

Please cite this document as:

Martins, Joaquim R. R. A., "Multidisciplinary Design Optimization of Aerospace Systems," *Advances and Trends in Optimization with Engineering Applications*, SIAM, 2017, pp. 249–257. doi:10.1137/1.9781611974683.ch19.

This document can be found at: <http://mdolab.engin.umich.edu>.

Multidisciplinary Design Optimization of Aerospace Systems

Joaquim R. R. A. Martins¹

Abstract Aerospace engineering has been at the forefront both in modeling and design optimization due to the demand for high performance. The objective of this chapter is to show how numerical optimization has been useful in the design of aerospace systems and to give an idea of the challenges involved.

1 Introduction

Aerospace engineering has been at the forefront both in the modeling and design optimization due to the demand for high performance. The objective of this chapter is to show how numerical optimization has been useful in the design of aerospace systems, and to give an idea of the challenges involved.

In the design of aerospace systems, mass is especially critical for performance. In aircraft applications, every unit of mass must be compensated with a corresponding increase in lift. This additional lift incurs additional drag and thus thrust, which in turn increases the required amount of fuel, leading to a further increase in total mass. Thus, one unit increase in structural or payload weight can lead to up to a four unit increase overall due to the additional fuel required. In space applications, there is a high cost associated with mass due to the large amount of potential and kinetic energy required to put spacecraft in orbit. This is evidenced by the launch cost per unit weight. Since there is such a high premium on mass, it is economically advantageous to invest in mass reduction through research and development in the modeling and design process.

Aerospace systems require the integrations of multiple disciplines for a successful design. In aircraft design, for example, aerodynamics, structures, propulsion and control all come into play. Any design changes within one discipline impacts the other disciplines, sometimes in significant and nonintuitive ways. Thus it is important to couple all disciplines when modeling the performance of aerospace systems. In addition, the design optimization must be performed by considering the design variables in all disciplines simultaneously to make sure that the true multidisciplinary optimum is found.

The field of multidisciplinary design optimization (MDO) was born out of this necessity, as aerospace engineers invested in the application of numerical optimization to the design of multidisciplinary aerospace systems. One of the first applications was wing design, where the coupling of aerodynamics and structures is particularly important [Haf77]. Since then, the application of MDO has been extended to a wide range of other engineering systems [ML13].

We start the remainder of this chapter by giving a background on the types of optimization programs that are typically found in aerospace applications (objective functions, design variables, constraints, and models). We then present three different applications: aerodynamic shape optimization applied to wing design, simultaneous optimization of aerodynamic shape and structure for this same application, and finally,

¹Professor, Department of Aerospace Engineering, University of Michigan, Ann Arbor, Michigan, USA

a satellite case that includes both the design optimization of the satellite as well as the optimization of its operations.

2 Background

2.1 Computational Models

The computational models used in aerospace engineering applications range from simple explicit algebraic expressions to the solution of nonlinear partial differential equations (PDE) in three-dimensional domains, which could also be time-dependent. We denote the computational models by the residual of a system of k $R(x, y(x)) = 0$, where x are fixed parameters and y are the k state variables that are implicitly determined by solving the governing equations. The number of state variables in the applications presented herein is up to $\mathcal{O}(10^6)$. These governing equations are usually solved using specialized solvers that were developed to solve the physics at hand.

One of the challenges of MDO is the solution of coupled systems of governing equations that pertain to different physical models (e.g., fluid vs. structure). While we might be tempted to use a single solver for the coupled system, specialized segregated solvers are usually used for practical reasons. In Secs. 3–5 we specify the governing equations and respective solvers used in the respective applications.

2.2 Optimization Problem

Most problems in the design optimization of aerospace systems are NLPs of the same form as presented in Ch. 1, where both the objective functions f and constraints h, g are usually nonlinear and non-convex. The design variables can be both continuous or discrete, but we restrict ourselves to continuous variables in the applications presented in this chapter.

