A modular adjoint approach to aircraft mission analysis and optimization

Jason Y. Kao, * John T. Hwang, * Joaquim R. R. A. Martins, †
University of Michigan, Ann Arbor, Michigan, 48109, United States
Justin S. Gray, ‡
NASA Glenn Research Center, Cleveland, Ohio, 44135, United States
and Kenneth T. Moore §
DB Consulting Inc., Berkshire, United Kingdom

Aircraft design and trajectory optimization are typically performed sequentially, which can lead to suboptimal results. To address this, we develop a new mission analysis and trajectory optimization tool that is efficient, robust, and modular. This enables large-scale optimization in problems involving trajectory and other disciplines. The most important feature that sets this tool apart from existing mission analysis software is the use of a computational framework to provide benefits in efficiency and modularity. Through different test cases, we are able to demonstrate the efficiency and the robustness of the developed approach. The generated mission analysis results match well with data from other tools. Runge oscillations are evident in trajectory optimization cases with insufficient number of altitude design variables. Therefore, we provide a relation between the minimum number of altitude design variables needed and the range of the mission for avoiding these oscillations efficiently.

I. Introduction

Commercial aviation is an integral part of the global economy, and a catalyst of world trade. However, its impact on the environment is significant and cannot be neglected. Currently, commercial aviation is responsible for around 2% of the total global carbon dioxide production, and the industry has collectively realized the need to reduce its share of carbon emissions. In 2010, the International Civil Aviation Organization (ICAO) formulated industry-wide goals for reducing carbon emissions. These goals can be summarized by two main objectives: to establish carbon-neutral growth beyond the year 2020, and to further reduce carbon emissions to half of the current levels by the year 2050.

One way of reducing the overall carbon emissions of the industry is by changing the operation of an aircraft. For example, many commercial operators have adopted the policy of single-engine taxiing, where the carbon emissions from these aircraft are reduced for the duration of their taxi and wait time on the ground. However, since taxiing represents only a small fraction of the total operating time of any particular aircraft, it would be beneficial if the fuel burn for other segments of a particular flight can be similarly reduced. The flight management system of current aircraft already provides a feature for the crew to enter a "cost index", which relates the time-cost of a specific airline to the current cost of fuel, and outputs the optimal cruise Mach number for the aircraft. This system works within the confines of the current Air Traffic Control (ATC) rules, and provides only the optimal cruise speed. However, with the advent of the Next Generation Air Transportation System (NextGen), more fuel-efficient, and therefore more environmentally-friendly trajectories may be flown by future commercial aircraft. As a result, determining the optimal trajectory for commercial aircraft is important for reducing the total environment impact of the industry.

The design and configuration choice of a new aircraft also affect the overall carbon emissions of the industry. As a result, this particular topic has received much attention from researchers and aeronautical engineers seeking to apply the methods of numerical optimization. Their efforts primarily focus on optimizing the aerodynamic properties of the aircraft, as well as the sizing of structural components. Lyu *et al.* [1] have performed aerodynamic shape optimization on a conventional aircraft configuration using the Common Research Model (CRM) as the initial design. Aerodynamic shape optimization has also been performed on unconventional configurations such as a flying wing [2]. Similarly, Lyu and Martins have applied such optimization methods to the blended-wing-body configuration [3]. Despite the success of these purely aerodynamic optimization demonstrations, aeronautical engineers must consider a fully coupled aerostructural model and optimize the aerodynamic properties, as well as the sizing of structural components of

*Ph.D. Candidate, Department of Aerospace Engineering, AIAA Student Member
†Associate Professor, Department of Aerospace Engineering, AIAA Associate Fellow
‡OpenMDAO Project Team Leader, NASA Glenn Research Center
§Senior Analyst, DB Consulting Inc.

an aircraft simultaneously to obtain a realistic optimal design. This has been performed by Kenway and Martins [4] with the CRM configuration.

However, some of these design cases are single-point optimization problems, meaning that the design is only optimized about a single flight condition. Single-point optimization is known to inflate the benefits of numerical optimization, since the optimizer may seek to improve only specified point at the cost of heavy penalties at other potentially significant flight conditions. To address this, Lyu and Martins formulated and applied a multi-point optimization problem with the blended-wing-body configuration to increase the robustness of the optimized design [3]. Kenway and Martins have also performed multi-point optimization on the CRM configuration using different with varying set of optimized points [5]. The choice of these flight conditions remains arbitrary, and may not reflect what is significant during the actual operation of the aircraft. Liem *et al.* took the approach of using the actual flight data of a certain type of aircraft over one year, and representing the total overall usage with a smaller subset of missions and corresponding numerical weights [6]. Ultimately, the goal of aeronautical engineers is to optimize the design of an aircraft based on its total expected usage, and to optimize this usage by altering the flown trajectories.

In addition to optimizing individual trajectories for a particular aircraft, we also wish to simultaneously optimize the allocation of a particular aircraft given a network of routes. Coupling the ability to solve this optimization problem with the aircraft design optimization problem provides the ability for the aircraft manufacturers to come up with the ultimate optimal aircraft design given the existing network of routes flown by the airlines. The solution of these three coupled problems will yield a design with the biggest impact in reducing carbon emissions across the entire sector. However, several challenges prevent the solution of these coupled problems. The first is the need for an efficient, modular mission analysis tool that can be used to analyze a variety of trajectories for a particular aircraft model, and provide accurate estimates for the fuel burn. Therefore, the goal of this paper is to present a tool capable of performing mission analysis and optimization efficiently and robustly in conjunction with other disciplines. In the following section, a comparison of currently available tools for mission analysis is presented, and a brief description for each is provided.

II. Literature Survey

Tools for the analysis and optimization of flight trajectories have been successfully implemented in the past. A brief survey is provided on the prevalent mission analysis tools currently available.

McCullers [7] implemented the Flight Optimization System (FLOPS), which uses a modular approach to allow the user to select the appropriate modules for a given mission. The aerodynamic results are provided by data generated using the empirical drag estimation technique, with modifications to improve accuracy [8]. The Sommer and Short T' method is implemented in the aerodynamic module to account for skin friction drag [9]. The aerodynamic module also allows the user to input aerodynamic data in drag-polar forms, which are corrected by scaling factors to address changes in wing area and engine sizing during optimization. A step integration technique is used within the mission performance module to calculate the state variables, which are: amount of fuel consumed, elapsed time, distance covered, and changes in speed and altitude. The primary mission can be composed of any combination of the built-in flight segment types. FLOPS allows for the use of a number of numerical optimization schemes including: Davidon–Fletcher–Powell, Broyden–Fletcher–Goldfarb–Shano, and a Quadratic Extended Interior Penalty method. However, the gradients required for these schemes are computed using a finite-difference method, which is inefficient for large numbers of variables and is subject to round-off errors.

Another widely-used tool for trajectory analysis and optimization is the Transport Aircraft System Optimization (TASOPT) by Drela [10]. Unlike FLOPS, TASOPT performs most predictions using low-order models derived from physics-based models rather than historical correlations. This ensures that the optimized design is realizable, and not an artifact of extrapolated data fits. The aerodynamic model relies on 2D viscous and inviscid CFD results for a range of C_L and Mach numbers. The standard trajectory equations are integrated over a parameterized mission profile to obtain the weight of the fuel required, and the flight equilibrium equations are then enforced at the profile points. The design space is filled with regions of feasibility and infeasibility created by the stability and the sizing constraints. The regions of infeasibility are exacerbated by the lack of robustness for some of the design modules, which result in the lack of convergence. Therefore, the gradient and Hessian information is not always available from design computations, which necessitates genetic algorithms with initial designs chosen using uniform Monte Carlo sampling of the design space to perform design optimization.

