}$, the approximation of $\nabla^2\mathcal{L}(f, g, h, \mathbf{x}, \mu)$
	Compute $f(x_0)$, $g_i(x_0)$, and $h_j(x_0)$
	Compute $\nabla f(x_0)$, $\nabla g_i(x_0)$, and $\nabla h_j(x_0)$
Step 1	Construct the QP sub-problem
	Solve the QP sub-problem for $\mathbf{d}_y$ and $\mathbf{d}_\mu$
Step 2	Find step size α_k via (inexact) line search
Step 3	Set $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_x$
	Set $\mu_{k+1} = \mu_k + \alpha_k \mathbf{d}_\mu$
Step 4	Compute $f(x_{k+1})$, $g_i(x_{k+1})$, and $h_j(x_{k+1})$
Step 5	Compute $\nabla f(x_{k+1})$, $\nabla g_i(x_{k+1})$, and $\nabla h_j(x_{k+1})$
Step 6	Store $f(\mathbf{x}_{k+1})$, $g_i(\mathbf{x}_{k+1})$, $h_j(\mathbf{x}_{k+1})$, $\nabla f(\mathbf{x}_{k+1})$, $\nabla g_i(\mathbf{x}_{k+1})$, and $\nabla h_j(\mathbf{x}_{k+1})$
Step 7	Compute $\nabla\mathcal{L}_k(f, g, h, \mathbf{x}_k, \mu_{k+1})$
	Compute $\nabla\mathcal{L}_{k+1}(f, g, h, \mathbf{x}_{k+1}, \mu_{k+1})$
Step 8	If $\nabla\mathcal{L}_{k+1}(f, g, h, \mathbf{x}_{k+1}, \mu_{k+1}) < \varepsilon_{GL}$: Converged, Terminate
	Elseif $\|\|d_x\|\| < \varepsilon_{dx}$ [80] : Converged, Terminate
	Elseif If $k > k_{max}$: Iteration Limit Reached, Terminate
	Else: Go to Step 9
Step 9	Perform damped BFGS update on matrix $\mathbf{B}$
	Set $k = k + 1$
	Go to Step 1

Step 0: Initialization

The first step is to convert the optimization problem into the standard form (Equation 3.11). Once complete, the solution process of all iterative algorithms begin with some guess $\mathbf{x}_0$ that initializes the problem. In applications relating to design, an initial guess is usually relatively

easy to identify based on previous designs or the result of some kind of manual trade space exploration study. Lagrange multipliers must be initialized, but the actual value does not matter since the values of μ_{k+1} are always used in subsequent steps. The matrix $\mathbf{B}$ can be initialized as any positive semi-definite matrix, and is most commonly chosen to be the identity matrix. Many implementations of SQP implement a reduced memory form of the $\mathbf{B}$ matrix, which speeds computation for highly decoupled problems.

Why is matrix $\mathbf{B}$ approximated and not simply computed at each iteration? The simplest answer is that approximated methods such as BFGS [1] work better in practice. But in general, computing the Hessian of the Lagrangian $\nabla^2\mathcal{L}(f, g, h, \mathbf{x}, \mu)$ is computationally expensive, often prohibitively so. In addition, the BFGS method ensures that the matrix $\mathbf{B}$ stays positive semi-definite throughout the solve. Other methods exist that do not preserve positive semi-definiteness such as the Symmetric Rank 1 (SR1) update [1], but the BFGS method as by far the most common for SQP applications.

Step 1: Generating and Solving the QP Sub-Problem

Given the function values ($f(x_0)$, $g_i(x_0)$, $h_j(x_0)$), gradient information ($\nabla f(x_0)$, $\nabla g_i(x_0)$, and $\nabla h_j(x_0)$), and the matrix $\mathbf{B}$, the quadratic programming sub-problem can be easily constructed:

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T \mathbf{d}_x + \frac{1}{2}\mathbf{d}_x^T \mathbf{B}_k \mathbf{d}_x \\ \text{subject to} \quad & g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k)^T \mathbf{d}_x \leq 0,\ i = 1, \ldots, N \\ & h_j(\mathbf{x}_k) + \nabla h_j(\mathbf{x}_k)^T \mathbf{d}_x = 0,\ j = 1, \ldots, M \\ & \mathbf{d}_x = \mathbf{x} - \mathbf{x}^* \end{aligned} \tag{B.1}$$

Solving this problem yields $\mathbf{d}_x$, which is the search direction for the next point $\mathbf{x}_{k+1}$. The solution to the dual problem yields μ_{QP}, which is theoretically the same as μ_{k+1}, though this would only be accurate if a full step was taken. Thus, $\mathbf{d}_\mu$ is computed as $\mathbf{d}_\mu = \mu_{QP} - \mu_k$.

The linearizations of SQP sometimes produce an infeasible sub-problem [1], and so the constraints are relaxed to ensure a feasible solution can be achieved. In this work, the solved

sub-problem is instead:

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T \mathbf{d}_x + \frac{1}{2}\mathbf{d}_x^T \mathbf{B}_k \mathbf{d}_x + K \sum_{i=1}^{N+M} \sigma_i^2 \\ \text{subject to} \quad & g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k)^T \mathbf{d}_x \leq \sigma_i, \ i = 1, \ldots, N \\ & h_j(\mathbf{x}_k) + \nabla h_j(\mathbf{x}_k)^T \mathbf{d}_x = \sigma_{j+N}, \ j = 1, \ldots, M \\ & \mathbf{d}_x = \mathbf{x} - \mathbf{x}^* \end{aligned} \tag{B.2}$$

where constant K is some arbitrarily large value. This modified formulation minimizes constraint violation in addition to producing a step direction $\mathbf{d}_x$. Other methods exist for handling this problem, but the formulation in Equation B.2 represented a simple approach that avoided the issues associated with active sets.

Step 2: Line Search for Optimal Step Size

Due to the inexact nature of matrix $\mathbf{B}$ in representing the Hessian of the Lagrangian $\nabla^2 \mathcal{L}(f, g, h, \mathbf{x}, \mu)$, the vector $\mathbf{d}_x$ may not always be the ideal step. Thus, rather than accepting the step size recommended by $\mathbf{d}_x$, an independent line search is conducted to minimize the some merit function in the direction of $\mathbf{d}_x$. Exact line searches are possible, but require a prohibitive number of calls to the objective and constraint functions. Thus, an inexact line search procedure is performed.

The line search process in this work seeks to sufficiently decrease a merit function $\phi(\mathbf{y}_k + \alpha_k \mathbf{d}_y; \rho_k)$. A step size is accepted if it satisfies the inequality:

$$\phi(\mathbf{x}_k + \alpha_k \mathbf{d}_x; \rho_k) \leq \phi(\mathbf{x}_k; \rho_k) + \eta \alpha_k D(\phi(\mathbf{x}_k; \rho_k); \mathbf{d}_x) \tag{B.3}$$

Which is the constrained variation of Armijo's Rule (also known as the First Wolfe Condition). The condition states that the point at the end of the step must produce a function value less than the value of the affine approximation (scaled by constant η). Nocedal and Wright recommend a small value of $\eta = 1 \times 10^{-4}$ [1] which is used in this work. The merit

function is [80]:

$$\phi(\mathbf{x};\rho_k) = f(\mathbf{x}) + \rho_k^T \mathbf{c} \tag{B.4}$$

thus the directional derivative D is:

$$D(\phi(\mathbf{x}_k;\rho_k);\mathbf{d}_y) = \nabla f(\mathbf{x}_k)^T \mathbf{d}_x - \rho_k^T \mathbf{c} \tag{B.5}$$

and the elements of the vector c are the magnitude of the constraint violations:

$$c_i = |\max(0, g_i(\mathbf{x}))| \tag{B.6}$$

and:

$$c_{j+N} = |h_j(\mathbf{x})| \tag{B.7}$$

The penalty constants ρ_k are computed via [80]:

$$\rho_{k+1} = \max\left(|\mu_k|, \frac{\rho_k + |\mu_k|}{2}\right) \tag{B.8}$$

Though this line search method is effective, there is an inherent flaw in the line search procedure that can cause steps to be rejected that otherwise would result in high rates of convergence. This phenomenon, called the Maratos Effect [1], must be corrected for during the search, and so under most conditions, the step $\alpha = 1$ is accepted regardless of whether Armijo's Rule (Equation B.3) has been satisfied. However when an iteration fails Armijo's Rule, it is bookmarked and the metadata saved to memory. If Armijo's rule is violated in a series of sequential iterations, implying that the algorithm is not successfully decreasing the value of the merit function, the algorithm is reverted back to the bookmarked location and Armijo's rule is enforced. This approach is called the Watchdog Strategy [1] and is usually limited to approximately five sequential failures.

When the algorithm rejects the step size (usually upon reaching the maximum number of iterations allowed by the Watchdog), the line search algorithm decreases the step size α

using a geometric scaling factor τ (ie, the first trial is $\alpha = 1$, the second is $\alpha = \tau$, the third is $\alpha = \tau^2$, etc.) until Armijo's rule is satisfied, then it accepts the decreased step. In this work, $\tau = 0.9$ is used in accordance with the recommendation of Nocedal and Wright [1]. Note that each step size trial requires an additional evaluation of the objective and constraint functions, and so there is a strong desire to keep these number of function calls to a minimum. The Watchdog Strategy has an advantage here in that it minimizes the number of times the algorithm must waste function evaluations determining a smaller step size.

Step 3: Update the Design Point and Lagrange Multipliers

Once the step size has been computed, the primal and dual variables can be updated:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{x}_k + \alpha_k \mathbf{d}_x \\ \mu_{k+1} &= \mu_k + \alpha_k \mathbf{d}_\mu \end{aligned} \tag{B.9}$$

Step 4: Computing the Values of the Objective and Constraint Functions

The values of the objective and constraint functions ($f(x_{k+1})$, $g_i(x_{k+1})$, $h_j(x_{k+1})$) are computed using the new point $\mathbf{x}_{k+1}$

Step 5: Computing the Gradients of the Objective and Constraint Functions

Gradient computation is typically the most computationally expensive step in the SQP algorithm. As a result, gradient computation has received a great deal of attention in the literature, and a number of methods for gradient computation have arisen. These are the most popular [2]:

- Analytical Derivatives
- Finite Difference
- Complex Step
- Automatic Differentiation

- Adjoint Methods

Analytical or symbolic derivatives are the preferred method if available, but highly complex analysis models have made analytic derivatives difficult, if not impossible to obtain. Automatic or algorithmic differentiation has surged in recent years with packages like Tensorflow [81] becoming more widespread across the computational realms. Adjoint and Complex Step are power methods that have gained widespread use in the aerospace sector, but require some expertise to use effectively. In general, finite difference methods are the last resort, only used when true black boxes must be implemented.

For this work, it is assumed that some form of derivative is available to the optimization algorithm.

Step 6: Store Quantities for the Next Iteration

This step is included only for clarity, and to explicitly call attention to the fact that the values of $f(\mathbf{x}_{k+1})$, $g_i(\mathbf{x}_{k+1})$, $h_j(\mathbf{x}_{k+1})$, $\nabla f(\mathbf{x}_{k+1})$, $\nabla g_i(\mathbf{x}_{k+1})$, and $\nabla h_j(\mathbf{x}_{k+1})$ only need to be computed once.

Step 7: Compute Gradient of the Lagrangian

Both Lagrangians $\nabla\mathcal{L}_k$ and $\nabla\mathcal{L}_{k+1}$ are straightforward to compute with the currently known information. The first:

$$\nabla\mathcal{L}_k = \nabla f(\mathbf{x}_k) + \sum_{i=1}^{N} \mu_i \nabla g_i(\mathbf{x}_k) + \sum_{j=1}^{M} \mu_{j+N} \nabla h_j(\mathbf{x}_k) \tag{B.10}$$

is computed using the stored values from the previous iteration, while the second:

$$\nabla\mathcal{L}_{k+1} = \nabla f(\mathbf{x}_{k+1}) + \sum_{i=1}^{N} \mu_i \nabla g_i(\mathbf{x}_{k+1}) + \sum_{j=1}^{M} \mu_{j+N} \nabla h_j(\mathbf{x}_{k+1}) \tag{B.11}$$

is computed using the recently computed gradients. Note that in both cases, the newly compute Lagrange multipliers μ_{k+1} are used, however the subscript is dropped for the sake of simplicity in the notation.

Step 8: First Order Optimality Check and Termination

Four termination criteria exist in the SQP literature:

1. First Order Optimality (constrained version of $\frac{df}{d\mathbf{x}} = 0$) $\nabla\mathcal{L}_{k+1}(\mathbf{x}_{k+1}, \mu_{k+1}) < \varepsilon_{GL}$ [1]

2. Small Step Size $||d_x|| < \varepsilon_{dx}$ [80]

3. Relative Change in the Objective $|f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)| < \varepsilon_{rel}$ [63]

4. $k > k_{max}$

The implementation here utilizes only termination conditions 1, 2, and 4, as the relative convergence metric is not particularly mathematically rigorous.

Step 9: Updating the Approximation of the Lagrangian Hessian Matrix

The algorithm uses the Damped BFGS update outlined by Nocedal and Wright [1]. The new matrix $\mathbf{B}_{k+1}$ is computed as:

$$\mathbf{B}_{k+1} = \mathbf{B}_k - \frac{\mathbf{B}_k s_k s_k^T \mathbf{B}_k}{s_k^T \mathbf{B}_k s_k} + \frac{r_k r_k^T}{s_k^T r_k} \tag{B.12}$$

The vector s_k is the step taken from k to $k+1$:

$$s_k = \alpha_k \mathbf{d}_x \tag{B.13}$$

Vector r_k is a function of an intermediate vector z_k which is the difference in the gradients of the Lagrangian:

$$z_k = \nabla\mathcal{L}_{k+1}(f, g, h, \mathbf{x}_{k+1}, \mu_{k+1}) - \nabla\mathcal{L}_k(f, g, h, \mathbf{x}_k, \mu_{k+1}) \tag{B.14}$$

The vector r_k is then:

$$r_k = \theta_k z_k + (1 - \theta_k)\mathbf{B}_k s_k \tag{B.15}$$

where θ_k is determined via:

$$\theta_k = \begin{cases} 1 & \text{if } s_k^T z_k \geq 0.2 s_k^T \mathbf{B}_k s_k \\ (0.8 s_k^T \mathbf{B}_k s_k)/(s_k^T \mathbf{B}_k s_k - s_k^T z_k) & \text{if } s_k^T z_k < 0.2 s_k^T \mathbf{B}_k s_k \end{cases} \tag{B.16}$$

How this method converges is covered in Fletcher and Powell [82], though readers interested in a full discussion of the motivation for this update should reference Nocedal and Wright Chapter 18 [1]. For this work it suffices to know that the update performs well in practice.

B.2 The Logspace Sequential Quadratic Programming Algorithm

LSQP can be implemented in two ways. The simplest approach is to write the problem in the modified standard form (Equation 4.3), apply the log transformation (Equation 4.4), and then solve iteratively by using a series of standard SQP sub-problems (Equation 3.12). The final solution $\mathbf{x}^*$ is then obtained by a simple reverse transformation of $\mathbf{y}^*$. The benefit of this approach is that existing SQP algorithms can be used to solve the problem, an approach demonstrated by Kirschen [51] in limited form. But this method only works well if all of the functions $f(\mathbf{x})$, $g_i(\mathbf{x})$, and $h_j(\mathbf{x})$ are known explicitly, and if the optimization framework enables the log transformations to be easily implemented. Additionally, the practical numeric considerations of log-sum-exp functions [47] may cause issues in this type of implementation.

A more tailored algorithm that is based on an understanding of the underlying mathematics of LSQP has four key benefits. First is usability as a practical algorithm. For software tools like Matlab and Python, it is important for a user to be able to swap between algorithms with minimal effort. A tailored algorithm hides the log transformation 'under the hood' which facilitates easy integration with existing frameworks.

Second, true black boxes can be more easily integrated. Constructing SQP sub-problems from Equation 4.4 will require gradients $\partial \log f(e^{\mathbf{y}})/\partial y_i$, which have been eliminated in Equation 4.13 using Equation 4.7. Black boxes can therefore be directly integrated into the log quadratic sub-problem of Equation 4.13 without modification, which constitutes a significant advantage.

Third, the construction of standard form for LSQP is not always straightforward. Consider even the simple constraint $y = x$. A natural construction of standard form might be to first zero out the constraint and then add one, resulting in $y - x + 1 = 1$. But $x/y = 1$ is a far superior constraint since it is linear after the log transformation. Allowing the algorithm to test for such structure enables higher quality solutions to be produced more frequently.

Fourth, modifications to the standard SQP algorithm may be necessary in future work to improve the algorithm. Consider that while iteration points $\mathbf{x}_k$ need not be feasible, the conditions $f(\mathbf{x}_k) > 0$, $g_i(\mathbf{x}_k) > 0$, and $h_j(\mathbf{x}_k) > 0$ must all hold true at the initial condition and at each subsequent $\mathbf{x}_k$. As will be discussed in the section on results, this is most easily enforced by modifying the line search phase of the traditional SQP algorithm to enforce these conditions. Such modification is not generally realistic for off the shelf SQP algorithms without significant expertise and effort.

Despite the differences, the close relationship between SQP and LSQP means that the existing SQP literature can be heavily utilized. The algorithm proposed here is for the most part similar to the SQP described in Section B.2, and so a full discussion of the steps is redundant (see Section B.1). But the key differences are highlighted here.

Algorithm for Logspace Sequential Quadratic Programming (LSQP)

Step 0	Construct standard form
	Given $\mathbf{x}_0$, and $k = 0$
	Initialize logspace Lagrange multipliers, $\mu_0 = 1$
	Initialize the matrix $\mathbf{B} = \mathbf{I}$, the approximation of $\nabla^2 \mathcal{L}(f, g, h, \mathbf{y}, \mu)$ [1]
	Compute $\log f(x_0)$, $\log g_i(x_0)$, and $\log h_j(x_0)$
	Compute $\nabla f(x_0)$, $\nabla g_i(x_0)$, and $\nabla h_j(x_0)$
	Compute $\nabla \log f(e^{y_0})$, $\nabla \log g_i(e^{y_0})$, and $\nabla \log h_j(e^{y_0})$ via Equation 4.7
Step 1	Compute $\mathbf{y}_k = \log(\mathbf{x}_k)$
Step 2	Solve the QP sub-problem, obtain $\mathbf{d}_y$ and $\mathbf{d}_\mu$ [1]
Step 3	Compute the step size α_k via inexact line search [1, 80]
Step 4	Set $\mathbf{y}_{k+1} = \mathbf{y}_k + \alpha_k \mathbf{d}_y$
	Set $\mathbf{x}_{k+1} = \exp(\mathbf{y}_{k+1})$
	Set $\mu_{k+1} = \mu_k + \alpha_k \mathbf{d}_\mu$
Step 5	Compute $f(x_{k+1})$, $g_i(x_{k+1})$, and $h_j(x_{k+1})$
	Compute $\log f(x_{k+1})$, $\log g_i(x_{k+1})$, and $\log h_j(x_{k+1})$
Step 6	Compute $\nabla f(x_{k+1})$, $\nabla g_i(x_{k+1})$, and $\nabla h_j(x_{k+1})$
	Compute $\nabla \log f(e^{y_{k+1}})$, $\nabla \log g_i(e^{y_{k+1}})$, and $\nabla \log h_j(e^{y_{k+1}})$ via Equation 4.7
Step 7	Store $\log f(x_{k+1})$, $\log g_i(x_{k+1})$, and $\log h_j(x_{k+1})$
	Store $\nabla f(x_{k+1})$, $\nabla g_i(x_{k+1})$, and $\nabla h_j(x_{k+1})$
Step 8	Compute $\nabla \mathcal{L}_k(\mathbf{y}_k, \mu_{k+1})$ [1]
	Compute $\nabla \mathcal{L}_{k+1}(\mathbf{y}_{k+1}, \mu_{k+1})$ [1]
Step 9	If $\nabla \mathcal{L}_{k+1}(\mathbf{y}_{k+1}, \mu_{k+1}) < \varepsilon_{GL}$ [1] : Converged, Terminate
	Elseif $\|\|d_y\|\| < \varepsilon_{dy}$ [80] : Converged, Terminate
	Elseif If $k > k_{max}$: Iteration Limit Reached, Terminate
	Else: Go to Step 10
Step 10	Perform damped BFGS update on matrix $\mathbf{B}$
	Set $k = k + 1$
	Go to Step 1

Step 1: Transformation of Variables

Fundamentally, the LSQP algorithm is the application of the SQP algorithm using log-transformed variables:

$$\mathbf{y}_k = \log(\mathbf{x}_k) \tag{B.17}$$

thus all of the steps in the algorithm reference $\mathbf{y}_k$ rather than $\mathbf{x}_k$.

Step 2: Generating and Solving the QP Sub-Problem

Just as with SQP, the problem of inconsistent constraints must be handled in the sub-problem. Thus, the algorithm solves the sub-problem:

$$\begin{aligned}
\underset{\mathbf{d}}{\text{minimize}} \quad & \log f(\mathbf{x}_k) + \frac{1}{f(\mathbf{x}_k)} \left(\mathbf{x}_k \odot \nabla f(\mathbf{x}_k)\right)^T \mathbf{d} + \frac{1}{2}\mathbf{d}^T \nabla^2 \mathcal{L}(\mathbf{y}_k)\mathbf{d} + K \sum_{i=1}^{N+M} \sigma_i^2 \\
\text{subject to} \quad & \log g_i(\mathbf{x}_k) + \frac{1}{g_i(\mathbf{x}_k)} \left(\mathbf{x}_k \odot \nabla g_i(\mathbf{x}_k)\right)^T \mathbf{d} \leq \sigma_i, \; i = 1, \ldots, N \\
& \log h_j(\mathbf{x}_k) + \frac{1}{h_j(\mathbf{x}_k)} \left(\mathbf{x}_k \odot \nabla h_j(\mathbf{x}_k)\right)^T \mathbf{d} = \sigma_{j+N}, \; j = 1, \ldots, M \\
& \mathbf{d} = \mathbf{y} - \log \mathbf{x}_k \\
& \mathbf{y} = \log \mathbf{x}
\end{aligned} \tag{B.18}$$

Rather than the one presented in Equation 4.13.

Step 3: Line Search for Optimal Step Size

Though the line search procedure remains the same for LSQP as it was for SQP, consider that LSQP requires that all $f(\mathbf{x}_k) > 0$, $g_i(\mathbf{x}_k) > 0$, and $h_j(\mathbf{x}_k) > 0$ at the first iteration $k = 0$ and for all subsequent iterations. Assuming the constraints hold at $k = 0$, the LSQP algorithm presented here can be modified such that the line search for step size must result in all $f(\mathbf{x}_k) > 0$, $g_i(\mathbf{x}_k) > 0$, and $h_j(\mathbf{x}_k) > 0$, or else the step size must be decreased according to the τ schedule. This correction has not been implemented in this work because it requires more calls to the objective and constraint functions at any iteration that must decrease step size to satisfy $f(\mathbf{x}_k) > 0$, $g_i(\mathbf{x}_k) > 0$, and $h_j(\mathbf{x}_k) > 0$. A comparison between the number of

iterations for SQP and LSQP would then not be a true apples to apples comparison, as the number of black box calls might not be the same for two identical iteration counts. Thus, this correction will be made in future work, when the algorithm has been benchmarked (see Chapter 5).

Steps 8, 9 and 10: Lagrangians and Matrix B

It is important to note that since the LSQP algorithm operates in logspace, the Lagrangian is constructed under the log transformation:

$$\mathcal{L}(\mathbf{y}, \mu) = \log f(e^{\mathbf{y}}) + \mu \log \mathbf{g}(e^{\mathbf{y}}) + \mu \log \mathbf{h}(e^{\mathbf{y}}) \tag{B.19}$$

and ***not*** in the original space:

$$\mathcal{L}(\mathbf{x}, \mu) = f(\mathbf{x}) + \mu \mathbf{g}(\mathbf{x}) + \mu \mathbf{h}(\mathbf{x}) \tag{B.20}$$

Thus the matrix $\mathbf{B}$ is an approximation of the Hessian of Equation B.19 and ***not*** of Equation B.20.

