

# An Efficient Monolithic Solution Algorithm for High-Fidelity Aerostructural Analysis and Optimization

Zimi J. Zhang\* and David W. Zingg†

*Institute for Aerospace Studies, University of Toronto*

*4925 Dufferin Street, Toronto, Ontario, M3H 5T6, Canada*

An efficient and robust solution algorithm for the aerostructural analysis and coupled adjoint problems is crucial to the success of high-fidelity aerostructural optimization. The objective of the present paper is to investigate ways to maximize the efficiency of a monolithic solution method and further quantify its benefits in the context of aerostructural optimization. A Newton-Krylov method is used for the aerostructural analysis, and a preconditioned Krylov subspace method is used for the coupled adjoint solution. Several aspects of the monolithic solution method have been investigated. These include appropriate strategies for scaling and matrix-vector product evaluations, as well as block Jacobi and block Gauss-Seidel preconditioning techniques that preserve the modularity between subproblems. The monolithic solution method is applied to problems with varying degrees of fluid-structure coupling, as well as a wing span optimization study. In most cases, the monolithic solution algorithm requires 20%–70% less computing time than its partitioned counterpart. This advantage increases with increasing wing flexibility. Robustness of the monolithic solution method is shown via its reduced sensitivity to the choice of problem dependent solution parameters, as well as its ability to converge when the partitioned method fails.

---

\*Ph.D. Candidate, AIAA Student Member, [jennyzm.zhang@utoronto.ca](mailto:jennyzm.zhang@utoronto.ca)

†University of Toronto Distinguished Professor of Computational Aerodynamics and Sustainable Aviation, Director, Centre for Research in Sustainable Aviation, Associate Fellow AIAA, [dwz@oddjob.utias.utoronto.ca](mailto:dwz@oddjob.utias.utoronto.ca)

## Nomenclature

$\mathbb{A}$	Jacobian of the aerostructural residual	$\hat{\mathbf{b}}_{\Delta}$	Locally scaled mesh state
$\mathbf{b}_{\Delta}$	Mesh state vector	$c_{\text{scl},*}$	Jacobian block column scaling
$E$	Young's modulus	$\mathbf{R}_{\text{AS}}$	Aerostructural residual vector
$\mathbf{f}_{\text{M}\Delta}$	Force vector in the mesh equation	$\mathbf{R}_{\text{A}}$	Aerodynamic residual vector
$\mathbf{f}_{\text{S}}$	Force vector in the structural equation	$\hat{\mathbf{R}}_{\text{A}}$	Locally scaled aerodynamic residual
$\mathbf{K}_{\text{M}\Delta}$	Stiffness matrix for the mesh equation	$\mathbf{R}_{\text{M}\Delta}$	Mesh residual vector for the deflected shape
$\hat{\mathbf{K}}_{\text{M}\Delta}$	Locally scaled mesh stiffness matrix	$\hat{\mathbf{R}}_{\text{M}\Delta}$	Locally scaled mesh residual
$\mathbf{K}_{\text{S}}$	Stiffness matrix for the structural equation	$\mathbf{R}_{\text{S}}$	Structural residual vector
$m_{\Delta}$	Number of mesh movement increments used for aerostructural analysis	$r_{\text{scl},*}$	Jacobian block row scaling
$\mathbf{M}_{\text{A}}$	Flow block preconditioner	$\mathbf{u}$	Structural state vector
$\mathbf{M}_{\text{M}}$	Mesh block preconditioner	$\beta$	Scalar parameter used to vary emphasis between weight and drag in the objective
$\mathbf{M}_{\text{S}}$	Structural block preconditioner	$\Psi_{\text{A}}$	Aerodynamic adjoint vector
$n$	Newton or partitioned adjoint iteration index	$\Psi_{\text{M}\Delta}$	Mesh adjoint vector corresponding to the deflected geometry
$q_{\infty}$	Freestream dynamic pressure	$\Psi_{\text{S}}$	Structural adjoint vector
$\mathbf{q}$	Flow state vector	$\theta$	Under-relaxation factor

## I. Introduction

Numerical optimization based on high-fidelity analysis is becoming a valuable tool during the preliminary and detailed design phases of new aircraft. Although computationally more expensive, high-fidelity analysis can capture nonlinear effects under conditions where low-fidelity models may be inaccurate. This allows for more reliable assessments of novel design features even in the absence of the designers' past experience. Optimization based on tightly integrated high-fidelity aerostructural analysis is particularly useful because many objectives and constraints relevant to aircraft design include both aerodynamic and structural functionals. Performing high-fidelity aerostructural analysis allows the values of these functionals to be determined accurately by including the effects of structural deflections and a full stress analysis based on aerodynamic loading. Thus aerodynamic efficiency improvements can be studied in relation to the possible increase in structural weight. A more rigorous prediction of structural failure can also provide more confidence in the feasibility of the design.