The pyACDT aircraft conceptual design toolkit developed by Perez and Martins [11] takes an object-oriented approach to modeling and analyzing design concepts. However, the mission analysis component of the tool is computed based on fuel fractions obtained empirically for each of the flight segments. The fuel fractions combined with the Breguet range equation are used to determine the amount of fuel consumed over each of the segments.

Another approach presented by Liem *et al.* [12] uses fuel fractions only for the startup, taxi, takeoff, and landing segments of the mission. The fuel burn for the climb, cruise, and descent segments are derived from the range equation and the flight equilibrium equations at each control point. Finally, the segments are linked together to form residual equations by enforcing that the fuel weights at the end points of two adjacent segments must be equal. The nonlinear residuals are solved using Newton's method stabilized by a line search, where the Jacobian required for the Newton iteration is formed by the use of finite differences.

A simpler approach to the mission analysis problem is presented by Koko [13], who utilizes the midpoint rule such that the values at the midpoint of each discretized flight interval is used to represent all position, speed, time, rate of climb, thrust, and fuel consumption associated with that particular flight interval. The flight equilibrium equations, as well as the fuel burn equations are only enforced at this midpoint. The only exception is that the aircraft mass is defined at the beginning of the flight interval, and is assumed to be constant until the start of the next flight interval, where it is subtracted by the fuel consumption computed at the midpoint of the interval. The derivatives required for the optimization of flight trajectories are provided by the complex step method [14].

The popular tool PIANO developed by Lissys Ltd [15] is another option for performing aircraft conceptual designs. The exact methods utilized to obtain the fuel consumption values as well as to satisfy the flight equilibrium equations are not known. The trajectory optimization is performed by the Nelder–Mead Sequential Simplex algorithm, which is a gradient-free heuristic method.

Finally, the Stanford University Aerospace Vehicle Environment (SUave) was recently developed [16], and is designed to be a modular architecture for the mission analysis of aerospace vehicles. SUave uses a pseudo-spectral approach in discretizing control points within each flight segment. The flight (or space) equilibrium equations are enforced at these control points, and the fuel consumption is obtained by integrating over the segments. SUave utilizes the OpenMDAO framework [17] to drive the trajectory optimization problem, and provides the derivatives required for gradient-based schemes though the use of algorithmic differentiation.

The brief survey of the popular mission analysis tools highlights several features that we propose to improve upon. The first and the most crucial requirement is the need for an efficient mission analysis tool that is able to generate solutions and the total derivatives required for gradient-based optimization algorithms, while guaranteeing robustness. This requirement ensures that the large-scale optimization problem featuring the simultaneous consideration of aircraft design, airline allocation, and mission trajectory remain computationally tractable. The second requirement is a modular approach to the mission analysis problem. This requires us to separate the aerodynamic relations from the flight equilibrium equations to allow for the interchangeable use between high-fidelity aerodynamic solvers and an aerodynamic surrogate. Furthermore, the modularity requirement allows us to experiment with different solution strategies, yielding improvements in both efficiency and robustness.

For these reasons, we chose a direct transcription with collocation approach similar to the one presented by Betts and Cramer [18] and implemented this method within a prototype framework. In addition, we also implemented a cubic B-spline interpolant for the necessary aerodynamic data, as suggested by Betts and Cramer. The computational framework formulates the entire mission analysis problem as a nonlinear mathematical system, and hierarchically decomposes the problem to enable the effective use of block Gauss–Seidel type methods. Furthermore, the framework automates the computation of the desired total derivatives required for gradient-based optimization by applying the adjoint method. Other approaches for the trajectory optimization problem can be found in Betts' comprehensive survey of the different methods [19].

III. Methodology

In this section, we first formulate the equations governing the trajectory of an aircraft in the Theory subsection. We then provide the numerical background of the prototype framework in the Numerical Framework subsection, and describe the implementation of the proposed nonlinear system of equations within the framework. Much of the core of the prototype framework developed by Hwang and Martins [20] has been implemented in the NASA OpenMDAO framework. The mission analysis tool has been incorporated into the OpenMDAO framework upon successful development and implementation within our prototype framework.

A. Theory

We start with the altitude and Mach number profiles specified by the user. In the case of trajectory optimization, these profiles would be chosen by the optimizer as design variables. From these specified profiles, we generate B-spline interpolants to ensure the accuracy of the collocation approach. We are then able to compute the flight conditions, such as temperature, density, and true airspeed at each of the discrete collocation points generated from the B-spline interpolants. The U.S. Standard Atmosphere is used to explicitly calculate temperature, density, and pressure for

altitudes up to 20 km. However, one modification is made to the temperature change near the tropopause at 11 km altitude. Instead of an abrupt change from a negative slope to a zero slope, a cubic function is fitted to the region to smoothly vary temperature as altitude is increased from 10.5 km to 11.5 km. The continuity in slope is desirable for gradient-based optimization, as discontinuities may cause the optimization process to be stuck in a loop. The equation of the cubic fit is determined to be the following:

$$T(h) = (2.00 \times 10^{-11})h^3 + (2.59 \times 10^{-6})h^2 - (6.75 \times 10^{-2})h + 6.20 \times 10^{-2} \tag{1}$$

for

$$10500 \leq h \leq 11500 \tag{2}$$

Having determined the altitude airspeed and flight conditions, we can now enforce the flight equilibrium equations at these points, and compute the state of the aircraft. We start by writing out the full flight equilibrium equations:

$$L + (W_e + W_p + W_f)\cos\gamma - T\sin\alpha + \frac{W_e + W_p + W_f}{g} v^2 \cos\gamma \frac{\mathrm{d}\gamma}{\mathrm{d}x} = 0 \tag{3}$$

$$T\cos\alpha + D + (W_e + W_p + W_f)\sin\gamma + \frac{W_e + W_p + W_f}{g} v \cos\gamma \frac{\mathrm{d}v}{\mathrm{d}x} = 0 \tag{4}$$

$$M - I\left(\frac{\mathrm{d}^2\gamma}{\mathrm{d}x^2}(v\cos\gamma)^2 + \frac{\mathrm{d}\gamma}{\mathrm{d}x}\frac{\mathrm{d}v}{\mathrm{d}x} v(\cos\gamma)^2\right) = 0 \tag{5}$$

where W_e represents the operating empty weight of the aircraft, W_p represents the weight of the payload carried by the aircraft, and W_f represents the weight of the fuel carried at a particular time instance. The direction of forces and the flight angles are shown on Figure 1. As shown on the figure, the flight path angle is represented by γ, and the angle of attack is represented by α. ϕ_T represents the angle at which the engines are mounted with respect to the horizon. It is assumed to be small, therefore neglected in our formulation. From these equations, we assume each of the collocation points to be in a steady flight condition. Although this assumption may not necessarily be valid, the high-fidelity solvers and the aerodynamic surrogate models used have already made the steady flight assumption, resulting in the loss of unsteady aerodynamic effects. Therefore, even by including the unsteady terms in the flight equilibrium equations, we are still not able to model the dynamics accurately. In addition to neglecting the unsteady terms, we have also substituted in the target values for the lift, drag, and moment coefficients: $\tilde{C_L}$, $\tilde{C_D}$, $\tilde{C_M}$ in the following equations:

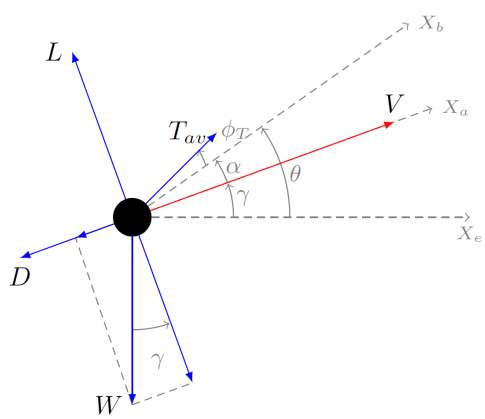

Figure 1: Free body diagram of flight equilibrium.

