Enabling Modular Aerostructural Optimization: Individual Discipline Feasible without the Jacobians

Alp Dener* and Jason E. Hicken†
Rensselaer Polytechnic Institute
Mechanical, Aerospace and Nuclear Engineering Department

Gaetan K. W. Kenway‡
Science and Technology Corporation

Joaquim R. R. A. Martins§
University of Michigan, Ann Arbor
Department of Aerospace Engineering

The individual discipline feasible (IDF) formulation is a modular multidisciplinary design optimization architecture that promotes the use of disparate discipline analysis tools. IDF achieves modularity by introducing coupling variables and coupling constraints into the optimization problem, which enables the state equations for each discipline to be solved independently at each optimization iteration. However, the increased number of optimization variables and nonlinear state-based constraints poses a significant challenge to conventional matrix-explicit optimization algorithms. In this paper, we apply a reduced-space inexact-Newton-Krylov (RSNK) algorithm to a large-scale, high-fidelity aerostructural optimization of a commercial airliner wing. Our findings demonstrate that the RSNK algorithm, paired with a novel matrix-free preconditioner, can solve the IDF problem at least as fast as its multidisciplinary feasible counterpart. In particular, the IDF preconditioner remains highly effective even in the presence of thousands of coupling constraints.

I. Introduction

The advent of high-performance computing has enabled researchers in both industry and academia to create predictive computational models for a variety of physical phenomena. The availability of these tools has made simulation-based design indispensable for the development of new technologies and products by helping engineers efficiently evaluate more prototypes than would be possible with physical experimentation alone. However, further design improvements and scientific discoveries beyond what is afforded by single-discipline analysis requires the consideration of complex multiphysics interactions. Accurate prediction of these systems usually requires the solution of coupled partial-differential equations (PDEs). This poses a challenge to traditional workflows based on the independent analysis and design of disparate subsystems.

Multidisciplinary design optimization (MDO) is a powerful methodology that seeks to inform the design of these complex multiphysics systems. MDO practitioners develop and apply numerical optimization techniques to reveal trade-offs in the design that may otherwise remain undiscovered. In doing so, however, one would like to take advantage of existing (well-validated) discipline-specific analysis tools that employ specialized solution methods for their governing equations. The individual discipline feasible (IDF) formulation [1–3] is an MDO architecture that supports the use of such disparate PDE solvers by decoupling the disciplines at each optimization iteration. The coupling is instead addressed within the optimization problem through the introduction of coupling variables and constraints.

Despite its advantages in terms of modularity, IDF has remained largely unused since its introduction in the early 1990s. One reason is that the large number of nonlinear IDF constraints and optimization variables has proven to be challenging for conventional optimization algorithms. Dener and Hicken [4] developed a reduced-space

*Ph.D. Alumnus (Graduated: December 2017), Postdoctoral Appointee at Argonne National Laboratory, AIAA Student Member
†Assistant Professor, AIAA Member
‡Research Scientist/Engineer, AIAA Member
§Professor, AIAA Member

inexact-Newton-Krylov (RSNK) algorithm and a matrix-free preconditioner tailored for IDF problems to address these limitations, and demonstrated their effectiveness on two small IDF problems. The findings in [4] indicate that the RSNK algorithm with the IDF preconditioner can solve IDF problems at least as efficiently as problems posed in the (less modular) multidisciplinary feasible (MDF) formulation. In the present work, we apply this approach to a high-fidelity aerostructural optimization test case and evaluate whether our previous observations can be replicated in a large-scale problem.

The remainder of the paper is organized as follows. First, we introduce the IDF formulation and review the relevant literature. Next, we provide an overview of both the RSNK algorithm and the IDF preconditioner. Then, we present the numerical results from the large-scale aerostructural optimization of a commercial airliner wing. We conclude the paper with a summary of our findings.

II. Individual Discipline Feasible

Our IDF-based optimization framework is applicable to any multidisciplinary system; however, we will describe the framework in the context of a generic reduced-space aerostructural optimization of a wing:

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & \mathcal{F}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}, \boldsymbol{v}), \boldsymbol{v}(\boldsymbol{x}, \boldsymbol{u})), \\ \text{subject to} \quad & C(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}, \boldsymbol{v}), \boldsymbol{v}(\boldsymbol{x}, \boldsymbol{u})) = \boldsymbol{0}, \\ \text{governed by} \quad & \mathcal{R} = \begin{pmatrix} \mathcal{R}_{aero}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{v}) \\ \mathcal{R}_{struct}(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{u}) \end{pmatrix} = \boldsymbol{0}, \end{aligned}$$

where $\boldsymbol{x}$ are the design variables (e.g. wing twist, sweep angle, structural sizing), $\boldsymbol{u}$ are elastic deformations of the wing box, and $\boldsymbol{v}$ are the flow states. The objective function, $\mathcal{F}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{v})$, represents a scalar cost metric that is to be minimized; for example, a weighted combination of induced drag and wing mass. The general constraints, $C(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{0}$, are typically imposed on functionals such as lift, pitching moment, or Von Mises stresses in the wing box. In the present work, we restrict ourselves to equality constraints.

In the reduced-space formulation, the state variables are expressed as functions of the design variables by using the implicit function theory. The "governed by" notation above signifies that the relationship between the states and the design variables is described by the PDE residuals for each discipline, $\mathcal{R}_{aero} = \boldsymbol{0}$ and $\mathcal{R}_{struct} = \boldsymbol{0}$, which are solved fully at each optimization iteration. In the context of MDO, this formulation is also referred to as multidisciplinary feasible (MDF), owing to the fact that design points traversed in the optimization always remain feasible with respect to the fully coupled analysis problem.

The MDF formulation has been successfully applied to high-fidelity aeroelastic problems [5–8]. However, this approach requires a coupled multidisciplinary analysis and, in the context of gradient-based optimization, a coupled adjoint solution at each optimization iteration. High-fidelity engineering analysis is traditionally performed by discipline-specific solvers developed by domain experts, and coupling these disparate solvers presents trade-offs and challenges in software development. Stationary iterative methods, such as block Jacobi or Gauss-Seidel, offer an easy way to implement multidisciplinary coupling, but they are not competitive in computational cost compared with monolithic approaches such as Newton-Krylov [9]. On the other hand, Newton-Krylov coupling requires access to the source code of the underlying solvers and may be impractical for legacy or commercial software.