The governing equations $R(x, y(x)) = 0$ introduced in Sec. 2.1 could be considered as equality constraints in the NLP problem, where y would also be a vector of design variables. However, as previously mentioned, we tend to use specialized solvers to solve the governing equations separately. At each optimization iteration, the solver finds the y that satisfies $R(x, y(x)) = 0$ for a given x , and thus $y(x)$ is effectively an implicit function of x . The dimension of the states equals that of the governing equations, and there is usually a unique solution for this problem. Once the governing equations are satisfied and y is computed, we can evaluate the objective function and the constraints that are not governing equations. Hence this is an optimization problem that is solved in a reduced space, which can be expressed as:

$$\min_{x \in \mathbb{R}^n} f(x, y(x)) \tag{1a}$$

$$\text{s.t. } h(x, y(x)) = 0 \tag{1b}$$

$$g(x, y(x)) \leq 0 \tag{1c}$$

where y satisfies $R(x, y(x)) = 0$ for each x . While we could consider a full space approach where the optimizer is responsible for satisfying the governing equations, it is unlikely that a general purpose nonlinear optimizer would be efficient enough in the solution of nonlinear PDEs with $\mathcal{O}(10^6)$ state variables. Furthermore, the implementation of a full-space method is intrusive and requires an extensive development effort.

2.3 Optimization Algorithms

There are two fundamental choices when it comes to optimization algorithms: derivative-free or derivative-based methods. The design optimization applications presented herein are afflicted by two compounding challenges: large numbers of design variables ($\mathcal{O}(10^2)$ variables or more), and a high cost of evaluating the objective and constraints (which involve the solution of governing equations with $\mathcal{O}(10^6)$ state variables).

Since the number of iterations required by derivative-free methods does not scale well with the number of optimization variables, we chose to use a derivative-based method. Given the efficiency of derivative-

based methods, it is possible to address the two compounding challenges mentioned above, provided we can compute the required derivatives efficiently (addressed in Sec. 2.4).

Derivative-based methods are not without some drawbacks. The two main drawbacks are that: (1) convergence to local (as opposed to global) optima, and (2) sensitivity to discontinuities in the design space. When it comes to local versus global minima, most engineers are satisfied by having their designs improved relative to the initial design. Furthermore, in the problems considered herein it is impossible to guarantee convergence to a global minimum even when using a global optimizer. Discontinuities of the objective and constraints with respect to the design variables can be either caused by discontinuities in y with respect to x due to the solution of the governing equations $R(x, y(x)) = 0$, or by discontinuities in $f(x, y(x))$, $h(x, y(x))$ or $g(x, y(x))$ with respect to x . In our experience, there are few cases where there are discontinuities that have an underlying physical justification, and many discontinuities are introduced in the modeling. Thus, we need to minimize the introduction of such discontinuities when developing solvers and constructing the objective and constraint functions. Even when there are physical discontinuities, these can be smoothed out without loss of fidelity in the physical model (see Sec. 5)

There are a number of derivative-based nonlinear optimization packages available. We use SNOPT [GMS05], an implementation of the sequential quadratic programming algorithm suitable for general nonlinear constrained problems. The search directions are determined by minimizing a quadratic model of the Lagrangian function subject to linearized constraints. The line search seeks to reduce an augmented Lagrangian merit function to ensure convergence to a local optimum from any starting point. To facilitate the use of SNOPT and its integration with the various solvers considered herein, we use the Python interface pyOpt [PJM12], which also enables the use of other optimizers with few modifications to the main program.

2.4 Computing Gradients Efficiently

Given the choice of a gradients-based optimizer, the efficiency of the overall optimization hinges on an efficient evaluation of the gradients of the f and c with respect to the design variables. A number of methods are available for evaluating derivatives of PDE systems: finite differences, the complex-step method, algorithmic differentiation (forward or reverse mode), and analytic methods (direct or adjoint) [MH13]. The computational cost of these methods is either proportional to the number of design variables, or proportional to the number of functions being differentiated.

Since we have a large number of design variables, the best options are the reverse mode algorithmic differentiation or the adjoint method. In our applications we tend to use a hybrid method that combines the adjoint method with algorithmic differentiation (both forward and reverse modes.)