$$\frac{1}{2}\rho v^2 S\tilde{C_L} + (W_e + W_p + W_f)\cos\gamma - \frac{1}{2}\rho v^2 S\tilde{C_T}\sin\alpha = 0 \tag{6}$$

$$\frac{1}{2}\rho v^2 S\tilde{C_T}\cos\alpha + \frac{1}{2}\rho v^2 S C_D + (W_e + W_p + W_f)\sin\gamma = 0 \tag{7}$$

$$\tilde{C_M} = 0 \tag{8}$$

These target values represent the values required to satisfy the equilibrium equations at the prescribed flight conditions with the current weight estimate. The target values for the three coefficients allow us to solve for the angle of attack α, throttle setting τ, and tail rotation η implicitly with the following relations:

$$\tilde{C_L} - C_L(h, M, \alpha, \eta) = 0 \tag{9}$$

$$\tilde{C_T} - C_T(h, M, t) = 0 \tag{10}$$

$$\tilde{C_M} - C_M(h, M, \alpha, \eta) = 0 \tag{11}$$

The functions C_L, C_T, and C_M represent the lift, thrust, and pitching moment coefficients, respectively, as calculated by the specified model given the current state of the other variables. By forcing these to be equal to their respective target values computed from the equilibrium equations, we effectively compute the angle of attack, throttle setting, and tail rotation angle needed to satisfy the equilibrium equations in a modular manner. A direct approach would substitute these relations into the equilibrium equations, and utilize callbacks when evaluating and solving the coupled system, but this would compromise our modularity requirement, and would prevent us from swapping between aerodynamic solvers and surrogate models easily.

Currently, the aerodynamic relations are solved using a surrogate model with data points generated by a panel code, Tripan [21]. A kriging model is built with the panel code outputs, which is then interpolated by B-splines for efficiency [22]. The additional layer of interpolation reduces the evaluation time of the model by an order of magnitude while incurring a small penalty in accuracy. A similar B-spline approach for interpolating aerodynamic models is used by Betts and Cramer [19].

The last of the coupled equations is the rate of fuel burn equation:

$$\frac{\mathrm{d}W_f}{\mathrm{d}x} = \frac{\mathrm{SFC}\frac{1}{2}\rho v^2 S C_T}{v\cos\gamma} \tag{12}$$

This ODE must be solved to compute the total weight of the aircraft at each of the collocation points. We have converted the rate of fuel burn in time to the rate of fuel burn in horizontal distance through the relations provided by airspeed and flight path angle. Since the amount of fuel carried at the end of the mission is known, and can be chosen *a priori*, we can use this as the initial condition for solving the ODE. The approach here is to apply the explicit Euler scheme starting from the end of the mission, and march backwards in distance to the start of the mission.

B. Numerical Framework

The most important contribution that sets this mission analysis tool apart from existing methods is the idea of utilizing a computational framework to provide benefits in terms of efficiency and modularity. The use of a computational framework allows us to uniquely tailor the solution strategy of the overall problem by selectively using different nonlinear and linear solvers for different blocks within the problem. The framework also automates the computation of total derivatives required for gradient-based optimization. The overall problem can be expressed as a system of nonlinear algebraic equations:

$$R(u) = 0 \tag{13}$$

Here, the vector u represents all variables within the system, and $R(u)$ represents the algebraic system of equations that is the unified formulation of any numerical model. The vector u^* which solves $R(u) = 0$ is defined as the solution of the overall problem.

The computation of total derivatives can be done efficiently by utilizing the unifying chain rule [23], as shown in the following equation:

$$\frac{\partial R}{\partial u}\frac{\mathrm{d}u}{\mathrm{d}r} = \mathcal{I} = \frac{\partial R}{\partial u}^T \frac{\mathrm{d}u}{\mathrm{d}r}^T \tag{14}$$

where r represents the vector of outputs provided by the system of equations with some input vector u.

$$r = R(u) \tag{15}$$

The unifying chain rule equation is presented by Martins and Hwang [23]. From this equation, the total derivatives can be computed using either the direct method or the adjoint method. The computational cost of the adjoint method

is dependent on the number of quantities of interest rather than the number of variables. Therefore, for gradient-based optimization problems with large numbers of design variables, computing total derivatives using the adjoint method is advantageous. Both direct and adjoint methods have been implemented in a prototype of the computational framework, and the total derivatives are automatically calculated with the specification of the partial derivatives.

The other important feature of the prototype framework is the ability to hierarchically decompose the problem, which enables the implementation of different solution strategies. For example, block Gauss–Seidel solvers can be used on certain parts of the problem while Newton–Krylov solvers are used to solve other parts monolithically. For large systems, Newton's method is the only tractable solution method. The lack of robustness of Newton's method can be addressed by implementing a line search or trust region method for selecting the sizes of the Newton steps. Gauss–Seidel methods can be useful by acting as preconditioners, as well as for solving a series of explicit systems.

The basic component of the framework is a mathematical *system*. A system is defined as a *compound system* if it contains subsystems, or an *elementary system* otherwise. Compound systems can be further classified into serial and parallel systems. For the mission analysis problem, only serial systems are used, since the problem size is generally not large enough to possess obvious advantages for parallel computing. Elementary systems can be distinguished between *independent systems*, *explicit systems*, and *implicit systems*. Independent systems consist of variables that are not dependent on other variables. Explicit systems include variables that can be determined exactly by an expression involving only variables from other systems. Implicit systems depend on both variables from within the system as well as variables from other systems.

The objective here, as motivated by previous sections, is to develop a modular mission analysis tool capable of performing the proposed simultaneous optimization of aircraft design, airline allocation and flight trajectories. Therefore, the three driving goals for the development of this tool are: efficiency, robustness, and modularity. Due to the anticipated large size of the overall problem, a gradient-based optimization scheme must be used to keep the problem tractable, which results in the need for total derivatives [23]. Many existing tools utilize finite-difference or complex-step methods to compute such derivatives, but with the anticipated size of the overall coupled optimization problem, the adjoint method must be used.

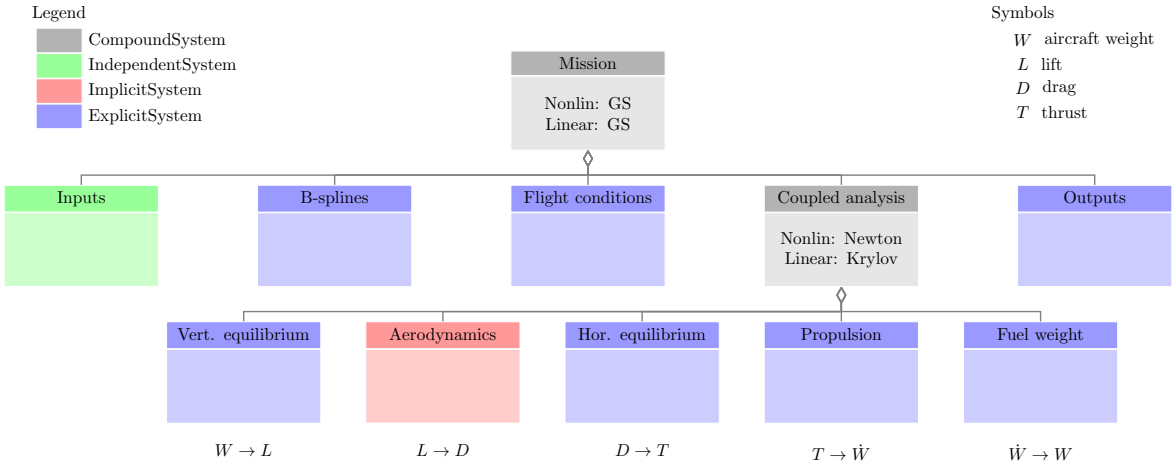

Figure 2: Hierarchical decomposition of the mission problem in the prototype framework.