Step 9: Termination

Most implementations of the SQP algorithm have one or more of four termination conditions:

1. First Order Optimality (constrained version of $\frac{df}{d\mathbf{x}} = 0$) $\nabla \mathcal{L}_{k+1}(\mathbf{x}_{k+1}, \mu_{k+1}) < \varepsilon_{GL}$ [1]
2. Small Step Size $||d_x|| < \varepsilon_{dx}$ [80]
3. Relative Change in the Objective $|f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)| < \varepsilon_{rel}$ [63]
4. $k > k_{max}$

For LSQP, the first two termination criteria can also be taken in the log transformed space, making the possible options:

1. $\nabla \mathcal{L}_{k+1}(\mathbf{y}_{k+1}, \mu_{k+1}) < \varepsilon_{GLy}$

2. $||d_y|| < \varepsilon_{dy}$

3. $\nabla \mathcal{L}_{k+1}(\mathbf{x}_{k+1}, \mu_{k+1}) < \varepsilon_{GLx}$

4. $||d_x|| < \varepsilon_{dx}$

5. $|f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)| < \varepsilon_{rel}$

6. $k > k_{max}$

This work uses 1.) $\nabla \mathcal{L}_{k+1}(\mathbf{y}_{k+1}, \mu_{k+1}) < \varepsilon_{GLy}$ 2.) $||d_x|| < \varepsilon_{dx}$ and 3.) $k > k_{max}$. In general, use of the logspace termination criteria will result in more rapid convergence due to consistency in the scaling.

B.3 The Sequential Log Convex Programming Algorithm

The SLCP algorithm is nearly identical to the LSQP algorithm in terms of outer loop execution, but is fundamentally different in that the sub-problem is no longer quadratic. Thus, existing SQP packages cannot be used to implement the SLCP algorithm, and an new algorithm must be created.

Algorithm for Sequential Log Convex Programming (SLCP)	
Step 0	Construct standard form
	Given $\mathbf{x}_0$, and $k = 0$
	Initialize logspace Lagrange multipliers, $\mu_0 = 1$
	Initialize the matrix $\mathbf{B} = \mathbf{I}$, the approximation of $\nabla^2 \mathcal{L}_R(f, g, h, \mathbf{y}, \mu)$ [1]
	Compute $\log f(x_0)$, $\log g_i(x_0)$, and $\log h_j(x_0)$
	Compute $\nabla f(x_0)$, $\nabla g_i(x_0)$, and $\nabla h_j(x_0)$
	Compute $\nabla \log f(e^{y_0})$, $\nabla \log g_i(e^{y_0})$, and $\nabla \log h_j(e^{y_0})$ via Equation 4.7
Step 1	Compute $\mathbf{y}_k = \log(\mathbf{x}_k)$
Step 2	Solve the convex sub-problem, obtain $\mathbf{d}_y$ and $\mathbf{d}_\mu$
Step 3	Compute the step size α_k via inexact line search [1, 80]
Step 4	Set $\mathbf{y}_{k+1} = \mathbf{y}_k + \alpha_k \mathbf{d}_y$, $\mathbf{x}_{k+1} = \exp(\mathbf{y}_{k+1})$, and $\mu_{k+1} = \mu_k + \alpha_k \mathbf{d}_\mu$
Step 5	Compute $f(x_{k+1})$, $g_i(x_{k+1})$, and $h_j(x_{k+1})$
	Compute $\log f(x_{k+1})$, $\log g_i(x_{k+1})$, and $\log h_j(x_{k+1})$
Step 6	Compute $\nabla f(x_{k+1})$, $\nabla g_i(x_{k+1})$, and $\nabla h_j(x_{k+1})$
	Compute $\nabla \log f(e^{y_{k+1}})$, $\nabla \log g_i(e^{y_{k+1}})$, and $\nabla \log h_j(e^{y_{k+1}})$ via Equation 4.7
Step 7	Store $\log f(x_{k+1})$, $\log g_i(x_{k+1})$, and $\log h_j(x_{k+1})$
	Store $\nabla f(x_{k+1})$, $\nabla g_i(x_{k+1})$, and $\nabla h_j(x_{k+1})$
Step 8	Compute $\nabla \mathcal{L}_k(\mathbf{y}_k, \mu_{k+1})$ [1]
	Compute $\nabla \mathcal{L}_{k+1}(\mathbf{y}_{k+1}, \mu_{k+1})$ [1]
Step 9	If $\nabla \mathcal{L}_{k+1}(\mathbf{y}_{k+1}, \mu_{k+1}) < \varepsilon_{GL}$ [1] : Converged, Terminate
	Elseif $\|\|d_y\|\| < \varepsilon_{dy}$ [80] : Converged, Terminate
	Elseif If $k > k_{max}$: Iteration Limit Reached, Terminate
	Else: Go to Step 10
Step 10	Perform modified damped BFGS update on matrix $\mathbf{B}$
	Set $k = k + 1$
	Go to Step 1

Step 2: Generating and Solving the QP Sub-Problem

Just as with SQP and LSQP, the problem of inconsistent constraints must be handled in the sub-problem by relaxation:

$$\begin{aligned}
\underset{\mathbf{d}}{\text{minimize}} \quad & \log f(\mathbf{x}_k) + \frac{1}{f(\mathbf{x}_k)} \left(\mathbf{x}_k \odot \nabla f(\mathbf{x}_k)\right)^T \mathbf{d} + \frac{1}{2}\mathbf{d}^T \nabla^2 \mathcal{L}_R(\mathbf{y}_k)\mathbf{d} + K \sum_{i=1}^{N+M} \sigma_i^2 \\
\text{subject to} \quad & \log\left(\sum_j \exp\left(P_j(\mathbf{d} + \log \mathbf{x}_k) + q_j\right)\right) \leq \sigma_i \\
& A_m(\mathbf{d} + \log \mathbf{x}_k) + b_m \leq \sigma_i \\
& \log g(\mathbf{x}_k) + \frac{1}{g(\mathbf{x}_k)} \left(\mathbf{x}_k \odot \nabla g(\mathbf{x}_k)\right)^T \mathbf{d} \leq \sigma_i \\
& \log h(\mathbf{x}_k) + \frac{1}{h(\mathbf{x}_k)} \left(\mathbf{x}_k \odot \nabla h(\mathbf{x}_k)\right)^T \mathbf{d} = \sigma_i \\
& \log \mathcal{B}_i(\mathbf{x}_k) + \frac{1}{\mathcal{B}_i(\mathbf{x}_k)} \left(\mathbf{x}_k \odot \nabla \mathcal{B}_i(\mathbf{x}_k)\right)^T \mathbf{d} \leq \sigma_i \\
& \log \mathcal{B}_e(\mathbf{x}_k) + \frac{1}{\mathcal{B}_e(\mathbf{x}_k)} \left(\mathbf{x}_k \odot \nabla \mathcal{B}_e(\mathbf{x}_k)\right)^T \mathbf{d} = \sigma_i \\
& \mathbf{d} = \mathbf{y} - \log \mathbf{x}_k \\
& \mathbf{y} = \log \mathbf{x}
\end{aligned} \tag{B.21}$$

Step 10: Updating Matrix B

The matrix $\mathbf{B}$ is now being used to approximate the Hessian of the Reduced Lagrangian:

$$\mathcal{L}_R(\mathbf{y}, \lambda) = \log f(\mathbf{x}) + \lambda \log \mathbf{g}(\mathbf{x}_k) + \lambda \log \mathbf{h}(\mathbf{x}_k) + \lambda \log \mathcal{B}_i(\mathbf{x}_k) + \lambda \log \mathcal{B}_e(\mathbf{x}_k) \tag{B.22}$$

and ***not*** the Full Lagrangian:

$$\begin{aligned}
\mathcal{L}(\mathbf{y}, \lambda) = \log f(\mathbf{x}) + \lambda \log \mathbf{p}(\mathbf{x}_k) + \lambda \log \mathbf{m}(\mathbf{x}_k) + \lambda \log \mathbf{g}(\mathbf{x}_k) + \lambda \log \mathbf{h}(\mathbf{x}_k) \\
+ \lambda \log \mathcal{B}_i(\mathbf{x}_k) + \lambda \log \mathcal{B}_e(\mathbf{x}_k)
\end{aligned} \tag{B.23}$$

which is still used in the termination conditions (Step 9). The procedure is therefore slightly

modified to the following (note the change in computing z_k):

$$\mathbf{B}_{k+1} = \mathbf{B}_k - \frac{\mathbf{B}_k s_k s_k^T \mathbf{B}_k}{s_k^T \mathbf{B}_k s_k} + \frac{r_k r_k^T}{s_k^T r_k} \tag{B.24}$$

$$s_k = \alpha_k \mathbf{d}_x \tag{B.25}$$

$$z_k = \nabla \mathcal{L}_{R_{k+1}}(f, g, h, \mathbf{y}_{k+1}, \mu_{k+1}) - \nabla \mathcal{L}_{R_k}(f, g, h, \mathbf{y}_k, \mu_{k+1}) \tag{B.26}$$

$$r_k = \theta_k z_k + (1 - \theta_k)\mathbf{B}_k s_k \tag{B.27}$$

$$\theta_k = \begin{cases} 1 & \text{if } s_k^T z_k \geq 0.2 s_k^T \mathbf{B}_k s_k \\ (0.8 s_k^T \mathbf{B}_k s_k)/(s_k^T \mathbf{B}_k s_k - s_k^T z_k) & \text{if } s_k^T z_k < 0.2 s_k^T \mathbf{B}_k s_k \end{cases} \tag{B.28}$$

Appendix C

Data Tables for the Benchmarking Studies

This appendix contains all of the aggregated data generated to benchmark the LSQP and SLCP algorithms against an SQP baseline. Each table reports the average value over 1000 random trials as described in Chapter 5. Values in parenthesis represent percent change from the SQP baseline values, with the exception of falures, where the value in parenthesis represents the precentage of initial conditions which resulted in a failed case.

C.1 The Boyd Problem

Table C.1: Results of Solving the Boyd Problem With a Good Initial Guess

	Optimum	SQP	Matlab SQP	LSQP	Matlab LT+SQP
Obj $[1/m^3]$	5.196e-03	5.199e-03 (+0.06%)	5.196e-03 (+0.00%)	5.196e-03 (-0.00%)	5.196e-03 (-0.00%)
d $[m]$	11.55	11.27 (-0.17%)	11.53 (-0.00%)	11.55 (+0.00%)	11.55 (+0.00%)
h $[m]$	2.89	2.92 (+0.09%)	2.89 (+0.00%)	2.89 (-0.00%)	2.89 (-0.00%)
w $[m]$	5.77	5.85 (+0.09%)	5.78 (-0.00%)	5.77 (-0.00%)	5.77 (-0.00%)
Iterations	-	6.94 (0.00%)	10.68 (+53.99%)	4.35 (-37.31%)	4.10 (-40.90%)
Failures	-	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)

Table C.2: Results of Solving the Boyd Problem With a Reasonable Initial Guess

	Optimum	SQP	Matlab SQP	LSQP	Matlab LT+SQP
Obj $[1/m^3]$	5.196e-03	5.200e-03 (+0.08%)	5.196e-03 (+0.00%)	5.196e-03 (-0.00%)	5.196e-03 (-0.00%)
d $[m]$	11.546	11.321 (-0.08%)	11.537 (-0.08%)	11.547 (+0.00%)	11.547 (+0.00%)
h $[m]$	2.887	2.916 (+0.04%)	2.888 (+0.04%)	2.887 (-0.00%)	2.887 (-0.00%)
w $[m]$	5.774	5.832 (+0.04%)	5.776 (+0.04%)	5.774 (-0.00%)	5.774 (-0.00%)
Iterations	-	14.65 (0.00%)	16.61 (+13.39%)	4.82 (-67.10%)	4.55 (-68.94%)
Failures	-	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)

Table C.3: Results of Solving the Boyd Problem With a Poor Initial Guess

	Optimum	SQP	Matlab SQP	LSQP	Matlab LT+SQP
Obj $[1/m^3]$	5.196e-03	5.203e-03 (+0.12%)	5.196e-03 (+0.00%)	5.196e-03 (-0.00%)	5.196e-03 (-0.00%)
d $[m]$	11.55	11.36 (-0.05%)	11.54 (-0.00%)	11.55 (+0.00%)	11.55 (+0.00%)
h $[m]$	2.89	2.91 (+0.03%)	2.89 (+0.00%)	2.89 (-0.00%)	2.89 (-0.00%)
w $[m]$	5.77	5.82 (+0.03%)	5.78 (+0.00%)	5.77 (-0.00%)	5.77 (-0.00%)
Iterations	-	16.94 (0.00%)	18.27 (+7.84%)	5.09 (-69.94%)	4.85 (-71.37%)
Failures	-	27 (2.70%)	0 (0.00%)	0 (0.00%)	0 (0.00%)

C.2 The Rosenbrock Problem

Note that this version of the Rosenbrock problem has a local minimum at the origin due to the imposed constraints. Cases that converged to this local minimum were counted as failures.

Table C.4: Results of Solving the Rosenbrock Problem With a Good Initial Guess

	Optimum	SQP	Matlab SQP	LSQP	Matlab LT+SQP
Obj [−]	1.00	1.00 (+0.00%)	1.00 (+0.00%)	1.00 (+0.00%)	1.00 (+0.00%)
x [−]	1.00	1.00 (-0.00%)	1.00 (-0.00%)	1.00 (-0.00%)	1.00 (-0.00%)
y [−]	1.00	1.00 (-0.00%)	1.00 (-0.00%)	1.00 (+0.00%)	1.00 (+0.00%)
Iterations	-	3.35 (0.00%)	3.61 (+7.84%)	4.75 (+42.07%)	4.63 (+38.48%)
Failures	-	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)

Table C.5: Results of Solving the Rosenbrock Problem With a Reasonable Initial Guess

	Optimum	SQP	Matlab SQP	LSQP	Matlab LT+SQP
Obj [−]	1.000	1.000 (+0.00%)	1.000 (+0.00%)	1.000 (+0.00%)	1.000 (+0.00%)
x [−]	1.000	1.000 (-0.00%)	1.000 (-0.00%)	1.000 (-0.00%)	1.000 (-0.00%)
y [−]	1.000	1.000 (-0.00%)	1.000 (-0.00%)	1.000 (+0.00%)	1.000 (+0.00%)
Iterations	-	4.22 (0.00%)	4.43 (+5.06%)	6.00 (+42.35%)	6.14 (+45.66%)
Failures	-	1 (0.10%)	29 (2.90%)	33 (3.30%)	167 (16.70%)

Table C.6: Results of Solving the Rosenbrock Problem With a Poor Initial Guess

	Optimum	SQP	Matlab SQP	LSQP	Matlab LT+SQP
Obj [−]	1.00	1.00 (+0.00%)	1.00 (+0.00%)	1.00 (+0.00%)	1.00 (+0.00%)
x [−]	1.00	1.00 (-0.00%)	1.00 (-0.00%)	1.00 (-0.00%)	1.00 (-0.00%)
y [−]	1.00	1.00 (-0.00%)	1.00 (-0.00%)	1.00 (+0.00%)	1.00 (+0.00%)
Iterations	-	4.77 (0.00%)	4.68 (-1.87%)	6.91 (+44.76%)	7.04 (+47.52%)
Failures	-	163 (16.30%)	250 (25.00%)	265 (26.50%)	437 (43.70%)

C.3 The Floudas Problem

Table C.7: Results of Solving the Floudas Problem With a Good Initial Guess

	Optimum	SQP	Matlab SQP	LSQP	Matlab LT+SQP	SLCP
Obj [−]	7049.2	7049.2 (-0.00%)	7049.2 (-0.00%)	7049.2 (-0.00%)	7049.2 (-0.00%)	7049.2 (-0.00%)
x_1 [−]	579.3	579.3 (-0.00%)	579.3 (-0.00%)	579.3 (-0.00%)	579.3 (-0.00%)	579.3 (-0.00%)
x_2 [−]	1360.0	1360.0 (-0.00%)	1360.0 (-0.00%)	1360.0 (-0.00%)	1360.0 (+0.00%)	1360.0 (-0.00%)
x_3 [−]	5110.0	5110.0 (-0.00%)	5110.0 (-0.00%)	5110.0 (-0.00%)	5110.0 (-0.00%)	5110.0 (-0.00%)
x_4 [−]	182.0	182.0 (-0.00%)	182.0 (-0.00%)	182.0 (-0.00%)	182.0 (-0.00%)	182.0 (-0.00%)
x_5 [−]	295.6	295.6 (+0.00%)	295.6 (+0.00%)	295.6 (+0.00%)	295.6 (+0.00%)	295.6 (+0.00%)
x_6 [−]	218.0	218.0 (+0.00%)	218.0 (+0.00%)	218.0 (+0.00%)	218.0 (+0.00%)	218.0 (+0.00%)
x_7 [−]	286.4	286.4 (-0.00%)	286.4 (-0.00%)	286.4 (-0.00%)	286.4 (-0.00%)	286.4 (-0.00%)
x_8 [−]	395.6	395.6 (+0.00%)	395.6 (+0.00%)	395.6 (+0.00%)	395.6 (+0.00%)	395.6 (+0.00%)
Iterations	-	25.87 (0.00%)	25.42 (-1.71%)	11.70 (-54.79%)	11.62 (-55.08%)	10.74 (-58.48%)
Failures	-	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)

Table C.8: Results of Solving the Floudas Problem With a Reasonable Initial Guess

	Optimum	SQP	Matlab SQP	LSQP	Matlab LT+SQP	SLCP
Obj [−]	7049.249	7049.248 (-0.00%)	7049.248 (-0.00%)	7050.363 (+0.02%)	7050.694 (+0.02%)	7049.248 (-0.00%)
x_1 [−]	579.307	579.307 (-0.00%)	579.306 (-0.00%)	577.683 (-0.28%)	579.137 (-0.03%)	579.307 (-0.00%)
x_2 [−]	1359.971	1359.971 (+0.00%)	1359.971 (+0.00%)	1358.035 (-0.14%)	1359.925 (-0.00%)	1359.971 (-0.00%)
x_3 [−]	5109.971	5109.971 (-0.00%)	5109.970 (-0.00%)	5114.644 (+0.09%)	5111.632 (+0.03%)	5109.971 (-0.00%)
x_4 [−]	182.018	182.018 (-0.00%)	182.018 (-0.00%)	181.843 (-0.10%)	181.886 (-0.07%)	182.018 (-0.00%)
x_5 [−]	295.601	295.601 (+0.00%)	295.601 (+0.00%)	295.414 (-0.06%)	295.536 (-0.02%)	295.601 (+0.00%)
x_6 [−]	217.982	217.982 (+0.00%)	217.982 (+0.00%)	218.157 (+0.08%)	218.108 (+0.06%)	217.982 (+0.00%)
x_7 [−]	286.417	286.417 (-0.00%)	286.416 (-0.00%)	286.429 (+0.00%)	286.351 (-0.02%)	286.417 (-0.00%)
x_8 [−]	395.601	395.601 (+0.00%)	395.601 (+0.00%)	395.414 (-0.05%)	395.536 (-0.02%)	395.601 (+0.00%)
Iterations	-	30.17 (0.00%)	29.90 (-0.88%)	14.30 (-52.60%)	14.53 (-51.82%)	12.92 (-57.15%)
Failures	-	0 (0.00%)	0 (0.00%)	69 (6.90%)	2 (0.20%)	0 (0.00%)

Table C.9: Results of Solving the Floudas Problem With a Poor Initial Guess

	Optimum	SQP	Matlab SQP	LSQP	Matlab LT+SQP	SLCP
Obj [−]	7049.2	7049.2 (-0.00%)	7049.2 (-0.00%)	7050.8 (+0.02%)	7049.4 (+0.00%)	7049.2 (-0.00%)
x_1 [−]	579.3	579.3 (-0.00%)	579.3 (-0.00%)	577.9 (-0.24%)	578.8 (-0.09%)	579.3 (-0.00%)
x_2 [−]	1360.0	1360.0 (+0.00%)	1360.0 (+0.00%)	1358.7 (-0.10%)	1360.4 (+0.03%)	1360.0 (-0.00%)
x_3 [−]	5110.0	5110.0 (-0.00%)	5110.0 (-0.00%)	5114.3 (+0.08%)	5110.2 (+0.00%)	5110.0 (-0.00%)
x_4 [−]	182.0	182.0 (-0.00%)	182.0 (-0.00%)	181.9 (-0.09%)	181.9 (-0.08%)	182.0 (-0.00%)
x_5 [−]	295.6	295.6 (+0.00%)	295.6 (+0.00%)	295.5 (-0.05%)	295.3 (-0.09%)	295.6 (+0.00%)
x_6 [−]	218.0	218.0 (+0.00%)	218.0 (+0.00%)	218.1 (+0.07%)	217.9 (-0.05%)	218.0 (+0.00%)
x_7 [−]	286.4	286.4 (-0.00%)	286.4 (-0.00%)	286.4 (-0.00%)	286.1 (-0.10%)	286.4 (-0.00%)
x_8 [−]	395.6	395.6 (+0.00%)	395.6 (+0.00%)	395.5 (-0.04%)	395.7 (+0.03%)	395.6 (+0.00%)
Iterations	-	32.79 (0.00%)	32.22 (-1.73%)	16.29 (-50.32%)	16.64 (-49.26%)	14.92 (-54.49%)
Failures	-	0 (0.00%)	0 (0.00%)	174 (17.40%)	8 (0.80%)	0 (0.00%)

C.4 The Kirschen-Ozturk Problem

Table C.10: Results of Solving the Kirschen-Ozturk Problem With a Good Initial Guess

	SP+DCA	SQP	Matlab SQP	LSQP	SLCP
Obj $[N]$	755.91	756.28 (+0.05%)	756.64 (+0.10%)	755.90 (-0.00%)	755.90 (-0.00%)
A [−]	6.52	6.46 (-0.89%)	6.55 (+0.50%)	6.51 (-0.11%)	6.51 (-0.11%)
C_D [−]	0.013	0.013 (-0.17%)	0.013 (+0.14%)	0.013 (-0.01%)	0.013 (-0.01%)
C_L [−]	0.234	0.233 (-0.65%)	0.235 (+0.53%)	0.234 (-0.06%)	0.234 (-0.06%)
C_f [−]	3.278e-03	3.273e-03 (-0.16%)	3.280e-03 (+0.05%)	3.277e-03 (-0.01%)	3.277e-03 (-0.01%)
D $[N]$	368.7	370.3 (+0.45%)	368.5 (-0.03%)	368.8 (+0.03%)	368.8 (+0.03%)
Re [−]	5.86e+06	5.92e+06 (+0.91%)	5.86e+06 (-0.06%)	5.869e+06 (+0.07%)	5.869e+06 (+0.07%)
S $[m^2]$	16.00	15.97 (-0.14%)	16.02 (+0.15%)	15.99 (-0.02%)	15.99 (-0.02%)
V $[m/s]$	54.18	54.40 (+0.40%)	54.12 (-0.13%)	54.21 (+0.03%)	54.21 (+0.03%)
V_f $[m^3]$	0.096	0.096 (+0.05%)	0.096 (+0.10%)	0.096 (-0.00%)	0.096 (-0.00%)
$V_{f_{avail}}$ $[m^3]$	9.584e-02	9.589e-02 (+0.05%)	9.593e-02 (+0.10%)	9.584e-02 (-0.00%)	9.584e-02 (-0.00%)
$V_{f_{fuse}}$ $[m^3]$	5.840e-03	5.454e-03 (-6.61%)	5.858e-03 (+0.30%)	5.656e-03 (-3.15%)	5.656e-03 (-3.15%)
$V_{f_{wing}}$ $[m^3]$	9.016e-02	9.043e-02 (+0.30%)	9.016e-02 (+0.00%)	9.018e-02 (+0.02%)	9.018e-02 (+0.02%)
W $[N]$	7140.2	7130.1 (-0.14%)	7151.0 (+0.15%)	7138.6 (-0.02%)	7138.6 (-0.02%)
W_f $[N]$	755.9	756.3 (+0.05%)	756.6 (+0.10%)	755.9 (-0.00%)	755.9 (-0.00%)
W_w $[N]$	1444.3	1433.8 (-0.73%)	1454.4 (+0.70%)	1442.7 (-0.11%)	1442.7 (-0.11%)
$W_{w_{strc}}$ $[N]$	720.8	711.3 (-1.32%)	729.8 (+1.24%)	719.4 (-0.20%)	719.4 (-0.20%)
$W_{w_{surf}}$ $[N]$	723.5	722.4 (-0.14%)	724.6 (+0.15%)	723.3 (-0.02%)	723.3 (-0.02%)
t $[min]$	307.6	306.5 (-0.35%)	308.2 (+0.22%)	307.5 (-0.03%)	307.5 (-0.03%)
Iterations	-	35.93 (0.00%)	47.67 (+32.68%)	15.36 (-57.24%)	13.34 (-62.88%)
Failures	-	0 (0.00%)	39 (3.90%)	0 (0.00%)	0 (0.00%)

Table C.11: Results of Solving the Kirschen-Ozturk Problem With a Reasonable Initial Guess