We now derive the adjoint method for evaluating the derivatives of a function of interest, $f(x, y(x))$ (which in our case are the objective function and constraints), with respect to the design variables x . The only independent variables are x , since y is determined implicitly by the solution of the governing equations, $R(x, y(x)) = 0$, for a given x . Using the implicit function theorem, the gradient of f and the with respect to x is,

$$\frac{df}{dx} = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial y} \frac{dy}{dx} \quad (2)$$

A similar expression can be written for the Jacobian of R ,

$$\frac{dR}{dx} = \frac{\partial R}{\partial x} + \frac{\partial R}{\partial y} \frac{dy}{dx} = 0 \quad \Rightarrow \quad \frac{\partial R}{\partial y} \frac{dy}{dx} = -\frac{\partial R}{\partial x} \quad (3)$$

We can now solve this linear system to evaluate the gradients of the state variables with respect to the design variables. Substituting this solution into the evaluation of the gradient of f (2) yields,

$$\frac{df}{dx} = \frac{\partial f}{\partial x} - \frac{\partial f}{\partial y} \left[\frac{\partial R}{\partial y} \right]^{-1} \frac{\partial R}{\partial x} \quad (4)$$

The adjoint method consists in factorizing the Jacobian $\partial R/\partial y$ with $\partial f/\partial y$, i.e., we solve the adjoint equations:

$$\left[\frac{\partial R}{\partial y} \right]^T \psi = -\frac{\partial f}{\partial y}, \quad (5)$$

where the adjoint vector ψ is the same for all x . We can then substitute the result into the total gradient equation (2),

$$\frac{df}{dx} = \frac{\partial f}{\partial x} + \psi^T \frac{\partial R}{\partial x} \quad (6)$$

to get the required gradient. The partial derivatives in the equations above are inexpensive to evaluate, since they do not require the solution of the governing equations. The computational cost of evaluating gradients with the adjoint method is independent of the number of design variables, but dependent on the number of functions of interest.

3 Aerodynamic Shape Optimization

Among the many parameters that an aircraft designer must decide on, the parameters determining the shape of the wing are among the most crucial and complex. Small changes have a large effect on both drag and lift, and a large number of parameters are required to reduce the drag as much as possible.

The typical wing aerodynamic shape optimization is to minimize the drag coefficient (C_D) by varying the design variables x (which consist on shape design variables and angle of attack) subject to a lift coefficient constraint, a moment constraint, and a number of geometric constraints. In our example, we have,

$$\min_{x \in \mathbb{R}^n} C_D(x, y(x)) \quad (7a)$$

$$\text{s.t. } C_L(x, y(x)) = 0.5 \quad (7b)$$

$$C_M(x, y(x)) \geq -0.17 \quad (7c)$$

$$g_{\text{geo}}(x) \leq 0 \quad (7d)$$

The geometric constraints g_{geo} are required to prevent the wing from becoming too thin. There is an strong incentive to decrease the thickness of the wing cross sections to reduce the drag at high subsonic speeds but, a thin wing makes for a heavy structure. Since in this case the structure is not considered in the modeling of the optimization, geometric constraints are required.

The solution of the PDEs governing the flow is implicit in this formulation: as previously mentioned, we use a reduced space approach, where the governing equations are solved for a given shape x at each optimization iteration. In these examples, we use the Sumb flow solver [?], a finite-volume, cell-centered multiblock solver for the Reynolds-averaged Navier–Stokes (RANS) equations. The discrete adjoint method for the RANS equations was developed by forming Eqns.(5) and (6), where the partial derivatives are implemented by performing algorithmic differentiation in the relevant parts of the original code [LKPM13].

We solve the RANS equations in the three-dimensional domain surrounding the wing, using a mesh with 450 560 cells, resulting in over 2.2 million state variables. The evaluation of the drag, lift, and moment coefficients consists in the numerical integration of the flow pressure distribution on the surface of the aircraft. The wing geometry is taken from the Common Research Model (CRM) aircraft configuration [Vas08]. The shape design variables (x) are the vertical positions of 768 points that control a free-form deformation volume, which allows for a fine control of the wing airfoil shapes. We enforce 750 thickness constraints distributed on the wing to be greater or equal than 25% of the initial thickness at the respective locations. The internal volume is also constrained to be greater than or equal to the initial volume. The reason for this volume constraint is that the wing also serves as the main fuel tank for the aircraft.