Numerous high-fidelity aerostructural optimization methodologies have been proposed in the past, with increasing levels of sophistication to solve complex design problems involving realistic flow features, detailed

structural models, and large numbers of load conditions as well as design variables.<sup>1–9</sup> Many of these methodologies take advantage of gradient-based optimization and adjoint methods for gradient calculations to make the problem more computationally tractable.<sup>10–12</sup> An efficient solution strategy for the coupled problem is another aspect of the algorithm design that is crucial for minimizing the cost. Robustness of the coupled solution algorithm is also needed to handle highly variable and unpredictable designs given the large number of design variables that is necessary for the exploration of novel geometric design features. The design of effective coupled solution strategies has been an active area of research for general fluid-structure interaction (FSI) problems. However, there are fewer such studies in the specific context of high-fidelity aerostructural optimization.

The partitioned approach is used most extensively in past research on aerostructural analysis and optimization.<sup>1–5</sup> Partitioned methods, also referred to as segregated or staggered methods, allow each of the flow, structures, and mesh equations in the coupled analysis and adjoint problems to be solved using existing routines within the respective modules. This reduces implementation cost and takes advantage of the software routines designed specifically for each subproblem. However, the performance of partitioned methods tends to deteriorate as the interaction between the flow and the structures intensifies.<sup>6,13–17</sup> Convergence becomes heavily dependent on the choice of relaxation parameter,<sup>6,13</sup> which is difficult to determine *a priori*. Even when convergence can be achieved with a conservative relaxation parameter, many iterations are often necessary, leading to high computational times especially with the use of high-fidelity flow and structural analysis. The efficiency of partitioned methods can be improved using a number of techniques,<sup>18–21</sup> but monolithic solution methods also provide a promising alternative.<sup>6,13–17,22</sup>

In this work, monolithic solution methods refer to iterative methods that fully couple variables across all disciplines. In other words, they do not iterate between subproblem solutions, unlike partitioned methods. Some of the literature on FSI has also used the phrase *monolithic* to refer to the development of new methodologies specifically for the purpose of solving coupled FSI problems, starting from the governing partial differential equations.<sup>17,23–28</sup> This is in contrast to a *modular* approach which can be constructed from software modules that are independently developed for each discipline. Although modularity is often associated with partitioned solution techniques, it is possible to develop solution methods that are both monolithic and modular.<sup>6,15,16,29,30</sup> Having made this clarification, the relevant monolithic solution methods for general steady and unsteady FSI analysis are reviewed below, followed by studies and techniques applied to aerostructural optimization.

Heil<sup>31</sup> and Heil *et al.*<sup>15</sup> showed that a monolithic method employing a Newton-Krylov solution algorithm performs well for steady and unsteady FSI simulations. The authors further demonstrated the importance of an effective preconditioner for the coupled linear system solution using the Generalized Minimal Residual

(GMRES) method at each Newton iteration. There is a modular approach that allows for the reuse of existing linear solvers specialized for the flow and structural subproblems. A block Gauss-Seidel preconditioner was shown to be more effective than a block Jacobi preconditioner.<sup>31</sup> The monolithic method is effective in strongly coupled FSI simulations where partitioned methods fail to converge, and it remains competitive when the coupling is weak.<sup>15</sup> Barker and Cai<sup>32</sup> and Wu and Cai<sup>17</sup> proposed a parallel non-modular monolithic solution strategy for two-dimensional and three-dimensional problems, respectively. A restricted additive Schwarz preconditioner is used with local incomplete point lower upper (LU) factorization which does not distinguish between the flow and structural subproblems. It was shown to perform well with thousands of processors in blood flow simulations where partitioned methods are known to have issues.<sup>17</sup> Crosetto *et al.*<sup>30</sup> further compared the performance of a modular block Gauss-Seidel preconditioner and a nonmodular application of the additive Schwarz preconditioner. The modular approach is more effective both in factorization time and the resulting number of GMRES iterations, as the number of processors is increased.