	Optimum	SQP	Matlab SQP	LSQP	SLCP
Obj $[N]$	755.907	758.549 (+0.35%)	768.787 (+1.70%)	755.896 (-0.00%)	755.896 (-0.00%)
A $[-]$	6.520	6.187 (-5.11%)	6.320 (-3.07%)	6.513 (-0.11%)	6.513 (-0.11%)
C_D $[-]$	0.013	0.013 (-1.12%)	0.013 (-0.04%)	0.013 (-0.01%)	0.013 (-0.01%)
C_L $[-]$	0.234	0.225 (-4.06%)	0.231 (-1.49%)	0.234 (-0.06%)	0.234 (-0.06%)
C_f $[-]$	3.278e-03	3.246e-03 (-0.96%)	3.285e-03 (+0.21%)	3.277e-03 (-0.01%)	3.277e-03 (-0.01%)
D $[N]$	368.660	379.405 (+2.91%)	383.848 (+4.12%)	368.760 (+0.03%)	368.759 (+0.03%)
Re $[-]$	5.864e+06	6.207e+06 (+5.83%)	6.033e+06 (+2.88%)	5.869e+06 (+0.07%)	5.869e+06 (+0.07%)
S $[m^2]$	15.992	15.864 (-0.80%)	16.080 (+0.55%)	15.988 (-0.02%)	15.988 (-0.02%)
V $[m/s]$	54.189	55.553 (+2.52%)	55.735 (+2.85%)	54.205 (+0.03%)	54.205 (+0.03%)
V_f $[m^3]$	0.096	0.096 (+0.35%)	0.097 (+1.70%)	0.096 (-0.00%)	0.096 (-0.00%)
$V_{f_{avail}}$ $[m^3]$	9.584e-02	9.625e-02 (+0.43%)	9.850e-02 (+2.77%)	9.584e-02 (-0.00%)	9.584e-02 (-0.00%)
$V_{f_{fuse}}$ $[m^3]$	5.840e-03	4.679e-03 (-19.89%)	7.682e-03 (+31.54%)	5.656e-03 (-3.15%)	5.656e-03 (-3.15%)
$V_{f_{wing}}$ $[m^3]$	9.016e-02	9.171e-02 (+1.71%)	9.398e-02 (+4.24%)	9.018e-02 (+0.02%)	9.018e-02 (+0.02%)
W $[N]$	7140.211	7083.257 (-0.80%)	7164.926 (+0.35%)	7138.567 (-0.02%)	7138.567 (-0.02%)
W_f $[N]$	755.907	758.549 (+0.35%)	768.787 (+1.70%)	755.896 (-0.00%)	755.896 (-0.00%)
W_w $[N]$	1444.304	1384.708 (-4.13%)	1446.693 (+0.17%)	1442.671 (-0.11%)	1442.671 (-0.11%)
$W_{w_{strc}}$ $[N]$	720.832	667.007 (-7.47%)	718.304 (-0.35%)	719.365 (-0.20%)	719.366 (-0.20%)
$W_{w_{surf}}$ $[N]$	723.472	717.702 (-0.80%)	728.277 (+0.66%)	723.306 (-0.02%)	723.306 (-0.02%)
t $[min]$	307.563	300.915 (-2.16%)	304.206 (-1.09%)	307.475 (-0.03%)	307.475 (-0.03%)
Iterations	-	72.55 (0.00%)	97.61 (+34.54%)	18.89 (-73.97%)	16.38 (-77.42%)
Failures	-	164 (16.40%)	287 (28.70%)	0 (0.00%)	0 (0.00%)

Table C.12: Results of Solving the Kirschen-Ozturk Problem With a Poor Initial Guess

	SP+DCA	SQP	Matlab SQP	LSQP	SLCP
Obj $[N]$	755.91	761.50 (+0.74%)	782.30 (+3.49%)	755.90 (-0.00%)	755.9 (-0.00%)
A $[-]$	6.52	6.02 (-7.71%)	6.06 (-7.11%)	6.51 (-0.11%)	6.51 (-0.11%)
C_D $[-]$	0.013	0.013 (-1.59%)	0.013 (-1.28%)	0.013 (-0.01%)	0.013 (-0.01%)
C_L $[-]$	0.234	0.220 (-5.95%)	0.223 (-4.75%)	0.234 (-0.06%)	0.234 (-0.06%)
C_f $[-]$	3.278e-03	3.229e-03 (-1.50%)	3.305e-03 (+0.83%)	3.277e-03 (-0.01%)	3.277e-03 (-0.01%)
D $[N]$	368.7	386.5 (+4.48%)	458.6 (+24.40%)	368.8 (+0.03%)	368.8 (+0.03%)
Re $[-]$	5.86e+06	6.447e+06 (+9.93%)	6.254e+06 (+6.65%)	5.869e+06 (+0.07%)	5.869e+06 (+0.07%)
S $[m^2]$	16.00	15.81 (-1.12%)	64.82 (+305.33%)	15.99 (-0.02%)	15.99 (-0.02%)
V $[m/s]$	54.18	56.34 (+3.96%)	117.10 (+116.10%)	54.21 (+0.03%)	54.21 (+0.03%)
V_f $[m^3]$	0.096	0.097 (+0.74%)	0.486 (+406.63%)	0.096 (-0.00%)	0.096 (-0.00%)
$V_{f_{avail}}$ $[m^3]$	9.584e-02	9.680e-02 (+1.00%)	4.919e-01 (+413.28%)	9.584e-02 (-0.00%)	9.584e-02 (-0.00%)
$V_{f_{fuse}}$ $[m^3]$	5.840e-03	4.535e-03 (-22.35%)	1.023e-02 (+75.11%)	5.656e-03 (-3.15%)	5.656e-03 (-3.15%)
$V_{f_{wing}}$ $[m^3]$	9.016e-02	9.272e-02 (+2.83%)	8.456e-01 (+837.91%)	9.018e-02 (+0.02%)	9.018e-02 (+0.02%)
W $[N]$	7140.2	7060.0 (-1.12%)	9454.5 (+32.41%)	7138.6 (-0.02%)	7138.6 (-0.02%)
W_f $[N]$	755.9	761.5 (+0.74%)	782.3 (+3.49%)	755.9 (-0.00%)	755.9 (-0.00%)
W_w $[N]$	1444.3	1358.5 (-5.94%)	3702.4 (+156.35%)	1442.7 (-0.11%)	1442.7 (-0.11%)
$W_{w_{strc}}$ $[N]$	720.8	643.2 (-10.77%)	763.3 (+5.89%)	719.4 (-0.20%)	719.4 (-0.20%)
$W_{w_{surf}}$ $[N]$	723.5	715.3 (-1.12%)	2938.0 (+306.09%)	723.3 (-0.02%)	723.3 (-0.02%)
t $[min]$	307.6	297.6 (-3.24%)	297.7 (-3.22%)	307.5 (-0.03%)	307.5 (-0.03%)
Iterations	-	91.18 (0.00%)	129.82 (+42.38%)	21.06 (-76.90%)	17.66 (-80.63%)
Failures	-	331 (33.10%)	468 (46.80%)	3 (0.30%)	0 (0.00%)

C.5 The Hoburg Problem with No Black Boxes

Table C.13: Results of Solving the Hoburg-0 Problem With a Good Initial Guess

	Optimum	LSQP	SLCP
Obj $[N]$	5911.3	5911.3 (-0.00%)	5911.3 (+0.00%)
AR $[-]$	20.39	20.39 (-0.00%)	20.39 (-0.00%)
C_{D_0} $[-]$	0.013	0.013 (+0.00%)	0.013 (+0.00%)
C_{D_1} $[-]$	0.013	0.013 (-0.00%)	0.013 (+0.00%)
C_{D_2} $[-]$	0.008	0.008 (+0.00%)	0.008 (+0.00%)
C_{Dfuse_0} $[-]$	0.002	0.002 (+0.00%)	0.002 (+0.00%)
C_{Dfuse_1} $[-]$	0.002	0.002 (+0.00%)	0.002 (+0.00%)
C_{Dfuse_2} $[-]$	0.002	0.002 (+0.00%)	0.002 (+0.00%)
C_{Di_0} $[-]$	0.005	0.005 (+0.00%)	0.005 (+0.00%)
C_{Di_1} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Di_2} $[-]$	2.3e-4	2.3e-4 (+0.00%)	2.3e-4 (+0.00%)
C_{Dp_0} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Dp_1} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Dp_2} $[-]$	0.006	0.006 (-0.00%)	0.006 (+0.00%)
C_{L_0} $[-]$	0.574	0.574 (-0.00%)	0.574 (+0.00%)
C_{L_1} $[-]$	0.574	0.574 (-0.00%)	0.574 (+0.00%)
C_{L_2} $[-]$	0.119	0.119 (-0.00%)	0.119 (-0.00%)
$I_{cap_{bar}}$ $[-]$	2.02e-5	2.02e-5 (-0.00%)	2.02e-5 (-0.00%)
$M_{r_{bar}}$ $[N]$	36170.8	36170.8 (-0.00%)	36170.8 (-0.00%)
P_{max} $[W]$	1.19e+6	1.19e+6 (-0.00%)	1.19e+6 (-0.00%)
R $[km]$	5000.0	5000.0 (+0.00%)	5000.0 (+0.00%)
Re_0 $[-]$	4.66e+6	4.66e+6 (+0.00%)	4.66e+6 (-0.00%)
Re_1 $[-]$	4.47e+6	4.47e+6 (+0.00%)	4.47e+6 (-0.00%)
Re_2 $[-]$	1.03e+7	1.03e+7 (+0.00%)	1.03e+7 (+0.00%)
S $[m^2]$	27.84	27.84 (-0.00%)	27.84 (-0.00%)
T_0 $[N]$	747.0	747.0 (-0.00%)	747.0 (-0.00%)
T_1 $[N]$	685.4	685.4 (+0.00%)	685.4 (-0.00%)
T_2 $[N]$	2209.6	2209.6 (-0.00%)	2209.6 (-0.00%)
V_0 $[m/s]$	68.3	68.3 (+0.00%)	68.3 (-0.00%)
V_1 $[m/s]$	65.3	65.3 (+0.00%)	65.3 (-0.00%)
V_2 $[m/s]$	150.0	150.0 (-0.00%)	150.0 (-0.00%)
V_{stall} $[m/s]$	38.0	38.0 (+0.00%)	38.0 (+0.00%)
W_0 $[N]$	33853.0	33853.0 (-0.00%)	33853.0 (-0.00%)
W_1 $[N]$	31024.2	31024.2 (-0.00%)	31024.2 (-0.00%)
W_2 $[N]$	33853.0	33853.0 (-0.00%)	33853.0 (-0.00%)
W_{MTO} $[N]$	36935.6	36935.6 (-0.00%)	36935.6 (-0.00%)
W_{cap} $[N]$	4103.8	4103.8 (-0.00%)	4103.8 (-0.00%)
W_{eng} $[N]$	2805.8	2805.8 (-0.00%)	2805.8 (-0.00%)
$W_{fuel_{out}}$ $[N]$	3082.6	3082.6 (-0.00%)	3082.6 (+0.00%)
$W_{fuel_{ref}}$ $[N]$	2828.7	2828.7 (-0.00%)	2828.7 (+0.00%)
W_{pay} $[N]$	4905.0	4905.0 (-0.00%)	4905.0 (-0.00%)
W_{tilde} $[N]$	22410.8	22410.8 (-0.00%)	22410.8 (-0.00%)
W_{web} $[N]$	202.9	202.9 (-0.00%)	202.9 (+0.00%)
W_{wing} $[N]$	8613.4	8613.4 (-0.00%)	8613.4 (-0.00%)
W_{zfw} $[N]$	31024.2	31024.2 (-0.00%)	31024.2 (-0.00%)
eta_{0_0} $[-]$	0.27	0.27 (+0.00%)	0.27 (-0.00%)
eta_{0_1} $[-]$	0.27	0.27 (+0.00%)	0.27 (-0.00%)
eta_{0_2} $[-]$	0.279	0.279 (+0.00%)	0.279 (-0.00%)
eta_{i_0} $[-]$	0.907	0.907 (+0.00%)	0.907 (-0.00%)
eta_{i_1} $[-]$	0.907	0.907 (+0.00%)	0.907 (-0.00%)
eta_{i_2} $[-]$	0.939	0.939 (+0.00%)	0.939 (-0.00%)
eta_{prop_0} $[-]$	0.771	0.771 (+0.00%)	0.771 (-0.00%)
eta_{prop_1} $[-]$	0.771	0.771 (+0.00%)	0.771 (-0.00%)
eta_{prop_2} $[-]$	0.798	0.798 (+0.00%)	0.798 (-0.00%)
nu $[-]$	0.786	0.786 (-0.00%)	0.786 (-0.00%)
p $[-]$	1.9	1.9 (-0.00%)	1.9 (-0.00%)
q $[-]$	1.45	1.45 (-0.00%)	1.45 (-0.00%)
$t_{cap_{bar}}$ $[-]$	0.005	0.005 (-0.00%)	0.005 (-0.00%)
$t_{web_{bar}}$ $[-]$	0.001	0.001 (-0.00%)	0.001 (-0.00%)
tau $[-]$	0.15	0.15 (-0.00%)	0.15 (-0.00%)
z_{bre_0} $[-]$	0.087	0.087 (+0.00%)	0.087 (+0.00%)
z_{bre_1} $[-]$	0.087	0.087 (+0.00%)	0.087 (+0.00%)
Iterations	-	13.63 (0.00%)	13.53 (-0.71%)
Failures	-	0 (0.00%)	0 (0.00%)

Table C.14: Results of Solving the Hoburg-0 Problem With a Reasonable Initial Guess

	Optimum	LSQP	SLCP
Obj $[N]$	5911.3	5911.3 (-0.00%)	5911.3 (+0.00%)
AR $[-]$	20.39	20.39 (+0.00%)	20.39 (-0.00%)
C_{D_0} $[-]$	0.013	0.013 (-0.00%)	0.013 (+0.00%)
C_{D_1} $[-]$	0.013	0.013 (-0.00%)	0.013 (+0.00%)
C_{D_2} $[-]$	0.008	0.008 (-0.00%)	0.008 (+0.00%)
C_{Dfuse_0} $[-]$	0.002	0.002 (-0.00%)	0.002 (+0.00%)
C_{Dfuse_1} $[-]$	0.002	0.002 (-0.00%)	0.002 (+0.00%)
C_{Dfuse_2} $[-]$	0.002	0.002 (-0.00%)	0.002 (+0.00%)
C_{Di_0} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Di_1} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Di_2} $[-]$	2.3e-4	2.3e-4 (-0.00%)	2.3e-4 (+0.00%)
C_{Dp_0} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Dp_1} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Dp_2} $[-]$	0.006	0.006 (-0.00%)	0.006 (+0.00%)
C_{L_0} $[-]$	0.574	0.574 (-0.00%)	0.574 (+0.00%)
C_{L_1} $[-]$	0.574	0.574 (-0.00%)	0.574 (+0.00%)
C_{L_2} $[-]$	0.119	0.119 (+0.00%)	0.119 (+0.00%)
$I_{cap_{bar}}$ $[-]$	2.02e-5	2.02e-5 (+0.00%)	2.02e-5 (-0.00%)
$M_{r_{bar}}$ $[N]$	36170.8	36170.8 (+0.00%)	36170.8 (-0.00%)
P_{max} $[W]$	1.19e+6	1.19e+6 (+0.00%)	1.19e+6 (+0.00%)
R $[km]$	5000.0	5000.0 (+0.00%)	5000.0 (+0.00%)
Re_0 $[-]$	4.66e+6	4.66e+6 (+0.00%)	4.66e+6 (-0.00%)
Re_1 $[-]$	4.47e+6	4.47e+6 (+0.00%)	4.47e+6 (-0.00%)
Re_2 $[-]$	1.03e+7	1.03e+7 (-0.00%)	1.03e+7 (+0.00%)
S $[m^2]$	27.84	27.84 (+0.00%)	27.84 (-0.00%)
T_0 $[N]$	747.0	747.0 (+0.00%)	747.0 (-0.00%)
T_1 $[N]$	685.4	685.4 (+0.00%)	685.4 (-0.00%)
T_2 $[N]$	2209.6	2209.6 (+0.00%)	2209.6 (+0.00%)
V_0 $[m/s]$	68.3	68.3 (+0.00%)	68.3 (-0.00%)
V_1 $[m/s]$	65.3	65.3 (+0.00%)	65.3 (-0.00%)
V_2 $[m/s]$	150.0	150.0 (-0.00%)	150.0 (+0.00%)
V_{stall} $[m/s]$	38.0	38.0 (+0.00%)	38.0 (+0.00%)
W_0 $[N]$	33853.0	33853.0 (+0.00%)	33853.0 (-0.00%)
W_1 $[N]$	31024.2	31024.2 (+0.00%)	31024.2 (-0.00%)
W_2 $[N]$	33853.0	33853.0 (+0.00%)	33853.0 (-0.00%)
W_{MTO} $[N]$	36935.6	36935.6 (+0.00%)	36935.6 (-0.00%)
W_{cap} $[N]$	4103.8	4103.8 (+0.00%)	4103.8 (-0.00%)
W_{eng} $[N]$	2805.8	2805.8 (-0.00%)	2805.8 (+0.00%)
$W_{fuel_{out}}$ $[N]$	3082.6	3082.6 (-0.00%)	3082.6 (+0.00%)
$W_{fuel_{ref}}$ $[N]$	2828.7	2828.7 (-0.00%)	2828.7 (+0.00%)
W_{pay} $[N]$	4905.0	4905.0 (-0.00%)	4905.0 (-0.00%)
W_{tilde} $[N]$	22410.8	22410.8 (+0.00%)	22410.8 (+0.00%)
W_{web} $[N]$	202.9	202.9 (+0.00%)	202.9 (+0.00%)
W_{wing} $[N]$	8613.4	8613.4 (+0.00%)	8613.4 (-0.00%)
W_{zfw} $[N]$	31024.2	31024.2 (+0.00%)	31024.2 (-0.00%)
eta_{0_0} $[-]$	0.27	0.27 (+0.00%)	0.27 (-0.00%)
eta_{0_1} $[-]$	0.27	0.27 (+0.00%)	0.27 (-0.00%)
eta_{0_2} $[-]$	0.279	0.279 (-0.00%)	0.279 (-0.00%)
eta_{i_0} $[-]$	0.907	0.907 (+0.00%)	0.907 (-0.00%)
eta_{i_1} $[-]$	0.907	0.907 (+0.00%)	0.907 (-0.00%)
eta_{i_2} $[-]$	0.939	0.939 (-0.00%)	0.939 (-0.00%)
eta_{prop_0} $[-]$	0.771	0.771 (+0.00%)	0.771 (-0.00%)
eta_{prop_1} $[-]$	0.771	0.771 (+0.00%)	0.771 (-0.00%)
eta_{prop_2} $[-]$	0.798	0.798 (-0.00%)	0.798 (-0.00%)
nu $[-]$	0.786	0.786 (-0.00%)	0.786 (-0.00%)
p $[-]$	1.9	1.9 (-0.00%)	1.9 (-0.00%)
q $[-]$	1.45	1.45 (-0.00%)	1.45 (-0.00%)
$t_{cap_{bar}}$ $[-]$	0.005	0.005 (+0.00%)	0.005 (-0.00%)
$t_{web_{bar}}$ $[-]$	0.001	0.001 (+0.00%)	0.001 (-0.00%)
tau $[-]$	0.15	0.15 (-0.00%)	0.15 (-0.00%)
z_{bre_0} $[-]$	0.087	0.087 (-0.00%)	0.087 (+0.00%)
z_{bre_1} $[-]$	0.087	0.087 (-0.00%)	0.087 (+0.00%)
Iterations	-	17.71 (0.00%)	16.23 (-8.33%)
Failures	-	0 (0.00%)	0 (0.00%)

Table C.15: Results of Solving the Hoburg-0 Problem With a Poor Initial Guess

	Optimum	LSQP	SLCP
Obj $[N]$	5911.3	5911.3 (-0.00%)	5911.3 (+0.00%)
AR $[-]$	20.39	20.39 (-0.00%)	20.39 (-0.00%)
C_{D_0} $[-]$	0.013	0.013 (-0.00%)	0.013 (+0.00%)
C_{D_1} $[-]$	0.013	0.013 (-0.00%)	0.013 (+0.00%)
C_{D_2} $[-]$	0.008	0.008 (+0.00%)	0.008 (+0.00%)
C_{Dfuse_0} $[-]$	0.002	0.002 (+0.00%)	0.002 (+0.00%)
C_{Dfuse_1} $[-]$	0.002	0.002 (+0.00%)	0.002 (+0.00%)
C_{Dfuse_2} $[-]$	0.002	0.002 (+0.00%)	0.002 (+0.00%)
C_{Di_0} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Di_1} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Di_2} $[-]$	2.3e-4	2.3e-4 (+0.00%)	2.3e-4 (+0.00%)
C_{Dp_0} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Dp_1} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Dp_2} $[-]$	0.006	0.006 (-0.00%)	0.006 (+0.00%)
C_{L_0} $[-]$	0.574	0.574 (-0.00%)	0.574 (+0.00%)
C_{L_1} $[-]$	0.574	0.574 (-0.00%)	0.574 (+0.00%)
C_{L_2} $[-]$	0.119	0.119 (+0.00%)	0.119 (-0.00%)
$I_{cap_{bar}}$ $[-]$	2.02e-5	2.02e-5 (-0.00%)	2.02e-5 (-0.00%)
$M_{r_{bar}}$ $[N]$	36170.8	36170.8 (-0.00%)	36170.8 (-0.00%)
P_{max} $[W]$	1.19e+6	1.19e+6 (-0.00%)	1.19e+6 (-0.00%)
R $[km]$	5000.0	5000.0 (+0.00%)	5000.0 (+0.00%)
Re_0 $[-]$	4.66e+6	4.66e+6 (+0.00%)	4.66e+6 (-0.00%)
Re_1 $[-]$	4.47e+6	4.47e+6 (+0.00%)	4.47e+6 (-0.00%)
Re_2 $[-]$	1.03e+7	1.03e+7 (+0.00%)	1.03e+7 (+0.00%)
S $[m^2]$	27.84	27.84 (-0.00%)	27.84 (-0.00%)
T_0 $[N]$	747.0	747.0 (+0.00%)	747.0 (-0.00%)
T_1 $[N]$	685.4	685.4 (+0.00%)	685.4 (-0.00%)
T_2 $[N]$	2209.6	2209.6 (-0.00%)	2209.6 (-0.00%)
V_0 $[m/s]$	68.3	68.3 (+0.00%)	68.3 (-0.00%)
V_1 $[m/s]$	65.3	65.3 (+0.00%)	65.3 (-0.00%)
V_2 $[m/s]$	150.0	150.0 (-0.00%)	150.0 (-0.00%)
V_{stall} $[m/s]$	38.0	38.0 (+0.00%)	38.0 (+0.00%)
W_0 $[N]$	33853.0	33853.0 (-0.00%)	33853.0 (-0.00%)
W_1 $[N]$	31024.2	31024.2 (-0.00%)	31024.2 (-0.00%)
W_2 $[N]$	33853.0	33853.0 (-0.00%)	33853.0 (-0.00%)
W_{MTO} $[N]$	36935.6	36935.6 (-0.00%)	36935.6 (-0.00%)
W_{cap} $[N]$	4103.8	4103.8 (-0.00%)	4103.8 (-0.00%)
W_{eng} $[N]$	2805.8	2805.8 (-0.00%)	2805.8 (-0.00%)
$W_{fuel_{out}}$ $[N]$	3082.6	3082.6 (-0.00%)	3082.6 (+0.00%)
$W_{fuel_{ref}}$ $[N]$	2828.7	2828.7 (-0.00%)	2828.7 (+0.00%)
W_{pay} $[N]$	4905.0	4905.0 (-0.00%)	4905.0 (-0.00%)
W_{tilde} $[N]$	22410.8	22410.8 (-0.00%)	22410.8 (-0.00%)
W_{web} $[N]$	202.9	202.9 (-0.00%)	202.9 (+0.00%)
W_{wing} $[N]$	8613.4	8613.4 (-0.00%)	8613.4 (-0.00%)
W_{zfw} $[N]$	31024.2	31024.2 (-0.00%)	31024.2 (-0.00%)
eta_{0_0} $[-]$	0.27	0.27 (+0.00%)	0.27 (-0.00%)
eta_{0_1} $[-]$	0.27	0.27 (+0.00%)	0.27 (-0.00%)
eta_{0_2} $[-]$	0.279	0.279 (+0.00%)	0.279 (+0.00%)
eta_{i_0} $[-]$	0.907	0.907 (+0.00%)	0.907 (-0.00%)
eta_{i_1} $[-]$	0.907	0.907 (+0.00%)	0.907 (-0.00%)
eta_{i_2} $[-]$	0.939	0.939 (+0.00%)	0.939 (+0.00%)
eta_{prop_0} $[-]$	0.771	0.771 (+0.00%)	0.771 (-0.00%)
eta_{prop_1} $[-]$	0.771	0.771 (+0.00%)	0.771 (-0.00%)
eta_{prop_2} $[-]$	0.798	0.798 (+0.00%)	0.798 (+0.00%)
nu $[-]$	0.786	0.786 (-0.00%)	0.786 (-0.00%)
p $[-]$	1.9	1.9 (-0.00%)	1.9 (-0.00%)
q $[-]$	1.45	1.45 (-0.00%)	1.45 (-0.00%)
$t_{cap_{bar}}$ $[-]$	0.005	0.005 (-0.00%)	0.005 (-0.00%)
$t_{web_{bar}}$ $[-]$	0.001	0.001 (-0.00%)	0.001 (-0.00%)
tau $[-]$	0.15	0.15 (-0.00%)	0.15 (-0.00%)
z_{bre_0} $[-]$	0.087	0.087 (+0.00%)	0.087 (+0.00%)
z_{bre_1} $[-]$	0.087	0.087 (+0.00%)	0.087 (+0.00%)
Iterations	-	19.36 (0.00%)	17.13 (-11.54%)
Failures	-	0 (0.00%)	0 (0.00%)