To demonstrate the robustness of the optimization process, we solve the optimization problem(7d) starting with a wing that is a random perturbation of the CRM wing. The solution of this design optimization

problem is shown in Fig. 1. The upper left quadrant shows a planform view of the wing with contours of the pressure coefficient, C_p . The left wing is the initial design, where we can see that the C_p distribution is erratic due to the random shape. The wing on the right side is the optimized one, where we can see that the C_p contours tend to be parallel to the spanwise direction of the wing. Below the planform view, we can see a front view of the wing showing the shock surface (in yellow), and the spanwise distributions of lift, wing geometric twist, and thickness-to-chord ratio. All the red lines refer to the starting wing, while the blue ones refer to the optimized one. The two columns on the right show the airfoil cross sectional shapes corresponding to the positions denoted by the respective letters shown on the wing planform view. The C_p distribution is shown above each airfoil. We can see the random nature of the airfoil shapes (in red) and the corresponding C_p distributions. The smooth optimized airfoils (in blue) result in a shock-free solution with a much lower drag.

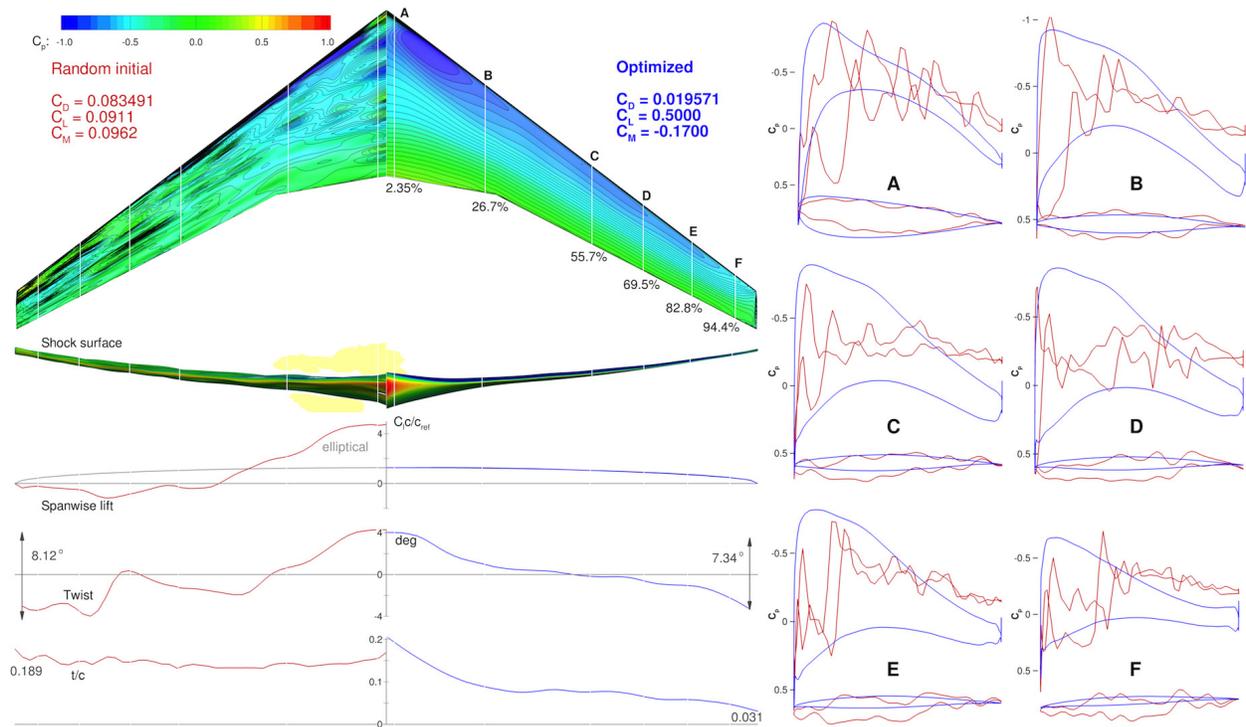


Figure 1: The optimization starts from a random geometry and converges to an optimal wing.