The hierarchical structure of the mission analysis problem will now be explained. As shown in Figure 2, the overall problem is contained within a serial system named mission. Mission contains 5 separate subsystems, and solves them in sequence using one Gauss–Seidel iteration. The first subsystem is composed of input variables such as altitude and Mach number control points. These are implemented as independent systems, and are initialized with a single block Gauss–Seidel iteration. The second subsystem uses these inputs to generate B-spline interpolants, which allow us to reduce the number of input variables (which are design variables during optimization) while maintaining the accuracy of the collocation method. The B-spline implementation is similar to the approach taken by Hwang *et al.* for a small satellite design optimization problem [20]. The third subsystem takes the parameterized input profiles, and computes the corresponding flight conditions at each collocation point explicitly. This is done by solving explicit systems sequentially once using the block Gauss–Seidel solver.

The fourth subsystem contains the nonlinear coupled system of equilibrium equations, as well as the aerodynamic relations, and the fuel-burn equation. The ordering of subsystems within the coupled analysis block is determined in

a logical way to ensure the convergence of block Gauss–Seidel methods. Although the Newton solver is chosen to solve the coupled analysis system in a monolithic way, the ordering is still important for the linear block Gauss–Seidel preconditioner, which speeds up the Krylov subspace solver for smaller problems. The final subsystem within the mission system contains the output variables necessary for optimization, which are the objective function (total fuel burn of the mission), and the constraints. The constraints used for trajectory optimization will be discussed further in the Results section.

C. Implementation

The hierarchical structure of the problem described in the previous subsection is implemented in a prototype computational framework during the development of the proposed mission analysis tool. The main advantage provided by the prototype framework is the ability to rapidly change solution methods for solving the mathematical system and rapidly add components as well as compute derivatives. For example, we were able to explore the effects of solving different parts of the problem with Newton–Krylov and block Gauss–Seidel methods. Additionally, we were able to utilize the linear block Gauss-Seidel method as a preconditioner for the Krylov subspace solver. However, the usage of the Gauss–Seidel preconditioner slows down the solution process for problems of non-trivial size. Furthermore, we have experimented with a globalized solver in which the nonlinear block Gauss–Seidel solver is used prior to the Newton–Krylov solver. The strategy was to improve the condition of a problem by using a partially-solved state as the initial starting point for the Newton solver. However, this approach is also detrimental to the performance of the solution process.

The core of the prototype framework presented here has been implemented in NASA's OpenMDAO framework along with the proposed mission analysis tool. The results generated from this implementation will be presented in the following section.

IV. Results

In this section, we will present verification results for mission analysis, followed by optimization results for a range of different missions. As mentioned in the previous section, the results presented here were generated using the OpenMDAO framework. For verification purposes, a benchmark mission is analyzed, and the results are compared with known data from PASS, TASOPT, and SUave. The test entails a 2950-nautical mile mission flown with a Boeing 737-800 aircraft at a cruise Mach number of 0.78. The altitude and Mach profiles are approximated with linear variation of the B-spline control points for climb and descent segments, and a constant cruise segment. The profiles are specified by 100 B-spline control points, which span the length of the mission using a cosine distribution. These 100 B-spline control points are used to generate 501 collocation points. It is noted here that an aerodynamic surrogate model for this aircraft was not available, resulting in the need to express the aerodynamic relations with a simple drag polar using estimated coefficients. The engine model uses available information of thrust and SFC for the CFM56-7B27 gas turbine engine. The results of this analysis is presented on Figure 3.

This particular benchmark mission was chosen due to the existence of data from the currently available tools produced by the SUave team [24]. Table 1 shows a comparison of the results obtained with our proposed mission analysis tool with the others. The time taken for the evaluation of this mission was 0.37 seconds on a single processor, which satisfies our efficiency requirements.

Parameter	PASS	TASOPT	SUave	**pyMission**
Cruise Mach	0.78	0.78	0.78	0.78
Cruise altitude (ft)	35 000	35 000	35 000	35 000
Mission range (nmi)	2 950	2 950	2 950	2 950
Payload (lb)	36 540	36 540	36 540	36 540
Block fuel (lb)	38 180	41 238	39 556	38 687
Time taken (s)				**0.37**

Table 1: Mission analysis verification results [24].

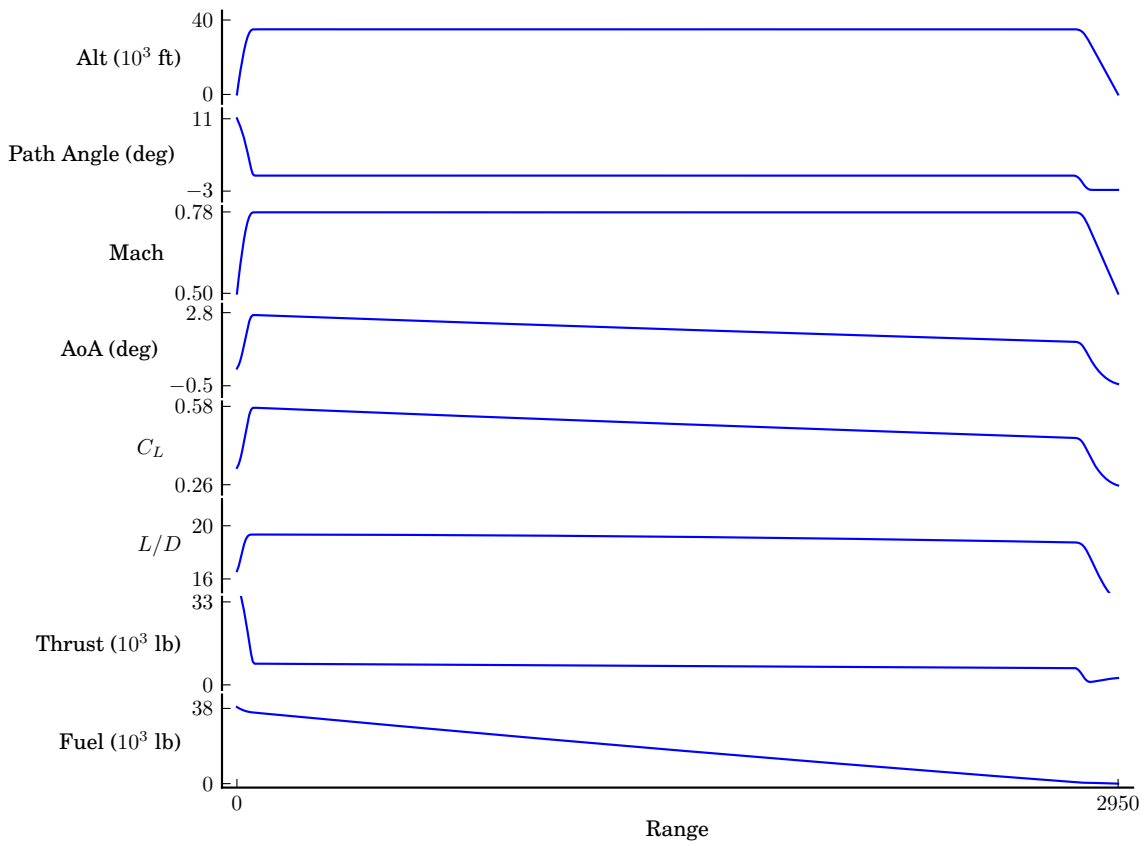

Figure 3: Boeing 737-800 mission analysis verification results.