C.6 The Hoburg Problem with One Black Box

Table C.16: Results of Solving the Hoburg-1 Problem With a Good Initial Guess

	Optimum	LSQP	SLCP
Obj $[N]$	5911.3	5911.3 (+0.00%)	5911.3 (+0.00%)
AR $[-]$	20.39	20.39 (-0.00%)	20.39 (-0.00%)
C_{D_0} $[-]$	0.013	0.013 (-0.00%)	0.013 (-0.00%)
C_{D_1} $[-]$	0.013	0.013 (-0.00%)	0.013 (-0.00%)
C_{D_2} $[-]$	0.008	0.008 (+0.00%)	0.008 (+0.00%)
C_{Dfuse_0} $[-]$	0.002	0.002 (-0.00%)	0.002 (-0.00%)
C_{Dfuse_1} $[-]$	0.002	0.002 (-0.00%)	0.002 (-0.00%)
C_{Dfuse_2} $[-]$	0.002	0.002 (-0.00%)	0.002 (-0.00%)
C_{Di_0} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Di_1} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Di_2} $[-]$	2.3e-4	2.3e-4 (+0.00%)	2.3e-4 (+0.00%)
C_{Dp_0} $[-]$	0.005	0.005 (-0.00%)	0.005 (-0.00%)
C_{Dp_1} $[-]$	0.005	0.005 (-0.00%)	0.005 (-0.00%)
C_{Dp_2} $[-]$	0.006	0.006 (+0.00%)	0.006 (+0.00%)
C_{L_0} $[-]$	0.574	0.574 (-0.00%)	0.574 (-0.00%)
C_{L_1} $[-]$	0.574	0.574 (-0.00%)	0.574 (-0.00%)
C_{L_2} $[-]$	0.119	0.119 (-0.00%)	0.119 (-0.00%)
$I_{cap_{bar}}$ $[-]$	2.02e-5	2.02e-5 (-0.00%)	2.02e-5 (-0.00%)
$M_{r_{bar}}$ $[N]$	36170.8	36170.9 (+0.00%)	36170.9 (+0.00%)
P_{max} $[W]$	1.19e+6	1.19e+6 (+0.00%)	1.19e+6 (+0.00%)
R $[km]$	5000.0	5000.0 (+0.00%)	5000.0 (+0.00%)
Re_0 $[-]$	4.66e+6	4.66e+6 (+0.00%)	4.66e+6 (+0.00%)
Re_1 $[-]$	4.47e+6	4.47e+6 (+0.00%)	4.47e+6 (+0.00%)
Re_2 $[-]$	1.03e+7	1.03e+7 (+0.00%)	1.03e+7 (+0.00%)
S $[m^2]$	27.84	27.84 (+0.00%)	27.84 (+0.00%)
T_0 $[N]$	747.0	747.0 (+0.00%)	747.0 (+0.00%)
T_1 $[N]$	685.4	685.4 (+0.00%)	685.4 (+0.00%)
T_2 $[N]$	2209.6	2209.7 (+0.00%)	2209.7 (+0.00%)
V_0 $[m/s]$	68.3	68.3 (+0.00%)	68.3 (+0.00%)
V_1 $[m/s]$	65.3	65.3 (+0.00%)	65.3 (+0.00%)
V_2 $[m/s]$	150.0	150.0 (-0.00%)	150.0 (+0.00%)
V_{stall} $[m/s]$	38.0	38.0 (+0.00%)	38.0 (+0.00%)
W_0 $[N]$	33853.0	33853.1 (+0.00%)	33853.1 (+0.00%)
W_1 $[N]$	31024.2	31024.3 (+0.00%)	31024.3 (+0.00%)
W_2 $[N]$	33853.0	33853.1 (+0.00%)	33853.1 (+0.00%)
W_{MTO} $[N]$	36935.6	36935.7 (+0.00%)	36935.7 (+0.00%)
W_{cap} $[N]$	4103.8	4103.8 (+0.00%)	4103.8 (+0.00%)
W_{eng} $[N]$	2805.8	2805.9 (+0.00%)	2805.9 (+0.00%)
$W_{fuel_{out}}$ $[N]$	3082.6	3082.6 (+0.00%)	3082.6 (+0.00%)
$W_{fuel_{ref}}$ $[N]$	2828.7	2828.7 (+0.00%)	2828.7 (+0.00%)
W_{pay} $[N]$	4905.0	4905.0 (-0.00%)	4905.0 (-0.00%)
W_{tilde} $[N]$	22410.8	22410.9 (+0.00%)	22410.9 (+0.00%)
W_{web} $[N]$	202.9	202.9 (+0.00%)	202.9 (+0.00%)
W_{wing} $[N]$	8613.4	8613.4 (+0.00%)	8613.4 (+0.00%)
W_{zfw} $[N]$	31024.2	31024.3 (+0.00%)	31024.3 (+0.00%)
eta_{0_0} $[-]$	0.27	0.27 (-0.00%)	0.27 (-0.00%)
eta_{0_1} $[-]$	0.27	0.27 (-0.00%)	0.27 (-0.00%)
eta_{0_2} $[-]$	0.279	0.279 (-0.00%)	0.279 (-0.00%)
eta_{i_0} $[-]$	0.907	0.907 (-0.00%)	0.907 (-0.00%)
eta_{i_1} $[-]$	0.907	0.907 (-0.00%)	0.907 (-0.00%)
eta_{i_2} $[-]$	0.939	0.939 (-0.00%)	0.939 (-0.00%)
eta_{prop_0} $[-]$	0.771	0.771 (-0.00%)	0.771 (-0.00%)
eta_{prop_1} $[-]$	0.771	0.771 (-0.00%)	0.771 (-0.00%)
eta_{prop_2} $[-]$	0.798	0.798 (-0.00%)	0.798 (-0.00%)
nu $[-]$	0.786	0.786 (-0.00%)	0.786 (-0.00%)
p $[-]$	1.9	1.9 (-0.00%)	1.9 (-0.00%)
q $[-]$	1.45	1.45 (-0.00%)	1.45 (-0.00%)
$t_{cap_{bar}}$ $[-]$	0.005	0.005 (-0.00%)	0.005 (-0.00%)
$t_{web_{bar}}$ $[-]$	0.001	0.001 (-0.00%)	0.001 (-0.00%)
tau $[-]$	0.15	0.15 (-0.00%)	0.15 (-0.00%)
z_{bre_0} $[-]$	0.087	0.087 (+0.00%)	0.087 (+0.00%)
z_{bre_1} $[-]$	0.087	0.087 (+0.00%)	0.087 (+0.00%)
Iterations	-	14.08 (0.00%)	14.27 (+1.36%)
Failures	-	0 (0.00%)	0 (0.00%)

Table C.17: Results of Solving the Hoburg-1 Problem With a Reasonable Initial Guess

	Optimum	LSQP	SLCP
Obj $[N]$	5911.3	5911.3 (+0.00%)	5911.3 (+0.00%)
AR $[-]$	20.39	20.39 (-0.00%)	20.39 (-0.00%)
C_{D_0} $[-]$	0.013	0.013 (-0.00%)	0.013 (-0.00%)
C_{D_1} $[-]$	0.013	0.013 (-0.00%)	0.013 (-0.00%)
C_{D_2} $[-]$	0.008	0.008 (+0.00%)	0.008 (+0.00%)
C_{Dfuse_0} $[-]$	0.002	0.002 (-0.00%)	0.002 (-0.00%)
C_{Dfuse_1} $[-]$	0.002	0.002 (-0.00%)	0.002 (-0.00%)
C_{Dfuse_2} $[-]$	0.002	0.002 (-0.00%)	0.002 (-0.00%)
C_{Di_0} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Di_1} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Di_2} $[-]$	2.3e-4	2.3e-4 (+0.00%)	2.3e-4 (+0.00%)
C_{Dp_0} $[-]$	0.005	0.005 (-0.00%)	0.005 (-0.00%)
C_{Dp_1} $[-]$	0.005	0.005 (-0.00%)	0.005 (-0.00%)
C_{Dp_2} $[-]$	0.006	0.006 (+0.00%)	0.006 (+0.00%)
C_{L_0} $[-]$	0.574	0.574 (-0.00%)	0.574 (-0.00%)
C_{L_1} $[-]$	0.574	0.574 (-0.00%)	0.574 (-0.00%)
C_{L_2} $[-]$	0.119	0.119 (-0.00%)	0.119 (-0.00%)
$I_{cap_{bar}}$ $[-]$	2.02e-5	2.02e-5 (-0.00%)	2.02e-5 (-0.00%)
$M_{r_{bar}}$ $[N]$	36170.8	36170.9 (+0.00%)	36170.9 (+0.00%)
P_{max} $[W]$	1.19e+6	1.19e+6 (+0.00%)	1.19e+6 (+0.00%)
R $[km]$	5000.0	5000.0 (+0.00%)	5000.0 (+0.00%)
Re_0 $[-]$	4.66e+6	4.66e+6 (+0.00%)	4.66e+6 (+0.00%)
Re_1 $[-]$	4.47e+6	4.47e+6 (+0.00%)	4.47e+6 (+0.00%)
Re_2 $[-]$	1.03e+7	1.03e+7 (+0.00%)	1.03e+7 (+0.00%)
S $[m^2]$	27.84	27.84 (+0.00%)	27.84 (+0.00%)
T_0 $[N]$	747.0	747.0 (+0.00%)	747.0 (+0.00%)
T_1 $[N]$	685.4	685.4 (+0.00%)	685.4 (+0.00%)
T_2 $[N]$	2209.6	2209.7 (+0.00%)	2209.7 (+0.00%)
V_0 $[m/s]$	68.3	68.3 (+0.00%)	68.3 (+0.00%)
V_1 $[m/s]$	65.3	65.3 (+0.00%)	65.3 (+0.00%)
V_2 $[m/s]$	150.0	150.0 (-0.00%)	150.0 (+0.00%)
V_{stall} $[m/s]$	38.0	38.0 (+0.00%)	38.0 (+0.00%)
W_0 $[N]$	33853.0	33853.1 (+0.00%)	33853.1 (+0.00%)
W_1 $[N]$	31024.2	31024.3 (+0.00%)	31024.3 (+0.00%)
W_2 $[N]$	33853.0	33853.1 (+0.00%)	33853.1 (+0.00%)
W_{MTO} $[N]$	36935.6	36935.7 (+0.00%)	36935.7 (+0.00%)
W_{cap} $[N]$	4103.8	4103.8 (+0.00%)	4103.8 (+0.00%)
W_{eng} $[N]$	2805.8	2805.9 (+0.00%)	2805.9 (+0.00%)
$W_{fuel_{out}}$ $[N]$	3082.6	3082.6 (+0.00%)	3082.6 (+0.00%)
$W_{fuel_{ref}}$ $[N]$	2828.7	2828.7 (+0.00%)	2828.7 (+0.00%)
W_{pay} $[N]$	4905.0	4905.0 (-0.00%)	4905.0 (-0.00%)
W_{tilde} $[N]$	22410.8	22410.9 (+0.00%)	22410.9 (+0.00%)
W_{web} $[N]$	202.9	202.9 (+0.00%)	202.9 (+0.00%)
W_{wing} $[N]$	8613.4	8613.4 (+0.00%)	8613.4 (+0.00%)
W_{zfw} $[N]$	31024.2	31024.3 (+0.00%)	31024.3 (+0.00%)
eta_{0_0} $[-]$	0.27	0.27 (-0.00%)	0.27 (-0.00%)
eta_{0_1} $[-]$	0.27	0.27 (-0.00%)	0.27 (-0.00%)
eta_{0_2} $[-]$	0.279	0.279 (-0.00%)	0.279 (-0.00%)
eta_{i_0} $[-]$	0.907	0.907 (-0.00%)	0.907 (-0.00%)
eta_{i_1} $[-]$	0.907	0.907 (-0.00%)	0.907 (-0.00%)
eta_{i_2} $[-]$	0.939	0.939 (-0.00%)	0.939 (-0.00%)
eta_{prop_0} $[-]$	0.771	0.771 (-0.00%)	0.771 (-0.00%)
eta_{prop_1} $[-]$	0.771	0.771 (-0.00%)	0.771 (-0.00%)
eta_{prop_2} $[-]$	0.798	0.798 (-0.00%)	0.798 (-0.00%)
nu $[-]$	0.786	0.786 (-0.00%)	0.786 (-0.00%)
p $[-]$	1.9	1.9 (-0.00%)	1.9 (-0.00%)
q $[-]$	1.45	1.45 (-0.00%)	1.45 (-0.00%)
$t_{cap_{bar}}$ $[-]$	0.005	0.005 (-0.00%)	0.005 (-0.00%)
$t_{web_{bar}}$ $[-]$	0.001	0.001 (-0.00%)	0.001 (-0.00%)
tau $[-]$	0.15	0.15 (-0.00%)	0.15 (-0.00%)
z_{bre_0} $[-]$	0.087	0.087 (+0.00%)	0.087 (+0.00%)
z_{bre_1} $[-]$	0.087	0.087 (+0.00%)	0.087 (+0.00%)
Iterations	-	17.82 (0.00%)	16.66 (-6.54%)
Failures	-	0 (0.00%)	0 (0.00%)

Table C.18: Results of Solving the Hoburg-1 Problem With a Poor Initial Guess

	Optimum	LSQP	SLCP
Obj $[N]$	5911.3	5911.3 (+0.00%)	5911.3 (+0.00%)
AR $[-]$	20.39	20.39 (-0.00%)	20.39 (-0.00%)
C_{D_0} $[-]$	0.013	0.013 (-0.00%)	0.013 (-0.00%)
C_{D_1} $[-]$	0.013	0.013 (-0.00%)	0.013 (-0.00%)
C_{D_2} $[-]$	0.008	0.008 (+0.00%)	0.008 (+0.00%)
C_{Dfuse_0} $[-]$	0.002	0.002 (-0.00%)	0.002 (-0.00%)
C_{Dfuse_1} $[-]$	0.002	0.002 (-0.00%)	0.002 (-0.00%)
C_{Dfuse_2} $[-]$	0.002	0.002 (-0.00%)	0.002 (-0.00%)
C_{Di_0} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Di_1} $[-]$	0.005	0.005 (-0.00%)	0.005 (+0.00%)
C_{Di_2} $[-]$	2.3e-4	2.3e-4 (+0.00%)	2.3e-4 (+0.00%)
C_{Dp_0} $[-]$	0.005	0.005 (-0.00%)	0.005 (-0.00%)
C_{Dp_1} $[-]$	0.005	0.005 (-0.00%)	0.005 (-0.00%)
C_{Dp_2} $[-]$	0.006	0.006 (+0.00%)	0.006 (+0.00%)
C_{L_0} $[-]$	0.574	0.574 (-0.00%)	0.574 (-0.00%)
C_{L_1} $[-]$	0.574	0.574 (-0.00%)	0.574 (-0.00%)
C_{L_2} $[-]$	0.119	0.119 (-0.00%)	0.119 (-0.00%)
$I_{cap_{bar}}$ $[-]$	2.02e-5	2.02e-5 (-0.00%)	2.02e-5 (-0.00%)
$M_{r_{bar}}$ $[N]$	36170.8	36170.9 (+0.00%)	36170.9 (+0.00%)
P_{max} $[W]$	1.19e+6	1.19e+6 (+0.00%)	1.19e+6 (+0.00%)
R $[km]$	5000.0	5000.0 (+0.00%)	5000.0 (+0.00%)
Re_0 $[-]$	4.66e+6	4.66e+6 (+0.00%)	4.66e+6 (+0.00%)
Re_1 $[-]$	4.47e+6	4.47e+6 (+0.00%)	4.47e+6 (+0.00%)
Re_2 $[-]$	1.03e+7	1.03e+7 (+0.00%)	1.03e+7 (+0.00%)
S $[m^2]$	27.84	27.84 (+0.00%)	27.84 (+0.00%)
T_0 $[N]$	747.0	747.0 (+0.00%)	747.0 (+0.00%)
T_1 $[N]$	685.4	685.4 (+0.00%)	685.4 (+0.00%)
T_2 $[N]$	2209.6	2209.7 (+0.00%)	2209.7 (+0.00%)
V_0 $[m/s]$	68.3	68.3 (+0.00%)	68.3 (+0.00%)
V_1 $[m/s]$	65.3	65.3 (+0.00%)	65.3 (+0.00%)
V_2 $[m/s]$	150.0	150.0 (-0.00%)	150.0 (+0.00%)
V_{stall} $[m/s]$	38.0	38.0 (+0.00%)	38.0 (+0.00%)
W_0 $[N]$	33853.0	33853.1 (+0.00%)	33853.1 (+0.00%)
W_1 $[N]$	31024.2	31024.3 (+0.00%)	31024.3 (+0.00%)
W_2 $[N]$	33853.0	33853.1 (+0.00%)	33853.1 (+0.00%)
W_{MTO} $[N]$	36935.6	36935.7 (+0.00%)	36935.7 (+0.00%)
W_{cap} $[N]$	4103.8	4103.8 (+0.00%)	4103.8 (+0.00%)
W_{eng} $[N]$	2805.8	2805.9 (+0.00%)	2805.9 (+0.00%)
$W_{fuel_{out}}$ $[N]$	3082.6	3082.6 (+0.00%)	3082.6 (+0.00%)
$W_{fuel_{ref}}$ $[N]$	2828.7	2828.7 (+0.00%)	2828.7 (+0.00%)
W_{pay} $[N]$	4905.0	4905.0 (-0.00%)	4905.0 (-0.00%)
W_{tilde} $[N]$	22410.8	22410.9 (+0.00%)	22410.9 (+0.00%)
W_{web} $[N]$	202.9	202.9 (+0.00%)	202.9 (+0.00%)
W_{wing} $[N]$	8613.4	8613.4 (+0.00%)	8613.4 (+0.00%)
W_{zfw} $[N]$	31024.2	31024.3 (+0.00%)	31024.3 (+0.00%)
eta_{0_0} $[-]$	0.27	0.27 (-0.00%)	0.27 (-0.00%)
eta_{0_1} $[-]$	0.27	0.27 (-0.00%)	0.27 (-0.00%)
eta_{0_2} $[-]$	0.279	0.279 (-0.00%)	0.279 (-0.00%)
eta_{i_0} $[-]$	0.907	0.907 (-0.00%)	0.907 (-0.00%)
eta_{i_1} $[-]$	0.907	0.907 (-0.00%)	0.907 (-0.00%)
eta_{i_2} $[-]$	0.939	0.939 (-0.00%)	0.939 (-0.00%)
eta_{prop_0} $[-]$	0.771	0.771 (-0.00%)	0.771 (-0.00%)
eta_{prop_1} $[-]$	0.771	0.771 (-0.00%)	0.771 (-0.00%)
eta_{prop_2} $[-]$	0.798	0.798 (-0.00%)	0.798 (-0.00%)
nu $[-]$	0.786	0.786 (-0.00%)	0.786 (-0.00%)
p $[-]$	1.9	1.9 (-0.00%)	1.9 (-0.00%)
q $[-]$	1.45	1.45 (-0.00%)	1.45 (-0.00%)
$t_{cap_{bar}}$ $[-]$	0.005	0.005 (-0.00%)	0.005 (-0.00%)
$t_{web_{bar}}$ $[-]$	0.001	0.001 (-0.00%)	0.001 (-0.00%)
tau $[-]$	0.15	0.15 (-0.00%)	0.15 (-0.00%)
z_{bre_0} $[-]$	0.087	0.087 (+0.00%)	0.087 (+0.00%)
z_{bre_1} $[-]$	0.087	0.087 (+0.00%)	0.087 (+0.00%)
Iterations	-	19.46 (0.00%)	17.73 (-8.93%)
Failures	-	0 (0.00%)	0 (0.00%)

C.7 The Hoburg Problem with Three Black Boxes

Table C.19: Results of Solving the Hoburg-3 Problem With a Good Initial Guess

	Optimum	LSQP	SLCP
Obj $[N]$	5911.3	5911.1 (-0.00%)	5911.1 (-0.00%)
AR $[-]$	20.39	20.39 (+0.00%)	20.39 (+0.00%)
C_{D_0} $[-]$	0.013	0.013 (-0.01%)	0.013 (-0.01%)
C_{D_1} $[-]$	0.013	0.013 (+0.00%)	0.013 (+0.00%)
C_{D_2} $[-]$	0.008	0.008 (+0.00%)	0.008 (+0.00%)
C_{Dfuse_0} $[-]$	0.002	0.002 (+0.00%)	0.002 (+0.00%)
C_{Dfuse_1} $[-]$	0.002	0.002 (+0.00%)	0.002 (+0.00%)
C_{Dfuse_2} $[-]$	0.002	0.002 (+0.00%)	0.002 (+0.00%)
C_{Di_0} $[-]$	0.005	0.005 (-0.01%)	0.005 (-0.01%)
C_{Di_1} $[-]$	0.005	0.005 (+0.00%)	0.005 (+0.00%)
C_{Di_2} $[-]$	2.3e-4	2.3e-4 (+0.00%)	2.3e-4 (+0.00%)
C_{Dp_0} $[-]$	0.005	0.005 (-0.03%)	0.005 (-0.03%)
C_{Dp_1} $[-]$	0.005	0.005 (+0.01%)	0.005 (+0.01%)
C_{Dp_2} $[-]$	0.006	0.006 (+0.00%)	0.006 (+0.00%)
C_{L_0} $[-]$	0.574	0.574 (-0.00%)	0.574 (-0.00%)
C_{L_1} $[-]$	0.574	0.574 (+0.00%)	0.574 (+0.00%)
C_{L_2} $[-]$	0.119	0.119 (+0.00%)	0.119 (+0.00%)
$I_{cap_{bar}}$ $[-]$	2.02e-5	2.02e-5 (+0.00%)	2.02e-5 (+0.00%)
$M_{r_{bar}}$ $[N]$	36170.8	36171.2 (+0.00%)	36171.2 (+0.00%)
P_{max} $[W]$	1.19e+6	1.19e+6 (+0.00%)	1.19e+6 (+0.00%)
R $[km]$	5000.0	5000.0 (+0.00%)	5000.0 (+0.00%)
Re_0 $[-]$	4.66e+6	4.66e+6 (+0.00%)	4.66e+6 (+0.00%)
Re_1 $[-]$	4.47e+6	4.47e+6 (-0.00%)	4.47e+6 (-0.00%)
Re_2 $[-]$	1.03e+7	1.03e+7 (-0.00%)	1.03e+7 (-0.00%)
S $[m^2]$	27.84	27.84 (-0.00%)	27.84 (-0.00%)
T_0 $[N]$	747.0	746.9 (-0.01%)	746.9 (-0.01%)
T_1 $[N]$	685.4	685.4 (+0.00%)	685.4 (+0.00%)
T_2 $[N]$	2209.6	2209.7 (+0.00%)	2209.7 (+0.00%)
V_0 $[m/s]$	68.3	68.3 (+0.00%)	68.3 (+0.00%)
V_1 $[m/s]$	65.3	65.3 (-0.00%)	65.3 (-0.00%)
V_2 $[m/s]$	150.0	150.0 (-0.00%)	150.0 (-0.00%)
V_{stall} $[m/s]$	38.0	38.0 (+0.00%)	38.0 (+0.00%)
W_0 $[N]$	33853.0	33853.3 (+0.00%)	33853.3 (+0.00%)
W_1 $[N]$	31024.2	31024.4 (+0.00%)	31024.4 (+0.00%)
W_2 $[N]$	33853.0	33853.3 (+0.00%)	33853.3 (+0.00%)
W_{MTO} $[N]$	36935.6	36935.5 (-0.00%)	36935.5 (-0.00%)
W_{cap} $[N]$	4103.8	4103.8 (+0.00%)	4103.8 (+0.00%)
W_{eng} $[N]$	2805.8	2805.9 (+0.00%)	2805.9 (+0.00%)
$W_{fuel_{out}}$ $[N]$	3082.6	3082.2 (-0.01%)	3082.2 (-0.01%)
$W_{fuel_{ref}}$ $[N]$	2828.7	2828.9 (+0.00%)	2828.9 (+0.00%)
W_{pay} $[N]$	4905.0	4905.0 (-0.00%)	4905.0 (-0.00%)
W_{tilde} $[N]$	22410.8	22410.9 (+0.00%)	22410.9 (+0.00%)
W_{web} $[N]$	202.9	202.9 (+0.00%)	202.9 (+0.00%)
W_{wing} $[N]$	8613.4	8613.5 (+0.00%)	8613.5 (+0.00%)
W_{zfw} $[N]$	31024.2	31024.4 (+0.00%)	31024.4 (+0.00%)
eta_{0_0} $[-]$	0.27	0.27 (+0.00%)	0.27 (+0.00%)
eta_{0_1} $[-]$	0.27	0.27 (-0.00%)	0.27 (-0.00%)
eta_{0_2} $[-]$	0.279	0.279 (-0.00%)	0.279 (-0.00%)
eta_{i_0} $[-]$	0.907	0.908 (+0.00%)	0.908 (+0.00%)
eta_{i_1} $[-]$	0.907	0.907 (-0.00%)	0.907 (-0.00%)
eta_{i_2} $[-]$	0.939	0.939 (-0.00%)	0.939 (-0.00%)
eta_{prop_0} $[-]$	0.771	0.771 (+0.00%)	0.771 (+0.00%)
eta_{prop_1} $[-]$	0.771	0.771 (-0.00%)	0.771 (-0.00%)
eta_{prop_2} $[-]$	0.798	0.798 (-0.00%)	0.798 (-0.00%)
nu $[-]$	0.786	0.786 (-0.00%)	0.786 (-0.00%)
p $[-]$	1.9	1.9 (-0.00%)	1.9 (-0.00%)
q $[-]$	1.45	1.45 (-0.00%)	1.45 (-0.00%)
$t_{cap_{bar}}$ $[-]$	0.005	0.005 (+0.00%)	0.005 (+0.00%)
$t_{web_{bar}}$ $[-]$	0.001	0.001 (+0.00%)	0.001 (+0.00%)
tau $[-]$	0.15	0.15 (-0.00%)	0.15 (-0.00%)
z_{bre_0} $[-]$	0.087	0.087 (-0.01%)	0.087 (-0.01%)
z_{bre_1} $[-]$	0.087	0.087 (+0.00%)	0.087 (+0.00%)
Iterations	-	14.47 (0.00%)	14.34 (-0.90%)
Failures	-	0 (0.00%)	0 (0.00%)

