10.2514/6.2016-1662

AIAA SciTech Forum
4-8 January 2016, San Diego, California, USA
57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference

Allocation-mission-design optimization of next-generation aircraft using a parallel computational framework

John T. Hwang [*] and Joaquim R. R. A. Martins [†]

Department of Aerospace Engineering, University of Michigan, Ann Arbor, Michigan, 48109, United States

Typically, computational design optimization of commercial aircraft is performed considering a small number of representative operating conditions. These conditions are based on the types of missions and the Mach number, altitude, and other operational profiles for which the aircraft will be flown. However, the design also influences which routes and mission parameters are optimal, so there is coupling that is ignored when using this approach. Here, we aim to simultaneously optimize the aircraft design, mission profiles, and the allocation of aircraft to routes in an airline network. To enable this, we use a gradient-based optimization approach with a parallel computational framework that facilitates the computation of derivatives and the multidisciplinary analysis. We use a surrogate model for the CFD analysis that is re-trained in each optimization iteration given the new set of shape design variables. The resulting optimization problem contains over 6,000 design variables and 23,000 constraints, and we solve it in roughly 10 hours on 128 processors. The optimization results show a 27% increase in airline profit when comparing the allocation-mission-design optimization to allocation-only optimization. The resulting aircraft shape design differs from a conventional multipoint design optimization, which confirms the need for the new approach. Its significance is that it produces aircraft designs that are optimal with respect to the airline-level profit, and the algorithm can quantify in these terms the potential benefits of next-generation aircraft.

I. Introduction

In the commercial aviation industry, there is significant interest in next-generation aircraft configurations because of aggressive goals for fuel burn, emissions, and noise reduction [1]. One of the challenges associated with the introduction of unconventional configurations is that the lack of prior experience and data on these new designs necessitates the use of higher-fidelity modeling tools early in the design process. For instance, the wing of a conventional tube-and-wing design is often modeled using a simple panel code for the aerodynamics and empirical relations for estimating structural weight as a function of parameters such as span and sweep. However, a concept such as the truss-braced wing requires 3-D computational fluid dynamics (CFD) to capture the interference drag where the struts, wing, and fuselage intersect [2], and a detailed structural sizing using finite element analysis (FEA) must be carried out to estimate the wing weight because historical data is simply not available.

Unfortunately, the comparison between unconventional configurations, and the high-level design of the layout of these new concepts are carried out during conceptual design, where high-fidelity simulations such as CFD are usually not suitable. The conceptual design stage is characterized by an iterative process that alternates between requirement specification and actual design [3]. The requirements define, among other things, a design mission that includes the range, cruise Mach number, and payload. From these requirements, overall sizing is carried out and refined analysis provides feedback with which to update the requirements based on the customer's true needs and the perceived difficulty of satisfying the current requirements. Because this process is iterative, it is necessary that the computational models used for sizing and analysis at this stage be fast and robust, which is usually not the case for methods such as CFD and FEA.

As a result, design optimization studies carried out using CFD and FEA are not designed to be part of such an iterative process. Rather, they are typically run assuming a fixed set of requirements. For instance, aerodynamic shape optimization using CFD might be run at a fixed, pre-determined set of lift coefficients that represent typical values of similarly-sized aircraft during actual operation [4]. Similarly, aerostructural optimization that couples CFD and FEA designs the aerodynamic shape and structural sizing simultaneously, while considering fuel burn for a limited number of flight conditions [5, 6].

[*] Postdoctoral Research Fellow, AIAA Member
[†] Professor, AIAA Associate Fellow

Herein lies the high-level motivation for this paper—we aim to widen the scope of the high-fidelity optimization to include some of the requirement-definition part of conceptual design. Instead of optimizing the aircraft design for a set of representative missions—or a design range, design Mach number, and payload—we let the algorithm choose on which routes in a network it is optimal to fly this next-generation aircraft, and which cruise Mach numbers are optimal on those routes. Thus, we view the problem from the airline's perspective, since the airlines are the customers who define the requirements in the conceptual-design sense.

In this new formulation, the objective function is the profit margin for the airline. We assume an airline network—a set of routes that the airline operates—as well as a finite-sized fleet of existing and next-generation aircraft that the airline can deploy on those routes. The optimizer chooses how to allocate the fleet of aircraft, selects the optimal mission profile for each aircraft flying on each route, and designs the next-generation aircraft, all at the same time. Thus, we call this formulation *allocation-mission-design optimization*, since all three are designed or chosen simultaneously within a single optimization problem. In this approach, the effective design range and payload for the next-generation aircraft is implicitly determined through optimization. The optimizer has the ability to design a smaller aircraft that is particularly efficient for short-range missions, or it could size the aircraft for long-range missions and cover the short-range routes for which it is over-designed using existing aircraft.

This paper describes the first implementation and solution of the aforementioned allocation-mission-design optimization problem. On the design side, we use 3-D Euler analysis of a swept, linearly-tapered wing with its twist distribution and shape parametrized as design variables. On the mission side, we optimize the shape of the altitude profiles for each mission, and we also make the cruise Mach number for each mission a design variable with a linear Mach number variation during climb and descent. On the allocation side, we consider a hypothetical 128-route network with fixed passenger demand on each route and a fleet consisting of 4 existing types of aircraft in addition to the simultaneous design of the next-generation aircraft.

There are several benefits to the allocation-mission-design optimization (AMD) approach compared to the traditional approach that optimizes or considers each one separately. These benefits result from the fact that the AMD approach uses a formulation that is much closer to the 'true' design problem because it captures the coupling among the airline allocation, mission analysis, and design. There are at least three sources of coupling between these disciplines. The first is due to the fact that changing the aircraft design can change on which routes it is optimal to operate the aircraft, and the route of course affects the optimal design. Thus, there is coupling between the allocation and design that is captured by the AMD approach. Another source of coupling exists between the allocation and mission disciplines, and it relates to the Mach number. A higher Mach number decreases the aerodynamic performance, thus increasing fuel burn, but it also reduces operating costs, enables the operation of potentially more flights, and increases passenger demand. A third source of coupling is due to next-generation technologies such as morphing wings [7, 8, 9] and continuous descent approaches [10, 11]. Given an aircraft design, it is possible to find an optimal descent profile or an optimal scheduling for wing morphing; however, the use of these technologies also impacts which aircraft design is optimal, so the mission and design are coupled as well.

To make the initial implementation of the AMD approach tractable, we limit the scope of this paper by fixing the aircraft sizing, which allows us to avoid including many additional disciplines and models. Fixing the sizing negates some of the benefits of AMD, especially the ability to exploit the coupling between allocation and design. However, we view this paper as a stepping stone towards future work that will incorporate the necessary detail and models to widen the scope of this problem to include all of the aforementioned benefits.

The remainder of the paper is organized as follows. First, we outline the overall approach by describing the computational challenges and their respective solutions. Second, we list and describe the various computational tools used in the AMD optimization. Next, we summarize the optimization problem, the wing geometry, and allocation problem. Finally, we present the optimization results and conclude with a summary and discussion of expected significance.

II. Overall approach

In this section, we present the overall approach by listing the computational challenges and the corresponding measures taken to address each. The challenges are the large number of design variables, the complexity of coupling many computational models from multiple disciplines, the large number of aerodynamic simulations that must be run, and the computational cost of the aerodynamic and mission analyses.

A. Large number of variables—adjoint-based optimization

The AMD optimization has design variables contributed from each discipline. The allocation problem parametrizes which aircraft is selected to run on each route by including as a design variable the number of flights per day for each route and for each type of aircraft. Since the network we consider in Sec. V has 128 routes, the allocation problem alone contributes hundreds of design variables. In the mission discipline, the continuous altitude profile is optimized for each route and each type of aircraft, and the curves are parametrized using between 10 and 50 B-spline control points, yielding thousands of design variables in total. The design problem also contributes $\mathcal{O}(10)$ to $\mathcal{O}(100)$ design variables because we are performing high-fidelity shape design optimization.

With thousands of design variables modeled using expensive simulations, the only feasible approach is to use gradient-based optimization. Moreover, it is necessary to use the adjoint method to compute the gradients for two reasons [12]. First, the accuracy of the analytic derivatives helps minimize the number of optimization iterations required by enabling the optimizer to find good search directions, and it also helps to achieve a high level of convergence. Tight convergence makes the algorithm more robust with respect to scaling issues among design variables and constraint functions—with poor scaling, a tighter overall convergence is required to ensure all the design variables are sufficiently converged. Otherwise, more manual effort is required *a priori* to ensure the variables are well-scaled relative to each other. Second, the adjoint method allows us to avoid the explosion of gradient computation time that would otherwise exist in a problem with such a large number of design variables.

B. Large number of disciplines—the MAUD architecture

The adjoint method can be difficult and time-consuming to implement in many situations, but this is exacerbated in a problem involving a large number of disciplines—i.e., a large number of heterogeneous computational models. Furthermore, in a problem such as the AMD optimization, there is a combination of computational models that internally solve a nonlinear system of equations (which we will call *implicit models*) and those that simply evaluate explicit functions (which we will call *explicit models*). Implicit models are the ones that are typically associated with the adjoint method, and there is a method equivalent to the adjoint method for multidisciplinary problems with only explicit models. However, for the general case involving both implicit and explicit models, neither can be directly applied.

The general case is addressed by the *modular analysis and unified derivatives* (MAUD) architecture [13]. MAUD does this by formulating the multidisciplinary computational model as a single system of nonlinear equations. We define a single vector u as the concatenation of all the variables in the computational model, whether its type is input, output, design, state, objective, constraint, and so on. We then define a residual for each variable, the form of which depends on the variable's type. If it is an input variable x whose value at the current optimization iteration is x^*, the residual is defined as $R_x = x - x^*$; if it is an implicitly defined state variable y, we simply take the residual R_y that is already defined; and if it is an explicitly computed output variable $z = f(u_1, \ldots, u_n)$, the residual is defined as $R_z = z - f(u_1, \ldots, u_n)$. If we concatenate all the residuals into a single residual function R, we obtain a nonlinear system $R(u) = 0$ that, once solved, represents the successful convergence and evaluation of our multidisciplinary computational model, accounting for all the coupled dependencies. This formulation leads to a unified equation for computing derivatives [12], given by

$$\frac{\partial R}{\partial u}\frac{\mathrm{d}u}{\mathrm{d}r} = \mathcal{I} = \frac{\partial R}{\partial u}^T \frac{\mathrm{d}u}{\mathrm{d}r}^T , \qquad (1)$$

which generalizes both the adjoint method and the related method for groups of explicit models that was mentioned previously. Equation (1). can handle any combination of implicit and explicit models, and in fact, it unifies all existing methods for computing derivatives—i.e., the chain rule, algorithmic differentiation, and the adjoint method can all be derived from this equation. Figure 1 shows how the adjoint method can be derived from Eq. (1) [12].