The individual discipline feasible (IDF) formulation [1, 2] offers a modular alternative to MDF. For the aerostructural problem under consideration, the IDF problem statement is as follows:

$$\begin{aligned} \underset{\boldsymbol{x}, \bar{\boldsymbol{u}}, \bar{\boldsymbol{v}}}{\text{minimize}} \quad & \mathcal{F}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}, \bar{\boldsymbol{v}}), \boldsymbol{v}(\boldsymbol{x}, \bar{\boldsymbol{u}})), \\ \text{subject to} \quad & C(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}, \bar{\boldsymbol{v}}), \boldsymbol{v}(\boldsymbol{x}, \bar{\boldsymbol{u}})) = \boldsymbol{0}, \\ & C_{idf}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}, \bar{\boldsymbol{v}}), \boldsymbol{v}(\boldsymbol{x}, \bar{\boldsymbol{u}})) = \begin{pmatrix} C_{\bar{u}} \\ C_{\bar{v}} \end{pmatrix} = \begin{pmatrix} \mathcal{U}(\boldsymbol{u}) - \bar{\boldsymbol{u}} \\ \mathcal{V}(\boldsymbol{v}) - \bar{\boldsymbol{v}} \end{pmatrix} = \boldsymbol{0}, \\ \text{governed by} \quad & \mathcal{R} = \begin{pmatrix} \mathcal{R}_{aero}(\boldsymbol{x}, \boldsymbol{u}, \bar{\boldsymbol{v}}) \\ \mathcal{R}_{struct}(\boldsymbol{x}, \boldsymbol{v}, \bar{\boldsymbol{u}}) \end{pmatrix} = \boldsymbol{0}, \end{aligned}$$

where $\mathcal{U}(\boldsymbol{u})$ and $\mathcal{V}(\boldsymbol{u})$ are functions that compute terms responsible for coupling between disciplines. In the

aerostructural problem considered in this work, $\mathcal{U}(\boldsymbol{u})$ would extract only the structural deformations on the wing surface, while $\mathcal{V}(\boldsymbol{v})$ would compute the aerodynamic forces on the wing based on the flow states.

The IDF formulation introduces coupling variables, $\bar{\boldsymbol{u}}$ and $\bar{\boldsymbol{v}}$, which stand in for the actual coupling terms in the multidisciplinary analysis. Consequently, at a given design point $(\boldsymbol{x}, \bar{\boldsymbol{u}}, \bar{\boldsymbol{v}})$, the state equations for each discipline can be solved independently from each other and in parallel. The coupling variables are then driven by the optimization algorithm to be equal to the actual coupling terms they represent.

Although IDF offers a significant advantage in terms of modularity, this advantage comes at the cost of a much larger optimization problem. For a high-fidelity aerostructural problem, such as the one explored in this paper, the IDF formulation may introduce on the order of 10^4 additional optimization variables and just as many nonlinear state-based constraints. This poses a significant challenge to conventional gradient-based algorithms that require the explicit constraint Jacobian. In the reduced space, the constraint Jacobian is a total derivative, and its explicit assembly involves one linear adjoint solution for each individual constraint. For the IDF problem, this may entail thousands of adjoint solutions at each optimization iteration, which is computationally impractical in high-fidelity applications.

III. Reduced-Space Inexact Newton-Krylov

To solve the IDF problem efficiently, we adopt an inexact Newton-Krylov approach, which is known to exhibit favorable algorithmic scaling with respect to the number of variables on a range of nonlinear problems. In particular, we draw inspiration from inexact-Newton methods that have been successfully applied to large, full-space single-discipline optimization problems [10–14].

We have introduced this RSNK algorithm in previous work where we studied its algorithmic scaling on a single-discipline high-fidelity problem [15] and demonstrated its efficacy on a pair of small IDF test problems [4]. In the present work, we will briefly review the algorithm's major components and apply it to a large-scale, high-fidelity aerostructural test case.

To facilitate the review, we first introduce a general reduced-space PDE-governed optimization problem:

$$\begin{aligned} \underset{\boldsymbol{y}}{\text{minimize}} \quad & \mathcal{F}(\boldsymbol{y}, \boldsymbol{w}(\boldsymbol{y})), \\ \text{subject to} \quad & C(\boldsymbol{y}, \boldsymbol{w}(\boldsymbol{y})) = \boldsymbol{0}, \\ \text{governed by} \quad & \mathcal{R}(\boldsymbol{y}, \boldsymbol{w}) = \boldsymbol{0}. \end{aligned}$$

This generic problem can encapsulate the IDF formulation by defining $\boldsymbol{y}^T = (\boldsymbol{x}^T, \bar{\boldsymbol{u}}^T, \bar{\boldsymbol{v}}^T)$ and $\boldsymbol{w}^T = (\boldsymbol{u}^T, \boldsymbol{v}^T)$. Similarly, the state equations and the IDF coupling constraints are combined under $\mathcal{R}$ and C, respectively. This general optimization statement ensures that the RSNK algorithm is broadly applicable to all reduced-space PDE-governed optimization problems with equality constraints.

Next, we formulate the reduced-space Lagrangian,

$$\mathcal{L}(\boldsymbol{y}, \boldsymbol{\lambda}) = \mathcal{F}(\boldsymbol{y}) + \boldsymbol{\lambda}^T C(\boldsymbol{y}),$$

and differentiate it to obtain the first–order necessary conditions,

$$\begin{aligned} \frac{d\mathcal{L}}{d\boldsymbol{y}} &= \frac{d\mathcal{F}}{d\boldsymbol{y}} + \boldsymbol{\lambda}^T \frac{dC}{d\boldsymbol{y}} = \boldsymbol{0}, \\ \frac{d\mathcal{L}}{d\boldsymbol{\lambda}} &= C = \boldsymbol{0}, \end{aligned} \tag{1}$$

where $d/d\boldsymbol{y}$ and $d/d\boldsymbol{\lambda}$ denote total derivatives with respect to the design variables and Lagrange multipliers, respectively. In reduced-space PDE-governed optimization, these total derivatives can be efficiently computed via the adjoint method [16, 17].

In the RSNK algorithm, we apply Newton's method to (1) to produce the primal-dual system,

$$\begin{bmatrix} \mathsf{W} & \mathsf{A}^T \\ \mathsf{A} & 0 \end{bmatrix} \begin{pmatrix} \boldsymbol{p}_y \\ \boldsymbol{p}_\lambda \end{pmatrix} = - \begin{pmatrix} \frac{d\mathcal{L}}{d\boldsymbol{y}} \\ \frac{d\mathcal{L}}{d\boldsymbol{\lambda}} \end{pmatrix}, \tag{2}$$

where $\mathsf{W} = d^2\mathcal{L}/d\boldsymbol{y}^2$ is the Hessian of the Lagrangian and $\mathsf{A} = dC/d\boldsymbol{y}$ is the constraint Jacobian. We solve this system inexactly at each Newton iteration using a Krylov solver, and we leverage the fact that Krylov solvers require only

matrix-vector-product and preconditioning operations. This approach enables the RSNK algorithm to avoid explicit and computationally intractable assembly of the constraint Jacobian and the Hessian.

Matrix-vector products with the primal-dual matrix in (2) are computed by a second-order adjoint formulation [4, 15, 18–20]. This approach entails two linear-system solutions, based on the PDE Jacobian, for each product, irrespective of the number of design variables and nonlinear constraints. In our previous numerical experiments on an aerodynamic shape optimization problem, we demonstrated that this matrix-free product is an important factor in achieving good algorithmic cost scaling with respect to the size of the design space [15]. This adjoint-based product is summarized in Appendix A.

The choice of Krylov solver in the RSNK algorithm is also critically important. Popular solvers such as generalized minimum residual (GMRES) [21] can be used only in convex cases where the Hessian of the Lagrangian, W, is positive-definite in the null space of the constraint Jacobian, A. In general nonconvex applications, however, GMRES can converge to stationary points that are not necessarily local minimizers. To account for this possibility, we use a Krylov method, called flexible equality-constrained subproblem (FLECS) solver [22], that utilizes a quadratic penalty and a trust radius constraint.