Table C.20: Results of Solving the Hoburg-3 Problem With a Reasonable Initial Guess

	Optimum	LSQP	SLCP
Obj $[N]$	5911.3	5911.1 (-0.00%)	5911.1 (-0.00%)
AR $[-]$	20.39	20.39 (+0.00%)	20.39 (+0.00%)
C_{D_0} $[-]$	0.013	0.013 (-0.01%)	0.013 (-0.01%)
C_{D_1} $[-]$	0.013	0.013 (+0.00%)	0.013 (+0.00%)
C_{D_2} $[-]$	0.008	0.008 (+0.00%)	0.008 (+0.00%)
C_{Dfuse_0} $[-]$	0.002	0.002 (+0.00%)	0.002 (+0.00%)
C_{Dfuse_1} $[-]$	0.002	0.002 (+0.00%)	0.002 (+0.00%)
C_{Dfuse_2} $[-]$	0.002	0.002 (+0.00%)	0.002 (+0.00%)
C_{Di_0} $[-]$	0.005	0.005 (-0.01%)	0.005 (-0.01%)
C_{Di_1} $[-]$	0.005	0.005 (+0.00%)	0.005 (+0.00%)
C_{Di_2} $[-]$	2.3e-4	2.3e-4 (+0.00%)	2.3e-4 (+0.00%)
C_{Dp_0} $[-]$	0.005	0.005 (-0.03%)	0.005 (-0.03%)
C_{Dp_1} $[-]$	0.005	0.005 (+0.01%)	0.005 (+0.01%)
C_{Dp_2} $[-]$	0.006	0.006 (+0.00%)	0.006 (+0.00%)
C_{L_0} $[-]$	0.574	0.574 (-0.00%)	0.574 (-0.00%)
C_{L_1} $[-]$	0.574	0.574 (+0.00%)	0.574 (+0.00%)
C_{L_2} $[-]$	0.119	0.119 (+0.00%)	0.119 (+0.00%)
$I_{cap_{bar}}$ $[-]$	2.02e-5	2.02e-5 (+0.00%)	2.02e-5 (+0.00%)
$M_{r_{bar}}$ $[N]$	36170.8	36171.2 (+0.00%)	36171.2 (+0.00%)
P_{max} $[W]$	1.19e+6	1.19e+6 (+0.00%)	1.19e+6 (+0.00%)
R $[km]$	5000.0	5000.0 (+0.00%)	5000.0 (+0.00%)
Re_0 $[-]$	4.66e+6	4.66e+6 (+0.00%)	4.66e+6 (+0.00%)
Re_1 $[-]$	4.47e+6	4.47e+6 (-0.00%)	4.47e+6 (-0.00%)
Re_2 $[-]$	1.03e+7	1.03e+7 (-0.00%)	1.03e+7 (-0.00%)
S $[m^2]$	27.84	27.84 (-0.00%)	27.84 (-0.00%)
T_0 $[N]$	747.0	746.9 (-0.01%)	746.9 (-0.01%)
T_1 $[N]$	685.4	685.4 (+0.00%)	685.4 (+0.00%)
T_2 $[N]$	2209.6	2209.7 (+0.00%)	2209.7 (+0.00%)
V_0 $[m/s]$	68.3	68.3 (+0.00%)	68.3 (+0.00%)
V_1 $[m/s]$	65.3	65.3 (-0.00%)	65.3 (-0.00%)
V_2 $[m/s]$	150.0	150.0 (-0.00%)	150.0 (+0.00%)
V_{stall} $[m/s]$	38.0	38.0 (+0.00%)	38.0 (+0.00%)
W_0 $[N]$	33853.0	33853.3 (+0.00%)	33853.3 (+0.00%)
W_1 $[N]$	31024.2	31024.4 (+0.00%)	31024.4 (+0.00%)
W_2 $[N]$	33853.0	33853.3 (+0.00%)	33853.3 (+0.00%)
W_{MTO} $[N]$	36935.6	36935.5 (-0.00%)	36935.5 (-0.00%)
W_{cap} $[N]$	4103.8	4103.8 (+0.00%)	4103.8 (+0.00%)
W_{eng} $[N]$	2805.8	2805.9 (+0.00%)	2805.9 (+0.00%)
$W_{fuel_{out}}$ $[N]$	3082.6	3082.2 (-0.01%)	3082.2 (-0.01%)
$W_{fuel_{ref}}$ $[N]$	2828.7	2828.9 (+0.00%)	2828.9 (+0.00%)
W_{pay} $[N]$	4905.0	4905.0 (-0.00%)	4905.0 (-0.00%)
W_{tilde} $[N]$	22410.8	22410.9 (+0.00%)	22410.9 (+0.00%)
W_{web} $[N]$	202.9	202.9 (+0.00%)	202.9 (+0.00%)
W_{wing} $[N]$	8613.4	8613.5 (+0.00%)	8613.5 (+0.00%)
W_{zfw} $[N]$	31024.2	31024.4 (+0.00%)	31024.4 (+0.00%)
eta_{0_0} $[-]$	0.27	0.27 (+0.00%)	0.27 (+0.00%)
eta_{0_1} $[-]$	0.27	0.27 (-0.00%)	0.27 (-0.00%)
eta_{0_2} $[-]$	0.279	0.279 (-0.00%)	0.279 (-0.00%)
eta_{i_0} $[-]$	0.907	0.908 (+0.00%)	0.908 (+0.00%)
eta_{i_1} $[-]$	0.907	0.907 (-0.00%)	0.907 (-0.00%)
eta_{i_2} $[-]$	0.939	0.939 (-0.00%)	0.939 (-0.00%)
eta_{prop_0} $[-]$	0.771	0.771 (+0.00%)	0.771 (+0.00%)
eta_{prop_1} $[-]$	0.771	0.771 (-0.00%)	0.771 (-0.00%)
eta_{prop_2} $[-]$	0.798	0.798 (-0.00%)	0.798 (-0.00%)
nu $[-]$	0.786	0.786 (-0.00%)	0.786 (-0.00%)
p $[-]$	1.9	1.9 (-0.00%)	1.9 (-0.00%)
q $[-]$	1.45	1.45 (-0.00%)	1.45 (-0.00%)
$t_{cap_{bar}}$ $[-]$	0.005	0.005 (+0.00%)	0.005 (+0.00%)
$t_{web_{bar}}$ $[-]$	0.001	0.001 (+0.00%)	0.001 (+0.00%)
tau $[-]$	0.15	0.15 (-0.00%)	0.15 (-0.00%)
z_{bre_0} $[-]$	0.087	0.087 (-0.01%)	0.087 (-0.01%)
z_{bre_1} $[-]$	0.087	0.087 (+0.00%)	0.087 (+0.00%)
Iterations	-	17.80 (0.00%)	16.95 (-4.76%)
Failures	-	0 (0.00%)	0 (0.00%)

Table C.21: Results of Solving the Hoburg-3 Problem With a Poor Initial Guess

	Optimum	LSQP	SLCP
Obj $[N]$	5911.3	5911.1 (-0.00%)	5911.1 (-0.00%)
AR $[-]$	20.39	20.39 (+0.00%)	20.39 (+0.00%)
C_{D_0} $[-]$	0.013	0.013 (-0.01%)	0.013 (-0.01%)
C_{D_1} $[-]$	0.013	0.013 (+0.00%)	0.013 (+0.00%)
C_{D_2} $[-]$	0.008	0.008 (+0.00%)	0.008 (+0.00%)
C_{Dfuse_0} $[-]$	0.002	0.002 (+0.00%)	0.002 (+0.00%)
C_{Dfuse_1} $[-]$	0.002	0.002 (+0.00%)	0.002 (+0.00%)
C_{Dfuse_2} $[-]$	0.002	0.002 (+0.00%)	0.002 (+0.00%)
C_{Di_0} $[-]$	0.005	0.005 (-0.01%)	0.005 (-0.01%)
C_{Di_1} $[-]$	0.005	0.005 (+0.00%)	0.005 (+0.00%)
C_{Di_2} $[-]$	2.3e-4	2.3e-4 (+0.00%)	2.3e-4 (+0.00%)
C_{Dp_0} $[-]$	0.005	0.005 (-0.03%)	0.005 (-0.03%)
C_{Dp_1} $[-]$	0.005	0.005 (+0.01%)	0.005 (+0.01%)
C_{Dp_2} $[-]$	0.006	0.006 (+0.00%)	0.006 (+0.00%)
C_{L_0} $[-]$	0.574	0.574 (-0.00%)	0.574 (-0.00%)
C_{L_1} $[-]$	0.574	0.574 (+0.00%)	0.574 (+0.00%)
C_{L_2} $[-]$	0.119	0.119 (+0.00%)	0.119 (+0.00%)
$I_{cap_{bar}}$ $[-]$	2.02e-5	2.02e-5 (+0.00%)	2.02e-5 (+0.00%)
$M_{r_{bar}}$ $[N]$	36170.8	36171.2 (+0.00%)	36171.2 (+0.00%)
P_{max} $[W]$	1.19e+6	1.19e+6 (+0.00%)	1.19e+6 (+0.00%)
R $[km]$	5000.0	5000.0 (+0.00%)	5000.0 (+0.00%)
Re_0 $[-]$	4.66e+6	4.66e+6 (+0.00%)	4.66e+6 (+0.00%)
Re_1 $[-]$	4.47e+6	4.47e+6 (-0.00%)	4.47e+6 (-0.00%)
Re_2 $[-]$	1.03e+7	1.03e+7 (-0.00%)	1.03e+7 (-0.00%)
S $[m^2]$	27.84	27.84 (-0.00%)	27.84 (-0.00%)
T_0 $[N]$	747.0	746.9 (-0.01%)	746.9 (-0.01%)
T_1 $[N]$	685.4	685.4 (+0.00%)	685.4 (+0.00%)
T_2 $[N]$	2209.6	2209.7 (+0.00%)	2209.7 (+0.00%)
V_0 $[m/s]$	68.3	68.3 (+0.00%)	68.3 (+0.00%)
V_1 $[m/s]$	65.3	65.3 (-0.00%)	65.3 (-0.00%)
V_2 $[m/s]$	150.0	150.0 (-0.00%)	150.0 (+0.00%)
V_{stall} $[m/s]$	38.0	38.0 (+0.00%)	38.0 (+0.00%)
W_0 $[N]$	33853.0	33853.3 (+0.00%)	33853.3 (+0.00%)
W_1 $[N]$	31024.2	31024.4 (+0.00%)	31024.4 (+0.00%)
W_2 $[N]$	33853.0	33853.3 (+0.00%)	33853.3 (+0.00%)
W_{MTO} $[N]$	36935.6	36935.5 (-0.00%)	36935.5 (-0.00%)
W_{cap} $[N]$	4103.8	4103.8 (+0.00%)	4103.8 (+0.00%)
W_{eng} $[N]$	2805.8	2805.9 (+0.00%)	2805.9 (+0.00%)
$W_{fuel_{out}}$ $[N]$	3082.6	3082.2 (-0.01%)	3082.2 (-0.01%)
$W_{fuel_{ref}}$ $[N]$	2828.7	2828.9 (+0.00%)	2828.9 (+0.00%)
W_{pay} $[N]$	4905.0	4905.0 (-0.00%)	4905.0 (-0.00%)
W_{tilde} $[N]$	22410.8	22410.9 (+0.00%)	22410.9 (+0.00%)
W_{web} $[N]$	202.9	202.9 (+0.00%)	202.9 (+0.00%)
W_{wing} $[N]$	8613.4	8613.5 (+0.00%)	8613.5 (+0.00%)
W_{zfw} $[N]$	31024.2	31024.4 (+0.00%)	31024.4 (+0.00%)
eta_{0_0} $[-]$	0.27	0.27 (+0.00%)	0.27 (+0.00%)
eta_{0_1} $[-]$	0.27	0.27 (-0.00%)	0.27 (-0.00%)
eta_{0_2} $[-]$	0.279	0.279 (-0.00%)	0.279 (-0.00%)
eta_{i_0} $[-]$	0.907	0.908 (+0.00%)	0.908 (+0.00%)
eta_{i_1} $[-]$	0.907	0.907 (-0.00%)	0.907 (-0.00%)
eta_{i_2} $[-]$	0.939	0.939 (-0.00%)	0.939 (-0.00%)
eta_{prop_0} $[-]$	0.771	0.771 (+0.00%)	0.771 (+0.00%)
eta_{prop_1} $[-]$	0.771	0.771 (-0.00%)	0.771 (-0.00%)
eta_{prop_2} $[-]$	0.798	0.798 (-0.00%)	0.798 (-0.00%)
nu $[-]$	0.786	0.786 (-0.00%)	0.786 (-0.00%)
p $[-]$	1.9	1.9 (-0.00%)	1.9 (-0.00%)
q $[-]$	1.45	1.45 (-0.00%)	1.45 (-0.00%)
$t_{cap_{bar}}$ $[-]$	0.005	0.005 (+0.00%)	0.005 (+0.00%)
$t_{web_{bar}}$ $[-]$	0.001	0.001 (+0.00%)	0.001 (+0.00%)
tau $[-]$	0.15	0.15 (-0.00%)	0.15 (-0.00%)
z_{bre_0} $[-]$	0.087	0.087 (-0.01%)	0.087 (-0.01%)
z_{bre_1} $[-]$	0.087	0.087 (+0.00%)	0.087 (+0.00%)
Iterations	-	19.25 (0.00%)	18.03 (-6.33%)
Failures	-	0 (0.00%)	0 (0.00%)

Appendix D

Full Report of Design Results for the Hoburg Problem

D.1 The Original Geometric Program

```
Objective
---------
   5911.3147112438 newton

Variables
---------
   AR          :        20.3872467861       [-]         aspect ratio
   C_D_0       :         0.0126678997       [-]         Drag Coefficient, segment 0
   C_D_1       :         0.0126820388       [-]         Drag Coefficient, segment 1
   C_D_2       :         0.0077600000       [-]         Drag Coefficient, segment 2
   C_Dfuse_0   :         0.0017959321       [-]         Fuselage Drag Coefficient, segment 0
   C_Dfuse_1   :         0.0017959321       [-]         Fuselage Drag Coefficient, segment 1
   C_Dfuse_2   :         0.0017959321       [-]         Fuselage Drag Coefficient, segment 2
   C_Di_0      :         0.0054170272       [-]         Induced Drag Coefficient, segment 0
   C_Di_1      :         0.0054164263       [-]         Induced Drag Coefficient, segment 1
   C_Di_2      :         0.0002323015       [-]         Induced Drag Coefficient, segment 2
   C_Dp_0      :         0.0054549404       [-]         Wing Profile Drag Coefficient, segment 0
   C_Dp_1      :         0.0054696804       [-]         Wing Profile Drag Coefficient, segment 1
   C_Dp_2      :         0.0057317664       [-]         Wing Profile Drag Coefficient, segment 2
   C_L_0       :         0.5741118849       [-]         Lift Coefficient, segment 0
   C_L_1       :         0.5740800428       [-]         Lift Coefficient, segment 1
   C_L_2       :         0.1188891116       [-]         Lift Coefficient, segment 2
   I_cap_bar   :         0.0000202099       [-]         Area moment of inertia of cap on 2D cross section, normalized by chord^4
   M_r_bar     :     36170.8491356912      newton       Root Bending unit per unit chord
   P_max       :   1186091.8306055914       watt        Maximum Engine Power
   R           :      4999.9999999588    kilometer      Single Segment range
   Re_0        :   4664753.2886835784       [-]         Reynolds Number, segment 0
   Re_1        :   4465734.2417956591       [-]         Reynolds Number, segment 1
   Re_2        :  10250756.2650345117       [-]         Reynolds Number, segment 2
   S           :        27.8406965819    meter ** 2     total wing area
   T_0         :       746.9730729405      newton       Thrust, segment 0
   T_1         :       685.3584916379      newton       Thrust, segment 1
   T_2         :      2209.6144895427      newton       Thrust, segment 2
   V_0         :        68.2596459520   meter / second   Velocity, segment 0
   V_1         :        65.3473869669   meter / second   Velocity, segment 1
   V_2         :       150.0000000190   meter / second   Velocity, segment 2
   V_stall     :        37.9999999967   meter / second   stall speed
   W_0         :     33852.9772371071      newton       Weight, segment 0
   W_1         :     31024.2413894609      newton       Weight, segment 1
   W_2         :     33852.9772370677      newton       Weight, segment 2
   W_MTO       :     36935.5561313216      newton       Maximum Takeoff Weight
   W_cap       :      4103.7934642141      newton       Weight of Wing Spar Cap
   W_eng       :      2805.8209062652      newton       Engine Weight
   W_fuel_out  :      3082.5788637203      newton       Weight of fuel, outbound
   W_fuel_ret  :      2828.7358475192      newton       Weight of fuel, return
   W_pay       :      4905.0000004167      newton       Payload Weight
   W_tilde     :     22410.8209029816      newton       Dry Weight, no wing
   W_web       :       202.9167795476      newton       Weight of Wing Spar Shear Web
   W_wing      :      8613.4204859094      newton       wing weight
   W_zfw       :     31024.2413888748      newton       Zero Fuel Weight
   eta_0_0     :         0.2699809027       [-]         Overall Efficiency, segment 0
   eta_0_1     :         0.2699553935       [-]         Overall Efficiency, segment 1
   eta_0_2     :         0.2794405671       [-]         Overall Efficiency, segment 2
   eta_i_0     :         0.9074988326       [-]         Inviscid Propeller Efficiency, segment 0
   eta_i_1     :         0.9074130873       [-]         Inviscid Propeller Efficiency, segment 1
```

```
   eta_i_2     :         0.9392960238        [-]         Inviscid Propeller Efficiency, segment 2
   eta_prop_0  :         0.7713740077        [-]         Propeller Efficiency, segment 0
   eta_prop_1  :         0.7713011242        [-]         Propeller Efficiency, segment 1
   eta_prop_2  :         0.7984016202        [-]         Propeller Efficiency, segment 2
   nu          :         0.7859963824        [-]         Placeholder, (1+lam_w+lam_w**2)/(1+lam_w)**2
   p           :         1.9000000001        [-]         Dummy Variable (1+2*lam_w)
   q           :         1.4500000001        [-]         Dummy Variable (1+lam_w)
   t_cap_bar   :         0.0045441492        [-]         Spar cap thickness per unit chord
   t_web_bar   :         0.0009986252        [-]         Spar web thickness per unit chord
   tau         :         0.1500000000        [-]         airfoil thickness to chord ratio
   z_bre_0     :         0.0871477567        [-]         Breguet Range Factor, segment 0
   z_bre_1     :         0.0872581093        [-]         Breguet Range Factor, segment 1

Constants
---------
   A_prop           :         0.7850000000          meter ** 2          Disk area of propeller
   CDA0             :         0.0500000000          meter ** 2          fuselage drag area
   C_Lmax           :         1.5000000000             [-]              max CL with flaps down
   N_lift           :         6                        [-]              Ultimate Load Factor
   W_fixed          :     14700                       newton           fixed weight
   e                :         0.9500000000             [-]              Oswald efficiency factor
   eta_eng          :         0.3500000000             [-]              Engine Efficiency
   eta_v            :         0.8500000000             [-]              Propeller Viscous Efficiency
   f_wadd           :         2                        [-]              Added Weight Fraction
   g                :         9.8100000000     meter / second ** 2      Gravitational constant
   h_fuel           :  46000000                  joule / kilogram       Fuel Specific energy density
   k_ew             :         0.0372000000    newton / watt ** 0.803    Constant for engine weight
   mu               :         0.0000155460  kilogram / meter / second   viscosity of air
   r_h              :         0.7500000000             [-]              Ratio of height at rear spar to maximum wing height
   rho              :         0.9091220000    kilogram / meter ** 3     density of air
   rho_SL           :         1.2250000000    kilogram / meter ** 3     density of air, sea level
   rho_cap          :      2700               kilogram / meter ** 3     Density of wing cap material (aluminum)
   rho_web          :      2700               kilogram / meter ** 3     Density of wing web material (aluminum)
   sigma_max        :       310                     megapascal          Ultimate tensile stress of aluminum
   sigma_max_shear  :       167                     megapascal          Ultimate shear stress of aluminum
   w_bar            :         0.5000000000             [-]              Ratio of spar box width to chord length

Sensitivities
-------------
   Sensitivities not computed

Solve Report
------------
   Solve Method           :  Geometric Program
   Classification         :  Convex
   Solver                 :  cvxopt
```