More details on this work, as well as other design optimization cases (including a case with 144 million state variables, multiple flight conditions, and different thickness constraints), and a more thorough analysis of the results are presented by Lyu et al. [LKM14].

4 Aerostructural Design Optimization

In the previous example, we performed aerodynamic shape optimization of an aircraft wing, where we had to impose thickness constraints to prevent the wing from becoming too thin and thus infeasible from the structural point of view. In this section, we include a structural model coupled to the same aerodynamic model of the previous section to perform aerostructural design optimization. This enables the optimizer to evaluate trades between aerodynamic and structural performance, providing we define an objective function representative of the overall aircraft performance. With aerostructural optimization it is possible to eliminate the geometric thickness constraints, and to include wing planform shape (e.g., sweep and span). In addition, coupling the aerodynamic and structural models is essential for predicting the performance of even

moderately flexible wings, since the structural deflections have a strong effect on the aerodynamic loads and vice-versa.

For the aerostructural design optimization objective, we chose to use a linear combination of fuel burn (FB) and take-off gross weight (TOGW). The fuel burn affects mainly the aircraft operating cost, while the weight affects mainly the aircraft acquisition cost. Airlines look for a balance of these two metrics, and this balance is different depending on the airlines. We state the optimization problem as follows:

$$\min_{x \in \mathbb{R}^n} \quad \beta \text{FB}(x, y(x)) + (1 - \beta) \text{TOGW}(x, y(x)) \quad (8a)$$

$$\text{s.t.} \quad L(x, y(x)) = n_i W(x, y(x)) \quad i = 1, 2, 3 \quad (8b)$$

$$C_M(x, y(x)) \geq -0.17 \quad (8c)$$

$$\text{KS}_{\text{struct}}(x, y(x)) \leq 1 \quad (8d)$$

$$g_{\text{geo}}(x) \leq 0 \quad (8e)$$

where β defines the balance between the performance metrics. The design variables consist in angle of attack, airfoil shape, planform shape (sweep, span and taper), and structural thicknesses. KS is an aggregation function representing the failure of the structure. Since the structural failure originally includes a large number of constraints, and there is no efficient way of computing the Jacobian of a large number of constraints with respect to a large number of design variables, we aggregate all the constraints into a few Kreisselmeier–Steinhauser (KS) functions so that we can use the adjoint method to evaluate the gradients of KS [?]. Three different flight conditions are considered for which the lift must be equal to the weight ($L = W$): $n = 1$ corresponds to the cruise condition, which is used to evaluate the objective function, while $n = 2.5$ and $n = -1$ are maneuver conditions, which are used to enforce the structural failure constraints.

The aerostructural model consists in the RANS aerodynamic model described in the previous example coupled to a structural finite-element model of the wing. The structural solver is a parallel direct solver that can also evaluate derivatives using the adjoint method [?]. The coupled system is solved using a Newton–Krylov method. The adjoint method described in the previous section can also be implemented to the coupled problem to obtain all the derivatives required in the aerostructural design optimization problem.

Figure 2 shows the result of solving the aerostructural design optimization problem for $\beta = 0$ (left wing) and $\beta = 1$ (right wing), which corresponds to minimizing take-off weight, and fuel burn, respectively. As in Fig. 1, we show the contours of pressure coefficient on a planform view of the wing in the upper left quadrant, but now we add another planform view below that shows the wing box structural thickness distribution. The front view now shows the actual deflected shapes of the wings for both the cruise and maneuver conditions.

As we can see, the fuel burn minimization converged to larger span than the take-off weight minimization. This is due to the fact that fuel burn minimization trades off a further decrease in drag for a penalty in structural weight.