Using this equation as a foundation, the MAUD architecture enables a matrix-free computational framework that can semi-automatically compute derivatives using the adjoint method if the user provides partial derivatives for each component. MAUD allows the integration and coupling of computational models of multiple disciplines in an efficient way that does not introduce significant framework overhead in computational time or memory. This is a key enabling aspect of the AMD optimization we present in this paper—the allocation, mission, and aerodynamic models are all implemented as a collection of components that follow the same template and interface. The tasks of handling the complex network of data connections between these components, solving the coupled nonlinear and linear systems of equations, and many other smaller details are automated centrally by a computational framework implementation that uses the MAUD architecture. This computational framework implementation of MAUD is built on top of the Portable, Extensible Toolkit for Scientific Computation (PETSc) [14].

The MAUD architecture has been adopted as part of the core of NASA's OpenMDAO framework [15]. OpenMDAO is an open-source computational framework designed to facilitate the solution of multidisciplinary optimization problems. Since it uses MAUD, OpenMDAO provides coupled adjoint-based derivative computation, as one of its key features. The mission analysis and allocation models used in this paper have been implemented in OpenMDAO, and other areas in which OpenMDAO has been applied include wind turbine design [16] and the design of the Hyperloop concept [17].

$$\begin{bmatrix} \mathcal{I} & 0 & 0 \\ -\dfrac{\partial R}{\partial x} & -\dfrac{\partial R}{\partial y} & 0 \\ -\dfrac{\partial F}{\partial x} & -\dfrac{\partial F}{\partial y} & \mathcal{I} \end{bmatrix} \begin{bmatrix} \mathcal{I} & 0 & 0 \\ \dfrac{\mathrm{d}y}{\mathrm{d}x} & \dfrac{\mathrm{d}y}{\mathrm{d}r} & 0 \\ \dfrac{\mathrm{d}f}{\mathrm{d}x} & \dfrac{\mathrm{d}f}{\mathrm{d}r} & \mathcal{I} \end{bmatrix} = \begin{bmatrix} \mathcal{I} & 0 & 0 \\ 0 & \mathcal{I} & 0 \\ 0 & 0 & \mathcal{I} \end{bmatrix}$$

Figure 1: The expansion of Eq. (1) for the direct or adjoint methods. In this case, u breaks down into input variables x, state variables y, and output variables f [13].

C. Large number of aerodynamic evaluations—dynamically-trained surrogate model

The solution of the AMD optimization necessarily involves millions of evaluations of the aerodynamic model. Each mission profile is discretized using $\mathcal{O}(100)$ points, but the coupling between the fuel weight differential equation and the aircraft equations of motion introduces at least another factor of $\mathcal{O}(10)$ because the aerodynamic model must be evaluated at each point within each iteration. In this problem, there are $\mathcal{O}(100)$ missions to evaluate, and the optimizer requires another $\mathcal{O}(100)$ iterations to converge.

Since performing tens of millions of evaluations of a 3-D CFD code is infeasible, we use a surrogate model as done by Liem et al. [18] However, the inputs to the surrogate model cannot include the $\mathcal{O}(10)$ to $\mathcal{O}(100)$ shape design variables because a space with that many dimensions would again require an inordinate number of CFD evaluations just to build the surrogate model. The solution is to re-train the surrogate model each optimization iteration so that the only inputs to the surrogate model are the aerodynamic operating conditions such as angle of attack α and Mach number M. A two-dimensional space with M and α can be populated with only $\mathcal{O}(10)$ points, with the downside that $\mathcal{O}(10)$ evaluations of the CFD model would be required at the start of each optimization iteration. Though this approach of using a dynamically-trained surrogate model is not inexpensive, it is tractable and certainly a significant improvement over the naive approach.

The AMD optimization problem data flow is shown in Fig. 2. At the start of each optimization iteration, the latest values of the twist and shape variables are passed to the geometry and mesh warping component. The updated mesh is passed to the CFD solver, which is run to compute the lift and drag coefficients at 16 points in M-α space. The results are assembled and given to the surrogate model as training data. During the 128 mission analyses, this aerodynamic surrogate model is now called for each point in the climb, cruise, or descent segments in the mission multiple times to converge the coupled mission equations, where the surrogate model provides the lift and drag coefficients as functions of Mach number and angle of attack. The fuel burn and block time information computed by the mission analyses are given to the allocation component to compute airline profit and the allocation constraint values.

The interpolation method using for the aerodynamic surrogate model is an energy-minimizing tensor-product spline interpolant. It can smoothly interpolate unstructured data in an arbitrary number of dimensions, though it is best suited for low-dimensional problems such as this one. Because it minimizes energy, it has the favorable property that it limits spurious oscillations, even with data sets that are not evenly spaced or arranged in a grid.

D. Large computational cost of the models—parallelization

Though the solutions to the previous challenges reduce the computational cost, the fact remains that the CFD evaluations and the coupled mission analyses are costly to run because they solve large systems of equations. The last aspect to the approach for solving the AMD optimization problem is to use distributed-memory parallelization at multiple levels. Within an optimization iteration, the first step is to evaluate the CFD model to generate the training points for the surrogate model (16 in this case). The 128-route AMD problem is run with 128 processors, so the MAUD framework automatically splits the 128 processors by 16, assigning 8 processors to each CFD evaluation. The 128 processors broadcast the data at the 16 training points to each other and re-combine to update the surrogate model globally. They are then split up once again, one processor per mission analysis, before re-combining to evaluate the

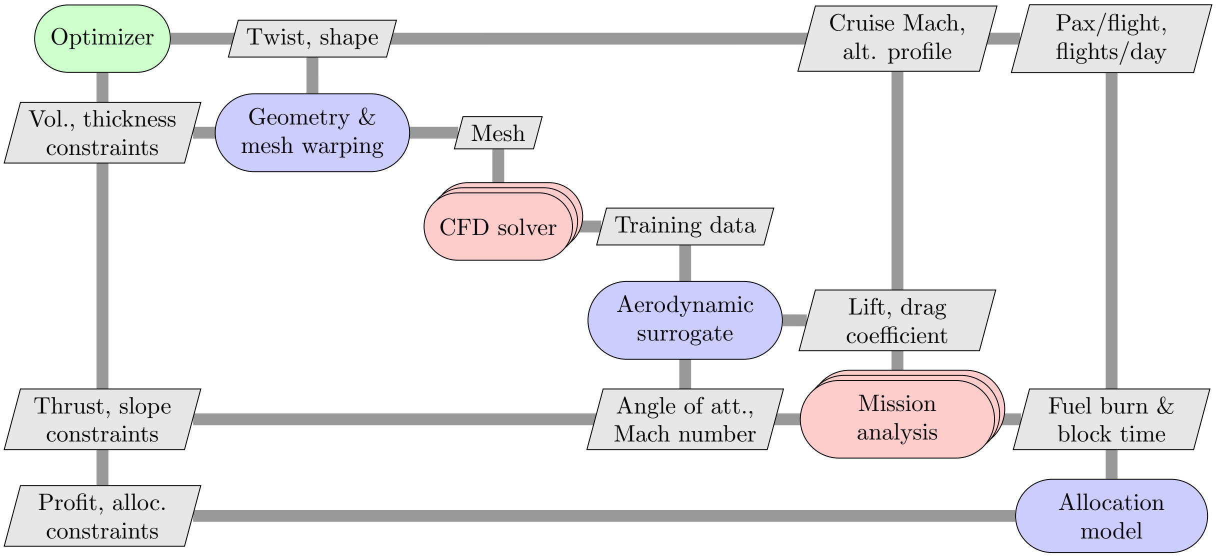

Figure 2: Extended design structure matrix (XDSM) diagram of the data flow in the allocation-mission-design optimization.

allocation model and pass the outputs to the optimizer.

This parallelization is fully automated in the MAUD framework. MAUD allows us to build the AMD problem in a hierarchical fashion where, for instance, each of the components in a mission analysis assembly are grouped together, then all of the 128 mission assemblies are grouped together in parallel, and the collective assembly of missions is grouped in sequence with the allocation, CFD, and surrogate modeling assemblies. MAUD also adjusts accordingly if the problem is run with a different number of processors or a different number of missions. The automatic handling of parallelism is an enabling feature that eliminates a large amount of time-consuming and error-prone development.

Another detail relevant to the parallelization is the fact that adjoints are computed in parallel. For outputs involving all processors, such as the profit objective, the parallel solution of the linear system for the adjoint method is a natural extension of the parallel solution of the nonlinear system for evaluating the multidisciplinary computational model. However, for outputs that do not involve all processors, such as the individual mission constraints, a naive implementation of Eq. (1) would compute the adjoint vector for each mission constraint in series, one right-hand side at a time. In this case, all processors would always be idling except the one processor that owns the mission analysis of the current constraint. As Fig. 3 illustrates, this issue is resolved by solving groups of multiple right-hand sides simultaneously [19]. By applying the linear block-Jacobi solver across the mission analyses, each processor would solve the one non-zero right-hand side it has in its group, all simultaneously, rather than waiting for the previous mission analysis to complete its adjoint solution.