The Newton step is globalized by a trust-region approach [23] that marries effectively to the trust radius constraint in the FLECS subproblem. Step acceptance is determined through a simple filter [24], and a rejection leads to an efficient re-solve that recycles the Krylov subspace built up during the original solution of the Newton step.

These building blocks are combined into the complete RSNK algorithm shown in Alg. 1 in Appendix B, which originally appeared in [4].

IV. Matrix-Free Preconditioner for Individual Discipline Feasible

A preconditioner is an operator that approximates the inverse of a target matrix and is relatively inexpensive to apply. The application of a preconditioner is meant to improve the conditioning of linear systems of equations, and thereby improve the rate of convergence of iterative methods used to solve these systems. In the present work, we adopt a left-preconditioning approach that transforms the linear system $\mathsf{K}\boldsymbol{x} = \boldsymbol{b}$ into $\mathsf{P}^{-1}\mathsf{K}\boldsymbol{x} = \mathsf{P}^{-1}\boldsymbol{b}$ where $\mathsf{P}^{-1} \approx \mathsf{K}^{-1}$.

The RSNK algorithm uses a Krylov iterative method to solve the primal-dual system in (2), which is an ill-conditioned saddle-point system [25] that needs to be preconditioned in order for the Krylov solver to not only produce high-quality step directions but also achieve good rates of convergence. This preconditioning is particularly important in the large-scale, high-fidelity applications targeted in the present work, where the size of the Krylov subspace (and, hence, the number of iterations the Krylov solver is allowed to take) is often restricted to 10 to 20 vectors because of the computational expense of each matrix-vector product and potential memory limitations of the computer.

To that end, we have developed a novel matrix-free preconditioner for the IDF architecture based on approximately inverting the IDF constraint Jacobian, which has the form

$$\mathsf{A}_{\bar{\boldsymbol{w}}} = \frac{dC_{idf}}{d\bar{\boldsymbol{w}}} = \begin{bmatrix} \frac{dC_{\bar{\boldsymbol{u}}}}{d\bar{\boldsymbol{u}}} & \frac{dC_{\bar{\boldsymbol{u}}}}{d\bar{\boldsymbol{v}}} \\ \frac{dC_{\bar{\boldsymbol{v}}}}{d\bar{\boldsymbol{u}}} & \frac{dC_{\bar{\boldsymbol{v}}}}{d\bar{\boldsymbol{v}}} \end{bmatrix} = \begin{bmatrix} -\mathsf{I} & \frac{dC_{\bar{\boldsymbol{u}}}}{d\bar{\boldsymbol{v}}} \\ \frac{dC_{\bar{\boldsymbol{v}}}}{d\bar{\boldsymbol{u}}} & -\mathsf{I} \end{bmatrix} \tag{3}$$

for a two-discipline problem where $\bar{\boldsymbol{w}}^T = (\bar{\boldsymbol{u}}^T, \bar{\boldsymbol{v}}^T)$. Applying this preconditioner to an arbitrary vector $\boldsymbol{z}^T = (\boldsymbol{z}_{\boldsymbol{x}}^T, \boldsymbol{z}_{\bar{\boldsymbol{w}}}^T, \boldsymbol{z}_{\boldsymbol{\lambda}}^T)$ involves approximately solving a preconditioner system,

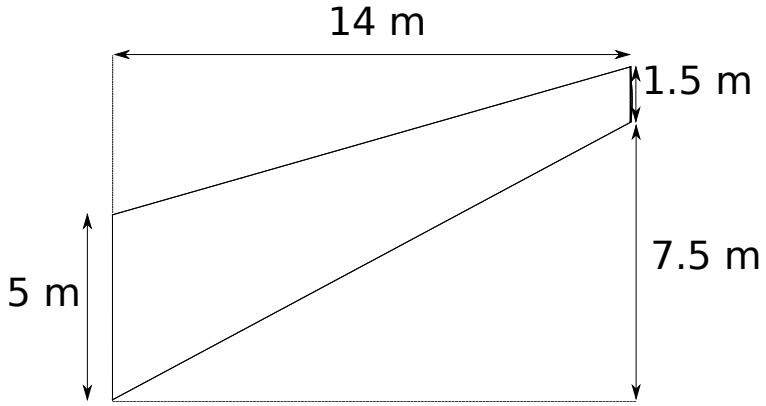
$$\begin{bmatrix} \mathsf{I} & 0 & \mathsf{A}_{\boldsymbol{x}}^T \\ 0 & 0 & \mathsf{A}_{\bar{\boldsymbol{w}}}^T \\ \mathsf{A}_{\boldsymbol{x}} & \mathsf{A}_{\bar{\boldsymbol{w}}} & 0 \end{bmatrix} \begin{pmatrix} \boldsymbol{p}_{\boldsymbol{x}} \\ \boldsymbol{p}_{\bar{\boldsymbol{w}}} \\ \boldsymbol{p}_{\boldsymbol{\lambda}} \end{pmatrix} = \begin{pmatrix} \boldsymbol{z}_{\boldsymbol{x}} \\ \boldsymbol{z}_{\bar{\boldsymbol{w}}} \\ \boldsymbol{z}_{\boldsymbol{\lambda}} \end{pmatrix}, \tag{4}$$

where $\mathsf{A}_{\boldsymbol{x}} = dC/d\boldsymbol{x}$ denotes the total derivative of the general constraints with respect to the design variables.

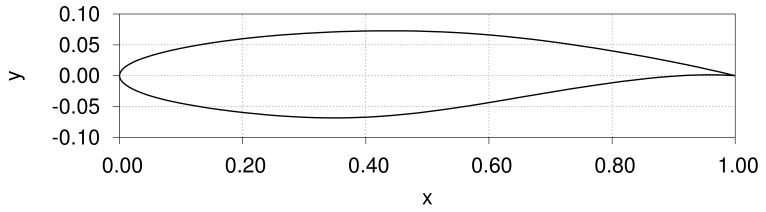
We solve (4) with a three step process.

1) Solve $\mathsf{A}_{\bar{\boldsymbol{w}}}^T \boldsymbol{p}_{\boldsymbol{\lambda}} = \boldsymbol{z}_{\bar{\boldsymbol{w}}}$ for the preconditioned dual $\boldsymbol{p}_{\boldsymbol{\lambda}}$.
2) Compute the preconditioned design $\boldsymbol{p}_{\boldsymbol{x}} = \boldsymbol{z}_{\boldsymbol{x}} - \mathsf{A}_{\boldsymbol{x}}^T \boldsymbol{p}_{\boldsymbol{\lambda}}$.
3) Solve $\mathsf{A}_{\bar{\boldsymbol{w}}} \boldsymbol{p}_{\bar{\boldsymbol{w}}} = \boldsymbol{z}_{\boldsymbol{\lambda}} - \mathsf{A}_{\boldsymbol{x}} \boldsymbol{p}_{\boldsymbol{x}}$ for the preconditioned coupling variables $\boldsymbol{p}_{\bar{\boldsymbol{w}}}$.