D.2 Using MSES with NACA 24XX

```
Objective
---------
   6018.5962509451 newton

Variables
---------
   AR                :        20.1478173279        [-]         aspect ratio
   Alpha_p20_MSES_0  :        22.9619956179      degree        Angle of Attack + 20deg from MSES, segment 0
   Alpha_p20_MSES_1  :        22.7872678948      degree        Angle of Attack + 20deg from MSES, segment 1
   Alpha_p20_MSES_2  :        18.8715848356      degree        Angle of Attack + 20deg from MSES, segment 2
   C_D_0             :         0.0128388105        [-]         Drag Coefficient, segment 0
   C_D_1             :         0.0124483493        [-]         Drag Coefficient, segment 1
   C_D_2             :         0.0074695690        [-]         Drag Coefficient, segment 2
   C_Dfuse_0         :         0.0018066754        [-]         Fuselage Drag Coefficient, segment 0
   C_Dfuse_1         :         0.0018066754        [-]         Fuselage Drag Coefficient, segment 1
   C_Dfuse_2         :         0.0018066754        [-]         Fuselage Drag Coefficient, segment 2
   C_Di_0            :         0.0053463531        [-]         Induced Drag Coefficient, segment 0
   C_Di_1            :         0.0049967085        [-]         Induced Drag Coefficient, segment 1
   C_Di_2            :         0.0002339627        [-]         Induced Drag Coefficient, segment 2
   C_Dp_0            :         0.0056857820        [-]         Wing Profile Drag Coefficient, segment 0
   C_Dp_1            :         0.0056449654        [-]         Wing Profile Drag Coefficient, segment 1
   C_Dp_2            :         0.0054289308        [-]         Wing Profile Drag Coefficient, segment 2
   C_L_0             :         0.5669752449        [-]         Lift Coefficient, segment 0
   C_L_1             :         0.5480078573        [-]         Lift Coefficient, segment 1
   C_L_2             :         0.1186107444        [-]         Lift Coefficient, segment 2
   C_L_MSES_0        :         0.5669752449        [-]         Lift Coefficient from MSES, segment 0
   C_L_MSES_1        :         0.5480078573        [-]         Lift Coefficient from MSES, segment 1
   C_L_MSES_2        :         0.1186107444        [-]         Lift Coefficient from MSES, segment 2
   I_cap_bar         :         0.0000197648        [-]         Area moment of inertia of cap on 2D cross section, normalized by chord^4
   M_r_bar           :     35581.7259094296      newton        Root Bending unit per unit chord
   P_max             :   1132102.1152377108       watt         Maximum Engine Power
   R                 :      5000.0000000002    kilometer      Single Segment range
   Re_0              :   4702364.6341194399        [-]         Reynolds Number, segment 0
   Re_1              :   4574400.4904455515        [-]         Reynolds Number, segment 1
   Re_2              :  10280781.6066629589        [-]         Reynolds Number, segment 2
   S                 :        27.6751449713    meter ** 2      total wing area
   T_0               :       760.2393574526      newton        Thrust, segment 0
   T_1               :       697.3348547844      newton        Thrust, segment 1
   T_2               :      2114.2684424621      newton        Thrust, segment 2
   V_0               :        68.6090864268   meter / second   Velocity, segment 0
   V_1               :        66.7419296097   meter / second   Velocity, segment 1
   V_2               :       150.0000000014   meter / second   Velocity, segment 2
   V_stall           :        38.0000000000   meter / second   stall speed
   W_0               :     33572.8823879721      newton        Weight, segment 0
   W_1               :     30697.3267036892      newton        Weight, segment 1
   W_2               :     33572.8823879721      newton        Weight, segment 2
   W_MTO             :     36715.9229547196      newton        Maximum Takeoff Weight
   W_cap             :      3994.5694530491      newton        Weight of Wing Spar Cap
   W_eng             :      2702.7948120874      newton        Engine Weight
   W_fuel_out        :      3143.0405666647      newton        Weight of fuel, outbound
   W_fuel_ret        :      2875.5556842804      newton        Weight of fuel, return
   W_pay             :      4905.0000000050      newton        Payload Weight
   W_tilde           :     22307.7948120833      newton        Dry Weight, no wing
   W_web             :       200.1964928109      newton        Weight of Wing Spar Shear Web
   W_wing            :      8389.5318916055      newton        wing weight
   W_zfw             :     30697.3267036870      newton        Zero Fuel Weight
```

```
   eta_0_0          :         0.2698107120       [-]         Overall Efficiency, segment 0
   eta_0_1          :         0.2705141528       [-]         Overall Efficiency, segment 1
   eta_0_2          :         0.2801339763       [-]         Overall Efficiency, segment 2
   eta_i_0          :         0.9069267632       [-]         Inviscid Propeller Efficiency, segment 0
   eta_i_1          :         0.9092912700       [-]         Inviscid Propeller Efficiency, segment 1
   eta_i_2          :         0.9416268111       [-]         Inviscid Propeller Efficiency, segment 2
   eta_prop_0       :         0.7708877487       [-]         Propeller Efficiency, segment 0
   eta_prop_1       :         0.7728975795       [-]         Propeller Efficiency, segment 1
   eta_prop_2       :         0.8003827894       [-]         Propeller Efficiency, segment 2
   nu               :         0.7859963823       [-]         Placeholder, (1+lam_w+lam_w**2)/(1+lam_w)**2
   p                :         1.9000000000       [-]         Dummy Variable (1+2*lam_w)
   q                :         1.4500000000       [-]         Dummy Variable (1+lam_w)
   t_cap_bar        :         0.0044366688       [-]         Spar cap thickness per unit chord
   t_web_bar        :         0.0009882367       [-]         Spar web thickness per unit chord
   tau              :         0.1499999974       [-]         airfoil thickness to chord ratio
   z_bre_0          :         0.0894919118       [-]         Breguet Range Factor, segment 0
   z_bre_1          :         0.0895431422       [-]         Breguet Range Factor, segment 1

Constants
---------
   A_prop           :         0.7850000000          meter ** 2          Disk area of propeller
   CDA0             :         0.0500000000          meter ** 2          fuselage drag area
   C_Lmax           :         1.5000000000             [-]              max CL with flaps down
   N_lift           :         6                        [-]              Ultimate Load Factor
   W_fixed          :     14700                       newton            fixed weight
   e                :         0.9500000000             [-]              Oswald efficiency factor
   eta_eng          :         0.3500000000             [-]              Engine Efficiency
   eta_v            :         0.8500000000             [-]              Propeller Viscous Efficiency
   f_wadd           :         2                        [-]              Added Weight Fraction
   g                :         9.8100000000     meter / second ** 2      Gravitational constant
   h_fuel           :  46000000                  joule / kilogram       Fuel Specific energy density
   k_ew             :         0.0372000000    newton / watt ** 0.803    Constant for engine weight
   mu               :         0.0000155460   kilogram / meter / second   viscosity of air
   r_h              :         0.7500000000             [-]              Ratio of height at rear spar to maximum wing height
   rho              :         0.9091220000    kilogram / meter ** 3     density of air
   rho_SL           :         1.2250000000    kilogram / meter ** 3     density of air, sea level
   rho_cap          :      2700               kilogram / meter ** 3     Density of wing cap material (aluminum)
   rho_web          :      2700               kilogram / meter ** 3     Density of wing web material (aluminum)
   sigma_max        :       310                     megapascal          Ultimate tensile stress of aluminum
   sigma_max_shear  :       167                     megapascal          Ultimate shear stress of aluminum
   w_bar            :         0.5000000000             [-]              Ratio of spar box width to chord length

Sensitivities
-------------
   Sensitivities not available for optimizations of type Sequential Log Convex Program

Solve Report
------------
   Solve Method           :  Sequential Log Convex Program
   Classification         :  Iterative
   Solver                 :  cvxopt
   Number of Iterations   :  29
   Termination Status     :  Converged, relative change in the objective funciton (5.54e-06) is witin specified tolerance (1.00e-05)
```

D.3 Using MSES with NACA 4-Series

```
Objective
---------
   5097.5288053139 newton

Variables
---------
   AR                :       22.1000559505        [-]         aspect ratio
   Alpha_p20_MSES_0  :       18.3031296730      degree        Angle of Attack + 20deg from MSES, segment 0
   Alpha_p20_MSES_1  :       18.2611842557      degree        Angle of Attack + 20deg from MSES, segment 1
   Alpha_p20_MSES_2  :       14.2354295622      degree        Angle of Attack + 20deg from MSES, segment 2
   C_D_0             :        0.0110919436        [-]         Drag Coefficient, segment 0
   C_D_1             :        0.0110537481        [-]         Drag Coefficient, segment 1
   C_D_2             :        0.0066762703        [-]         Drag Coefficient, segment 2
   C_Dfuse_0         :        0.0018010837        [-]         Fuselage Drag Coefficient, segment 0
   C_Dfuse_1         :        0.0018010837        [-]         Fuselage Drag Coefficient, segment 1
   C_Dfuse_2         :        0.0018010837        [-]         Fuselage Drag Coefficient, segment 2
   C_Di_0            :        0.0051521723        [-]         Induced Drag Coefficient, segment 0
   C_Di_1            :        0.0050955638        [-]         Induced Drag Coefficient, segment 1
   C_Di_2            :        0.0002199937        [-]         Induced Drag Coefficient, segment 2
   C_Dp_0            :        0.0041386875        [-]         Wing Profile Drag Coefficient, segment 0
   C_Dp_1            :        0.0041571007        [-]         Wing Profile Drag Coefficient, segment 1
   C_Dp_2            :        0.0046551929        [-]         Wing Profile Drag Coefficient, segment 2
   C_L_0             :        0.5829039784        [-]         Lift Coefficient, segment 0
   C_L_1             :        0.5789020857        [-]         Lift Coefficient, segment 1
   C_L_2             :        0.1204190186        [-]         Lift Coefficient, segment 2
   C_L_MSES_0        :        0.5829039784        [-]         Lift Coefficient from MSES, segment 0
   C_L_MSES_1        :        0.5789020857        [-]         Lift Coefficient from MSES, segment 1
   C_L_MSES_2        :        0.1204190186        [-]         Lift Coefficient from MSES, segment 2
   I_cap_bar         :        0.0000234052        [-]         Area moment of inertia of cap on 2D cross section, normalized by chord^4
   M_r_bar           :    38611.3201799042      newton        Root Bending unit per unit chord
   P_max             :  1009102.1473296646       watt         Maximum Engine Power
   R                 :     5000.0000000001    kilometer      Single Segment range
   Re_0              :  4471197.8479656521        [-]         Reynolds Number, segment 0
   Re_1              :  4325851.7076423634        [-]         Reynolds Number, segment 1
   Re_2              :  9833763.6514182929        [-]         Reynolds Number, segment 2
   S                 :       27.7652457798    meter ** 2      total wing area
   T_0               :      650.7948450018      newton        Thrust, segment 0
   T_1               :      606.1782010429      newton        Thrust, segment 1
   T_2               :     1895.4147285424      newton        Thrust, segment 2
   V_0               :       68.1960475703   meter / second   Velocity, segment 0
   V_1               :       65.9832696009   meter / second   Velocity, segment 1
   V_2               :      150.0000000017   meter / second   Velocity, segment 2
   V_stall           :       37.9999800100   meter / second   stall speed
   W_0               :    34195.9383054526      newton        Weight, segment 0
   W_1               :    31737.8689248389      newton        Weight, segment 1
   W_2               :    34195.9383054526      newton        Weight, segment 2
   W_MTO             :    36835.4192206835      newton        Maximum Takeoff Weight
   W_cap             :     4626.5285021231      newton        Weight of Wing Spar Cap
   W_eng             :     2464.2950697346      newton        Engine Weight
   W_fuel_out        :     2639.4594247056      newton        Weight of fuel, outbound
   W_fuel_ret        :     2458.0693806083      newton        Weight of fuel, return
   W_pay             :     4905.0000000043      newton        Payload Weight
   W_tilde           :    22069.2950697271      newton        Dry Weight, no wing
   W_web             :      207.7584255239      newton        Weight of Wing Spar Shear Web
   W_wing            :     9668.5738551770      newton        wing weight
   W_zfw             :    31737.8689248365      newton        Zero Fuel Weight
```

```
   camberLocation    :        0.7348781854        [-]         location of maximum airfoil camber scaled by chord
   eta_0_0           :        0.2729391463        [-]         Overall Efficiency, segment 0
   eta_0_1           :        0.2730186247        [-]         Overall Efficiency, segment 1
   eta_0_2           :        0.2817516676        [-]         Overall Efficiency, segment 2
   eta_i_0           :        0.9174425087        [-]         Inviscid Propeller Efficiency, segment 0
   eta_i_1           :        0.9177096629        [-]         Inviscid Propeller Efficiency, segment 1
   eta_i_2           :        0.9470644288        [-]         Inviscid Propeller Efficiency, segment 2
   eta_prop_0        :        0.7798261324        [-]         Propeller Efficiency, segment 0
   eta_prop_1        :        0.7800532135        [-]         Propeller Efficiency, segment 1
   eta_prop_2        :        0.8050047645        [-]         Propeller Efficiency, segment 2
   maxCamber         :        0.0440906942        [-]         maximum airfoil camber scaled by chord
   nu                :        0.7859963823        [-]         Placeholder, (1+lam_w+lam_w**2)/(1+lam_w)**2
   p                 :        1.9000000000        [-]         Dummy Variable (1+2*lam_w)
   q                 :        1.4500000000        [-]         Dummy Variable (1+lam_w)
   t_cap_bar         :        0.0053539658        [-]         Spar cap thickness per unit chord
   t_web_bar         :        0.0010709946        [-]         Spar web thickness per unit chord
   tau               :        0.1496872210        [-]         airfoil thickness to chord ratio
   z_bre_0           :        0.0743603678        [-]         Breguet Range Factor, segment 0
   z_bre_1           :        0.0746131637        [-]         Breguet Range Factor, segment 1

Constants
---------
   A_prop            :        0.7850000000          meter ** 2          Disk area of propeller
   CDA0              :        0.0500000000          meter ** 2          fuselage drag area
   C_Lmax            :        1.5000000000             [-]              max CL with flaps down
   N_lift            :        6                        [-]              Ultimate Load Factor
   W_fixed           :    14700                       newton            fixed weight
   e                 :        0.9500000000             [-]              Oswald efficiency factor
   eta_eng           :        0.3500000000             [-]              Engine Efficiency
   eta_v             :        0.8500000000             [-]              Propeller Viscous Efficiency
   f_wadd            :        2                        [-]              Added Weight Fraction
   g                 :        9.8100000000     meter / second ** 2      Gravitational constant
   h_fuel            : 46000000                  joule / kilogram       Fuel Specific energy density
   k_ew              :        0.0372000000    newton / watt ** 0.803    Constant for engine weight
   mu                :        0.0000155460  kilogram / meter / second   viscosity of air
   r_h               :        0.7500000000             [-]              Ratio of height at rear spar to maximum wing height
   rho               :        0.9091220000    kilogram / meter ** 3     density of air
   rho_SL            :        1.2250000000    kilogram / meter ** 3     density of air, sea level
   rho_cap           :     2700               kilogram / meter ** 3     Density of wing cap material (aluminum)
   rho_web           :     2700               kilogram / meter ** 3     Density of wing web material (aluminum)
   sigma_max         :      310                     megapascal          Ultimate tensile stress of aluminum
   sigma_max_shear   :      167                     megapascal          Ultimate shear stress of aluminum
   w_bar             :        0.5000000000             [-]              Ratio of spar box width to chord length

Sensitivities
-------------
   Sensitivities not available for optimizations of type Sequential Log Convex Program

Solve Report
------------
   Solve Method           :  Sequential Log Convex Program
   Classification         :  Iterative
   Solver                 :  cvxopt
   Number of Iterations   :  85
   Termination Status     :  Converged, relative change in the objective funciton (4.03e-06) is witin specified tolerance (1.00e-05)
```

D.4 Using MSES with NACA 4-Series and Moment Constraint

```
Objective
---------
   5957.7220529740 newton

Variables
---------
   AR                :        20.4265249816        [-]         aspect ratio
   Alpha_p20_MSES_0  :        22.7588849909      degree        Angle of Attack + 20deg from MSES, segment 0
   Alpha_p20_MSES_1  :        22.8469113191      degree        Angle of Attack + 20deg from MSES, segment 1
   Alpha_p20_MSES_2  :        18.5227789084      degree        Angle of Attack + 20deg from MSES, segment 2
   C_D_0             :         0.0130076255        [-]         Drag Coefficient, segment 0
   C_D_1             :         0.0132032967        [-]         Drag Coefficient, segment 1
   C_D_2             :         0.0076034477        [-]         Drag Coefficient, segment 2
   C_Dfuse_0         :         0.0017956336        [-]         Fuselage Drag Coefficient, segment 0
   C_Dfuse_1         :         0.0017956336        [-]         Fuselage Drag Coefficient, segment 1
   C_Dfuse_2         :         0.0017956336        [-]         Fuselage Drag Coefficient, segment 2
   C_Di_0            :         0.0056307192        [-]         Induced Drag Coefficient, segment 0
   C_Di_1            :         0.0058105573        [-]         Induced Drag Coefficient, segment 1
   C_Di_2            :         0.0002315110        [-]         Induced Drag Coefficient, segment 2
   C_Dp_0            :         0.0055812727        [-]         Wing Profile Drag Coefficient, segment 0
   C_Dp_1            :         0.0055971058        [-]         Wing Profile Drag Coefficient, segment 1
   C_Dp_2            :         0.0055763032        [-]         Wing Profile Drag Coefficient, segment 2
   C_L_0             :         0.5858313763        [-]         Lift Coefficient, segment 0
   C_L_1             :         0.5949616867        [-]         Lift Coefficient, segment 1
   C_L_2             :         0.1187998042        [-]         Lift Coefficient, segment 2
   C_L_MSES_0        :         0.5858313763        [-]         Lift Coefficient from MSES, segment 0
   C_L_MSES_1        :         0.5949616867        [-]         Lift Coefficient from MSES, segment 1
   C_L_MSES_2        :         0.1187998042        [-]         Lift Coefficient from MSES, segment 2
   I_cap_bar         :         0.0000202413        [-]         Area moment of inertia of cap on 2D cross section, normalized by chord^4
   M_r_bar           :     36163.3763863633      newton        Root Bending unit per unit chord
   P_max             :   1161024.6518796829       watt         Maximum Engine Power
   R                 :      5000.0000000142    kilometer      Single Segment range
   Re_0              :   4612710.7365797032        [-]         Reynolds Number, segment 0
   Re_1              :   4381223.2951073041        [-]         Reynolds Number, segment 1
   Re_2              :  10241816.5322786700        [-]         Reynolds Number, segment 2
   S                 :        27.8454246688    meter ** 2      total wing area
   T_0               :       751.2612645370      newton        Thrust, segment 0
   T_1               :       687.6730763156      newton        Thrust, segment 1
   T_2               :      2165.3982982561      newton        Thrust, segment 2
   V_0               :        67.5563043154   meter / second   Velocity, segment 0
   V_1               :        64.1656178875   meter / second   Velocity, segment 1
   V_2               :       150.0000000087   meter / second   Velocity, segment 2
   V_stall           :        37.9999999993   meter / second   stall speed
   W_0               :     33833.3028164319      newton        Weight, segment 0
   W_1               :     30984.1067167489      newton        Weight, segment 1
   W_2               :     33833.3028164319      newton        Weight, segment 2
   W_MTO             :     36941.8287711316      newton        Maximum Takeoff Weight
   W_cap             :      4107.8048136983      newton        Weight of Wing Spar Cap
   W_eng             :      2758.1035525138      newton        Engine Weight
   W_fuel_out        :      3108.5259533277      newton        Weight of fuel, outbound
   W_fuel_ret        :      2849.1960996462      newton        Weight of fuel, return
   W_pay             :      4905.0000001349      newton        Payload Weight
   W_tilde           :     22363.1035526690      newton        Dry Weight, no wing
```

```
   W_web             :       202.6967683251       newton       Weight of Wing Spar Shear Web
   W_wing            :      8621.0031637666       newton       wing weight
   W_zfw             :     30984.1067165409       newton       Zero Fuel Weight
   camberLocation    :         0.3763171022        [-]         location of maximum airfoil camber scaled by chord
   eta_0_0           :         0.2693670702        [-]         Overall Efficiency, segment 0
   eta_0_1           :         0.2690194797        [-]         Overall Efficiency, segment 1
   eta_0_2           :         0.2797613033        [-]         Overall Efficiency, segment 2
   eta_i_0           :         0.9054355302        [-]         Inviscid Propeller Efficiency, segment 0
   eta_i_1           :         0.9042671586        [-]         Inviscid Propeller Efficiency, segment 1
   eta_i_2           :         0.9403741287        [-]         Inviscid Propeller Efficiency, segment 2
   eta_prop_0        :         0.7696202007        [-]         Propeller Efficiency, segment 0
   eta_prop_1        :         0.7686270848        [-]         Propeller Efficiency, segment 1
   eta_prop_2        :         0.7993180094        [-]         Propeller Efficiency, segment 2
   maxCamber         :         0.0237335258        [-]         maximum airfoil camber scaled by chord
   negCMp1_0         :         1.0577324910        [-]         Negative CM about 0.25c + 1, segment 0
   negCMp1_1         :         1.0577700394        [-]         Negative CM about 0.25c + 1, segment 1
   negCMp1_2         :         1.0590378842        [-]         Negative CM about 0.25c + 1, segment 2
   nu                :         0.7859963824        [-]         Placeholder, (1+lam_w+lam_w**2)/(1+lam_w)**2
   p                 :         1.9000000000        [-]         Dummy Variable (1+2*lam_w)
   q                 :         1.4500000000        [-]         Dummy Variable (1+lam_w)
   t_cap_bar         :         0.0045517791        [-]         Spar cap thickness per unit chord
   t_web_bar         :         0.0009982444        [-]         Spar web thickness per unit chord
   tau               :         0.1499999998        [-]         airfoil thickness to chord ratio
   z_bre_0           :         0.0878992398        [-]         Breguet Range Factor, segment 0
   z_bre_1           :         0.0879716556        [-]         Breguet Range Factor, segment 1

Constants
---------
   A_prop            :         0.7850000000          meter ** 2          Disk area of propeller
   CDA0              :         0.0500000000          meter ** 2          fuselage drag area
   C_Lmax            :         1.5000000000             [-]              max CL with flaps down
   N_lift            :         6                        [-]              Ultimate Load Factor
   W_fixed           :     14700                       newton            fixed weight
   e                 :         0.9500000000             [-]              Oswald efficiency factor
   eta_eng           :         0.3500000000             [-]              Engine Efficiency
   eta_v             :         0.8500000000             [-]              Propeller Viscous Efficiency
   f_wadd            :         2                        [-]              Added Weight Fraction
   g                 :         9.8100000000     meter / second ** 2      Gravitational constant
   h_fuel            :  46000000                  joule / kilogram       Fuel Specific energy density
   k_ew              :         0.0372000000    newton / watt ** 0.803    Constant for engine weight
   mu                :         0.0000155460   kilogram / meter / second   viscosity of air
   r_h               :         0.7500000000             [-]              Ratio of height at rear spar to maximum wing height
   rho               :         0.9091220000    kilogram / meter ** 3     density of air
   rho_SL            :         1.2250000000    kilogram / meter ** 3     density of air, sea level
   rho_cap           :      2700               kilogram / meter ** 3     Density of wing cap material (aluminum)
   rho_web           :      2700               kilogram / meter ** 3     Density of wing web material (aluminum)
   sigma_max         :       310                     megapascal          Ultimate tensile stress of aluminum
   sigma_max_shear   :       167                     megapascal          Ultimate shear stress of aluminum
   w_bar             :         0.5000000000             [-]              Ratio of spar box width to chord length

Sensitivities
-------------
   Sensitivities not available for optimizations of type Sequential Log Convex Program

Solve Report
------------
   Solve Method           :  Sequential Log Convex Program
   Classification         :  Iterative
```

```
Solver                 :  cvxopt
Number of Iterations   :  31
Termination Status     :  Converged, relative change in the objective funciton (1.17e-06) is witin specified tolerance (1.00e-05)
```