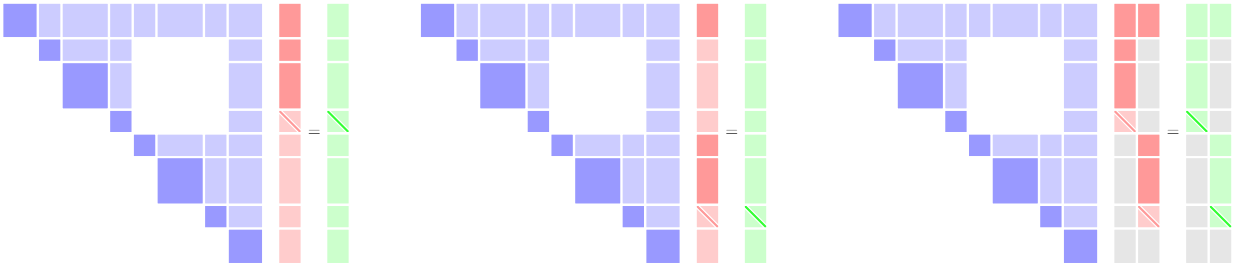

Figure 3: The adjoint system from Eq. (1) for a notional two-mission problem where a constraint from the first mission is solved (left), a constraint from the second mission is solved (center), or both constraints are solved simultaneously (right) using a strategy that solves the linear equation with multiple right-hand sides simultaneously [19].

III. Software overview

The software components that are integrated to solve the AMD optimization problem include: an unstructured CFD mesh deformation algorithm, a CFD solver, a mission analysis model, an aircraft allocation model, and an optimizer.

A. CFD mesh deformation

The CFD mesh deformation algorithm we use propagates the displacements and rotations from the deformed surface to the full CFD volume mesh by using inverse-distance weights [20]. More specifically, the deformation of each node in the CFD volume mesh is the sum of the displacements and rotations of all the nodes on the deformed surface, weighted by the inverse of the distance to that surface node. In numerical experiments, this approach has proven to be a robust strategy for CFD mesh deformation in both aerodynamic and aerostructural optimization problems [7], though we consider only aerodynamic optimization in this paper.

B. CFD solver

The CFD solver is SUMad, which is a structured multi-block finite-volume solver with multigrid [21]. SUMad was originally developed at Stanford University and an adjoint using reverse-mode automatic differentiation was developed at the University of Michigan [22]. SUMad solves the Reynolds-averaged Navier–Stokes (RANS) equations or the Euler equations in parallel using the fourth-order Runge–Kutta scheme or the diagonally dominant alternating direction implicit (DDADI) scheme, combined with the Newton–Krylov method. In this paper, we solve the Euler equations using SUMad to reduce computation time.

C. Mission analysis

The mission analysis tool [23] discretizes the full mission profile using a collocation approach and solves the vertical and horizontal equilibrium equations, the trim condition, and the ordinary differential equation for fuel weight. The equations are all coupled to the aircraft performance model, which is represented using the aforementioned surrogate model for lift and drag coefficients as functions of Mach number and angle of attack. The mission analysis takes the altitude and Mach number profiles as inputs, parametrized using B-splines, and its outputs are fuel burn, block time, and aggregated values for idle and maximum thrust constraints.

There are two approaches used for mission profile optimization: direct and indirect [24]. The direct approach applies the optimality conditions after discretizing the equilibrium conditions, while the indirect approach differentiates the equilibriums first and then discretizes. The mission analysis tool takes the direct approach as it is more amenable to coupling with other disciplines, and the implementation is similar to that of Betts and Cramer [24], which also uses gradient-based optimization.

One of the advantages of optimizing the altitude profiles is that we are able to consider continuous descent approach (CDA), also known as optimized descent profile (ODP). ODP is part of the FAA's next generation air transportation system (NextGen), currently in development to achieve reductions in noise and fuel burn. To this end, continuous descent tests have been conducted recently at several airports [10, 11, 25, 26].

Figure 4 lists all the variables in the mission analysis, including the shapes of the Jacobians in each connection between variables. The variables are organized into five groups, also known as *assemblies*, both for convenience and to take advantage of hierarchical solution algorithms in the MAUD framework—i.e., we use nonlinear block Gauss–Seidel across the 5 groups and within all but the *CoupledAnalysis* assembly, which internally uses a Newton solver to resolve the coupling.

The inputs are the B-spline control points for the altitudes and their corresponding horizontal positions. The resolution depends on the mission range, but typically we use between 10 and 50 control points and 100 to 250 discretization points. The *Bsplines* assembly then computes the actual values of the altitudes, horizontal coordinates, and slopes. The *AtmosphericProperties* assembly parametrizes the Mach number, temperature, pressure, density, and airspeed in terms of the altitude. The thrust-specific fuel consumption is also a simple function of altitude, but more elaborate models can be used in future work.

The *CoupledAnalysis* assembly contains the core of the mission analysis model. Given an initial guess for the fuel weight at every point in the mission, it applies the vertical equilibrium equation to solve for the required lift coefficient using

$$C_L = \frac{W \cos\gamma}{\frac{1}{2}\rho v^2 S} - C_T \sin\alpha. \tag{2}$$

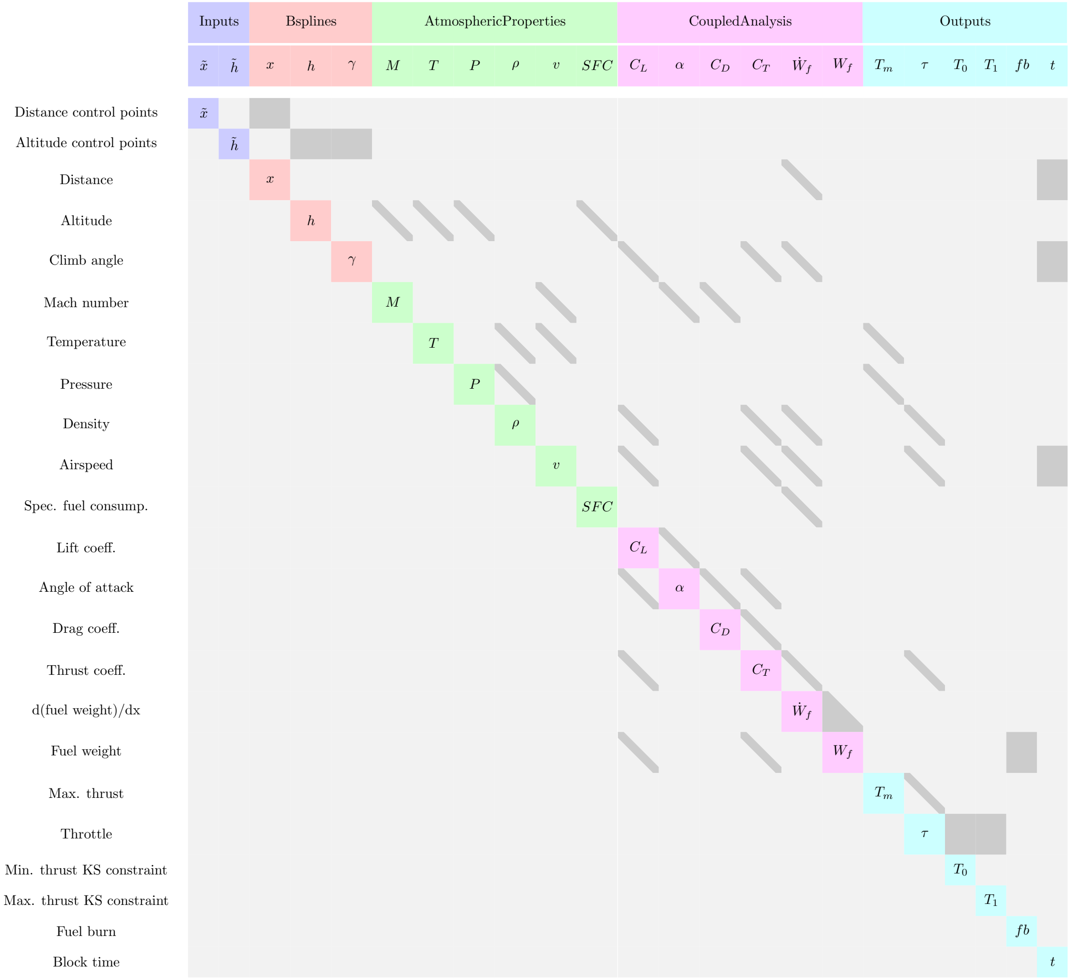

Figure 4: Dependency graph for all the variables in a single mission analysis.

Next, the surrogate model for C_L is used to solve an implicit function to determine the angle of attack that produces this target C_L with the residual,

$$\tilde{C}_L(\alpha, M) - C_L = 0, \tag{3}$$

where $\tilde{C}_L$ is the surrogate model. The surrogate model for drag coefficient, $\tilde{C}_D$, is then evaluated using the computed angle of attack to determine the resulting drag coefficient from

$$C_D = \tilde{C}_D(\alpha, M). \tag{4}$$

Next, the horizontal equilibrium equation is applied to compute the thrust coefficient from

$$C_T = \frac{C_D}{\cos\alpha} + \frac{W \sin\gamma}{\frac{1}{2}\rho v^2 S \cos\alpha}. \tag{5}$$

Finally, the ordinary differential equation for the fuel weight is given by

$$\dot{W}_f = SFC \frac{C_T \frac{1}{2}\rho v^2 S}{v \cos\gamma}. \tag{6}$$

This must be integrated backwards from the last mission point which has a small amount of reserve fuel, and this produces a new estimate of aircraft weight that requires us to iterate back to the first step of the *CoupledAnalysis* assembly.

The *Outputs* component performs a series of postprocessing computations, including the maximum thrust estimate at altitude, given by

$$T_m = T_{m,SL} \frac{P}{P_{SL}} \sqrt{\frac{T_{SL}}{T}}, \tag{7}$$

where the maximum thrust is expressed in terms of pressure and temperature, and the respective sea-level values. There is no guarantee that the engine was appropriately sized to fly the mission as evaluated by the *CoupledAnalysis* assembly; thus, we define nonlinear constraints that the throttle setting is between idle and maximum, and let the optimizer make sure this is satisfied. However, these thrust constraints must be satisfied at every point in the mission, so we aggregate them using the Kreisselmeier–Steinhauser function [27, 28]. The two outputs of the mission analysis that are of interest in the allocation problem are the fuel burn and block time, which are computed at the end.