The trajectory optimization problem is formulated with the fuel burn of the mission as the objective function. The design variables are the N altitude control points used to control the B-spline interpolant, resulting in M collocation points for the altitude profile. The first constraint for the optimization problem is that the initial and final altitude points must be at sea level. The second constraint ensures that the throttle remains between 0% and 100%. Due to the robustness requirement, the mission analysis tool allows the throttle setting to go beyond the physical limits. Therefore, constraints must be established during optimization to prevent unrealistic results. These constraints are implemented using the Kreisselmeier–Steinhauser (KS) function, which aggregates the M constraints into a single constraint function. Lastly, linear constraints are established between neighboring collocation points which limit the maximum and minimum path angles. This prevents the optimizer from taking large steps in altitude beyond normal operation limits, which slows down the analysis and the optimization considerably. The mathematical formulation of the optimization problem can be summarized by the relations below.

$$
\begin{aligned}
\text{minimize} \quad & \text{fuel burn} = W_{f0} \\
\text{with respect to} \quad & h_j \geq 0 && \text{for } j = 0, ..., N \\
\text{subject to} \quad & h_0 = 0 \\
& h_N = 0 \\
& 0 \leq \tau_k \leq 1 && \text{for } k = 0, ..., M \\
& |\gamma_k| \leq \gamma_{\text{max}} && \text{for } k = 0, ..., M \\
& TAS_k = TAS_{\text{fixed}} && \text{for } k = 0, ..., M
\end{aligned}
$$

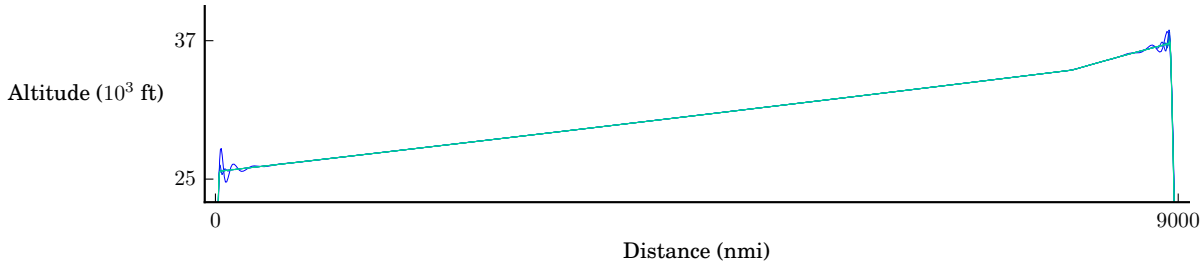

Figure 4: Visible Runge oscillations on optimized trajectories resulting from insufficient number of B-spline control points.

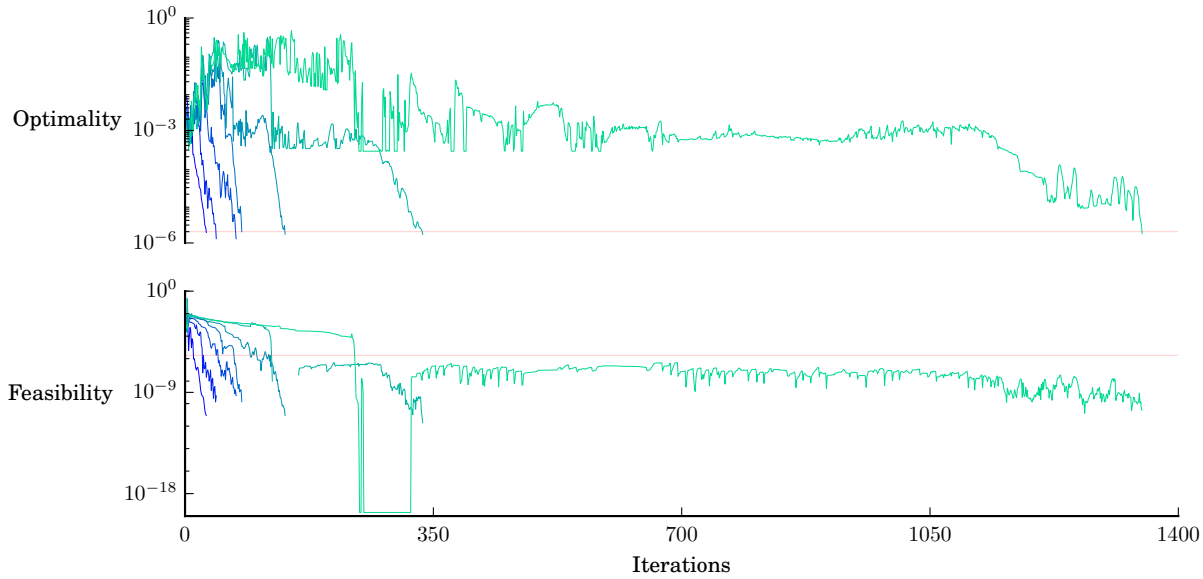

Figure 5: Optimization history with varying number of B-spline control points.

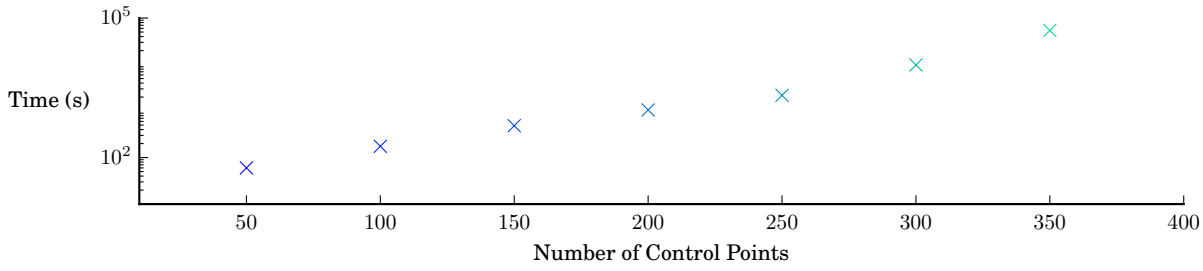

Figure 6: Time taken for each optimization problem.

In the case of altitude and airspeed optimization, Mach number control points used to control the B-spline interpolant for the airspeed profile are added to the list of design variables. Furthermore, the takeoff and landing speed constraints are added. Lastly, due to inaccuracies within our surrogate model, an upper bound for the Mach number control points must be specified also. This results from errors within the surrogate model where the wave drag appears to be absent when the angle of attack is small. Therefore, optimization performed without a realistic upper bound

yields inaccurate fuel burn estimates as well as unrealistic block times. The following is the optimization problem with Mach profile added as design variables:

$$
\begin{aligned}
\text{minimize} \quad & \text{fuel burn} = W_{f0} \\
\text{with respect to} \quad & h_j \geq 0 && \text{for } j = 0, ..., N \\
& 0 \leq \text{Mach}_j \leq \text{Mach}_{\text{max}} && \text{for } j = 0, ..., N \\
\text{subject to} \quad & h_0 = 0 \\
& h_N = 0 \\
& 0 \leq \tau_k \leq 1 && \text{for } k = 0, ..., M \\
& |\gamma_k| \leq \gamma_{\text{max}} && \text{for } k = 0, ..., M \\
& \text{Mach}_0 = \text{Mach}_{\text{takeoff}} \\
& \text{Mach}_N = \text{Mach}_{\text{landing}}
\end{aligned}
$$