The linear systems in Steps 1) and 2) are solved by GMRES, where the products with the constraint Jacobians are computed by a second-order adjoint formulation similar to the one used for the primal-dual matrix in (2). Additionally, a limited-memory quasi-Newton approximation could be used in place of the identity matrix replacing the Hessian block; however, in our experiments, this did not yield any noticeable improvement.



(a) Planform view.



(b) RAE 2822 airfoil[27].

Fig. 1. Geometry of the wing used in the aerostructural optimization.

For implementation details and the invertibility proof of these Jacobians, we refer the reader to our previous publications [4, 20].

V. Numerical Results

Having reviewed the RSNK algorithm and the IDF preconditioner, we will now exercise them on a large-scale aerostructural design optimization problem. In the following subsections, we analyze the efficacy of the IDF preconditioner and compare the RSNK convergence characteristics and computational cost of both the MDF and IDF architectures. Additionally, we benchmark the RSNK algorithm on the MDF problem against SNOPT [26], a robust quasi-Newton sequential quadratic optimization algorithm.

A. Optimization Problem Statement and Setup

The test case is based on a tapered swept wing with RAE 2822 airfoil sections, as shown in Fig. 1. The sizing of the problem is based on a Boeing 717-200HGW, with maximum take-off weight (MTOW) of 54, 884 kg, service ceiling of 11, 000 m, and cruise speed of 0.77 Mach (822 km/h).

The optimization statement for this problem is

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & \frac{1}{2}\frac{D}{D_0} + \frac{1}{2}\frac{m}{m_0}, \\ \text{subject to} \quad & L = \frac{MTOW}{2},\ KS\left(\frac{\sigma_{ribs,spars}}{\sigma_{yield}}\right) = 1,\ KS\left(\frac{\sigma_{skin,upper}}{\sigma_{yield}}\right) = 1,\ KS\left(\frac{\sigma_{skin,lower}}{\sigma_{yield}}\right) = 1, \\ \text{governed by} \quad & \mathcal{R}_{Aero} = \mathbf{0},\ \mathcal{R}_{Struct} = \mathbf{0}. \end{aligned}$$

The objective function is an equally weighted combination of the scaled drag, D/D_0, and scaled wing mass, m/m_0; the drag and mass are scaled by their initial values, D_0 and m_0, respectively. The optimizer manipulates 6 twist variables (at 0%, 20%, 40%, 60%, 80%, and 100% along the half-span) and 108 structural sizing variables (thicknesses of the spars, ribs, and skin elements) for a total of 114 design variables.

Table 1. Size of the aero-structural optimization problem for each MDO architecture.

	MDF	IDF
Aerodynamic design variables	6	6
Structural design variables	108	108
Coupling variables	0	8400
Total number of optimization variables	114	8514
Aerodynamic constraints	1	1
Structural constraints	4	4
Coupling constraints	0	8400
Total number of state-based constraints	5	8405

The lift, L, is constrained to be half the MTOW to account for a single-wing solution. The stress constraints are separated into three structural groups: the ribs and spars; the upper skin; and the lower skin. For each group, the scaled Von Mises stresses, σ_v/σ_{yield}, are aggregated by a Kreisselmeier-Steinhauser (KS) function [28]. For a vector of constraints, $\boldsymbol{c} \in \mathbb{R}^K$, the KS function is defined as

$$KS(\boldsymbol{c}) = \frac{1}{\rho} \ln\left(\sum_{k=1}^{K} \exp(\rho \boldsymbol{c}_k)\right),$$

where ρ is a user-defined parameter that controls the smoothness and accuracy of the KS aggregation. Higher ρ values increase the accuracy of the KS function in estimating $\boldsymbol{c}_{\max}$ but also increase the condition number of the Hessian. For the present work, we have adopted a recommended value of $\rho = 50$ [29–32].

In general, the stress constraints should be imposed as inequality constraints. However, the IDF-based RSNK algorithm currently supports equality constraints only. To account for this limitation, we investigated the inequality constrained problem using SNOPT and observed that the stress constraints are active at the optimum. Hence, the stress constraints can be safely converted to equalities without changing the optimum solution. For this conversion, we also take advantage of the fact that the KS function is a "conservative" aggregation method such that

$$\boldsymbol{c}_{\max} \leq KS(\boldsymbol{c}) \leq \boldsymbol{c}_{\max} + \frac{\ln(K)}{\rho}.$$

In this application, we set the aggregated KS constraints to be equal to 1, which signifies the boundary of the Von Mises yield criterion where $\sigma_v = \sigma_{yield}$. The overestimation of the Von Mises stresses by the KS aggregation acts as a built-in factor of safety on the structural design, ensuring that the discrete stress values do not exceed the yield stress. For the optimization results presented in this section, both RSNK and SNOPT solve the equality-constrained problem based on this approach.

The IDF implementation of this problem introduces an additional 8,400 coupling variables and coupling constraints: 4,200 accounting for aerodynamic forces on the wing surface, and 4,200 accounting for the structural deformations of the wing surface. The optimization problem sizes for both MDF and IDF are summarized in Table 1, and the complete list of optimization parameters used in this problem is provided in Table 2 in Appendix B.

1. CFD Solver

The flow solver used in the multidisciplinary analysis is ADflow (formerly known as SUMad or SUmb) [33], a finite-volume solver for compressible Euler, laminar Navier-Stokes, and RANS equations, with support for a variety of turbulence models. ADflow also provides PDE Jacobian products, partial derivative evaluations, and adjoint solutions via matrix-free automatic differentiation. Mesh movement is performed by an efficient analytic inverse distance method [34].

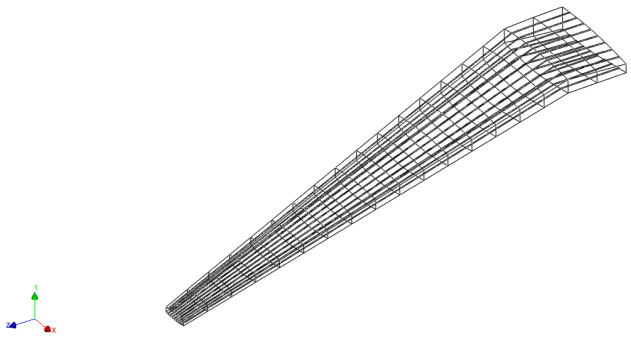


Fig. 2. Structural wing box used in the aerostructural problem.

For the aerostructural optimization case, the flow is modeled by the compressible Euler equations discretized on a structured grid with 96,767 cells. The Mach number and the angle of attack are fixed at $M = 0.77$ and $\alpha = 2.0°$, respectively.

2. Structural Solver

The elastic structural deformations under aerodynamic loads are predicted by the Toolkit for the Analysis of Composite Structures (TACS) [35]. The wing box, shown in Fig. 2, consists of 7,632 quadrilateral plate elements. The material of the wing box is Aluminum 2024, with a density of $\rho = 2780\ \mathrm{g/m^3}$, Young's modulus of $E = 73.1$ GPa, and yield stress of $\sigma_{yield} = 326$ MPa.

3. Multidisciplinary Solution for MDF

The coupled aerostructural analysis required by the MDF architecture is performed using a nonlinear block Gauss-Seidel approach with Aitken acceleration [36]. The aerodynamic forces and structural displacements are transferred by rigid links between the structural elements and the aerodynamic surface [37].