D. Allocation model

The allocation problem is formulated with two sets of design variables: the number of flights per day flown by a type of aircraft on a given route, and the number of passengers per flight for a given type of aircraft and a given route. This formulation was first used as an application of linear programming in 1956 [29]. Here, the formulation maximizes profit subject the operational and demand constraints [30, 31]. Govindaraju and Crossley [32] also solve this allocation problem simultaneously while optimizing the aircraft design; however, the key difference in this work is that we use CFD analysis, which enables us to perform high-fidelity shape optimization.

The airline profit is evaluated using estimates for ticket price and operating costs, using

$$\begin{aligned} \text{profit} = & \sum_{i}^{n_{rt}} \sum_{j}^{n_{ac}} \left[\text{price\_pax}_{i,j} \cdot \text{pax\_flt}_{i,j} \cdot \text{flt\_day}_{i,j}\right] \\ & - \sum_{i}^{n_{rt}} \sum_{j}^{n_{ac}} \left[(\text{cost\_flt}_{i,j} + \text{cost\_fuel} \cdot \text{fuel\_flt}_{i,j}) \cdot \text{flt\_day}_{i,j}\right], \end{aligned} \tag{8}$$

where $\text{price\_pax}_{i,j}$ is the ticket price per flight, $\text{cost\_flt}_{i,j}$ is the total cost of operating a flight minus fuel, cost_fuel is the cost per unit fuel, and fuel_flt is the total fuel burn on a flight.

The allocation problem contributes two inequality constraints to the allocation-mission optimization. The first ensures that the total number of people that fly on a given route on a given day is less than the total demand for that route, and it is given by

$$\text{total\_pax}_i = \sum_{j}^{n_{ac}} \left[\text{pax\_flt}_{i,j} \cdot \text{flt\_dat}_{i,j}\right] \le \text{demand}_i\ , \quad 1 \le i \le n_{rt}. \tag{9}$$

The second inequality constraint takes into consideration how many aircraft of a given type are actually owned by the airline, and it is given by

$$\text{total\_usage}_j = \sum_{i}^{n_{rt}} \left[\text{flt\_day}_{i,j} \cdot (\text{time\_flt}_{i,j}(1 + \text{maint}_j) + \text{turn\_flt})\right] \leq 12\text{hr} \cdot \text{num\_ac}_j \ , \quad 1 \leq j \leq n_{ac}, \tag{10}$$

where time_flt$_{i,j}$ is the block time for a flight, maint$_j$ is the maintenance time required as a multiple of block time, turn_flt is the turnaround time between flights, and num_ac$_j$ is the number of aircraft available for a given type.

E. Optimizer

The gradient-based optimizer we use in this paper is SNOPT [33], which is an active-set sequential quadratic programming (SQP) algorithm. SNOPT is designed to handle sparse nonlinear constrained optimization problems, so it is well-suited to the current application. In addition to the thousands of design variables, the AMD optimization also has tens of thousands of constraints, mostly from the 128 mission analysis, but the tens of thousands by thousands-sized Jacobian is sparse. This is efficiently handled by SNOPT because of its reduced-Hessian approach, and the fact that it uses a direct solver for its quadratic programming (QP) problems. In this paper, we interface to SNOPT in Python via a version of the pyOpt library [34], modified to handle sparse constraints.

IV. Problem description

In this section, we describe the wing geometry and parametrization, the Mach number parametrization, and the optimization problem.

A. Geometry parametrization

The geometry that we analyze in this paper is a swept, linearly-tapered wing that based on a scaled version of the Boeing 717 wing. Figure 5 shows the result of a CFD analysis of the wing using the Euler equations at a Mach number of 0.78 and an angle of attack of 3°. The mesh has roughly 100,000 cells with an O topology as shown in Fig. 5. Since we model only the wing, the additional sources of drag—fuselage drag, trim drag, viscous drag, and so on—are accounted for by subtracting the computed drag at a nominal condition from the known, expected drag for an aircraft of this size.

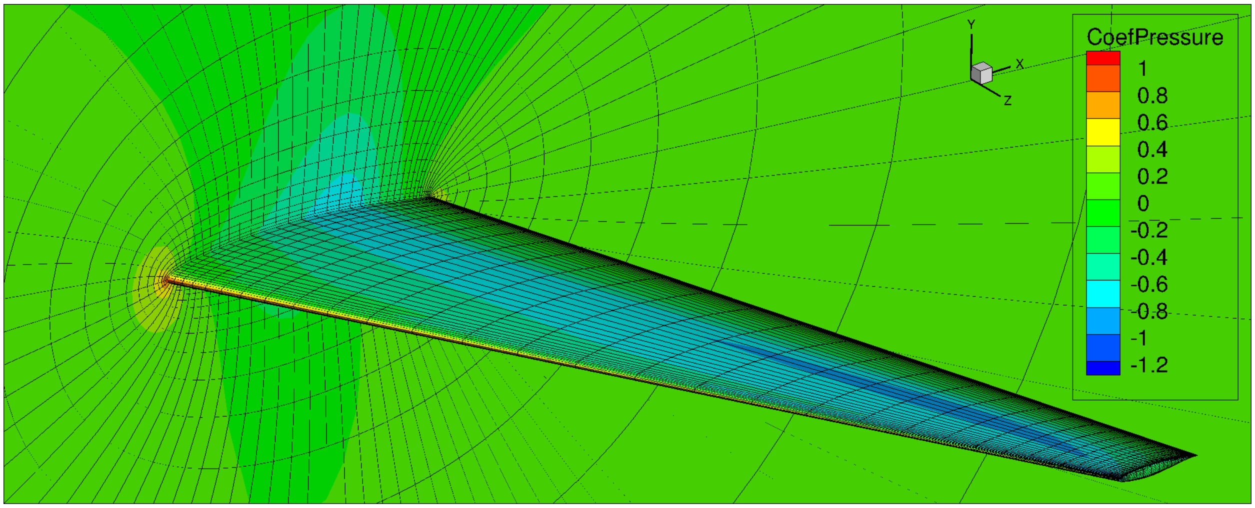

Figure 5: Euler analysis result of the geometry at M=0.78 and $\alpha = 0.3°$.

For the geometry parametrization, we use free-form deformation with 72 B-spline control points in total, as shown in Fig. 6 There are 6 sections with 6 control points on each of the upper and lower surfaces, and the control points define a 3-D tensor-product B-spline volume that morphs smoothly. Each of the 6 twist variables causes a non-shearing twist by rotating its group of 12 control points, and all 72 control points are free to move vertically; thus, there are 78 shape design variables in total. However, this 78-dimensional design space is narrowed by constraints on the wing volume, a 10×10 grid of thickness constraints, and linear constraints on the leading and trailing edge that eliminate redundant degrees of freedom—e.g., reproducing a twist-like shape change using only shape variables.

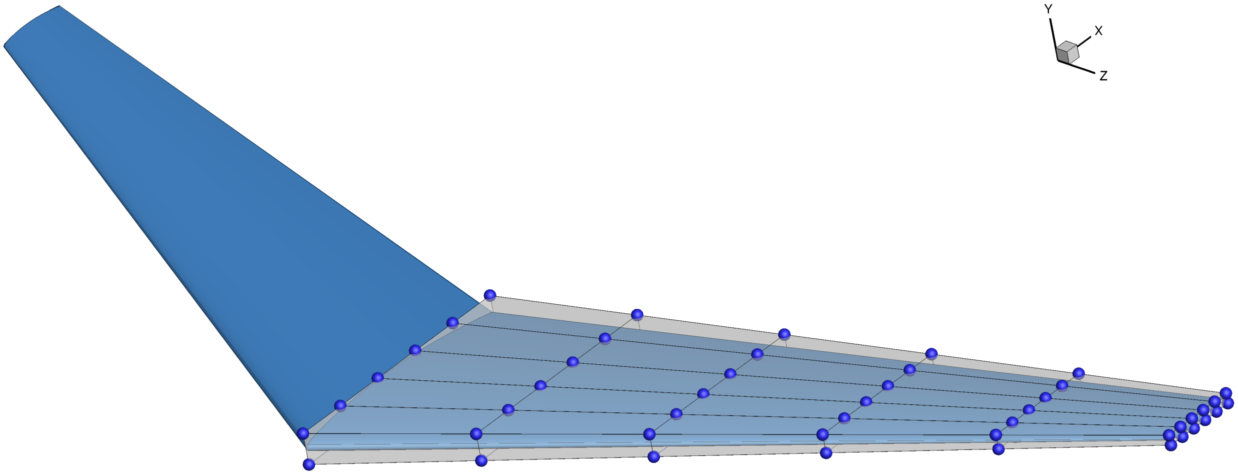

Figure 6: Geometry parametrization using free-form deformation with a tensor-product B-spline volume.

B. Cruise Mach number

Since we are analyzing the full mission profile, we need to discuss the parametrization of the cruise Mach number. The approach we adopt is based on the actual Mach number profiles that are flown. Pilots climb using the indicated airspeed (IAS) as the reference until the aircraft reaches the altitude at which the Mach number reaches the desired cruise Mach number. Here, we assume a constant IAS climb at 300 knots. This is shown in the 'true' curve in Fig. 7 with an assumed cruise Mach number of 0.78. For the AMD optimization, however, we use a piecewise linear approximation, labeled accordingly in Fig. 7, that is linear between a Mach number of 0.45 at sea level and the cruise Mach number at an altitude of 8 km.