We first conduct a refinement study on a 9000-nautical mile mission to establish a balance between obtaining accurate results and doing so in a timely manner. Furthermore, a refinement study can provide us with insights about the optimization process, as well as ways to improve its efficiency. Runge oscillations can be observed on the optimized trajectories when the number of B-spline control point is insufficient for the length of the mission, as demonstrated by Figure 4. The optimization history for each of the cases is presented in Figure 5, with the red line highlighting the tolerances set for optimality and feasibility. The time taken is shown in Figure 6 with the same color scheme as Figures 4 and 5 for the different resolutions. The number of B-spline control points varies linearly from 50 to 350, and the number of collocation points for these corresponding cases range from 251 to 1751. From the results, we can conclude that around 250 B-spline control points are needed for the optimization of a 9000-nautical mile mission. A similar study conducted for a 100-nautical mile mission shows that only 10 B-spline control points are needed. Based on these two numbers, we are now able to estimate the number of B-spline control points needed to avoid Runge oscillations for missions of any length. This useful relation is expressed by the following equation:

$$cp = \frac{12}{445}\text{range} + \frac{650}{89} \tag{16}$$

It is worth noting that the optimizer appears to struggle with the larger problems. This can be seen with the 350 control point example on Figure 5. The optimality parameter appears to be stuck on a plateau, while the feasibility parameter is consistently below the specified tolerance. One possible solution for improving the efficiency of larger problems is to apply a ”grid-sequencing” approach, where the same problem is first solved with a small number of control points, yielding a inaccurate ”optimum”. This state is then used as the starting point of the higher-resolution problem. This approach was tested during development with simplified aerodynamic models, and was found to yield no benefit in terms of efficiency. However, with the implementation of more sophisticated aerodynamic models, the approach may be worthwhile.

One of the crucial requirements for the proposed tool is the need for robustness. In order for the combined airline allocation and trajectory optimization problem to be solved successfully, the proposed tool must be able to optimize the trajectory for a given aircraft for any reasonable combination of range and payload. Therefore, we perform a test which sweeps the range and the payload of the missions, and the results are visualized with a carpet plot. These tests are applied to an altitude-only optimization, in which the Mach number of the entire mission is fixed. Although this fixed airspeed case is unrealistic, it represents a stepping stone for the altitude and airspeed optimization case. The results are shown on Figure 7 and Figure 9. Figure 7 shows the optimized results for missions ranging from 100 to 9000 nautical miles, where the color scheme varies from dark blue to light green as the mission length is increased. The number of B-spline control points and collocation points are chosen according to the findings of the refinement studies done previously. The ranges are normalized for visualization purposes. The optimization history for these missions is shown on Figure 8 with a color scheme that corresponds to the one used for the carpet plot. As expected, the shorter missions require less time, as the number of control points needed to accurately generate results is comparably small. Figure 9 shows the optimized trajectories for a 100-nautical mile mission with the weight of the aircraft varying from operating empty weight (OEW), represented by the dark blue line, to maximum take-off weight (MTOW), represented by the light green line. The lighter aircraft is able to fly a higher trajectory and consume less fuel relative to the heavier aircraft.

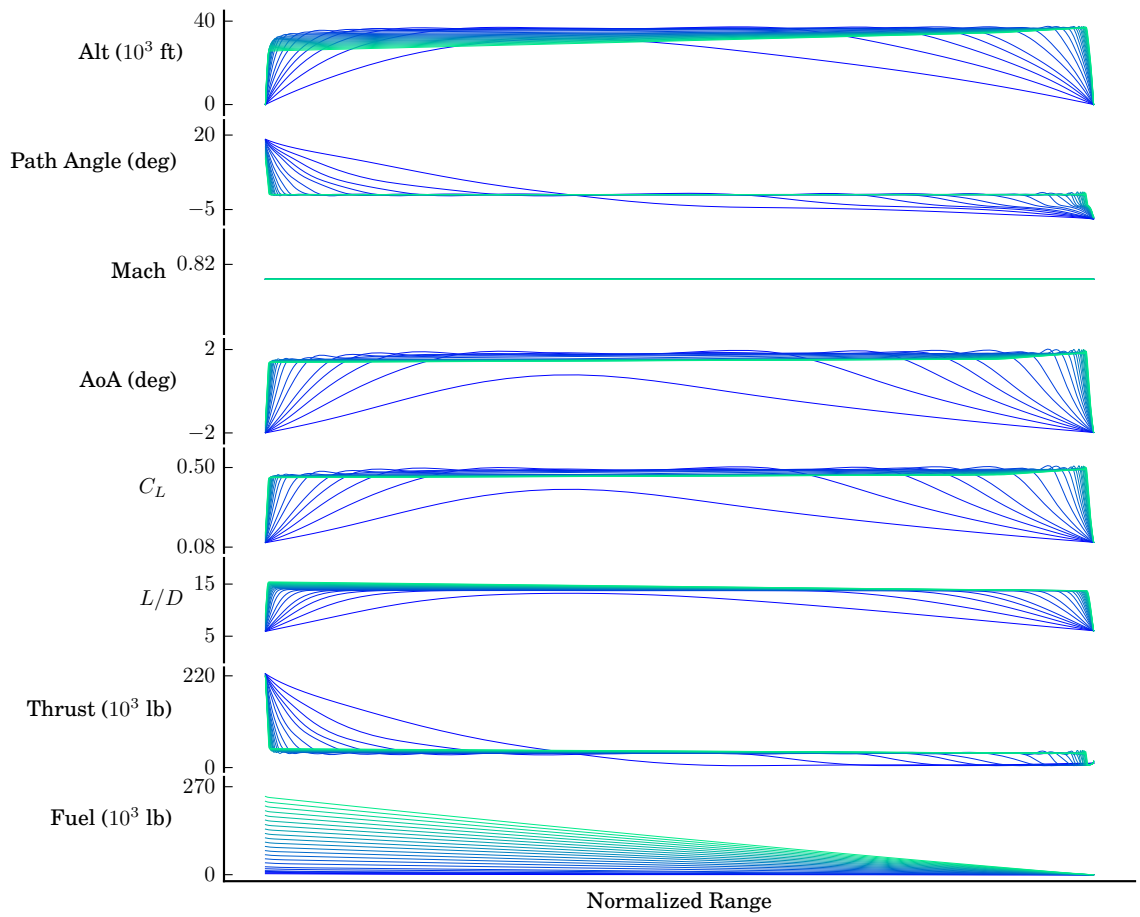

Figure 7: Optimized trajectories for missions ranging from 100 to 9000 nautical miles with constant Mach number.

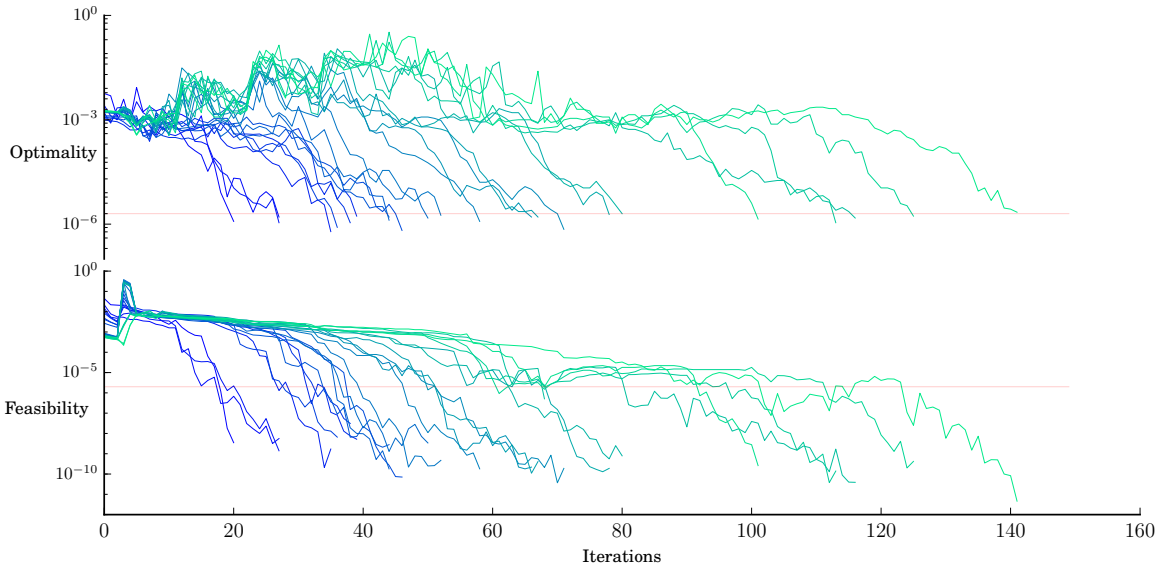

Figure 8: The optimization history of missions with different ranges. The red line indicates the tolerances for optimality and feasibility.