Coupled adjoint and forward linear systems are solved in a monolithic manner by flexible GMRES [38]. Both solutions are preconditioned by block Jacobi, which reuses the independent discipline preconditioners for the discipline forward and adjoint problems.

B. IDF Preconditioner Results

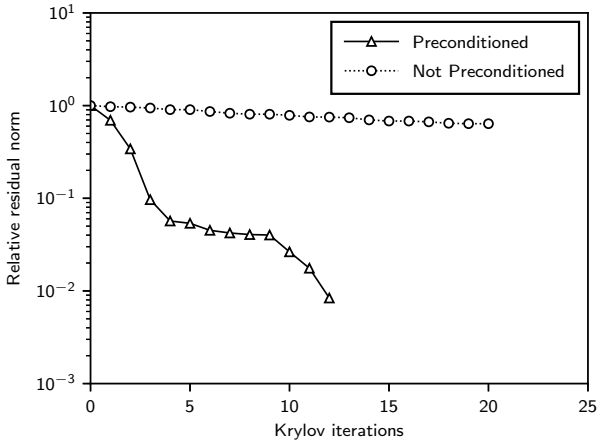
Fig. 3a shows the Krylov convergence history for the solution of the IDF KKT system with and without the IDF preconditioner. This particular history, drawn from the initial solution, is representative of the typical Krylov convergence behavior on this problem. We can see that the IDF preconditioner is critically important to the quality of the Newton step. Without the preconditioner, the KKT system cannot be solved to the required tolerance. This result demonstrates that the IDF preconditioner maintains its effectiveness even with thousands of coupling constraints present in the problem.

This observation prompts us to investigate Steps 1) and 3) in the preconditioner application more closely. Fig. 3b shows a sample convergence history for these nested solutions, which are representative of the typical nested solutions at all optimization iterations.

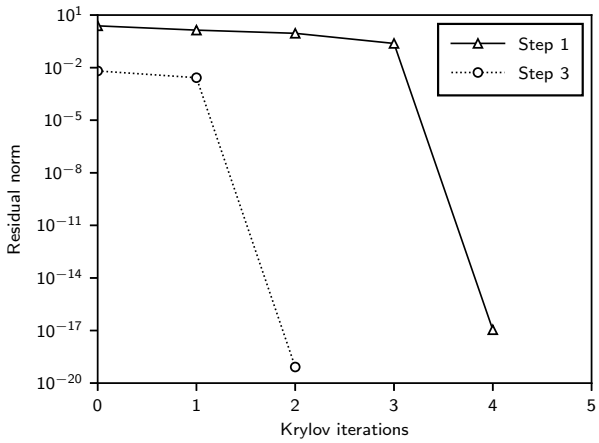
One can immediately see that the nested solutions converge rapidly, indicating that the IDF constraint Jacobian is well conditioned. To understand why, recall that the Jacobian of the IDF coupling constraints takes the following form for a two-discipline problem:

$$\frac{dC_{idf}}{d\bar{\boldsymbol{w}}} = \begin{bmatrix} \frac{dC_{\bar{u}}}{d\bar{u}} & \frac{dC_{\bar{u}}}{d\bar{v}} \\ \frac{dC_{\bar{v}}}{d\bar{u}} & \frac{dC_{\bar{v}}}{d\bar{v}} \end{bmatrix} = \begin{bmatrix} -\mathrm{I} & \frac{dC_{\bar{u}}}{d\bar{v}} \\ \frac{dC_{\bar{v}}}{d\bar{u}} & -\mathrm{I} \end{bmatrix}.$$

The rapid convergence of the linear systems that use this Jacobian suggests that the identity matrix blocks dominate the off-diagonal blocks. We also observed the consistent trend that the initial residual norm for the Step 3) solution is



(a) KKT system solution at a representative nonlinear iteration.



(b) Nested solutions within the IDF preconditioner.

Fig. 3. Krylov convergence histories.

approximately two orders of magnitude lower than Step 1) and converges in fewer iterations as well. The reason for this is unknown, and requires further study. Ultimately, however, the IDF preconditioner's success on the present problem, with $8,400$ coupling variables and constraints, suggests that the preconditioner can be applied effectively to even larger problems in the future.

C. Optimization Results

The MDF and IDF algorithms recover approximately the same solution for the aerostructural problem, producing the same optimum coefficient of drag and mass to the fourth decimal place. This optimum offers a 74.4% reduction in mass from the initial design, with a 32.2% penalty in the total drag, for an overall 21.1% improvement in the objective.

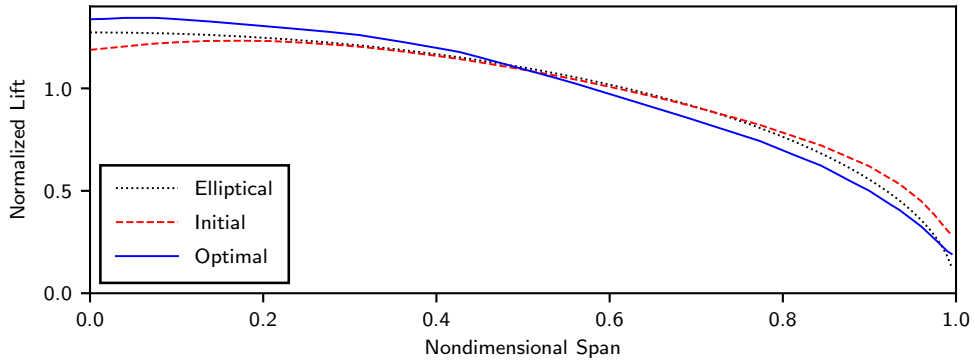
The initial and optimal lift and spanwise twist distributions are shown in Fig. 4. These results are plotted only for the RSNK solution of the MDF problem; the SNOPT solution of the MDF problem and the RSNK solution of the IDF problem are both visually indistinguishable from this result.

The initial lift distribution is close to elliptical, which is the aerodynamic optimum that minimizes induced drag. The multidisciplinary aerostructural optimum, however, increases the twist angle at the root and decreases it at the wingtip, thereby producing a wash-out that significantly reduces the stresses in the wing box and permits a lower mass structure.

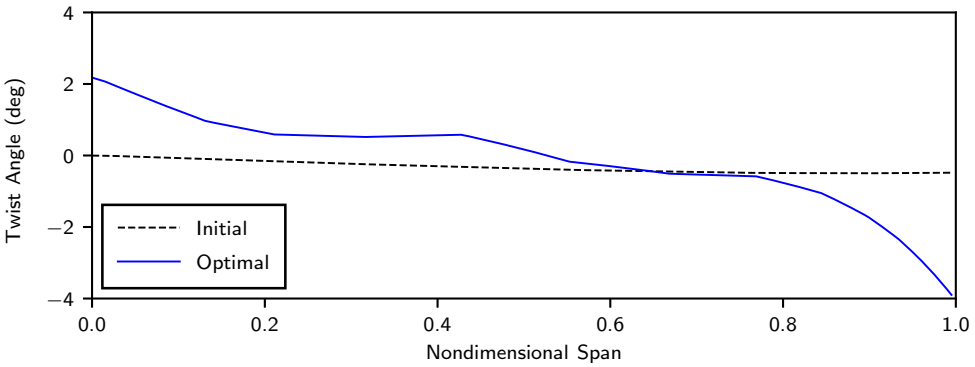
Fig. 5a shows the RSNK convergence histories for the MDF and IDF implementations of the aerostructural problem. The RSNK algorithm exhibits the expected superlinear asymptotic convergence in both cases. The computational cost for these optimization runs are measured in CPU time, normalized by the needed to perform a single multidisciplinary analysis at the initial design. The results demonstrate that the RSNK algorithm is able to solve the IDF problem at least as efficiently as MDF.