More advanced preconditioning techniques have been proposed which can better resolve the fluid-structure coupling than a block Jacobi or block Gauss-Seidel preconditioner. Some examples include the preconditioning method by Badia *et al.*<sup>33</sup> based on an inexact block LU factorization of the coupled Jacobian matrix, a block iterative preconditioner formulated using the Robin transmission conditions,<sup>29</sup> and another alternative by Gee *et al.*<sup>16</sup> based on algebraic multigrid. Although the above methodologies<sup>16,29,33</sup> have demonstrated improvements in comparison to a block Jacobi or block Gauss-Seidel preconditioner, their implementation is less trivial for problems of interest to this work. The applications considered also involve much stronger fluid-structure coupling than typically encountered in the aerostructural optimization of aircraft. Therefore, a further increase in the complexity of the coupled solution strategy may not be necessary.

In the context of aerostructural optimization, Elham *et al.*<sup>34</sup> applied the Newton method to the aerostructural analysis in a mid-fidelity optimization methodology. Kennedy and Martins<sup>7,35</sup> proposed a modular monolithic solution technique in coupling a high-fidelity linear structural analysis code with a panel method for aerodynamic analysis. A two-field formulation is used. The coupled linear system at each Newton iteration during analysis, as well as the coupled adjoint system, are solved via a flexible variant of GMRES (FGMRES). A block Jacobi preconditioner was proposed that reuses the distributed linear solvers in the flow and structural modules. The monolithic solution method demonstrates good parallel scalability up to 96 processors. A similar monolithic solution strategy was investigated for high-fidelity aerostructural optimization by Kenway *et al.*<sup>6</sup> The same structural analysis methodology is coupled to the three-dimensional flow analysis based on the Euler equations. The Jacobian-free Newton-Krylov algorithm for the aerostructural analysis is more efficient than the partitioned method by 9% and 25% for a  $1g$  and a  $2.5g$  load condition,

respectively. The monolithic adjoint calculations are more efficient by 19% and 29%, respectively, for the two load conditions.

The Schur-Newton-Krylov (SNK) method proposed by Barcelos *et al.*<sup>13</sup> for steady-state aerostructural analysis and optimization is an interface approach. Each matrix-vector product evaluation with the interface Jacobian matrix requires a linear system solution to each of the flow, mesh, and structural subproblems in a three-field formulation. The linearized interface problem is solved by GMRES without preconditioning. For the analysis of a stiff wing, the SNK approach requires a smaller number of iterations but almost twice as much processing time as the partitioned method. Applying the SNK approach to gradient calculations using a direct method is faster than the partitioned method by more than 30%. For a highly flexible wing, the SNK algorithm outperforms the partitioned method for both analysis and sensitivity calculations.

In contrast to the large number of studies demonstrating the advantages of monolithic solution methods for general FSI problems, only a handful of authors have examined their application to high-fidelity aerostructural optimization. The reported computational savings from using monolithic solution methods have been between 10% to 30% for wings with realistic deflections.<sup>6,13</sup> The results are promising, but it remains unclear whether these numbers represent the maximum possible efficiency improvements from the monolithic approach. The present paper therefore has two objectives. The first objective is to improve the efficiency of the monolithic approach for high-fidelity aerostructural optimization. This is accomplished by investigating an effective monolithic solution strategy for the methodology of Zhang *et al.*,<sup>9</sup> which uses an integrated geometry parameterization and mesh movement algorithm<sup>36</sup> to provide an analytical geometry representation while enabling efficient mesh movement for very large shape changes during aerostructural optimization. The present monolithic solution method is applied to the analysis and coupled adjoint problems using a three-field formulation without reducing them to the interface degrees of freedom. This allows the Jacobian matrix-vector products to be evaluated efficiently without needing to perform a mesh solution during each coupled Krylov iteration. An explicit treatment of the mesh subproblem leads to effective block iterative preconditioners that capture the effects of the flow grid deformation efficiently.<sup>37</sup> The block Gauss-Seidel preconditioner is studied and compared with a block Jacobi preconditioner. The second objective of this work is to quantify the benefits of the monolithic method in comparison with a partitioned method and their dependence on the nature of the problem. To this end, performance of the monolithic and partitioned methods is examined for several wing optimization problems with varying degrees of fluid-structure coupling.