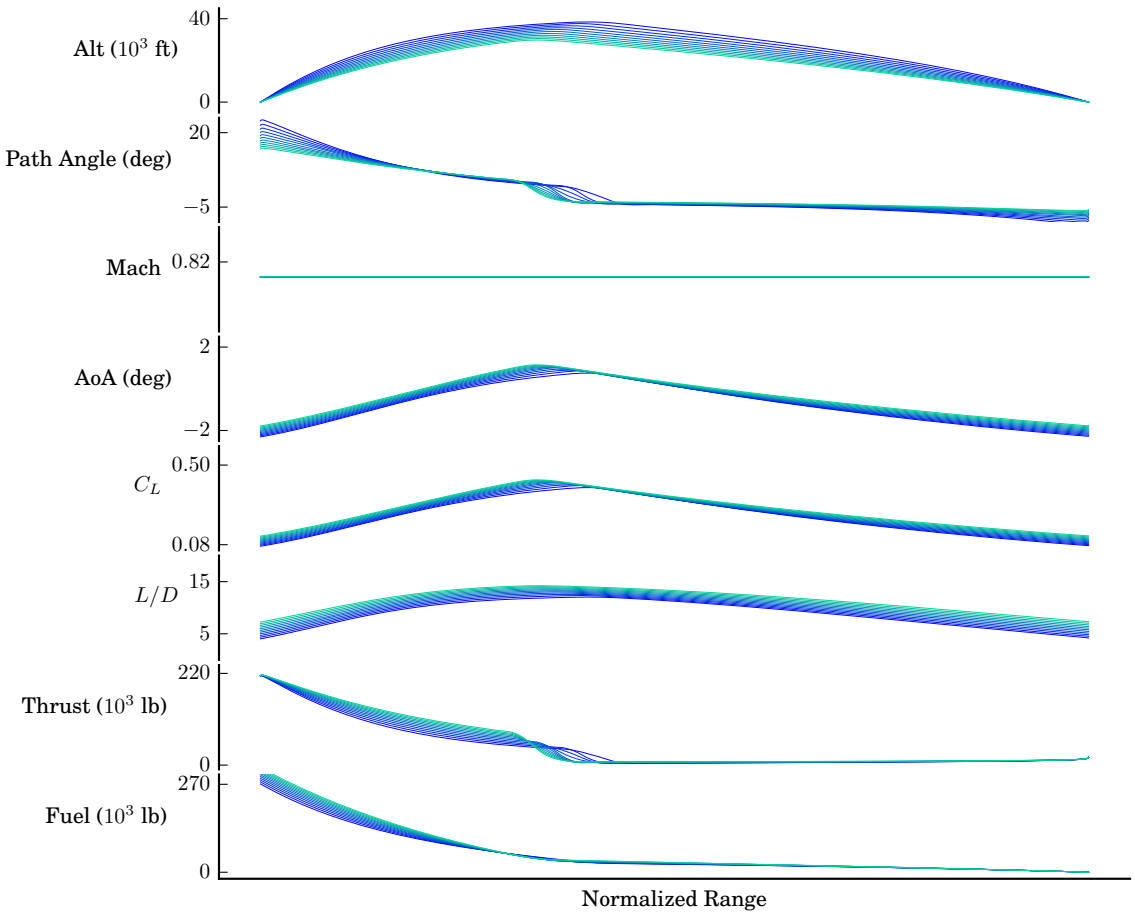

Figure 9: Optimized trajectories for aircraft weight varying from OEW to MTOW on a 100-nautical mile mission with constant Mach number.

Finally, optimization is performed by varying altitude and Mach number control points. Due to the low takeoff and landing speeds, we have had to modify the aerodynamic model to include the effects of high-lift devices when the aircraft is below a certain airspeed. Without these considerations, the angle of attack of the aircraft would increase to 12 degrees during takeoff and landing, since the aerodynamic surrogate does not model stall effects. For this particular mission, the optimizer has determined that the most fuel efficient method of flying a distance of 1000 nautical miles for the CRM aircraft is to cruise at just under 25000 ft with a Mach number of around 0.6. The resulting plot from a case with 30 B-spline control points and 151 collocation points is shown on Figure 10.

The low cruise Mach number chosen by the optimizer for this mission is not surprising. It is well known that the optimal cruise Mach number for modern airliners is around Mach 0.6, which is situated just prior to the wave drag rise. However, operators often have significant costs associated with the flight time of the aircraft, therefore an altitude-Mach optimization with a known cost-index relating the cost of fuel and time is proposed for the immediate future.

The results we generated are similar to what we expect to be optimal trajectories. All the profiles exhibit a high-power climb segment to reach the low-density altitudes quickly, a cruise-climb segment in which altitude increases as the aircraft loses weight from fuel burn, and a very low-power descent segment where the aircraft essentially glides to the destination. Therefore, our current effort involves attempting to parameterize these optimal trajectory curves accurately, and generating an effective surrogate model for the entire trajectory optimization problem. This would eliminate the majority of the computation time required for optimizing trajectory within our simultaneous optimization problem involving multiple disciplines. Additionally, by parameterizing these curves accurately, we can eliminate the need for performing trajectory optimization for all future applications.