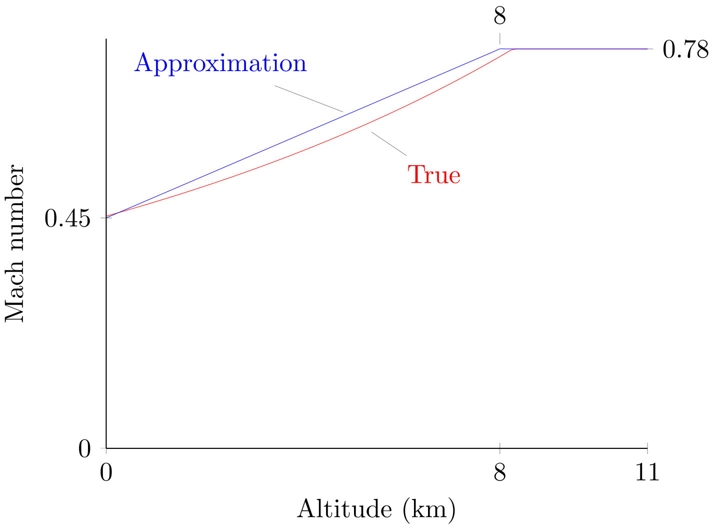

Figure 7: The true Mach number dependence on altitude and the linear approximation used in this paper for a cruise Mach number of 0.78.

C. Optimization problem

The full allocation-mission-design optimization problem is summarized in Table 1. As mentioned previously, there are 6 twist and 72 shape design variables with a minimum wing volume constraint and a 10×10 grid of points on the wing at which a minimum thickness is enforced. Each mission analysis contributes a cruise Mach number design variable and altitude control points, the number of which varies depending on the mission range. The mission analyses also contribute minimum and maximum thrust constraints aggregated using the Kreisselmeier–Steinhauser function and linear constraints on the maximum climb and descent angle. The allocation problem adds passengers per flight and flights per day design variables for each type of existing or new aircraft, for each of the 128 routes. There is also a demand constraint that limits the total number of passengers travelling on a route in a given day, and an aircraft utilization constraint that takes into account the finite number of aircraft of each type that the airline owns.

	Variable	Quantity
maximize	profit	
with respect to	twist	6
	shape	72
	cruise Mach number for each route (between 0.6 and 0.82)	1×128
	altitude control points for each route	4,575
	passengers per flight for each aircraft type and route	5×128
	flights per day for each aircraft type and route	5×128
	Total number of design variables	6,061
subject to	wing volume constraint	1
	wing thickness constraints	100
	idle thrust KS constraint for each route	128
	maximum thrust KS constraint for each route	128
	linear climb angle bounds for each mission	22,875
	demand constraint for each route	128
	total flight time constraint for each aircraft	5
	Total number of constraints	23,365

Table 1: The 128-route allocation-mission-design optimization problem.

D. Location of aerodynamic points

In Sec. V, we present not only the results of the AMD optimization, but also results from a multipoint design optimization to provide a baseline. Here, we discuss the locations of the aerodynamic analyses, in M-α space for the AMD optimization, and in M-C_L space for the design-only optimization.

The locations are shown in Fig. 8. For the AMD optimization, the Mach number range extends down to M=0.45 because of the climb and descent parts of the missions, as shown in Fig. 7. In fact, a Mach number below 0.45 is never required. At the upper end, we are constrained by CFD convergence failures because of which we do not include points above M=0.8. Moreover, cases with high M and high α are the most difficult to solve, which is why the corner point with M=0.8 and α=3° is omitted. The points are clustered at the high end of this range, and we keep remaining the corner points to ensure the surrogate model is accurate over the full range. For the design optimization, we use $C_L = 0.5$ and M=0.8 as the design condition and take the neighboring points in all directions except for the direction of higher M, again due to convergence issues.

Figure 9 shows lift and drag coefficient contours as a function of Mach number and angle of attack. These results can be used to verify that the Euler-based CFD analysis is converging at all the training points and is providing reasonable results, but the results can also indicate whether the surrogate model is performing well. At the very least, we observe that the energy-minimizing interpolant is able to avoid spurious oscillations despite the fact that the training points are not arranged in a grid and their spacing varies fairly significantly in different parts of the space.

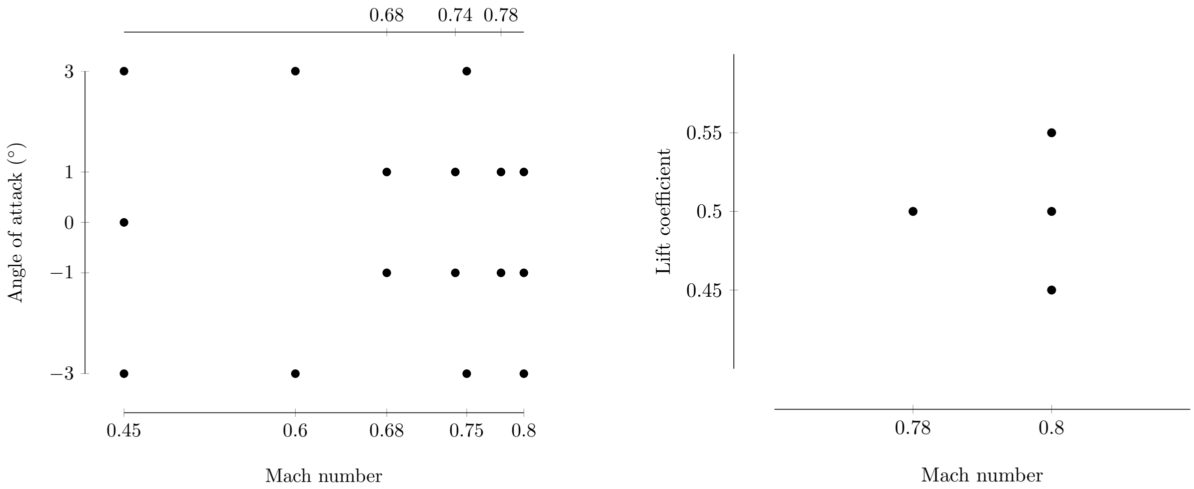

Figure 8: Locations of the aerodynamic training points for the AMD optimization (left) and the operating condition for the multipoint design-only optimization (right).

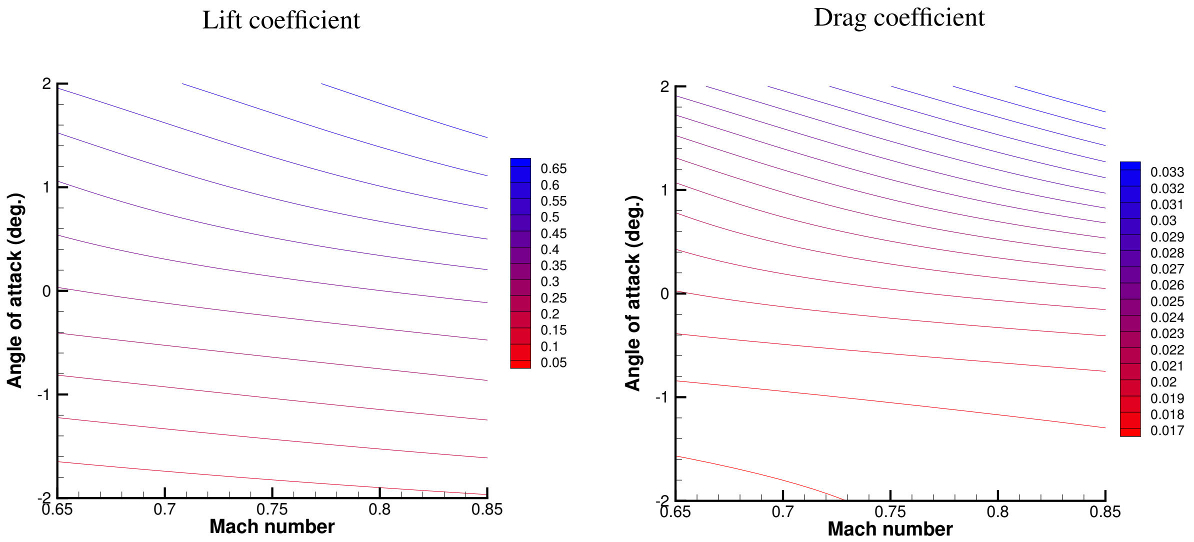

Figure 9: Output from the aerodynamic surrogate model generated from the multipoint design-optimized geometry.

V. Results

In this section, we present the AMD optimization results and compare them to the baseline design-only multipoint optimization where appropriate. We discuss the optimal values of the shape design variables, the optimal allocation variables, the optimal cruise Mach numbers, and the computational performance.

A. Optimal design

As we examine the optimized designs, the questions of interest are whether the AMD optimization result agrees with intuition and is reasonable, and whether the AMD optimization and design-only optimization are significantly different. First, we consider the twist and lift distributions, which are shown in Fig. 10. We observe that the twist distribution from the design-only optimization is smoother and less oscillatory. This is expected given that the design optimization is effectively a direct maximization of the lift-to-drag ratio albeit in a multipoint setting, while in the AMD optimization the output of the CFD analysis undergoes post-processing—first being interpolated by a surrogate model and then repeatedly called in the mission analyses to solve the coupled equations. Because of this, there is numerical noise downstream of the CFD analyses in the AMD optimization, and this likely contributes some error to the derivatives with respect to the shape and twist design variables, making it more difficult to converge to smooth distributions. In both the AMD and design optimization cases, the twist distribution is shown only for a single representative operating condition, but we observe that the lift distributions are close to elliptical in both cases.

Figure 11 shows slices of the wing at 6 span-wise locations that are approximately equally spaced. Beyond the differences in twist, the differences in shape between the AMD and design-only results are minor and irregular. We observe that they are both thicker overall and have thicker leading and trailing edges when compared to the initial design, though this result would likely be different if we were capturing viscous effects with the Reynolds-averaged Navier–Stokes (RANS) equations [35]. The AMD and design-only results are sufficiently different to imply that the coupling with allocation and mission analysis is not negligible, but further investigation is required to make a firm conclusion.

B. Optimal allocation

In this subsection, we look at the optimized values of the allocation variables, both from the allocation-only optimization and from the AMD optimization. Here, we look at the coupling between allocation and design from the perspective of the allocation problem—i.e., if we want to determine the optimal allocation, whether a flexible design significantly impacts the optimal allocation.

We observe a 27% increase in airline profit per day going from the allocation-only optimization to the AMD optimization. The result from the allocation-only optimization yields a daily profit of about \$23.4 million, and the AMD-optimized result has a profit of \$29.8 million, as shown in Fig. 12.