## II. Aerostructural Optimization Methodology

The discrete steady aerostructural analysis and the coupled adjoint problems are summarized here for the methodology by Zhang *et al.*<sup>9</sup> The remaining details can be found in the cited reference. A three-field

formulation is adopted for the analysis. The mesh equations,  $\mathbf{R}_{M\Delta}$ , appear explicitly in the aerostructural residual,  $\mathbf{R}_{AS}$ , along with the flow equations,  $\mathbf{R}_A$ , and the structural equations,  $\mathbf{R}_S$ :

$$\mathbf{R}_{AS} = \begin{bmatrix} \mathbf{R}_A(\mathbf{q}, \mathbf{b}_\Delta) \\ \mathbf{R}_{M\Delta}(\mathbf{b}_\Delta, \mathbf{u}) \\ \mathbf{R}_S(\mathbf{q}, \mathbf{b}_\Delta, \mathbf{u},) \end{bmatrix} = 0. \quad (1)$$

The flow, mesh, and structural state variables are denoted by  $\mathbf{q}$ ,  $\mathbf{b}_\Delta$ , and  $\mathbf{u}$ . The flow equations are given by the Euler equations governing three-dimensional compressible inviscid flow, which are discretized on the multi-block structured mesh using second-order summation-by-parts finite-difference operators.<sup>38</sup> The state of the flow grid deformation due to structural deflections is governed by  $\mathbf{R}_{M\Delta}$ . The mesh movement calculation can be broken into  $m_\Delta$  increments in the presence of large deflections. This translates to solving the following  $m_\Delta$  linear-elasticity mesh movement equations:

$$\mathbf{R}_{M\Delta}^{(i)}(\mathbf{b}_\Delta^{(i-1)}, \mathbf{b}_\Delta^{(i)}) = \mathbf{K}_{M\Delta}^{(i)}(\mathbf{b}_\Delta^{(i-1)})[\mathbf{b}_\Delta^{(i)} - \mathbf{b}_\Delta^{(i-1)}] - \mathbf{f}_{M\Delta}^{(i)}(\mathbf{u}) = 0, \quad \text{for } i = 1, \dots, m_\Delta. \quad (2)$$

The mesh state variables,  $\mathbf{b}_\Delta$ , are the control point coordinates of the B-spline volumes that parameterize and describe the computational grid for the flow calculations. The implicit force vector,  $\mathbf{f}_{M\Delta}$ , is determined based on changes in the geometry as a function of  $\mathbf{u}$ . The mesh stiffness matrix,  $\mathbf{K}_{M\Delta}^{(i)}$ , is a function of  $\mathbf{b}_\Delta^{(i-1)}$ . The mesh residual,  $\mathbf{R}_{M\Delta}$ , refers to the vector containing all the incremental mesh residuals, i.e.  $\mathbf{R}_{M\Delta} = [\mathbf{R}_{M\Delta}^{(1)}, \mathbf{R}_{M\Delta}^{(2)}, \dots, \mathbf{R}_{M\Delta}^{(m_\Delta)}]^T$ . A similar notation is used for  $\mathbf{b}_\Delta$  and  $\mathbf{f}_{M\Delta}$ . This also suggests a necessary shift in perspective. Although the incremental mesh states are normally updated in a sequential manner, it is important to recognize that they can be modified simultaneously during a monolithic solution process. For a linear finite element analysis, the structural equations are given by<sup>7,39</sup>

$$\mathbf{R}_S(\mathbf{q}, \mathbf{u}, \mathbf{b}_\Delta) = \mathbf{K}_S \mathbf{u} - \mathbf{f}_S(\mathbf{q}, \mathbf{b}_\Delta) = 0. \quad (3)$$

The force vector,  $\mathbf{f}_S$ , is a result of aerodynamic loading which depends on both  $\mathbf{q}$  and  $\mathbf{b}_\Delta$ . Finally, the flow, mesh, and structural equations are coupled by the transfer of forces and displacements.<sup>7</sup>