D.5 Using MSES with Kulfan CST-2

```
Objective
---------
   4538.1718287918 newton

Variables
---------
   AR                :        20.6661068243        [-]         aspect ratio
   Alower1           :         1.0537837424        [-]         Kulfan coefficient, lower surface mode 1
   Alower2           :         1.1633138959        [-]         Kulfan coefficient, lower surface mode 2
   Alpha_p20_MSES_0  :        20.8515970812      degree        Angle of Attack + 20deg from MSES, segment 0
   Alpha_p20_MSES_1  :        20.8649365804      degree        Angle of Attack + 20deg from MSES, segment 1
   Alpha_p20_MSES_2  :        16.6581653419      degree        Angle of Attack + 20deg from MSES, segment 2
   Aupper1           :         1.1811240794        [-]         Kulfan coefficient, upper surface mode 1
   Aupper2           :         1.4247810133        [-]         Kulfan coefficient, upper surface mode 2
   C_D_0             :         0.0082525115        [-]         Drag Coefficient, segment 0
   C_D_1             :         0.0085727819        [-]         Drag Coefficient, segment 1
   C_D_2             :         0.0077263695        [-]         Drag Coefficient, segment 2
   C_Dfuse_0         :         0.0018071063        [-]         Fuselage Drag Coefficient, segment 0
   C_Dfuse_1         :         0.0018071063        [-]         Fuselage Drag Coefficient, segment 1
   C_Dfuse_2         :         0.0018071063        [-]         Fuselage Drag Coefficient, segment 2
   C_Di_0            :         0.0037148591        [-]         Induced Drag Coefficient, segment 0
   C_Di_1            :         0.0038309205        [-]         Induced Drag Coefficient, segment 1
   C_Di_2            :         0.0002332968        [-]         Induced Drag Coefficient, segment 2
   C_Dp_0            :         0.0027305462        [-]         Wing Profile Drag Coefficient, segment 0
   C_Dp_1            :         0.0029347551        [-]         Wing Profile Drag Coefficient, segment 1
   C_Dp_2            :         0.0056859664        [-]         Wing Profile Drag Coefficient, segment 2
   C_L_0             :         0.4745944216        [-]         Lift Coefficient, segment 0
   C_L_1             :         0.4782072443        [-]         Lift Coefficient, segment 1
   C_L_2             :         0.1198426489        [-]         Lift Coefficient, segment 2
   C_L_MSES_0        :         0.4745944216        [-]         Lift Coefficient from MSES, segment 0
   C_L_MSES_1        :         0.4782072443        [-]         Lift Coefficient from MSES, segment 1
   C_L_MSES_2        :         0.1198426489        [-]         Lift Coefficient from MSES, segment 2
   I_cap_bar         :         0.0000199865        [-]         Area moment of inertia of cap on 2D cross section, normalized by chord^4
   M_r_bar           :     36607.1265167237      newton        Root Bending unit per unit chord
   P_max             :   1176417.5698024882       watt         Maximum Engine Power
   R                 :      5000.0000000121    kilometer      Single Segment range
   Re_0              :   5191219.7252135826        [-]         Reynolds Number, segment 0
   Re_1              :   5049188.4892131025        [-]         Reynolds Number, segment 1
   Re_2              :  10163981.4386706334        [-]         Reynolds Number, segment 2
   S                 :        27.6893642043    meter ** 2      total wing area
   T_0               :       584.6969284002      newton        Thrust, segment 0
   T_1               :       558.4735105175      newton        Thrust, segment 1
   T_2               :      2190.7964053254      newton        Thrust, segment 2
   V_0               :        76.3859642793   meter / second   Velocity, segment 0
   V_1               :        74.2306597197   meter / second   Velocity, segment 1
   V_2               :       150.0000000086   meter / second   Velocity, segment 2
   V_stall           :        37.8982021514   meter / second   stall speed
   W_0               :     33926.5925018776      newton        Weight, segment 0
   W_1               :     31706.3429721618      newton        Weight, segment 1
   W_2               :     33926.5925018776      newton        Weight, segment 2
   W_MTO             :     36532.4339435457      newton        Maximum Takeoff Weight
   W_cap             :      4455.5719745093      newton        Weight of Wing Spar Cap
   W_eng             :      2783.4376812099      newton        Engine Weight
   W_fuel_out        :      2317.9222993742      newton        Weight of fuel, outbound
   W_fuel_ret        :      2220.2495294177      newton        Weight of fuel, return
   W_pay             :      4905.0000001834      newton        Payload Weight
```

```
   W_tilde          :     22388.4376814480       newton       Dry Weight, no wing
   W_web            :       203.3806706059       newton       Weight of Wing Spar Shear Web
   W_wing           :      9317.9052903455       newton       wing weight
   W_zfw            :     31706.3429719504       newton       Zero Fuel Weight
   eta_0_0          :         0.2786548851        [-]         Overall Efficiency, segment 0
   eta_0_1          :         0.2780854705        [-]         Overall Efficiency, segment 1
   eta_0_2          :         0.2796048022        [-]         Overall Efficiency, segment 2
   eta_i_0          :         0.9366550759        [-]         Inviscid Propeller Efficiency, segment 0
   eta_i_1          :         0.9347410772        [-]         Inviscid Propeller Efficiency, segment 1
   eta_i_2          :         0.9398480746        [-]         Inviscid Propeller Efficiency, segment 2
   eta_prop_0       :         0.7961568145        [-]         Propeller Efficiency, segment 0
   eta_prop_1       :         0.7945299156        [-]         Propeller Efficiency, segment 1
   eta_prop_2       :         0.7988708634        [-]         Propeller Efficiency, segment 2
   negCMp1_0        :         1.5322718767        [-]         Negative CM about 0.25c + 1, segment 0
   negCMp1_1        :         1.4772267371        [-]         Negative CM about 0.25c + 1, segment 1
   negCMp1_2        :         1.2284495196        [-]         Negative CM about 0.25c + 1, segment 2
   nu               :         0.7859963824        [-]         Placeholder, (1+lam_w+lam_w**2)/(1+lam_w)**2
   p                :         1.9000000000        [-]         Dummy Variable (1+2*lam_w)
   q                :         1.4500000000        [-]         Dummy Variable (1+lam_w)
   t_cap_bar        :         0.0050076114        [-]         Spar cap thickness per unit chord
   t_web_bar        :         0.0010736501        [-]         Spar web thickness per unit chord
   tau              :         0.1423640367        [-]         airfoil thickness to chord ratio
   z_bre_0          :         0.0660291207        [-]         Breguet Range Factor, segment 0
   z_bre_1          :         0.0677316717        [-]         Breguet Range Factor, segment 1

Constants
---------
   A_prop           :         0.7850000000          meter ** 2          Disk area of propeller
   CDA0             :         0.0500000000          meter ** 2          fuselage drag area
   C_Lmax           :         1.5000000000             [-]              max CL with flaps down
   N_lift           :         6                        [-]              Ultimate Load Factor
   W_fixed          :     14700                       newton           fixed weight
   e                :         0.9500000000             [-]              Oswald efficiency factor
   eta_eng          :         0.3500000000             [-]              Engine Efficiency
   eta_v            :         0.8500000000             [-]              Propeller Viscous Efficiency
   f_wadd           :         2                        [-]              Added Weight Fraction
   g                :         9.8100000000     meter / second ** 2      Gravitational constant
   h_fuel           :  46000000                  joule / kilogram       Fuel Specific energy density
   k_ew             :         0.0372000000    newton / watt ** 0.803    Constant for engine weight
   mu               :         0.0000155460  kilogram / meter / second   viscosity of air
   r_h              :         0.7500000000             [-]              Ratio of height at rear spar to maximum wing height
   rho              :         0.9091220000    kilogram / meter ** 3     density of air
   rho_SL           :         1.2250000000    kilogram / meter ** 3     density of air, sea level
   rho_cap          :      2700               kilogram / meter ** 3     Density of wing cap material (aluminum)
   rho_web          :      2700               kilogram / meter ** 3     Density of wing web material (aluminum)
   sigma_max        :       310                     megapascal          Ultimate tensile stress of aluminum
   sigma_max_shear  :       167                     megapascal          Ultimate shear stress of aluminum
   w_bar            :         0.5000000000             [-]              Ratio of spar box width to chord length

Sensitivities
-------------
   Sensitivities not available for optimizations of type Sequential Log Convex Program

Solve Report
------------
   Solve Method           :  Sequential Log Convex Program
   Classification         :  Iterative
   Solver                 :  cvxopt
```

```
Number of Iterations    :  45
Termination Status      :  Converged, relative change in the objective funciton (5.78e-06) is witin specified tolerance (1.00e-05)
```

D.6 Using MSES with Kulfan CST-2 and Tripped Flow

```
Objective
---------
   5825.7326879038 newton

Variables
---------
   AR                :        19.9368938149        [-]         aspect ratio
   Alower1           :         1.0967889043        [-]         Kulfan coefficient, lower surface mode 1
   Alower2           :         1.0758992758        [-]         Kulfan coefficient, lower surface mode 2
   Alpha_p20_MSES_0  :        20.9999917294      degree        Angle of Attack + 20deg from MSES, segment 0
   Alpha_p20_MSES_1  :        20.9315664747      degree        Angle of Attack + 20deg from MSES, segment 1
   Alpha_p20_MSES_2  :        17.0817431741      degree        Angle of Attack + 20deg from MSES, segment 2
   Aupper1           :         1.3024848958        [-]         Kulfan coefficient, upper surface mode 1
   Aupper2           :         1.2925454448        [-]         Kulfan coefficient, upper surface mode 2
   C_D_0             :         0.0118445526        [-]         Drag Coefficient, segment 0
   C_D_1             :         0.0118026416        [-]         Drag Coefficient, segment 1
   C_D_2             :         0.0085117926        [-]         Drag Coefficient, segment 2
   C_Dfuse_0         :         0.0017981077        [-]         Fuselage Drag Coefficient, segment 0
   C_Dfuse_1         :         0.0017981077        [-]         Fuselage Drag Coefficient, segment 1
   C_Dfuse_2         :         0.0017981077        [-]         Fuselage Drag Coefficient, segment 2
   C_Di_0            :         0.0050009181        [-]         Induced Drag Coefficient, segment 0
   C_Di_1            :         0.0049015682        [-]         Induced Drag Coefficient, segment 1
   C_Di_2            :         0.0002387790        [-]         Induced Drag Coefficient, segment 2
   C_Dp_0            :         0.0050455269        [-]         Wing Profile Drag Coefficient, segment 0
   C_Dp_1            :         0.0051029657        [-]         Wing Profile Drag Coefficient, segment 1
   C_Dp_2            :         0.0064749059        [-]         Wing Profile Drag Coefficient, segment 2
   C_L_0             :         0.5447669785        [-]         Lift Coefficient, segment 0
   C_L_1             :         0.5395111285        [-]         Lift Coefficient, segment 1
   C_L_2             :         0.1190601871        [-]         Lift Coefficient, segment 2
   C_L_MSES_0        :         0.5447669785        [-]         Lift Coefficient from MSES, segment 0
   C_L_MSES_1        :         0.5395111285        [-]         Lift Coefficient from MSES, segment 1
   C_L_MSES_2        :         0.1190601871        [-]         Lift Coefficient from MSES, segment 2
   I_cap_bar         :         0.0000195038        [-]         Area moment of inertia of cap on 2D cross section, normalized by chord^4
   M_r_bar           :     35718.5398125232      newton        Root Bending unit per unit chord
   P_max             :   1305928.0280958987       watt         Maximum Engine Power
   R                 :      5000.0000000009    kilometer      Single Segment range
   Re_0              :   4868674.2369979974        [-]         Reynolds Number, segment 0
   Re_1              :   4683214.5612302395        [-]         Reynolds Number, segment 1
   Re_2              :  10368632.7008782625        [-]         Reynolds Number, segment 2
   S                 :        27.8146662104    meter ** 2      total wing area
   T_0               :       738.8436785783      newton        Thrust, segment 0
   T_1               :       681.8835083078      newton        Thrust, segment 1
   T_2               :      2419.5064447046      newton        Thrust, segment 2
   V_0               :        70.3790594637   meter / second   Velocity, segment 0
   V_1               :        67.7027089653   meter / second   Velocity, segment 1
   V_2               :       150.0000000016   meter / second   Velocity, segment 2
   V_stall           :        37.9999999982   meter / second   stall speed
   W_0               :     33871.3059023133      newton        Weight, segment 0
   W_1               :     31075.2221804880      newton        Weight, segment 1
   W_2               :     33871.3059023133      newton        Weight, segment 2
   W_MTO             :     36901.0222911632      newton        Maximum Takeoff Weight
   W_cap             :      4017.2126743196      newton        Weight of Wing Spar Cap
   W_eng             :      3030.7562208815      newton        Engine Weight
   W_fuel_out        :      3029.6489661543      newton        Weight of fuel, outbound
   W_fuel_ret        :      2796.0837217496      newton        Weight of fuel, return
   W_pay             :      4905.0000000384      newton        Payload Weight
```

```
   W_tilde          :     22635.7562209056       newton       Dry Weight, no wing
   W_web            :       202.5203058504       newton       Weight of Wing Spar Shear Web
   W_wing           :      8439.4659601785       newton       wing weight
   W_zfw            :     31075.2221804741       newton       Zero Fuel Weight
   eta_0_0          :         0.2715023120        [-]         Overall Efficiency, segment 0
   eta_0_1          :         0.2715782510        [-]         Overall Efficiency, segment 1
   eta_0_2          :         0.2779412123        [-]         Overall Efficiency, segment 2
   eta_i_0          :         0.9126128136        [-]         Inviscid Propeller Efficiency, segment 0
   eta_i_1          :         0.9128680707        [-]         Inviscid Propeller Efficiency, segment 1
   eta_i_2          :         0.9342561759        [-]         Inviscid Propeller Efficiency, segment 2
   eta_prop_0       :         0.7757208916        [-]         Propeller Efficiency, segment 0
   eta_prop_1       :         0.7759378601        [-]         Propeller Efficiency, segment 1
   eta_prop_2       :         0.7941177495        [-]         Propeller Efficiency, segment 2
   negCMp1_0        :         1.0710119367        [-]         Negative CM about 0.25c + 1, segment 0
   negCMp1_1        :         1.0714259792        [-]         Negative CM about 0.25c + 1, segment 1
   negCMp1_2        :         1.0749205732        [-]         Negative CM about 0.25c + 1, segment 2
   nu               :         0.7859963823        [-]         Placeholder, (1+lam_w+lam_w**2)/(1+lam_w)**2
   p                :         1.9000000000        [-]         Dummy Variable (1+2*lam_w)
   q                :         1.4500000000        [-]         Dummy Variable (1+lam_w)
   t_cap_bar        :         0.0044036520        [-]         Spar cap thickness per unit chord
   t_web_bar        :         0.0009891628        [-]         Spar web thickness per unit chord
   tau              :         0.1496272777        [-]         airfoil thickness to chord ratio
   z_bre_0          :         0.0857122576        [-]         Breguet Range Factor, segment 0
   z_bre_1          :         0.0862122052        [-]         Breguet Range Factor, segment 1

Constants
---------
   A_prop           :         0.7850000000          meter ** 2          Disk area of propeller
   CDA0             :         0.0500000000          meter ** 2          fuselage drag area
   C_Lmax           :         1.5000000000             [-]              max CL with flaps down
   N_lift           :         6                        [-]              Ultimate Load Factor
   W_fixed          :     14700                       newton           fixed weight
   e                :         0.9500000000             [-]              Oswald efficiency factor
   eta_eng          :         0.3500000000             [-]              Engine Efficiency
   eta_v            :         0.8500000000             [-]              Propeller Viscous Efficiency
   f_wadd           :         2                        [-]              Added Weight Fraction
   g                :         9.8100000000     meter / second ** 2      Gravitational constant
   h_fuel           :  46000000                  joule / kilogram       Fuel Specific energy density
   k_ew             :         0.0372000000    newton / watt ** 0.803    Constant for engine weight
   mu               :         0.0000155460  kilogram / meter / second   viscosity of air
   r_h              :         0.7500000000             [-]              Ratio of height at rear spar to maximum wing height
   rho              :         0.9091220000    kilogram / meter ** 3     density of air
   rho_SL           :         1.2250000000    kilogram / meter ** 3     density of air, sea level
   rho_cap          :      2700               kilogram / meter ** 3     Density of wing cap material (aluminum)
   rho_web          :      2700               kilogram / meter ** 3     Density of wing web material (aluminum)
   sigma_max        :       310                     megapascal          Ultimate tensile stress of aluminum
   sigma_max_shear  :       167                     megapascal          Ultimate shear stress of aluminum
   w_bar            :         0.5000000000             [-]              Ratio of spar box width to chord length

Sensitivities
-------------
   Sensitivities not available for optimizations of type Sequential Log Convex Program

Solve Report
------------
   Solve Method           :  Sequential Log Convex Program
   Classification         :  Iterative
   Solver                 :  cvxopt
```

```
Number of Iterations    :  61
Termination Status      :  Converged, relative change in the objective funciton (7.07e-06) is witin specified tolerance (1.00e-05)
```

D.7 Using MSES with Kulfan CST-4

```
Objective
---------
   5555.6177728669 newton

Variables
---------
   AR                :        20.4913030247        [-]         aspect ratio
   Alower1           :         1.2049881328        [-]         Kulfan coefficient, lower surface mode 1
   Alower2           :         1.1123192813        [-]         Kulfan coefficient, lower surface mode 2
   Alower3           :         1.1312093254        [-]         Kulfan coefficient, lower surface mode 3
   Alower4           :         1.0939372762        [-]         Kulfan coefficient, lower surface mode 4
   Alpha_p20_MSES_0  :        23.3276360308      degree        Angle of Attack + 20deg from MSES, segment 0
   Alpha_p20_MSES_1  :        23.2706150369      degree        Angle of Attack + 20deg from MSES, segment 1
   Alpha_p20_MSES_2  :        19.3590332651      degree        Angle of Attack + 20deg from MSES, segment 2
   Aupper1           :         1.1987277033        [-]         Kulfan coefficient, upper surface mode 1
   Aupper2           :         1.3108865243        [-]         Kulfan coefficient, upper surface mode 2
   Aupper3           :         1.2061315179        [-]         Kulfan coefficient, upper surface mode 3
   Aupper4           :         1.1763599936        [-]         Kulfan coefficient, upper surface mode 4
   C_D_0             :         0.0121014902        [-]         Drag Coefficient, segment 0
   C_D_1             :         0.0119641483        [-]         Drag Coefficient, segment 1
   C_D_2             :         0.0075478829        [-]         Drag Coefficient, segment 2
   C_Dfuse_0         :         0.0018189560        [-]         Fuselage Drag Coefficient, segment 0
   C_Dfuse_1         :         0.0018189560        [-]         Fuselage Drag Coefficient, segment 1
   C_Dfuse_2         :         0.0018189560        [-]         Fuselage Drag Coefficient, segment 2
   C_Di_0            :         0.0054263783        [-]         Induced Drag Coefficient, segment 0
   C_Di_1            :         0.0053213354        [-]         Induced Drag Coefficient, segment 1
   C_Di_2            :         0.0002332519        [-]         Induced Drag Coefficient, segment 2
   C_Dp_0            :         0.0048561559        [-]         Wing Profile Drag Coefficient, segment 0
   C_Dp_1            :         0.0048238569        [-]         Wing Profile Drag Coefficient, segment 1
   C_Dp_2            :         0.0054956750        [-]         Wing Profile Drag Coefficient, segment 2
   C_L_0             :         0.5749119239        [-]         Lift Coefficient, segment 0
   C_L_1             :         0.5691409226        [-]         Lift Coefficient, segment 1
   C_L_2             :         0.1194202853        [-]         Lift Coefficient, segment 2
   C_L_MSES_0        :         0.5749119239        [-]         Lift Coefficient from MSES, segment 0
   C_L_MSES_1        :         0.5691409226        [-]         Lift Coefficient from MSES, segment 1
   C_L_MSES_2        :         0.1194202853        [-]         Lift Coefficient from MSES, segment 2
   I_cap_bar         :         0.0000205901        [-]         Area moment of inertia of cap on 2D cross section, normalized by chord^4
   M_r_bar           :     36198.4833580894      newton        Root Bending unit per unit chord
   P_max             :   1136402.5306298169       watt         Maximum Engine Power
   R                 :      5000.0000000006   kilometer       Single Segment range
   Re_0              :   4642733.3016389618        [-]         Reynolds Number, segment 0
   Re_1              :   4478681.7680599103        [-]         Reynolds Number, segment 1
   Re_2              :  10160549.3083673455        [-]         Reynolds Number, segment 2
   S                 :        27.4890028694    meter ** 2      total wing area
   T_0               :       706.3443828110      newton        Thrust, segment 0
   T_1               :       650.0367484744      newton        Thrust, segment 1
   T_2               :      2121.6525592197      newton        Thrust, segment 2
   V_0               :        68.5266610272  meter / second    Velocity, segment 0
   V_1               :        66.1044357991  meter / second    Velocity, segment 1
   V_2               :       150.0000000034  meter / second    Velocity, segment 2
   V_stall           :        37.9999999992  meter / second    stall speed
   W_0               :     33574.4064938418      newton        Weight, segment 0
   W_1               :     30913.3551045718      newton        Weight, segment 1
   W_2               :     33574.4064938418      newton        Weight, segment 2
   W_MTO             :     36468.9728802121      newton        Maximum Takeoff Weight
   W_cap             :      4097.6369965494      newton        Weight of Wing Spar Cap
```

```
   W_eng             :      2710.5331704719       newton       Engine Weight
   W_fuel_out        :      2894.5663835584       newton       Weight of fuel, outbound
   W_fuel_ret        :      2661.0513893085       newton       Weight of fuel, return
   W_pay             :      4905.0000000206       newton       Payload Weight
   W_tilde           :     22315.5331704435       newton       Dry Weight, no wing
   W_web             :       201.2739706297       newton       Weight of Wing Spar Shear Web
   W_wing            :      8597.8219341541       newton       wing weight
   W_zfw             :     30913.3551045575       newton       Zero Fuel Weight
   eta_0_0           :         0.2713100457        [-]         Overall Efficiency, segment 0
   eta_0_1           :         0.2715570665        [-]         Overall Efficiency, segment 1
   eta_0_2           :         0.2800835885        [-]         Overall Efficiency, segment 2
   eta_i_0           :         0.9119665402        [-]         Inviscid Propeller Efficiency, segment 0
   eta_i_1           :         0.9127968621        [-]         Inviscid Propeller Efficiency, segment 1
   eta_i_2           :         0.9414574404        [-]         Inviscid Propeller Efficiency, segment 2
   eta_prop_0        :         0.7751715592        [-]         Propeller Efficiency, segment 0
   eta_prop_1        :         0.7758773328        [-]         Propeller Efficiency, segment 1
   eta_prop_2        :         0.8002388244        [-]         Propeller Efficiency, segment 2
   negCMp1_0         :         1.0410030208        [-]         Negative CM about 0.25c + 1, segment 0
   negCMp1_1         :         1.0410997983        [-]         Negative CM about 0.25c + 1, segment 1
   negCMp1_2         :         1.0411551919        [-]         Negative CM about 0.25c + 1, segment 2
   nu                :         0.7859963823        [-]         Placeholder, (1+lam_w+lam_w**2)/(1+lam_w)**2
   p                 :         1.9000000000        [-]         Dummy Variable (1+2*lam_w)
   q                 :         1.4500000000        [-]         Dummy Variable (1+lam_w)
   t_cap_bar         :         0.0046364558        [-]         Spar cap thickness per unit chord
   t_web_bar         :         0.0010121577        [-]         Spar web thickness per unit chord
   tau               :         0.1500000000        [-]         airfoil thickness to chord ratio
   z_bre_0           :         0.0826915256        [-]         Breguet Range Factor, segment 0
   z_bre_1           :         0.0825731402        [-]         Breguet Range Factor, segment 1

Constants
---------
   A_prop            :         0.7850000000          meter ** 2          Disk area of propeller
   CDA0              :         0.0500000000          meter ** 2          fuselage drag area
   C_Lmax            :         1.5000000000             [-]              max CL with flaps down
   N_lift            :         6                        [-]              Ultimate Load Factor
   W_fixed           :     14700                       newton            fixed weight
   e                 :         0.9500000000             [-]              Oswald efficiency factor
   eta_eng           :         0.3500000000             [-]              Engine Efficiency
   eta_v             :         0.8500000000             [-]              Propeller Viscous Efficiency
   f_wadd            :         2                        [-]              Added Weight Fraction
   g                 :         9.8100000000     meter / second ** 2      Gravitational constant
   h_fuel            :  46000000                  joule / kilogram       Fuel Specific energy density
   k_ew              :         0.0372000000    newton / watt ** 0.803    Constant for engine weight
   mu                :         0.0000155460  kilogram / meter / second   viscosity of air
   r_h               :         0.7500000000             [-]              Ratio of height at rear spar to maximum wing height
   rho               :         0.9091220000    kilogram / meter ** 3     density of air
   rho_SL            :         1.2250000000    kilogram / meter ** 3     density of air, sea level
   rho_cap           :      2700               kilogram / meter ** 3     Density of wing cap material (aluminum)
   rho_web           :      2700               kilogram / meter ** 3     Density of wing web material (aluminum)
   sigma_max         :       310                     megapascal          Ultimate tensile stress of aluminum
   sigma_max_shear   :       167                     megapascal          Ultimate shear stress of aluminum
   w_bar             :         0.5000000000             [-]              Ratio of spar box width to chord length

Sensitivities
-------------
   Sensitivities not available for optimizations of type Sequential Log Convex Program

Solve Report
```

```
------------
   Solve Method           :  Sequential Log Convex Program
   Classification         :  Iterative
   Solver                 :  cvxopt
   Number of Iterations   :  45
   Termination Status     :  Converged, relative change in the objective funciton (1.14e-06) is witin specified tolerance (1.00e-05)
```