We can see in Fig. 13 how the allocation-only and AMD optimization results differ. The most noteworthy difference between the two is that the AMD optimization flies the next-generation aircraft on the shorter-range routes much more. We see more directly in Fig. 14 that the next-generation aircraft enjoys the largest share of the total number of passengers flown per day, but this share increases in AMD optimization, as one would expect since the design of the next-generation aircraft improves in the AMD optimization.

C. Optimal cruise Mach numbers

Figure 15 shows the cruise Mach numbers at the solution to the AMD optimization problem. The cruise Mach numbers are given a lower bound of 0.6 and an upper bound of 0.82, as marked by the dotted lines. We see that for short-range missions, many of the optimal Mach numbers are on the low end, which is expected especially for missions so short that the aircraft does not reach cruising altitude. We also see that the majority of the Mach numbers stay at the initial value of 0.78. This is the case for all the missions on which the next-generation aircraft is not flown—likely the majority of the 128 routes.

However, we see that for many routes, and all above a range of 2000 nm, the optimal cruise Mach number goes to the upper bound. The advantage of a higher Mach number is that the block time is lower, which allows the airline to get more utilization out of the aircraft each day. This advantage is likely overestimated since we relax the number

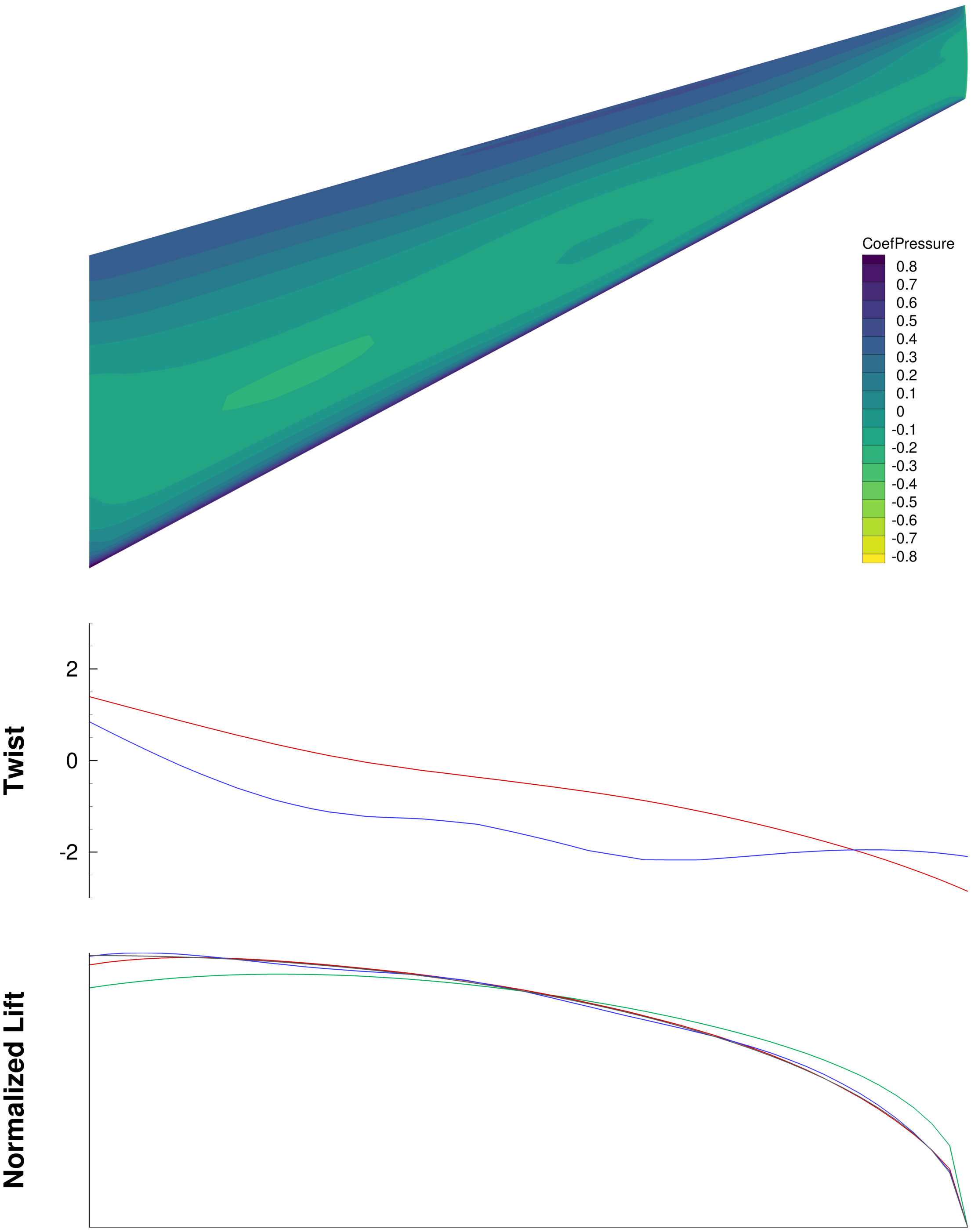

Figure 10: Planform view, optimal twist distribution, and optimized lift distribution. The AMD result is in blue, the design optimization result is in red, and green is the initial. The elliptical lift distribution is shown in gray.

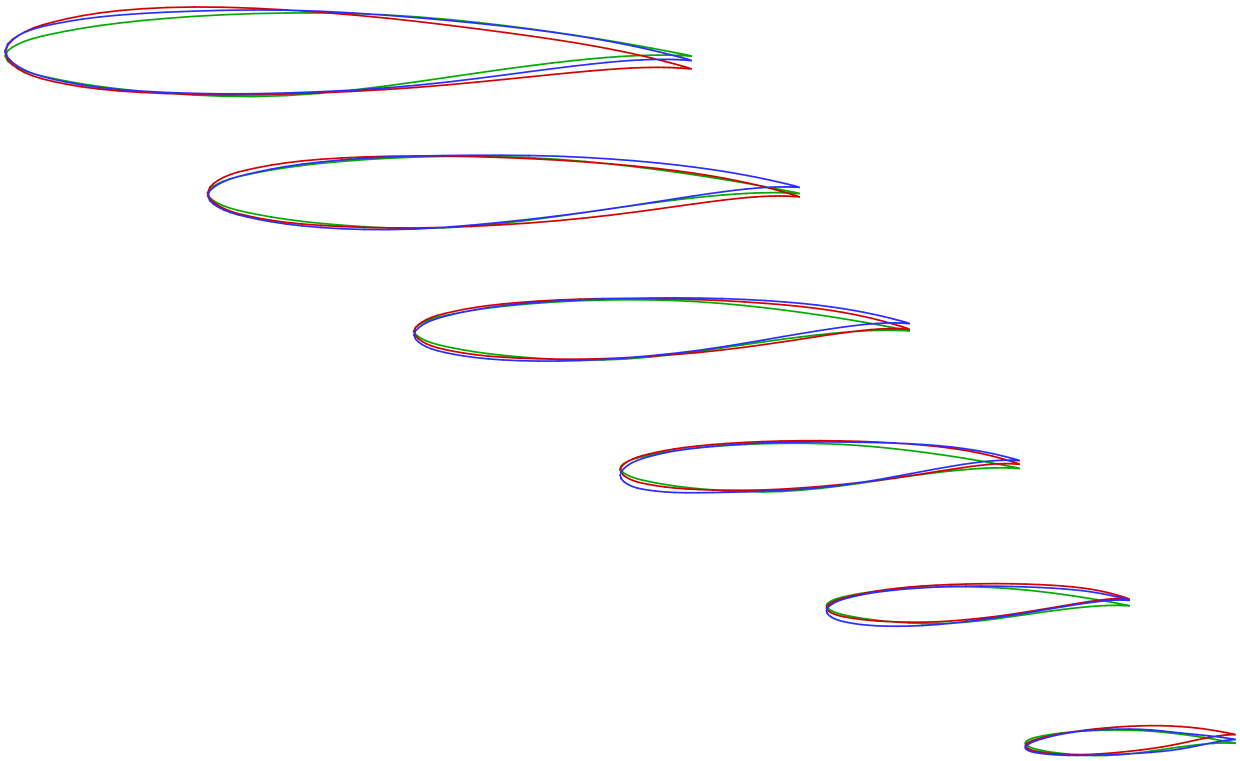

Figure 11: Airfoil shapes at 6 roughly uniformly spaced sections from the root to the tip of the wing. The AMD result is in blue, the design optimization result is in red, and green is the initial.

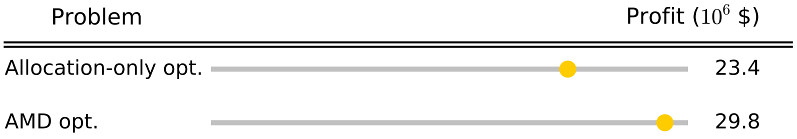

Figure 12: The AMD optimization produces 27% more profit than the allocation-only optimization.