A partitioned nonlinear block Gauss-Seidel solution to (1), which involves sub-iterations using existing solution routines within each module, is used as a basis for comparison with the monolithic solution technique. More specifically, an update to  $\mathbf{q}$  given  $\mathbf{b}_\Delta$  is obtained by solving  $\mathbf{R}_A$  using a parallel implicit Newton-Krylov-Schur algorithm.<sup>38</sup> Solution to  $\mathbf{R}_{M\Delta}^{(i)}$  at each increment is computed via a parallel conjugate gradient algorithm with an additive Schwarz preconditioner and local zero-fill incomplete lower-upper (ILU) factorization. Given  $\mathbf{f}_S$  at each partitioned iteration, an update to  $\mathbf{u}$  is obtained using a parallel direct fac-

torization of  $\mathbf{K}_S$  with GMRES for the iterative refinement of the direct solution.<sup>39</sup> The direct factorization of  $\mathbf{K}_S$  is computed at the start of the aerostructural analysis. Aitken acceleration<sup>40, 41</sup> is used to improve the stability and convergence of the partitioned analysis.

Gradient evaluation at each design iteration requires the solution to the following coupled adjoint problem:

$$\underbrace{\begin{bmatrix} \frac{\partial \mathbf{R}_A}{\partial \mathbf{q}} & 0 & \frac{\partial \mathbf{R}_S}{\partial \mathbf{q}} \\ \frac{\partial \mathbf{R}_A}{\partial \mathbf{b}_\Delta} & \frac{\partial \mathbf{R}_{M\Delta}}{\partial \mathbf{b}_\Delta} & \frac{\partial \mathbf{R}_S}{\partial \mathbf{b}_\Delta} \\ 0 & \frac{\partial \mathbf{R}_{M\Delta}}{\partial \mathbf{u}} & \frac{\partial \mathbf{R}_S}{\partial \mathbf{u}} \end{bmatrix}}_{\mathbb{A}^T} \begin{bmatrix} \Psi_A \\ \Psi_{M\Delta} \\ \Psi_S \end{bmatrix} = \begin{bmatrix} -\frac{\partial \mathcal{J}}{\partial \mathbf{q}} \\ -\frac{\partial \mathcal{J}}{\partial \mathbf{b}_\Delta} \\ -\frac{\partial \mathcal{J}}{\partial \mathbf{u}} \end{bmatrix}, \quad (4)$$

where the flow adjoint,  $\Psi_A$ , the mesh adjoint for the deflected shape,  $\Psi_{M\Delta}$ , and the structural adjoint,  $\Psi_S$ , are coupled by the transposed Jacobian of  $\mathbf{R}_{AS}$ ,  $\mathbb{A}^T$ , on the left-hand side. The partitioned solution to the coupled adjoint problem in (4) is achieved via a linear block Gauss-Seidel method. Each iteration solves the following equations in sequence:

$$\frac{\partial \mathbf{R}_A}{\partial \mathbf{q}} \Psi_A^{(n+1)} = -\frac{\partial \mathcal{J}}{\partial \mathbf{q}} - \frac{\partial \mathbf{R}_S}{\partial \mathbf{q}} \Psi_S^{(n)} \quad (5)$$

$$\frac{\partial \mathbf{R}_{M\Delta}}{\partial \mathbf{b}_\Delta} \Psi_{M\Delta}^{(n+1)} = -\frac{\partial \mathcal{J}}{\partial \mathbf{b}_\Delta} - \frac{\partial \mathbf{R}_A}{\partial \mathbf{b}_\Delta} \Psi_A^{(n+1)} - \frac{\partial \mathbf{R}_S}{\partial \mathbf{b}_\Delta} \Psi_S^{(n)}, \quad (6)$$

$$\mathbf{K}_S \Psi_S^{(n+1)} = -\frac{\partial \mathcal{J}}{\partial \mathbf{u}} - \frac{\partial \mathbf{R}_{M\Delta}}{\partial \mathbf{u}} \Psi_{M\Delta}^{(n+1)} \quad (7)$$

where  $n$  is the iteration index. The solution to (5) is obtained using GCROT( $m, k$ )<sup>42</sup> with an approximate Schur preconditioner. For mesh movement calculations performed in  $m_\Delta$  increments, (6) further consists of  $m_\Delta$  mesh adjoint equations solved in reverse order. Due to the symmetry in the mesh stiffness matrix,  $\mathbf{K}_{M\Delta}^{(i)}$ , at each increment and the structural stiffness matrix,  $\mathbf{K}_S$ , each mesh adjoint equation in (6) and the structural adjoint equation in (7) are solved in the same manner as for the aerostructural analysis. An appropriate relaxation factor is often necessary when updating the structural adjoint variable in (7) to ensure the robustness of the partitioned calculations. It will be shown later that the choice of relaxation factor is highly problem dependent, which is a disadvantage of the partitioned method.