5 Optimization of a Satellite’s Design and Operation

The previous two examples focused on wing design optimization, starting with one discipline (aerodynamics) in the first example, and then coupling this to a second discipline (structures) to perform MDO. The solution of this problem was enabled by the coupled adjoint method, which is able to evaluate gradients of the coupled system efficiently. We can take this idea further to many more disciplines. In this chapter we present a satellite MDO problem with seven disciplines: orbit dynamics, attitude dynamics, cell illumination, temperature, solar power, energy storage, and communication. Many of these disciplines include functions with discontinuities and non-smooth regions that are addressed to enable a numerically exact computation of derivatives for all the modeled variables. The wide-ranging time scales in the design problem, spanning 30 seconds to 1 year, are captured through a combination of multi-point optimization and the use of a small time step in the analyses.

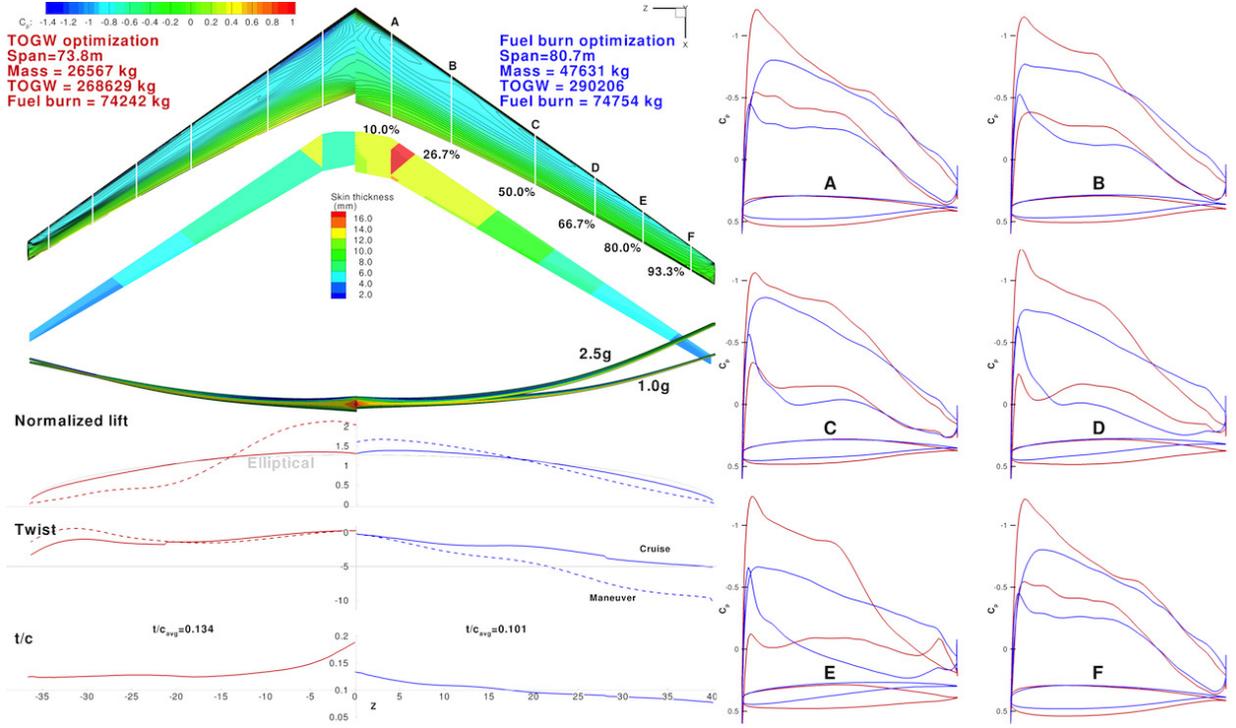


Figure 2: Comparison of two wing aerostructural design optimization problems: take-off weight minimization and fuel burn minimization.

This satellite design optimization problem can be stated as follows:

	Function/variable/	Description	Quantity
$\min_{x \in \mathbb{R}^n}$	$\sum_{i=1}^6 D_i$	Data downloaded	
with respect to	$0 \leq I_{solar} \leq 0.4$	Solar panel current	$300 \times 12 \times 6$
	$0 \leq \gamma \leq \pi/2$	Roll angle profile	300×6
	$0 \leq P_{comm} \leq 25$	Communication power	300×6
	$0 \leq cell \leq 1$	Cell vs. radiator	84
	$0 \leq \alpha_{fin} \leq \pi/2$	Fin angle	1
	$0 \leq \alpha_{ant} \leq \pi$	Antenna angle	1
	$0.2 \leq SOC_i \leq 1$	Initial state of charge	6
		Total	25292
subject to	$I_{bat} - 5 \leq 0$	Battery charge constraint	6
	$-10 - I_{bat} \leq 0$	Battery discharge constraint	6
	$0.2 - SOC \leq 0$	Battery capacity constraint	6
	$SOC - 1 \leq 0$	Battery capacity constraint	6
	$SOC_f - SOC_i = 0$	SOC periodicity constraint	6
		Total	30