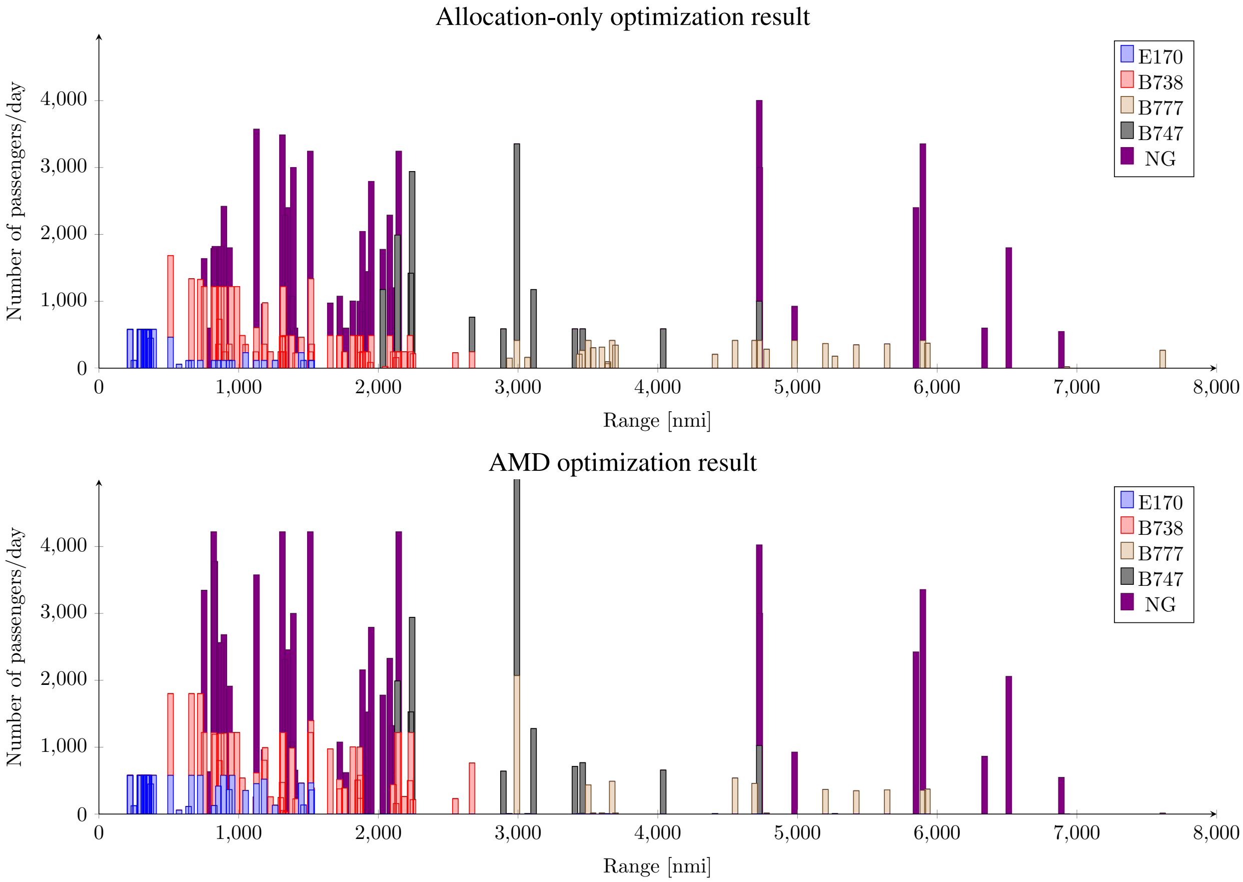

Figure 13: The total number of passengers flown per day for each of the 128 missions, shown according to their range.

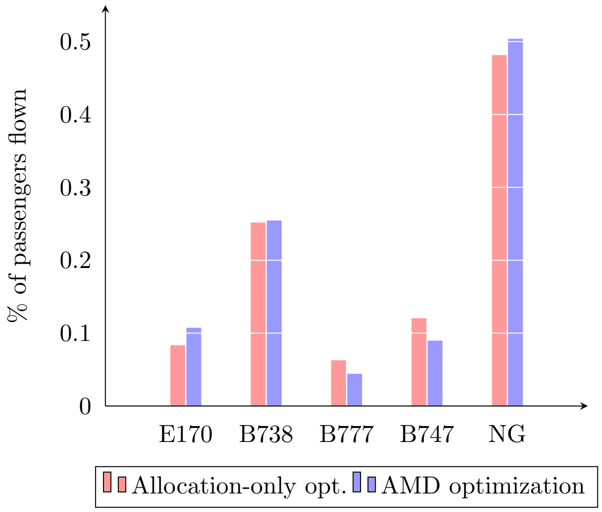

Figure 14: The percentage of the passengers flown on each type of aircraft.

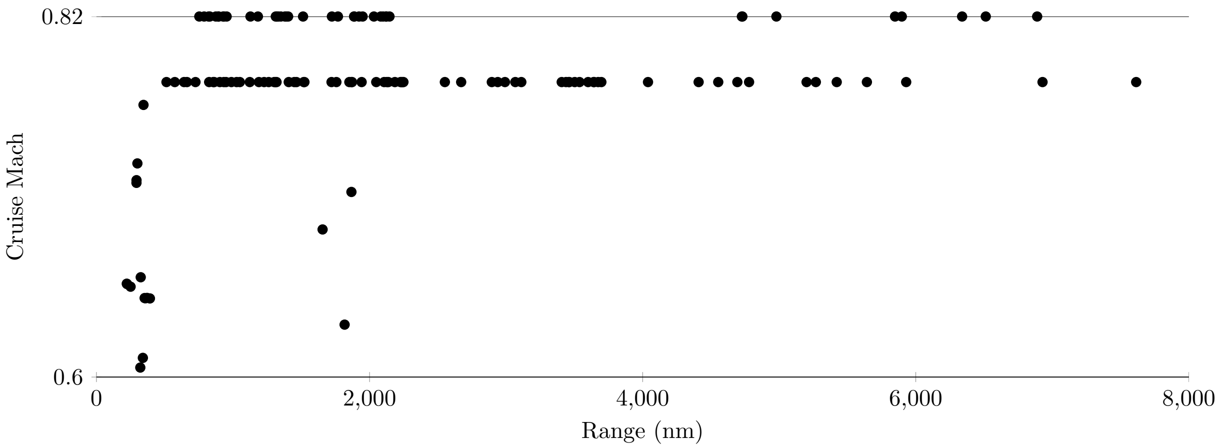

Figure 15: The distribution of optimal cruise Mach numbers found via AMD optimization.

of flights per day design variable and ignore the fact that it should be an integer variable. Moreover, the cruise Mach number is also likely to be higher because the transonic drag rise is not captured sufficiently accurately. This would be addressed partly by switching to a viscous CFD solver and also by including more training points for the surrogate model in this region of high Mach numbers.

D. Computational performance

The AMD optimization solution time is dominated by the CFD solver. The computation time per optimization iteration is highly irregular for three reasons. First, the CFD solver takes the most amount of time on the first iteration and is faster in all subsequent iterations because the design variable changes are typically small and the previous solution usually provides a good initial guess. Since we parallelize across the 16 points, each set of processors deals with only one flow condition—i.e., one Mach number and angle of attack combination—so the previous solution is always stored and available. Second, as is typical with high-fidelity shape optimization, the first 10–50 major iterations in the SQP optimization algorithm involves the largest design changes, and afterwards, the optimization accelerates significantly because each design variable change is small enough that only a few iterations are required to converge the flow equations. Finally, there is always a risk of convergence failure in the flow solver or mesh failures, which is part of the reason for a conservative selection of training points with no Mach numbers above 0.80.

The convergence history for the AMD optimization is shown in Fig. 16. The optimality, which is the norm of the gradient of the augmented Lagrangian, converges nearly two orders of magnitude, and feasibility is converged to within 10^{-5}. The merit function changes significantly in the first 5–10 iterations and is nearly converged by the 50th iteration. The number of superbasic variables—design variables neither at their bounds nor used to satisfy constraints—stays fairly consistently around 1600 after a steady increase initial because SNOPT limits the increase in the number of superbasic variables from iteration to iteration. Viewed together, all the convergence metrics suggest that the optimization problem converges within 50–100 iterations.

VI. Conclusion

In this paper, we presented an aircraft allocation-mission-design (AMD) optimization algorithm that maximizes an airline's profit by designing a next-generation aircraft and solving an allocation problem to decide how to allocate this and a fleet of existing aircraft to a 128-route network. The long-term motivation is that we want to perform design optimization while also accounting for next-generation operational technologies such as continuous descent approach, and to simultaneously determining the appropriate sizing, design mission, and optimal cruise Mach number. To make the current work tractable, however, we make the following high-level simplifications: we fix the aircraft sizing and layout, solve the Euler equations ignoring viscous drag, model only the wing, ignore the aircraft trim constraint, use only 16 points in the aerodynamic surrogate, use an empirically-based propulsion model, allow the integer allocation design variables to converge to continuous values, and ignore aeroelastic deflections.

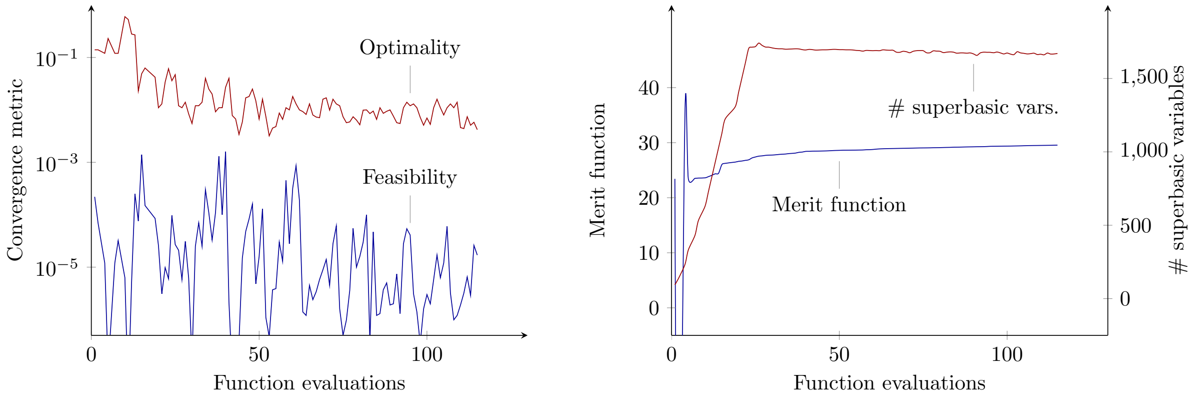

Figure 16: Optimization convergence history.

Instead, the current work focuses on addressing the computational challenges of coupling these disciplines together. We address the first challenge, dealing with over 6000 design variables, by using a gradient-based optimizer and computing the required derivatives using the adjoint method. The second challenge is that of integrating three disciplines that are each composed of many components. We use a computational framework that adopts the *modular analysis and unified derivatives* (MAUD) architecture to handle parallel data passing between components and to provide a semi-automatic implementation of the adjoint method to compute multidisciplinary derivatives. The third challenge is the need to evaluate the aerodynamic model tens of millions of times during the solution of the allocation-mission-design optimization problem. We deal with this issue by using a surrogate model that is retrained in the loop, at the start of each optimization iteration. Finally, we address the issue of the large computational cost of performing CFD and mission analysis by using efficient parallelization facilitated by the aforementioned computational framework.