As a final benchmark, we solve the MDF problem using SNOPT. The convergence history for this optimization is shown in Fig. 5b. SNOPT is able to solve this MDF problem, with 114 design variables, approximately 22% faster than RSNK using either the MDF or IDF formulation. This observation is in line with the cost scaling comparison we have performed in previous work [15]. Based on our studies, we expect the RSNK algorithm to become more competitive on larger problems, with more design variables and state-based constraints than are present in this particular aero-structural problem.

While SNOPT can solve the MDF formulation, we emphasize that solving the IDF problem with SNOPT is computationally intractable; SNOPT requires the constraint Jacobian to be provided explicitly, which involves 8400 adjoint solutions at every optimization iteration for this aero-structural test case. The RSNK-IDF approach, on the other hand, has a computational cost comparable to the SNOPT-MDF solution, while offering modularity in the underlying PDE solvers.

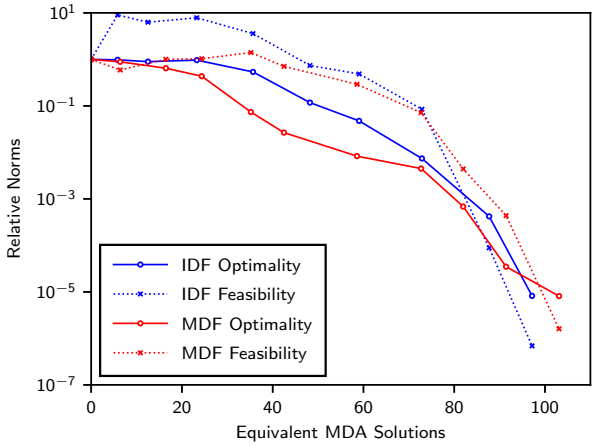


(a) Normalized lift distribution.

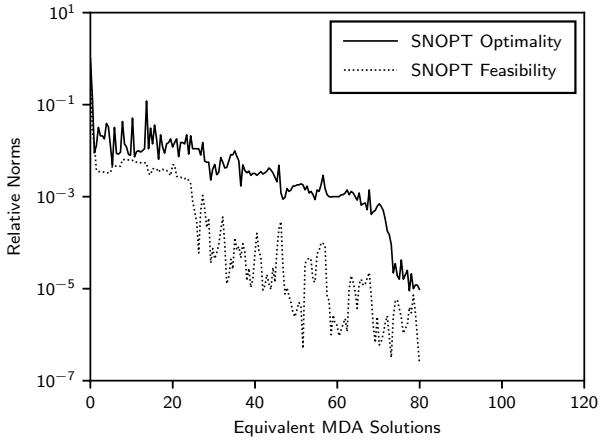


(b) Spanwise twist angle.

Fig. 4. Initial and optimal solutions for the aerostructural problem.



(a) RSNK on MDF and IDF formulations.



(b) SNOPT on the MDF formulation.

Fig. 5. Optimization convergence histories for the aerostructural problem.

VI. Conclusion

We solved a challenging large-scale aerostructural optimization problem using the reduced-space inexact Newton-Krylov (RSNK) algorithm implemented with both the multidisciplinary feasible (MDF) and individual discipline feasible (IDF) formulations. To the best of our knowledge, the present work is the first application of the IDF architecture to a high-fidelity aerostructural optimization problem.

Our results demonstrate that the RSNK algorithm can successfully solve a large-scale IDF problem at least as efficiently as its MDF counterpart. Additionally, the computational cost of the RSNK-IDF approach is comparable to the solution of the MDF problem with SNOPT [26], a quasi-Newton algorithm. This is a significant finding, since IDF offers valuable advantages in the modularity of the underlying PDE solvers and ease of implementation in the discipline coupling. Addressing the computational cost challenges associated with the added coupling variables and constraints enables the use of the IDF architecture as a practical and viable alternative to the commonly used MDF formulation.

We note that the matrix-free IDF preconditioner has been instrumental in achieving this result. We had previously established its efficacy using low-fidelity 2-D test problems featuring on the order of one hundred coupling constraints [4]. This aerostructural problem demonstrates that the IDF preconditioner remains effective with $8,400$ coupling constraints. This increases our confidence in the preconditioner's general applicability to high-fidelity, large-scale multidisciplinary optimization problems.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No.1332819. All computational work has been performed on the RSA cluster at the Center for Computational Innovations at Ressenlaer Polytechnic Institute. We gratefully acknowledge this support.

Appendix

A. Second-order Adjoint Formulation for KKT Matrix-Vector Products

We begin by recognizing that the KKT matrix in (2) is the Jacobian of the KKT conditions. Consequently, we can approximate products with this matrix and an arbitrary vector $z^T = (z_y^T, z_\lambda^T)$ via forward differencing on (1),

$$\mathsf{K}z = \begin{bmatrix} \mathsf{W} & \mathsf{A}^T \\ \mathsf{A} & 0 \end{bmatrix} \begin{pmatrix} z_y \\ z_\lambda \end{pmatrix} \approx \frac{1}{\epsilon} \begin{pmatrix} \mathcal{G}(y + \epsilon z_y, \lambda + \epsilon z_\lambda) - \mathcal{G}(y, \lambda) \\ C(y + \epsilon z_y) - C(y) \end{pmatrix}, \tag{5}$$

where $\mathcal{G} = d\mathcal{L}/dy$ is the total derivative of the Lagrangian with respect to the design variables and ϵ is the finite-difference step size.

In reduced-space formulations, the evaluation of the right-hand side in (5) involves the solution of nonlinear state equations as well as the first-order adjoint at the perturbed design point, $(y + \epsilon z_y)$. While the linear system solution associated with the first-order adjoint is unavoidable, we can adopt a second-order adjoint approach to replace the nonlinear state solution with a linear problem.

Let $(w + \epsilon\sigma)$ be the state corresponding the perturbed design point $(y + \epsilon z_y)$ such that

$$\mathcal{R}(y + \epsilon z_y, w + \epsilon\sigma) = \mathbf{0}. \tag{6}$$

For an infinitesimal perturbation ϵ, we can use a first-order Taylor expansion to expand the left -and side of (6) to arrive at

$$\mathcal{R}(y + \epsilon z_y, w + \epsilon\sigma) = \mathcal{R}(y, w) + \epsilon \frac{\partial \mathcal{R}}{\partial y} z_y + \epsilon \frac{\partial \mathcal{R}}{\partial w} \sigma + O(\epsilon^2) = \mathbf{0}. \tag{7}$$

Recognizing that $\mathcal{R}(y, w) = \mathbf{0}$ and taking the limit as $\epsilon \to 0$, we arrive at the linear system,

$$\frac{\partial \mathcal{R}}{\partial w} \sigma = -\frac{\partial \mathcal{R}}{\partial y} z_y, \tag{8}$$

where σ is a second-order adjoint, so-called because it helps us assemble second-derivative information associated with the KKT matrix.