The objective is to maximize the data downloaded from the satellite to ground station by varying both satellite design parameters (cell and radiator allocation, fin angle, and antenna angle), and satellite control variables (these are variables that vary in time: solar panel current, roll angle, and communication power).

The total number of design variables is 25,292. All the design constraints are related to limitations of the battery, and the governing equations for all disciplines are solved separately as before, i.e., we use a reduced space approach. The crucial combination of numerical methods that makes the solution of this large-scale nonlinear problem tractable is the combination of SNOPT, a Newton method for the solution of the coupled governing equations, and a coupled adjoint for evaluating the derivatives efficiently. Hwang et al. [HLCM13] provide much more detail on this optimization problem and the governing equations for each discipline.

6 Conclusion

The strong incentive to invest in improving the performance of aerospace systems, together with the complex multidisciplinary nature of these systems, led to the development of high-fidelity MDO. Because these design optimization problems involve large numbers of design variables, our approach was to use a derivative-based optimizer in conjunction with adjoint methods, which efficiently evaluate the derivatives for such problems.

The three examples presented here are not meant to span the wide variety of MDO problems in aerospace, but they show how it is possible to solve large-scale design optimization problems by combining a derivative-based optimizer, efficient methods for solving coupled systems of governing equations, and a coupled adjoint for evaluating the derivatives efficiently. Furthermore, these approaches to MDO can be generalized and applied to the design optimization of other engineering systems that involve multiple disciplines or components.

References

- [GMS05] Philip E. Gill, Walter Murray, and Michael A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM Review*, 47(1):99–131, 2005.
- [Haf77] Raphael T. Haftka. Optimization of flexible wing structures subject to strength and induced drag constraints. *AIAA Journal*, 15(8):1101–1106, 1977.
- [HLCM13] John T. Hwang, Dae Young Lee, James W. Cutler, and Joaquim R. R. A. Martins. Large-scale MDO of a small satellite using a novel framework for the solution of coupled systems and their derivatives. In *Proceedings of the 54th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Boston, MA, April 2013.
- [LKM14] Zhoujie Lyu, Gaetan K. W. Kenway, and Joaquim R. R. A. Martins. RANS-based aerodynamic shape optimization investigations of the common research model wing. In *Proceedings of the AIAA Science and Technology Forum and Exposition (SciTech)*, National Harbor, MD, January 2014. AIAA 2014-0567.
- [LKPM13] Zhoujie Lyu, Gaetan K. Kenway, Cody Paige, and Joaquim R. R. A. Martins. Automatic differentiation adjoint of the Reynolds-averaged Navier–Stokes equations with a turbulence model. In *21st AIAA Computational Fluid Dynamics Conference*, San Diego, CA, Jul. 2013.
- [MH13] Joaquim R. R. A. Martins and John T. Hwang. Review and unification of methods for computing derivatives of multidisciplinary computational models. *AIAA Journal*, 51(11):2582–2599, November 2013.
- [ML13] Joaquim R. R. A. Martins and Andrew B. Lambe. Multidisciplinary design optimization: A survey of architectures. *AIAA Journal*, 51(9):2049–2075, September 2013.

- [PJM12] Ruben E. Perez, Peter W. Jansen, and Joaquim R. R. A. Martins. pyOpt: A Python-based object-oriented framework for nonlinear constrained optimization. *Structural and Multidisciplinary Optimization*, 45(1):101–118, January 2012.
- [Vas08] J. Vassberg. Introduction: Drag prediction workshop. *Journal of Aircraft*, 45(3):737–737, Jun 2008.