We apply the allocation-mission-design optimization algorithm on a swept, linearly tapered wing parametrized with 6 twist and 72 shape design variables and analyzed using a 3-D Euler solver. The continuous altitude profiles and the cruise Mach numbers on each mission are also design variables, in addition to those coming from the allocation problem to represent how an existing fleet of aircraft is deployed on the hypothetical airline's 128 routes. The objective function, airline profit, depends on the block times and fuel burn values computed by each mission analysis.

The results show a 27% increase in airline profit going from the allocation-only optimization to the simultaneous allocation-mission-design optimization. Examining the optimized variables, we see that some of this additional profit comes from flying the next-generation aircraft on more routes overall and in particular on short-range routes. Moreover, we see differences in comparing the optimized wing geometry from the simultaneous optimization with that from a design-only optimization. We also see a distribution of optimal cruise Mach numbers depending on the mission range. There are several steps for future work including adding aircraft sizing, engine sizing, aeroelastic analysis, using RANS CFD, and applying this algorithm to unconventional configurations, where we see the highest potential impact for this approach.

VII. Acknowledgments

The authors gratefully acknowledge support from NASA through grant number NNX14AC73A—technical monitor Justin S. Gray. The authors also thank Dr. Gaetan W. Kenway for several insightful discussions.

References

[1] Del Rosario, R., Follen, G., Wahls, R., and Madavan, N., "Subsonic Fixed Wing Project Overview of Technical Challenges for Energy Efficient, Environmentally Compatible Subsonic Transport Aircraft," *Proceedings of the 50th AIAA Aerospace Sciences Meeting*, Nashville, TN, 2012.

[2] Ivaldi, D., Secco, N. R., Chen, S., Hwang, J. T., and Martins, J. R. R. A., "Aerodynamic shape optimization of a truss-braced-wing aircraft," *16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Jun 2015.

[3] Raymer, D. P., *Aircraft Design: A Conceptual Approach*, AIAA, 4th ed., 2006.

[4] Kenway, G. K. W. and Martins, J. R. R. A., "Multipoint Aerodynamic Shape Optimization Investigations of the Common Research Model Wing," *AIAA Journal*, 2015. doi:10.2514/1.J054154, (In press).

[5] Kenway, G. K. W. and Martins, J. R. R. A., "Multipoint 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.

[6] Liem, R., Kenway, G. K. W., and Martins, J. R. R. A., "Multimission Aircraft Fuel Burn Minimization via Multipoint Aerostructural Optimization," *AIAA Journal*, Vol. 53, No. 1, January 2015, pp. 104–122. doi:10.2514/1.J052940.

[7] Burdette, D. A., Kenway, G. K., and Martins, J. R. R. A., "Performance Evaluation of a Morphing Trailing Edge Using Multipoint Aerostructural Design Optimization," *57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, American Institute of Aeronautics and Astronautics, January 2016. doi:10.2514/6.2016-0159.

[8] Martins, J. R. R. A., *Green Aviation*, chap. Fuel burn reduction through wing morphing, Wiley, 2015, (In press).

[9] Kota, S., Osborn, R., Ervin, G., Maric, D., Flick, P., and Paul, D., "Mission Adaptive Compliant Wing–Design, Fabrication and Flight Test," *RTO Applied Vehicle Technology Panel (AVT) Symposium*, 2009.

[10] Clarke, J.-P. B., Ho, N. T., Ren, L., Brown, J. A., Elmer, K. R., Zou, K., Hunting, C., McGregor, D. L., Shivashankara, B. N., Tong, K.-O., et al., "Continuous descent approach: Design and flight test for Louisville International Airport," *Journal of Aircraft*, Vol. 41, No. 5, 2004, pp. 1054–1066.

[11] Shresta, S., Neskovic, D., and Williams, S., "Analysis of continuous descent benefits and impacts during daytime operations," *8th USA/Europe Air Traffic Management Research and Development Seminar (ATM2009), Napa, CA*, 2009.

[12] 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.

[13] Hwang, J. T., *A modular approach to large-scale design optimization of aerospace systems*, Ph.D. thesis, University of Michigan, 2015.

[14] Balay, S., Gropp, W. D., McInnes, L. C., and Smith, B. F., "Efficient Management of Parallelism in Object Oriented Numerical Software Libraries," *Modern Software Tools in Scientific Computing*, edited by E. Arge, A. M. Bruaset, and H. P. Langtangen, Birkhäuser Press, 1997, pp. 163–202.

[15] Gray, J., Moore, K. T., Hearn, T. A., and Naylor, B. A., "Standard Platform for Benchmarking Multidisciplinary Design Analysis and Optimization Architectures," *AIAA Journal*, Vol. 51, No. 10, 2013, pp. 2380–2394. doi:10.2514/1.J052160.

[16] Gray, J., Hearn, T., Moore, K., Hwang, J. T., Martins, J. R. R. A., and Ning, A., "Automatic Evaluation of Multidisciplinary Derivatives Using a Graph-Based Problem Formulation in OpenMDAO," *Proceedings of the 15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Atlanta, GA, June 2014. doi:10.2514/6.2014-2042.

[17] Chin, J., Gray, J., Jones, S., and Berton, J., "Open-Source Conceptual Sizing Models for the Hyperloop Passenger Pod," *2015 AIAA SciTech Conference*, American Institute of Aeronautics and Astronautics, Janurary 2015.

[18] Liem, R. P., Mader, C. A., and Martins, J. R. R. A., "Surrogate Models and Mixtures of Experts in Aerodynamic Performance Prediction for Aircraft Mission Analysis," *Aerospace Science and Technology*, Vol. 43, June 2015, pp. 126–151, 10.1016/j.ast.2015.02.019.

[19] Hwang, J. T. and Martins, J. R. R. A., "Parallel allocation-mission optimization of a 128-route network," *16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Jun 2015.

[20] Uyttersprot, L., *Inverse Distance Weighting Mesh Deformation*, Ph.D. thesis, Delft University of Technology, 2014.

[21] van der Weide, E., Kalitzin, G., Schluter, J., and Alonso, J. J., "Unsteady Turbomachinery Computations Using Massively Parallel Platforms," *Proceedings of the 44th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, NV, 2006, AIAA 2006-0421.

[22] Lyu, Z., Kenway, G. K., Paige, C., and Martins, J. R. R. A., "Automatic Differentiation Adjoint of the Reynolds-Averaged Navier–Stokes Equations with a Turbulence Model," *21st AIAA Computational Fluid Dynamics Conference*, San Diego, CA, Jul 2013. doi:10.2514/6.2013-2581.

[23] Kao, J. Y., Hwang, J. T., Martins, J. R. R. A., Gray, J. S., and Moore, K. T., "A modular adjoint approach to aircraft mission analysis and optimization," *56th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Jan 2015. doi:10.2514/6.2015-0136.

[24] Betts, J. T., "Survey of numerical methods for trajectory optimization," *Journal of guidance, control, and dynamics*, Vol. 21, No. 2, 1998, pp. 193–207.

[25] Clarke, J.-P., Brooks, J., Nagle, G., Scacchioli, A., White, W., and Liu, S., "Optimized Profile Descent Arrivals at Los Angeles International Airport," *Journal of Aircraft*, Vol. 50, No. 2, 2013, pp. 360–369.

[26] Micallef, M., Chircop, K., Zammit-Mangion, D., and Sammut, A., "Revised approach procedures to support optimal descents into Malta International Airport," *CEAS Aeronautical Journal*, 2014, pp. 1–15.

[27] Kreisselmeier, G. and Steinhauser, R., "Systematic Control Design by Optimizing a Vector Performance Index," *International Federation of Active Controls Syposium on Computer-Aided Design of Control Systems, Zurich, Switzerland*, 1979.

[28] Poon, N. M. K. and Martins, J. R. R. A., "An Adaptive Approach to Constraint Aggregation Using Adjoint Sensitivity Analysis," *Structural and Multidisciplinary Optimization*, Vol. 34, No. 1, 2007, pp. 61–73. doi:10.1007/s00158-006-0061-7.

[29] Ferguson, A. R. and Dantzig, G. B., "The allocation of aircraft to routes-an example of linear programming under uncertain demand," *Management science*, Vol. 3, No. 1, 1956, pp. 45–73.

[30] Hwang, J. T., Roy, S., Kao, J. Y., Martins, J. R. R. A., and Crossley, W. A., "Simultaneous aircraft allocation and mission optimization using a modular adjoint approach," *56th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Jan 2015. doi:10.2514/6.2015-0900.

[31] Zhao, J., Tetzloff, I. J., Tyagi, A., Dikshit, P., Mane, M., Agusdinata, D., Crossley, W. A., and DeLaurentis, D., "Assessing New Aircraft and Technology Impacts on Fleet-Wide Environmental Metrics including Future Scenarios," *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, Orlando, Florida*, 2010.

[32] Govindaraju, P. and Crossley, W. A., "Profit Motivated Airline Fleet Allocation and Concurrent Aircraft Design for Multiple Airlines," *Aviation Technology, Integration, and Operations Conference*, Aug 2013, AIAA 2013-4391.

[33] Gill, P., Murray, W., and Saunders, M., "SNOPT: An SQP algorithm for large–scale constraint optimization," *SIAM Journal of Optimization*, Vol. 12, No. 4, 2002, pp. 979–1006.

[34] Perez, R. E., Jansen, P. W., and Martins, J. R. R. A., "pyOpt: a Python-Based Object-Oriented Framework for Nonlinear Constrained Optimization," *Structural and Multidisciplinary Optimization*, Vol. 45, No. 1, January 2012, pp. 101–118. doi:10.1007/s00158-011-0666-3.

[35] Lyu, Z., Kenway, G. K., and Martins, J. R. R. A., "Aerodynamic Shape Optimization Studies on the Common Research Model Wing Benchmark," *AIAA Journal*, Vol. 53, No. 4, April 2015, pp. 968–985. doi:10.2514/1.J053318.