We can repeat this process for the first-order adjoint residual,

$$\mathcal{S}(y, w, \lambda, \psi) = \frac{\partial \mathcal{J}}{\partial w} + \lambda^T \frac{\partial C}{\partial w} + \psi^T \frac{\partial \mathcal{R}}{\partial w} = \mathbf{0}, \tag{9}$$

and produce the linear system,

$$\left(\frac{\partial \mathcal{R}}{\partial w}\right)^T \phi = -\left(\frac{\partial C}{\partial w}\right)^T z_\lambda - \begin{bmatrix} \frac{\partial \mathcal{S}}{\partial y} & \frac{\partial \mathcal{S}}{\partial w} \end{bmatrix} \begin{pmatrix} z_y \\ \sigma \end{pmatrix}, \tag{10}$$

where ϕ is another second-order adjoint that defines the first-order adjoint $(\psi + \epsilon\phi)$ corresponding to the perturbed primal-dual point $(y + \epsilon z_y, \lambda + \epsilon z_\lambda)$.

Solving (8) and (10) yields all the information necessary to assemble the KKT matrix-vector product in (5), at a fixed cost of two PDE linear system solves, independently from the number of design variables and number of constraints in the optimization problem.

B. Complete RSNK Algorithm

Algorithm 1: Reduced-space inexact Newton-Krylov with filter globalization.

Data: $\boldsymbol{y}_0$, $\boldsymbol{\lambda}_0$, μ_0, η_0, Δ_0, $\Delta_{\min}$, $\Delta_{\max}$, τ_p, τ_d
Result: estimate for the optimal solution $(\boldsymbol{y}^*, \boldsymbol{\lambda}^*)$

1 set $\mu = \mu_0$, $\eta = \eta_0$, $\Delta = \Delta_0$
2 **for** $k = 0, 1, 2, \ldots,$ *max_iter* **do** `// start Newton loop`
3 compute KKT conditions $\nabla\mathcal{L}_k = (\mathcal{G}_k^T, C_k^T)^T$
4 **if** $\|\mathcal{G}_k\| \leq \tau_p\|\mathcal{G}_0\|$ *and* $\|C_k\| \leq \tau_d\|C_0\|$ **then** `// check convergence`
5 set $(\boldsymbol{y}^*, \boldsymbol{\lambda}^*) = (\boldsymbol{y}_k, \boldsymbol{\lambda}_k)$ and return
6 **end**
7 set $\mu \leftarrow \max\left[\mu,\ \mu_0\|C_0\|/\min\left(\|\mathcal{G}_k\|,\ \|C_k\|\right)\right]$
8 set $\eta \leftarrow \max\left[\eta,\ \min\left(1.0,\ \sqrt{\nabla\mathcal{L}_k/\nabla\mathcal{L}_0}\right),\ \min\left(\tau_p/\|\mathcal{G}_k\|,\ \tau_d/\|C_k\|\right)\right]$
9 solve (2) for $\boldsymbol{s}_k = (\boldsymbol{p}^T, \boldsymbol{d}^T)^T$ using FLECS, with tolerance η, penalty μ, and trust radius Δ
10 **for** $i = 0, 1, 2, \ldots,$ *max_filter_iter* **do** `// start filter loop`
11 **if** $[\mathcal{F}(\boldsymbol{y}_k + \boldsymbol{p}),\ \|C(\boldsymbol{y}_k + \boldsymbol{p})\|]$ *is not dominated by filter* **then** `// step accepted by filter`
12 **if** $i = 0$ *and* $\|\boldsymbol{p}\| = \Delta$ **then** set $\Delta \leftarrow \min(2\Delta, \Delta_{\max})$
13 set filter_success = true, and exit filter loop
14 **else** `// step rejected by filter`
15 **if** $i = 0$ **then**
16 compute second-order correction $\boldsymbol{p}_c$
17 **if** $[\mathcal{F}(\boldsymbol{y}_k + \boldsymbol{p} + \boldsymbol{p}_c),\ \|C(\boldsymbol{y}_k + \boldsymbol{p} + \boldsymbol{p}_c)\|]$ *is not dominated by filter* **then**
18 set filter_success = true, set $\boldsymbol{p} \leftarrow \boldsymbol{p} + \boldsymbol{p}_c$, and exit filter loop
19 **end**
20 **end**
21 **if** $\Delta = \Delta_{\min}$ **then** optimization failed, raise error
22 set $\Delta \leftarrow \max(\Delta_{\min}, \frac{1}{4}\Delta)$, and re-solve (2) for $\boldsymbol{s}_k$ using FLECS with updated trust radius Δ
23 **end**
24 **end**
25 **if** *filter_success* **then** set $\boldsymbol{y}_{k+1} = \boldsymbol{y}_k + \boldsymbol{p}$ and $\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \boldsymbol{d}$
26 **else** set $\boldsymbol{y}_{k+1} = \boldsymbol{y}_k$ and $\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k$
27 **end**

Table 2. Optimization parameters for the aerostructural test case used in Alg. 1.

Parameter	Value
relative optimality tolerance (line 4), τ_p	10^{-5}
relative feasibility tolerance (line 4), τ_d	10^{-5}
initial penalty parameter (line 1), μ_0	1.0
initial Krylov tolerance (line 1), η_0	0.5
Krylov subspace size(line 9)	20
initial trust radius (lines 1), Δ_0	0.5
minimum trust radius (lines 21-22), $\Delta_{\min}$	10^{-4}
maximum trust radius (line 12), $\Delta_{\max}$	4.0

References

[1] Haftka, R. T., Sobieszczanski-Sobieski, J., and Padula, S. L., "On options for interdisciplinary analysis and design optimization," *Structural Optimization*, Vol. 4, No. 2, 1992, pp. 65–74.

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

[3] Martins, J. R. R. A., and Lambe, A. B., "Multidisciplinary design optimization: a survey of architectures," *AIAA Journal*, Vol. 51, No. 9, 2013, pp. 2049–2075.

[4] Dener, A., and Hicken, J. E., "Matrix-free algorithm for the optimization of multidisciplinary systems," *Structural and Multidisciplinary Optimization*, Vol. 56, 2017, pp. 1429–1446.

[5] Maute, K., Nikbay, M., and Farhat, C., "Coupled analytical sensitivity analysis and optimization of three-dimensional nonlinear aeroelastic systems," *AIAA Journal*, Vol. 39, No. 11, 2001, pp. 2051–2061.

[6] Maute, K., Nikbay, M., and Farhat, C., "Sensitivity analysis and design optimization of three-dimensional non-linear aeroelastic systems by the adjoint method," *International Journal for Numerical Methods in Engineering*, Vol. 56, No. 6, 2003, pp. 911–933.