Appendix E

Full Report of Design Results for the KYSP-SUGAR Test Problem

E.1 Results of the Original Signomial Program (Phase 1)

```
Objective
---------
   33877.36229 force_pound

Variables
---------
   AR_w           :        10.71392       [-]         Wing Aspect Ratio
   C_D_i_w_0      :         0.01115       [-]         Wing Induced Drag Coefficient, segment 0
   C_D_i_w_1      :         0.00897       [-]         Wing Induced Drag Coefficient, segment 1
   C_D_p_w_0      :         0.00607       [-]         Wing Pressure Drag Coefficient, segment 0
   C_D_p_w_1      :         0.00613       [-]         Wing Pressure Drag Coefficient, segment 1
   C_D_w_0        :         0.01722       [-]         Wing Drag Coefficient, segment 0
   C_D_w_1        :         0.01510       [-]         Wing Drag Coefficient, segment 1
   C_L_alpha_w_0  :         5.47898    1 / radian     Wing Lift Curve Slope, segment 0
   C_L_alpha_w_1  :         5.47898    1 / radian     Wing Lift Curve Slope, segment 1
   C_L_w_0        :         0.59564       [-]         Wing Lift Coefficient, segment 0
   C_L_w_1        :         0.53424       [-]         Wing Lift Coefficient, segment 1
   D_0            :     39618.54023      newton       Total aircraft drag, segment 0
   D_1            :     35107.82844      newton       Total aircraft drag, segment 1
   D_fuse_0       :     11333.45572      newton       Fuselage drag, segment 0
   D_fuse_1       :     10158.52094      newton       Fuselage drag, segment 1
   D_ht_0         :      3777.81857      newton       Horizontal tail drag, segment 0
   D_ht_1         :      3386.17365      newton       Horizontal tail drag, segment 1
   D_vt_0         :      3777.81857      newton       Vertical tail drag, segment 0
```

D_vt_1	:	3386.17365	newton	Vertical tail drag, segment 1
D_w_0	:	20729.44736	newton	Wing drag, segment 0
D_w_1	:	18176.96020	newton	Wing drag, segment 1
I_cap_w	:	0.00002	[-]	Area moment of inertia of cap on 2D cross section, normalized by chord^4
L_total_0	:	715327.88834	newton	Total Aircraft Lift, segment 0
L_total_1	:	641591.65363	newton	Total Aircraft Lift, segment 1
L_w_0	:	681264.65556	newton	Wing Lift, segment 0
L_w_1	:	611039.67012	newton	Wing Lift, segment 1
L_w_max	:	1888909.28702	newton	Maximum wing lift
Lam_w	:	0.56980	radian	Wing Sweep
M_0	:	0.78000	[-]	Mach number, segment 0
M_1	:	0.78000	[-]	Mach number, segment 1
M_r_w	:	1054042.77432	newton	Wing Root Bending unit per unit chord
R_0	:	1750.00000	nautical_mile	Range, segment 0
R_1	:	1750.00000	nautical_mile	Range, segment 1
Re_w_0	:	62276969.32742	[-]	Wing Reynolds Number, segment 0
Re_w_1	:	62276969.32899	[-]	Wing Reynolds Number, segment 1
S_w	:	112.93343	meter ** 2	Vehicle reference area (wing area)
V_inf_0	:	232.33662	meter / second	Cruise velocity, segment 0
V_inf_1	:	232.33662	meter / second	Cruise velocity, segment 1
W_avg_0	:	160812.10655	force_pound	Average aircraft weight, segment 0
W_avg_1	:	144235.54155	force_pound	Average aircraft weight, segment 1
W_cap_w	:	13510.34452	force_pound	Weight of Wing Spar Cap
W_dry	:	99790.11788	force_pound	Aircraft dry weight
W_end_0	:	152248.42372	force_pound	Weight at end of segment, segment 0
W_end_1	:	136644.37988	force_pound	Weight at end of segment, segment 1
W_fuel_0	:	17609.05645	force_pound	Weight of fuel burned, segment 0
W_fuel_1	:	15604.04383	force_pound	Weight of fuel burned, segment 1
W_fuel_prim	:	33213.10028	force_pound	Primary fuel weight (no reserve)
W_fuel_res	:	664.26201	force_pound	Reserve fuel weight
W_fuel_total	:	33877.36229	force_pound	Total fuel carried
W_max	:	169857.48017	force_pound	Maximum aircraft weight
W_pay	:	36190.00000	force_pound	Payload weight
W_start_0	:	169857.48017	force_pound	Weight at start of segment, segment 0
W_start_1	:	152248.42372	force_pound	Weight at start of segment, segment 1
W_struct_w	:	14868.97432	force_pound	Weight of Wing Spar
W_web_w	:	1358.62980	force_pound	Weight of Wing Spar Shear Web
W_wing	:	24385.11788	force_pound	Wing weight
alpha_w_0	:	0.10871	radian	Wing Angle of Attack, segment 0
alpha_w_1	:	0.09751	radian	Wing Angle of Attack, segment 1
b_w	:	34.78447	meter	Wing Span
c_bar_w	:	3.90134	meter	Wing Mean Aerodynamic Chord
c_root_w	:	5.77184	meter	Wing Root Chord
c_tip_w	:	0.72148	meter	Wing Tip Chord
cosLam_w	:	0.84203	[-]	Placeholder, cos(Lam_w)
e	:	0.94543	[-]	Oswald Efficiency Factor
lam_w	:	0.12500	[-]	Wing Taper Ratio
nu_w	:	0.90061	[-]	Placeholder, (1+lam_w+lam_w**2)/(1+lam_w)**2
p_w	:	1.25000	[-]	Dummy Variable (1+2*lam_w)
q_w	:	1.12500	[-]	Dummy Variable (1+lam_w)
t2Lam_w	:	0.40892	[-]	Placeholder, tan^2(Lam_w)
t_0	:	232.49312	minute	Flight time, segment 0
t_1	:	232.49312	minute	Flight time, segment 1
t_cap_w	:	0.00368	[-]	Spar cap thickness per unit chord
t_web_w	:	0.00244	[-]	Spar web thickness per unit chord
tau_w	:	0.12500	[-]	Wing Thickness Ratio
y_c_bar_w	:	10.43534	meter	Spanwise Location of Mean Aerodynamic Chord
z_bre_0	:	0.10945	[-]	Breguet parameter, segment 0

```
   z_bre_1        :         0.10814        [-]         Breguet parameter, segment 1

Constants
---------
   N_lift           :     2.50000              [-]              Ultimate Load Factor
   R_req            :  3500               nautical_mile         Required aircraft range
   TSFC             :     0.51000            1 / hour           Thrst specific fuel consumption
   a                :   297.86746         meter / second        Speed of sound
   alpha_w_max      :     0.25000             radian            Maximum Wing Angle of Attack
   eta_w            :     0.95000              [-]              Wing Lifting Efficiency
   f_L_total_wing   :     1.05000              [-]              Ratio of Total Lift to Wing Lift
   f_aileron        :     0.04000              [-]              Added Weight Fraction Due to Ailerons
   f_flap           :     0.20000              [-]              Added Weight Fraction Due to Flaps
   f_fuel_res       :     0.02000              [-]              Fuel reserve fraction
   f_lete           :     0.10000              [-]              Added Weight Fraction Due to LETE (Unknown Drela Term)
   f_ribs           :     0.15000              [-]              Added Weight Fraction Due to Ribs
   f_slat           :     0.10000              [-]              Added Weight Fraction Due to Slats
   f_spoiler        :     0.02000              [-]              Added Weight Fraction Due to Spoilers
   f_watt           :     0.03000              [-]              Added Weight Fraction Due to Attachments
   g                :     9.81000     meter / second ** 2       Gravitational constant
   lam_w_min        :     0.12500              [-]              Minimum Taper Ratio
   mu_inf           :     0.00001   kilogram / meter / second   Freestream Air Viscosity
   r_h_w            :     0.85000              [-]              Ratio of height of web to maximum thickness of wing
   r_w_c_w          :     0.70000              [-]              Ratio of chord to wing box width, wing
   rho_cap_w        :  2700           kilogram / meter ** 3     Density of wing cap material (aluminum)
   rho_web_w        :  2700           kilogram / meter ** 3     Density of wing web material (aluminum)
   sigma_max        :   310                megapascal           Ultimate tensile stress of aluminum
   sigma_max_shear  :   155                megapascal           Ultimate shear stress of aluminum

Sensitivities
-------------
   Sensitivities not available for optimizations of type Sequential Log Convex Program

Solve Report
------------
   Solve Method           :  Sequential Log Convex Program
   Classification         :  Iterative
   Solver                 :  cvxopt
   Number of Iterations   :  17
   Termination Status     :  Converged, relative change in the objective funciton (8.52e-05) is witin specified tolerance (1.00e-04)
```

E.2 Results of the Airfoil Design Problem (Phase 2)

```
Objective
---------
   29325.85797 force_pound

Variables
---------
   AR_w              :        9.10577        [-]         Wing Aspect Ratio
   Alower1           :        1.10249        [-]         Kulfan coefficient, lower surface mode 1
   Alower2           :        1.13671        [-]         Kulfan coefficient, lower surface mode 2
   Alower3           :        1.32729        [-]         Kulfan coefficient, lower surface mode 3
   Alower4           :        0.77955        [-]         Kulfan coefficient, lower surface mode 4
   Alpha_p20_MSES_0  :       19.44973      degree       Angle of Attack + 20deg from MSES, segment 0
   Alpha_p20_MSES_1  :       19.15566      degree       Angle of Attack + 20deg from MSES, segment 1
   Aupper1           :        1.12334        [-]         Kulfan coefficient, upper surface mode 1
   Aupper2           :        1.18417        [-]         Kulfan coefficient, upper surface mode 2
   Aupper3           :        1.15806        [-]         Kulfan coefficient, upper surface mode 3
   Aupper4           :        1.27029        [-]         Kulfan coefficient, upper surface mode 4
   C_L_w_0           :        0.42670        [-]         Wing Lift Coefficient, segment 0
   C_L_w_1           :        0.39154        [-]         Wing Lift Coefficient, segment 1
   C_L_w_MSES_0      :        0.42670        [-]         Lift Coefficient from MSES, segment 0
   C_L_w_MSES_1      :        0.39154        [-]         Lift Coefficient from MSES, segment 1
   D_0               :    34589.22841      newton       Total aircraft drag, segment 0
   D_1               :    30102.39882      newton       Total aircraft drag, segment 1
   D_fuse_0          :    10797.32043      newton       Fuselage drag, segment 0
   D_fuse_1          :     9771.60895      newton       Fuselage drag, segment 1
   D_ht_0            :     3599.10681      newton       Horizontal tail drag, segment 0
   D_ht_1            :     3257.20298      newton       Horizontal tail drag, segment 1
   D_vt_0            :     3599.10681      newton       Vertical tail drag, segment 0
   D_vt_1            :     3257.20298      newton       Vertical tail drag, segment 1
   D_w_0             :    16593.69436      newton       Wing drag, segment 0
   D_w_1             :    13816.38389      newton       Wing drag, segment 1
   I_cap_w           :        0.00001        [-]         Area moment of inertia of cap on 2D cross section, normalized by chord^4
   L_total_0         :   684777.21677      newton       Total Aircraft Lift, segment 0
   L_total_1         :   627009.27855      newton       Total Aircraft Lift, segment 1
   L_w_0             :   652168.77788      newton       Wing Lift, segment 0
   L_w_1             :   597151.69386      newton       Wing Lift, segment 1
   L_w_max           :  1799553.40507      newton       Maximum wing lift
   Lam_w             :        0.54343      radian       Wing Sweep
   M_0               :        0.78000        [-]         Mach number, segment 0
   M_1               :        0.78000        [-]         Mach number, segment 1
   M_r_w             :   854034.09158      newton       Wing Root Bending unit per unit chord
   R_0               :     1750.00000   nautical_mile    Range, segment 0
   R_1               :     1750.00000   nautical_mile    Range, segment 1
   S_w               :      151.27738     meter ** 2     Vehicle reference area (wing area)
   V_inf_0           :      232.33662   meter / second   Cruise velocity, segment 0
   V_inf_1           :      232.33662   meter / second   Cruise velocity, segment 1
   W_avg_0           :   153944.04236    force_pound     Average aircraft weight, segment 0
   W_avg_1           :   139598.83709    force_pound     Average aircraft weight, segment 1
   W_cap_w           :    11345.56663    force_pound     Weight of Wing Spar Cap
   W_dry             :    96306.42165    force_pound     Aircraft dry weight
   W_end_0           :   146449.67210    force_pound     Weight at end of segment, segment 0
   W_end_1           :   133071.43847    force_pound     Weight at end of segment, segment 1
   W_fuel_0          :    15372.60752    force_pound     Weight of fuel burned, segment 0
   W_fuel_1          :    13378.23362    force_pound     Weight of fuel burned, segment 1
   W_fuel_prim       :    28750.84115    force_pound     Primary fuel weight (no reserve)
   W_fuel_res        :      575.01682    force_pound     Reserve fuel weight
```

```
   W_fuel_total      :    29325.85797    force_pound     Total fuel carried
   W_max             :   161822.27962    force_pound     Maximum aircraft weight
   W_pay             :    36190.00000    force_pound     Payload weight
   W_start_0         :   161822.27962    force_pound     Weight at start of segment, segment 0
   W_start_1         :   146449.67210    force_pound     Weight at start of segment, segment 1
   W_struct_w        :    12744.76930    force_pound     Weight of Wing Spar
   W_web_w           :     1399.20267    force_pound     Weight of Wing Spar Shear Web
   W_wing            :    20901.42165    force_pound     Wing weight
   b_w               :       37.10101       meter        Wing Span
   c_bar_w           :        5.29415       meter        Wing Mean Aerodynamic Chord
   c_root_w          :        7.24450       meter        Wing Root Chord
   c_tip_w           :        0.90556       meter        Wing Tip Chord
   lam_w             :        0.12500        [-]         Wing Taper Ratio
   nu_w              :        0.90061        [-]         Placeholder, (1+lam_w+lam_w**2)/(1+lam_w)**2
   p_w               :        1.25001        [-]         Dummy Variable (1+2*lam_w)
   q_w               :        1.12500        [-]         Dummy Variable (1+lam_w)
   t_0               :      232.49312       minute       Flight time, segment 0
   t_1               :      232.49312       minute       Flight time, segment 1
   t_cap_w           :        0.00184        [-]         Spar cap thickness per unit chord
   t_web_w           :        0.00148        [-]         Spar web thickness per unit chord
   tau_w             :        0.12465        [-]         Wing Thickness Ratio
   y_c_bar_w         :       11.13029       meter        Spanwise Location of Mean Aerodynamic Chord
   z_bre_0           :        0.09975        [-]         Breguet parameter, segment 0
   z_bre_1           :        0.09565        [-]         Breguet parameter, segment 1

Constants
---------
   N_lift           :     2.50000            [-]            Ultimate Load Factor
   R_req            :  3500             nautical_mile       Required aircraft range
   TSFC             :     0.51000          1 / hour         Thrst specific fuel consumption
   a                :   297.86746       meter / second      Speed of sound
   eta_w            :     0.95000            [-]            Wing Lifting Efficiency
   f_L_total_wing   :     1.05000            [-]            Ratio of Total Lift to Wing Lift
   f_aileron        :     0.04000            [-]            Added Weight Fraction Due to Ailerons
   f_flap           :     0.20000            [-]            Added Weight Fraction Due to Flaps
   f_fuel_res       :     0.02000            [-]            Fuel reserve fraction
   f_lete           :     0.10000            [-]            Added Weight Fraction Due to LETE (Unknown Drela Term)
   f_ribs           :     0.15000            [-]            Added Weight Fraction Due to Ribs
   f_slat           :     0.10000            [-]            Added Weight Fraction Due to Slats
   f_spoiler        :     0.02000            [-]            Added Weight Fraction Due to Spoilers
   f_watt           :     0.03000            [-]            Added Weight Fraction Due to Attachments
   g                :     9.81000    meter / second ** 2    Gravitational constant
   lam_w_min        :     0.12500            [-]            Minimum Taper Ratio
   r_h_w            :     0.85000            [-]            Ratio of height of web to maximum thickness of wing
   r_w_c_w          :     0.70000            [-]            Ratio of chord to wing box width, wing
   rho_cap_w        :  2700          kilogram / meter ** 3   Density of wing cap material (aluminum)
   rho_web_w        :  2700          kilogram / meter ** 3   Density of wing web material (aluminum)
   sigma_max        :   310               megapascal        Ultimate tensile stress of aluminum
   sigma_max_shear  :   155               megapascal        Ultimate shear stress of aluminum

Sensitivities
-------------
   Sensitivities not available for optimizations of type Sequential Log Convex Program

Solve Report
------------
   Solve Method           :  Sequential Log Convex Program
   Classification         :  Iterative
```

```
Solver                 :  cvxopt
Number of Iterations   :  59
Termination Status     :  Converged, relative change in the objective funciton (1.68e-06) is witin specified tolerance (1.00e-04)
```

E.3 Results of the Wing Design Problem (Phase 3)

```
Objective
---------
   25493.18388 force_pound

Variables
---------
   Alpha_p20_MSES_0    :         20.40409           degree              Angle of attack, Segment 0
   Alpha_p20_MSES_1    :         20.11633           degree              Angle of attack, Segment 1
   D_0                 :      29437.22895           newton              Total Drag, Segment 0
   D_1                 :      26802.73540           newton              Total Drag, Segment 1
   D_fuse_0            :      10054.52783           newton              Fuselage Drag, Segment 0
   D_fuse_1            :       9181.64922           newton              Fuselage Drag, Segment 1
   D_ht_0              :       3351.50928           newton              Horizontal Tail Drag, Segment 0
   D_ht_1              :       3060.54974           newton              Horizontal Tail Drag, Segment 1
   D_vt_0              :       3351.50928           newton              Vertical Tail Drag, Segment 0
   D_vt_1              :       3060.54974           newton              Vertical Tail Drag, Segment 1
   D_w_0               :      12679.68256           newton              Wing Drag, Segment 0
   D_w_1               :      11499.98671           newton              Wing Drag, Segment 1
   L_total_0           :     640545.41372           newton              Total Lift, Segment 0
   L_total_1           :     585018.46696           newton              Total Lift, Segment 1
   L_w_0               :     610043.25116           newton              Wing Lift, Segment 0
   L_w_1               :     557160.44472           newton              Wing Lift, Segment 1
   L_w_max             :    1675754.63857           newton              Maximum Wing Lift
   Lam_w               :         24.64614           degree              Wing Mid Chord Sweep
   M_0                 :          0.78000        dimensionless          Mach Number, Segment 0
   M_1                 :          0.78000        dimensionless          Mach Number, Segment 1
   R_0                 :       1750.00000        nautical_mile          Range, Segment 0
   R_1                 :       1750.00000        nautical_mile          Range, Segment 1
   V_inf_0             :        232.33662        meter / second         Cruise Velocity, Segment 0
   V_inf_1             :        232.33662        meter / second         Cruise Velocity, Segment 1
   W_avg_0             :     144000.33746         force_pound           Average Weight, Segment 0
   W_avg_1             :     131517.38325         force_pound           Average Weight, Segment 1
   W_dry               :      89006.66780         force_pound           Dry aircraft weight
   W_end_0             :     137607.79041         force_pound           Ending Weight, Segment 0
   W_end_1             :     125696.53415         force_pound           Ending Weight, Segment 1
   W_fuel_0            :      13082.06126         force_pound           Fuel Weight Consumed, Segment 0
   W_fuel_1            :      11911.25627         force_pound           Fuel Weight Consumed, Segment 1
   W_fuel_prim         :      24993.31753         force_pound           Primary Fuel
   W_fuel_res          :        499.86635         force_pound           Reserve Fuel
   W_fuel_total        :      25493.18388         force_pound           Total Fuel
   W_max               :     150689.85168         force_pound           Maximum Aircraft Weight (MTOW)
   W_pay               :      36190.00000         force_pound           Payload Weight
   W_start_0           :     150689.85168         force_pound           Starting Weight, Segment 0
   W_start_1           :     137607.79041         force_pound           Starting Weight, Segment 1
   W_struct_w          :       8293.69988         force_pound           Weight of wing spar
   W_wing              :      13601.66780         force_pound           Weight of wing
   alpha_sec_p20_1_0   :         21.84714           degree              Angle of attack at the outboard section (+20 deg), Segment 0
   alpha_sec_p20_1_1   :         21.55937           degree              Angle of attack at the outboard section (+20 deg), Segment 1
   alpha_sec_p20_2_0   :         20.95863           degree              Angle of attack at the mid section (+20 deg), Segment 0
   alpha_sec_p20_2_1   :         20.67087           degree              Angle of attack at the mid section (+20 deg), Segment 1
   alpha_sec_p20_3_0   :         20.28781           degree              Angle of attack at the inboard section (+20 deg), Segment 0
   alpha_sec_p20_3_1   :         20.00004           degree              Angle of attack at the inboard section (+20 deg), Segment 1
   b_w                 :         34.59783           meter               Wingspan
   c_1                 :          1.70296           meter               Chord, outboard section
   c_2                 :          2.87933           meter               Chord, mid section
   c_3                 :          4.31439           meter               Chord, inboard section
```

```
   sigma_1             :  309999991.64651            pascal              Stress in spar cap, outboard section
   sigma_2             :  309999999.97857            pascal              Stress in spar cap, mid section
   sigma_3             :  309999999.99920            pascal              Stress in spar cap, inboard section
   sigma_shear_1       :   57925276.79019            pascal              Stress in spar web, outboard section
   sigma_shear_2       :   62318669.04190            pascal              Stress in spar web, mid section
   sigma_shear_3       :   60536962.70784            pascal              Stress in spar web, inboard section
   t_0                 :         232.49312           minute              Flight time, Segment 0
   t_1                 :         232.49312           minute              Flight time, Segment 1
   t_cap_1             :           0.00768        dimensionless          Spar cap thickness scaled by chord, outboard section
   t_cap_2             :           0.00881        dimensionless          Spar cap thickness scaled by chord, mid section
   t_cap_3             :           0.00662        dimensionless          Spar cap thickness scaled by chord, inboard section
   t_web_1             :           0.00518        dimensionless          Spar web thickness scaled by chord, outboard section
   t_web_2             :           0.00449        dimensionless          Spar web thickness scaled by chord, mid section
   t_web_3             :           0.00353        dimensionless          Spar web thickness scaled by chord, inboard section
   theta_p20_1         :          21.44305           degree              Twist (+20 deg), outboard section
   theta_p20_2         :          20.55445           degree              Twist (+20 deg), mid section
   theta_p20_3         :          19.88368           degree              Twist (+20 deg), inboard section
   z_bre_0             :           0.09082        dimensionless          Breguet range parameter, Segment 0
   z_bre_1             :           0.09054        dimensionless          Breguet range parameter, Segment 1

Constants
---------
   N_lift           :     2.50000            [-]            Ultimate Load Factor
   R_req            :  3500             nautical_mile       Required aircraft range
   TSFC             :     0.51000          1 / hour         Thrst specific fuel consumption
   a                :   297.86746       meter / second      Speed of sound
   eta_w            :     0.95000            [-]            Wing Lifting Efficiency
   f_L_total_wing   :     1.05000            [-]            Ratio of Total Lift to Wing Lift
   f_aileron        :     0.04000            [-]            Added Weight Fraction Due to Ailerons
   f_flap           :     0.20000            [-]            Added Weight Fraction Due to Flaps
   f_fuel_res       :     0.02000            [-]            Fuel reserve fraction
   f_lete           :     0.10000            [-]            Added Weight Fraction Due to LETE (Unknown Drela Term)
   f_ribs           :     0.15000            [-]            Added Weight Fraction Due to Ribs
   f_slat           :     0.10000            [-]            Added Weight Fraction Due to Slats
   f_spoiler        :     0.02000            [-]            Added Weight Fraction Due to Spoilers
   f_watt           :     0.03000            [-]            Added Weight Fraction Due to Attachments
   g                :     9.81000    meter / second ** 2    Gravitational constant
   rho_cap_w        :  2700          kilogram / meter ** 3   Density of wing cap material (aluminum)
   rho_web_w        :  2700          kilogram / meter ** 3   Density of wing web material (aluminum)
   sigma_max        :   310               megapascal        Ultimate tensile stress of aluminum
   sigma_max_shear  :   155               megapascal        Ultimate shear stress of aluminum

Solve Report
------------
   Solve Method           :  Sequential Log Convex Program
   Classification         :  Iterative
   Solver                 :  cvxopt
   Number of Iterations   :  25
   Termination Status     :  Terminated Manually with Relative Tolerance of 1.367e-05
```