### III. Newton-Krylov Solution to the Analysis Problem

Applying the Newton method to (1) requires the solution of the following linear system:

$$\frac{\partial \mathbf{R}_{AS}^{(n)}}{\partial [\mathbf{q}, \mathbf{b}_\Delta, \mathbf{u}]^{(n)}} [\Delta \mathbf{q}, \Delta \mathbf{b}_\Delta, \Delta \mathbf{u}]^{(n)T} = \mathbb{A}^{(n)} \delta^{(n)} = -\mathbf{R}_{AS}^{(n)}, \quad (8)$$

where  $\mathbb{A}$  is the Jacobian of  $\mathbf{R}_{AS}$ . The solution to (8) is used to update  $\mathbf{q}$ ,  $\mathbf{b}_\Delta$ , and  $\mathbf{u}$  simultaneously. Equation (8) is solved iteratively to some tolerance such that

$$\frac{\|\mathbb{A}^{(n)}\boldsymbol{\delta}^{(n)} + \mathbf{R}_{AS}^{(n)}\|}{\|\mathbf{R}_{AS}^{(n)}\|} \leq \eta^{(n)}, \quad \eta \in [0, 1), \quad (9)$$

where  $\eta^{(n)}$  is the forcing parameter. An appropriate initial guess for the Newton method is found using the partitioned method with Aitken acceleration to reduce the norms of  $\mathbf{R}_A$ ,  $\mathbf{R}_{M\Delta}$ , and  $\mathbf{R}_S$  below a relative tolerance of 0.05. This work focuses on the application of the Newton method to (8), which leads to a monolithic method for aerostructural analysis. It should be noted that the start-up phase of the monolithic analysis could suffer from some of the disadvantages of a fully partitioned analysis. While a potentially more effective way to globalize the Newton method is left for future work, using the Newton method during the latter phase of the calculations provides efficiency and robustness benefits, as will be shown below. It has been found that a fixed value of  $\eta^{(n)} = 0.01$  is more effective in minimizing CPU time than an adaptive forcing parameter such as the one recommended by Eisenstat and Walker.<sup>43</sup> FGMRES is used for the solution to (8). This allows for a non-stationary preconditioner that reuses existing linear solution routines from the flow, mesh, and structural modules.

### III.A. Scaling

During a Newton-Krylov solution procedure, it is important to ensure that the linear system is well-scaled. Failing to do so can affect the Newton convergence.<sup>44</sup> Scaling can additionally improve the condition number of the linear system and the accuracy of the iterative solution.<sup>45,46</sup> In the present algorithm, the coupled linear system at every Newton iteration is scaled as follows:

$$\underbrace{\begin{bmatrix} r_{\text{scl},A} \left( \frac{\partial \hat{\mathbf{R}}_A}{\partial \mathbf{q}} \right) c_{\text{scl},A} & r_{\text{scl},A} \left( \frac{\partial \hat{\mathbf{R}}_A}{\partial \hat{\mathbf{b}}_\Delta} \right) c_{\text{scl},M} & 0 \\ 0 & r_{\text{scl},M} \left( \frac{\partial \hat{\mathbf{R}}_{M\Delta}}{\partial \hat{\mathbf{b}}_\Delta} \right) c_{\text{scl},M} & r_{\text{scl},M} \left( \frac{\partial \hat{\mathbf{R}}_{M\Delta}}{\partial \mathbf{u}} \right) c_{\text{scl},S} \\ r_{\text{scl},S} \left( \frac{\partial \mathbf{R}_S}{\partial \mathbf{q}} \right) c_{\text{scl},A} & r_{\text{scl},S} \left( \frac{\partial \mathbf{R}_S}{\partial \hat{\mathbf{b}}_\Delta} \right) c_{\text{scl},M} & r_{\text{scl},S} \left( \frac{\partial \mathbf{R}_S}{\partial \mathbf{u}} \right) c_{\text{scl},S} \end{bmatrix}}_{\hat{\mathbb{A}}} \underbrace{\begin{bmatrix} c_{\text{scl},A}^{-1} \Delta \mathbf{q} \\ c_{\text{scl},M}^{-1} \Delta \hat{\mathbf{b}}_\Delta \\ c_{\text{scl},S}^{-1} \Delta \mathbf{u} \end{bmatrix}}_{\hat{\boldsymbol{\delta}}} = \underbrace{\begin{bmatrix} -r_{\text{scl},A} \hat{\mathbf{R}}_A \\ -r_{\text{scl},M} \hat{\mathbf{R}}_{M\Delta} \\ -r_{\text{scl},S} \mathbf{R}_S \end{bmatrix}}_{-\hat{\mathbf{R}}_{AS}}, \quad (10)$$