[7] Martins, J. R. R. A., Alonso, J. J., and Reuther, J. J., "High-fidelity aerostructural design optimization of a supersonic business jet," *Journal of Aircraft*, Vol. 41, No. 3, 2004, pp. 523–530.

[8] Kenway, G. K., Kennedy, G. J., and Martins, J. R. R. A., "Scalable parallel approach for high-fidelity steady-state aeroelastic analysis and adjoint derivative computations," *AIAA Journal*, Vol. 52, No. 5, 2014, pp. 935–951.

[9] Kennedy, G. J., and Martins, J. R. R. A., "Parallel solution methods for aerostructural analysis and design optimization," *13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*, 2010, p. 9308.

[10] Ta'asan, S., Kuruvila, G., and Salas, M., "Aerodynamic design and optimization in one shot," *30th Aerospace Sciences Meeting and Exhibit*, 1992, p. 25.

[11] Herskovits, J., Mappa, P., Goulart, E., and Soares, C. M., "Mathematical programming models and algorithms for engineering design optimization," *Computer Methods in Applied Mechanics and Engineering*, Vol. 194, No. 30, 2005, pp. 3244–3268.

[12] Biros, G., and Ghattas, O., "Parallel Lagrange–Newton–Krylov–schur methods for PDE-constrained optimization, part I: the Krylov–Schur solver," *SIAM Journal on Scientific Computing*, Vol. 27, No. 2, 2005, pp. 687–713.

[13] Biros, G., and Ghattas, O., "Parallel Lagrange–Newton–Krylov–schur methods for PDE-constrained optimization, part II: the Lagrange–Newton solver and its application to optimal control of steady viscous flows," *SIAM Journal on Scientific Computing*, Vol. 27, No. 2, 2005, pp. 714–739.

[14] Özkaya, E., and Gauger, N. R., "Single-step one-shot aerodynamic shape optimization," *Optimal Control of Coupled Systems of Partial Differential Equations*, Springer, 2009, pp. 191–204.

[15] Dener, A., Kenway, G. K., Hicken, J. E., and Martins, J. R. R. A., "Comparison of inexact-and quasi-newton algorithms for aerodynamic shape optimization," *53rd AIAA Aerospace Sciences Meeting*, 2015, p. 1945.

[16] Lions, J.-L., *Contrôle optimal de systèmes gouvernés par des équations aux dérivées partielles*, Etudes Mathématiques, Dunod; Gauthier-Villars, Collier-Macmillan Ltd., 1968.

[17] Jameson, A., "Aerodynamic design via control theory," *Recent Advances in Computational Fluid Dynamics*, Springer, 1989, pp. 377–401.

[18] Wang, Z., Navon, I. M., Le Dimet, F., and Zou, X., "The second order adjoint analysis: theory and applications," *Meteorology and Atmospheric Physics*, Vol. 50, No. 1-3, 1992, pp. 3–20.

[19] Wang, Z., Navon, I., Zou, X., and Le Dimet, F., "A truncated Newton optimization algorithm in meteorology applications with analytic Hessian/vector products," *Computational Optimization and Applications*, Vol. 4, No. 3, 1995, pp. 241–262.

[20] Dener, A., and Hicken, J. E., "Revisiting individual discipline feasible using matrix-free inexact-Newton-Krylov," *10th AIAA Multidisciplinary Design Optimization Conference, American Institute of Aeronautics and Astronautics*, 2014, pp. 2014–0110.

[21] Saad, Y., and Schultz, M. H., "GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems," *SIAM Journal on Scientific and Statistical Computing*, Vol. 7, No. 3, 1986, pp. 856–869.

[22] Hicken, J. E., and Dener, A., "A flexible iterative solver for nonconvex, equality-constrained quadratic subproblems," *SIAM Journal on Scientific Computing*, Vol. 37, No. 4, 2015, pp. A1801–A1824.

[23] Conn, A. R., Gould, N. I., and Toint, P. L., *Trust Region Methods*, SIAM, 2000.

[24] Fletcher, R., and Leyffer, S., "Nonlinear programming without a penalty function," *Mathematical Programming*, Vol. 91, No. 2, 2002, pp. 239–269.

[25] Benzi, M., Golub, G. H., and Liesen, J., "Numerical solution of saddle point problems," *Acta Numerica*, Vol. 14, 2005, pp. 1–137.

[26] Gill, P. E., Murray, W., and Saunders, M. A., "SNOPT: an SQP algorithm for large-scale constrained optimization," *SIAM Journal on Optimization*, Vol. 12, No. 4, 2002, pp. 979–1006.

[27] Iuliano, E., and Quagliarella, D., "Proper orthogonal decomposition, surrogate modelling and evolutionary optimization in aerodynamic design," *Computers & Fluids*, Vol. 84, 2013, pp. 327–350.

[28] Kreisselmeier, G., and Steinhauser, R., "Systematic control design by optimizing a vector performance index," *IFAC Proceedings Volumes*, Vol. 12, No. 7, 1979, pp. 113–117.

[29] Raspanti, C., Bandoni, J., and Biegler, L., "New strategies for flexibility analysis and design under uncertainty," *Computers & Chemical Engineering*, Vol. 24, No. 9, 2000, pp. 2193–2209.

[30] Akgun, M. A., Haftka, R. T., Wu, K. C., Walsh, J. L., and Garcelon, J. H., "Efficient structural optimization for multiple load cases using adjoint sensitivities," *AIAA Journal*, Vol. 39, No. 3, 2001, pp. 511–516.

[31] Kennedy, G. J., and Hicken, J. E., "Improved constraint-aggregation methods," *Computer Methods in Applied Mechanics and Engineering*, Vol. 289, No. 0, 2015, pp. 332 – 354.

[32] Lambe, A. B., Martins, J. R. R. A., and Kennedy, G. J., "An evaluation of constraint aggregation strategies for wing box mass minimization," *Structural and Multidisciplinary Optimization*, Vol. 55, No. 1, 2017, pp. 257–277.

[33] van der Weide, E., Kalitzin, G., Schluter, J., and Alonso, J., "Unsteady turbomachinery computations using massively parallel platforms," *44th AIAA Aerospace Sciences Meeting and Exhibit*, American Institute of Aeronautics and Astronautics, Reston, Virigina, 2012.

[34] Luke, E., Collins, E., and Blades, E., "A fast mesh deformation method using explicit interpolation," *Journal of Computational Physics*, Vol. 231, No. 2, 2012, pp. 586–601.

[35] Kennedy, G. J., and Martins, J. R. R. A., "A parallel finite-element framework for large-scale gradient-based design optimization of high-performance structures," *Finite Elements in Analysis and Design*, Vol. 87, 2014, pp. 56–73.

[36] Irons, B. M., and Tuck, R. C., "A version of the Aitken accelerator for computer iteration," *International Journal for Numerical Methods in Engineering*, Vol. 1, No. 3, 1969, pp. 275–277.

[37] Brown, S., "Displacement extrapolation for CFD+ CSM aeroelastic analysis," *AIAA Journal*, Vol. 1090, 1997, p. 1997.

[38] Saad, Y., "A flexible inner-outer preconditioned GMRES algorithm," *SIAM Journal on Scientific Computing*, Vol. 14, No. 2, 1993, pp. 461–469.