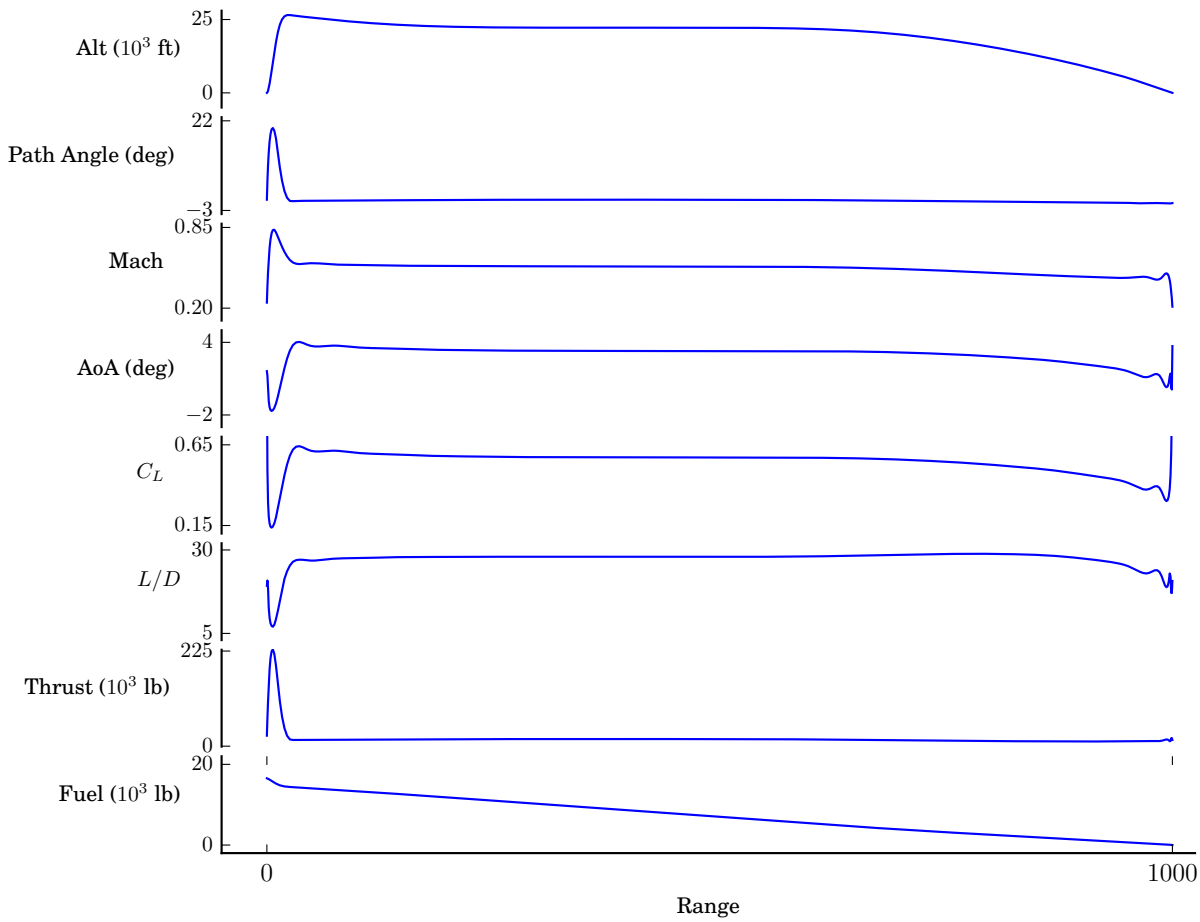

Figure 10: Altitude and Mach Number optimization for a 1000-nautical mile mission.

V. Conclusion

In this study, we have proposed a new mission analysis and trajectory optimization tool. The robustness of the tool is demonstrated by successfully optimizing missions with ranges varying from 100 to 9000 nautical miles, and aircraft weights varying from OEW to MTOW. The combination of the proposed tool and the use of a framework allows for efficient computation of total derivatives required for gradient-based optimization schemes. The mission analysis results generated using this tool match well with results generated by existing software. Altitude and Mach profile optimization was demonstrated. A relation between the number of B-spline control points required for an oscillation-free solution and the range of the mission is also presented. The progress of the proposed tool thus far has demonstrated its suitability for the eventual use in a simultaneous multidisciplinary optimization problem.

VI. Acknowledgments

The authors gratefully acknowledge support from NASA through grant number NNX14AC73A. The authors would also like to thank Rhea P. Liem for providing the aerodynamic surrogate models.

References

[1] Lyu, Z., Kenway, G. K., and Martins, J. R. R. A., "Aerodynamic Shape Optimization Studies on the Common Research Model Wing Benchmark," *AIAA Journal*, 2014, (In press).

[2] Mader, C. A. and Martins, J. R. R. A., "Stability-Constrained Aerodynamic Shape Optimization of Flying Wings," *Journal of Aircraft*, Vol. 50, No. 5, September 2013, pp. 1431–1449. doi:10.2514/1.C031956.

[3] Lyu, Z. and Martins, J. R. R. A., "Aerodynamic Design Optimization Studies of a Blended-Wing-Body Aircraft," *Journal of Aircraft*, 2014. doi:10.2514/1.C032491, (In press).

[4] Kenway, G. K. W. and Martins, J. R. R. A., "Multi-Point High-Fidelity Aerostructural Optimization of a Transport Aircraft Configuration," *Journal of Aircraft*, Vol. 51, No. 1, January 2014, pp. 144–160. doi:10.2514/1.C032150.

[5] Kenway, G. K. W. and Martins, J. R. R. A., "Multipoint Aerodynamic Shape Optimization Investigations of the Common Research Model Wing," *Proceedings of the AIAA Science and Technology Forum and Exposition (SciTech)*, Kissimmee, FL, January 2015.

[6] Liem, R., Kenway, G. K. W., and Martins, J. R. R. A., "Multi-mission Aircraft Fuel Burn Minimization via Multi-point Aerostructural Optimization," *AIAA Journal*, 2014, (Accepted subject to revisions).

[7] McCullers, L. A., "Aircraft Configuration Optimization Including Optimized Flight Profiles," *Recent Experiences in Multidisciplinary Analysis and Optimization*, Apr 1984.

[8] Feagin, R. C. and Morrison, W. D., "Delta Method, An Empirical Drag Buildup Technique," Tech. rep., NASA, 1978.

[9] Sommer, S. C. and Short, B. J., "Free-Flight Measurements of Turbulent-Boundary-Layer Skin Friction in the Presence of Severe Aerodynamic Heating at Mach Numbers From 2.8 to 7.0," Tech. rep., NASA, 1955.

[10] Greitzer, E. M., "Design Methodologies for Aerodynamics, Structures, Weight, and Thermodynamic Cycles," Tech. rep., Massachusetts Institute of Technology, Aurora Flight Sciences, Pratt & Whitney, March 2010.

[11] Perez, R. E. and Martins, J. R. R. A., "pyACDT: An Object- Oriented Framework for Aircraft Design Modelling and Multidisciplinary Optimization," 2010.

[12] Liem, R. P., Mader, C. A., Lee, E., and Martins, J. R. R. A., "Aerostructural design optimization of a 100-passenger regional jet with surrogate-based mission analysis," *2013 Aviation Technology, Integration, and Operations Conference*, Sep 2013. doi:10.2514/6.2013-4372.

[13] Koko, F., *Aerostructural and Trajectory Optimization of Morphing Wingtip Devices*, Master's thesis, Delft University of Technology, Oct. 2011.

[14] Martins, J. R. R. A., Sturdza, P., and Alonso, J. J., "The Complex-Step Derivative Approximation," *ACM Transactions on Mathematical Software*, Vol. 29, No. 3, 2003, pp. 245–262. doi:10.1145/838250.838251.

[15] Lissys, "PIANO User's Manual," 2010.

[16] Alonso, J. J., "Stanford University Aerospace Vehicle Environment," Jan 2014.

[17] Gray, J., Moore, K. T., and Naylor, B. A., "OpenMDAO: An Open Source Framework for Multidisciplinary Analysis and Optimization," *Proceedings of the 13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*, Fort Worth, TX, Sept. 2010, AIAA 2010-9101.

[18] Betts, J. T. and Cramer, E. J., "Application of Direct Transcription to Commercial Aircraft Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 1, 1995, pp. 151–159.

[19] Betts, J. T., "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 2, 1998, pp. 193–207.

[20] Hwang, J. T., Lee, D. Y., Cutler, J. W., and Martins, J. R. R. A., "Large-Scale Multidisciplinary Optimization of a Small Satellite's Design and Operation," *Journal of Spacecraft and Rockets*, 2014. doi:10.2514/1.A32751, (In press).

[21] Kennedy, G. J. and Martins, J. R. R. A., "A parallel aerostructural optimization framework for aircraft design studies," *Structural and Multidisciplinary Optimization*, 2014, (In press).

[22] Liem, R. P. and Martins, J. R. R. A., "Surrogate Models and Mixtures of Experts in Aerodynamic Performance Prediction for Mission Analysis," *15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2014.

[23] Martins, J. R. R. A. and Hwang, J. T., "Review and Unification of Methods for Computing Derivatives of Multidisciplinary Computational Models," *AIAA Journal*, Vol. 51, No. 11, November 2013, pp. 2582–2599. doi:10.2514/1.J052184.

[24] Alonso, J. J., "An Introduction to the Stanford University Aerospace Vehicle Environment (SUave)," FAA External Tools Meeting / Telecon, 2014.