where  $\hat{\mathbb{A}}\hat{\boldsymbol{\delta}} = -\hat{\mathbf{R}}_{AS}$  is the new linear system to be solved. Two types of scaling are applied in (10). Equations and variables in the flow and mesh subproblems are scaled using existing routines tailored to the respective equation within the flow and mesh modules. The scaled results are denoted by the  $\hat{\cdot}$  symbol. The 9 Jacobian blocks in  $\mathbb{A}$  are additionally scaled using scalars  $r_{\text{scl},*}$  and  $c_{\text{scl},*}$ . The importance of such Jacobian block scaling is demonstrated in Figure 1 using the wing optimization problem described in Appendix A. The

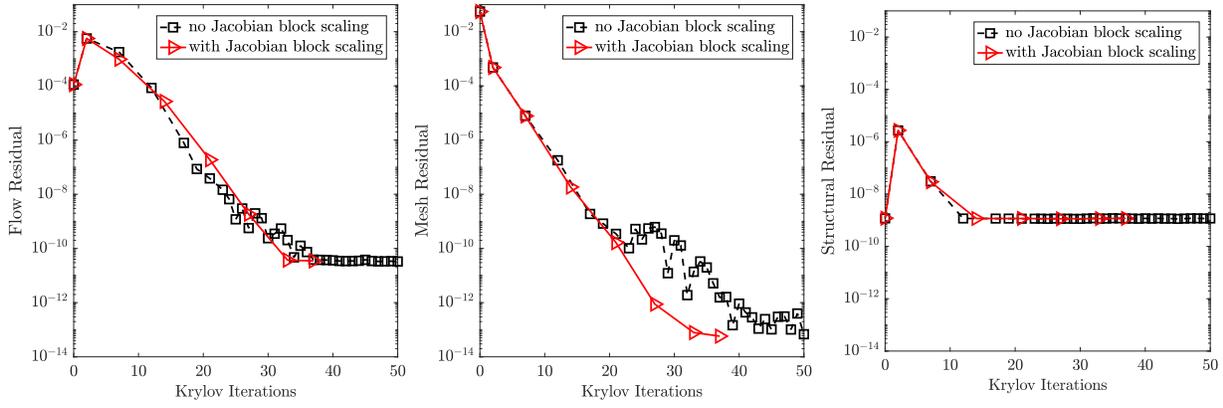


Figure 1: Newton convergence of  $\|\mathbf{R}_A\|_2$ ,  $\|\mathbf{R}_{M\Delta}\|_2$ , and  $\|\mathbf{R}_S\|_2$  for the problem described in Appendix A with and without the Jacobian block scaling in  $\hat{\mathbf{A}}$ . A block Gauss-Seidel preconditioner is used, and each symbol represents a Newton iteration.

aerostructural analysis is terminated when the flow, mesh, and structural equations have converged as tightly as possible, which occurs at relative residual values on the order of  $10^{-11}$ ,  $10^{-14}$ , and  $10^{-9}$ , respectively, based on experience from using the partitioned method. It is evident that when the difference in scaling between the three equations is left unadjusted, the resulting Newton update contributes little to reducing the flow and mesh components of  $\mathbf{R}_{AS}$ . Consequently, the result does not exhibit the rapid convergence expected of an inexact Newton method. For the scaled results in Figure 1,  $r_{scl,A}$  and  $c_{scl,A}$  are both set to 1. It can be assumed that the equation scaling already applied to  $\hat{\mathbf{R}}_A$  and the nondimensionalization of all flow variables ensure that flow subproblem is well scaled. The mesh and structural problems are scaled by choosing  $r_{scl,M} = n_M / \|\mathbf{R}_{M\Delta}^{(0)}\|_2$ ,  $r_{scl,S} = 1.0 / \|\mathbf{R}_S^{(0)}\|_2$ ,  $c_{scl,M} = m_\Delta \|\Delta \mathbf{b}_\Delta^{(n)}\|_2 / n_M$ , and  $c_{scl,S} = \|\mathbf{u}\|_2 / n_S$ , where  $n_M$  and  $n_S$  represent the size of the mesh and structural problems, respectively, and  $\Delta \mathbf{b}_\Delta^{(n)}$  represents the nodal displacements in the control volume between increments. The initial structural residual,  $\mathbf{R}_S^{(0)}$ , represents the forces acting on the structures, the norm of which does not tend to scale with  $n_S$ . On the other hand,  $\|\mathbf{R}_{M\Delta}^{(0)}\|_2$  is not a physical quantity, so it is necessary to include a factor of  $n_M$  in  $r_{scl,M}$  to prevent the scaled mesh equation from being too small. The variable or column scaling values,  $c_{scl,S}$  and  $c_{scl,M}$ , are based on the approximate average magnitude of the nodal structural deflection and mesh deformation, respectively. The chosen scaling method substantially improved the convergence rate of the inexact Newton method for all three equations. A total of 7 Newton iterations are required to fully converge all equations, in contrast to the 35 iterations required without any Jacobian block scaling. Although it did not translate to a significant increase in the total number coupled Krylov iterations in this case, failing to adjust the Jacobian block scaling has led to non-convergence of the Newton iterations in other problems. This shows that appropriate scaling is important for ensuring the robustness of the Newton-Krylov solution algorithm.

### III.B. Matrix-Vector Product

A routine to compute the matrix-vector product with the scaled matrix  $\hat{\mathbf{A}}$  in (10) is required by FGMRES. Such routine must be efficient and sufficiently accurate relative to the linear system solution tolerance required to ensure rapid convergence of the Newton iterations. In the present methodology, the matrix-vector products are evaluated using a combination of matrix-free finite-difference Jacobian approximation and matrix-explicit analytical differentiation. Using  $\mathbf{z} = [\mathbf{z}_A, \mathbf{z}_M, \mathbf{z}_S]^T$  to represent the vector to be multiplied by  $\hat{\mathbf{A}}$ , it is convenient to use the following scaled vector,

$$\hat{\mathbf{z}} = \left[ c_{\text{scl},A} \mathbf{z}_A, c_{\text{scl},M} \frac{\partial \mathbf{b}_\Delta}{\partial \hat{\mathbf{b}}_\Delta} \mathbf{z}_M, c_{\text{scl},S} \mathbf{z}_S \right]^T = [\hat{\mathbf{z}}_A, \hat{\mathbf{z}}_M, \hat{\mathbf{z}}_S]^T, \quad (11)$$

to calculate the Jacobian matrix-vector products. This allows the unscaled state variables to be perturbed when Jacobian-free approximations are used.

Matrix-vector products with the flow Jacobian blocks, as well as the Jacobian of the structural forces in  $\mathbf{R}_S$ , with respect to the flow and mesh states are approximated using first-order forward difference as follows:

$$r_{\text{scl},A} \left( \frac{\partial \hat{\mathbf{R}}_A^{(n)}}{\partial [\mathbf{q}, \mathbf{b}_\Delta]^{(n)}} \begin{bmatrix} \hat{\mathbf{z}}_A \\ \hat{\mathbf{z}}_M \end{bmatrix} \right) \approx \frac{r_{\text{scl},A} \left[ \hat{\mathbf{R}}_A(\mathbf{q}^{(n)} + \delta_{\text{FD}} \hat{\mathbf{z}}_A, \mathbf{b}_\Delta^{(n)} + \epsilon \hat{\mathbf{z}}_M) - \hat{\mathbf{R}}_A(\mathbf{q}^{(n)}, \mathbf{b}_\Delta^{(n)}) \right]}{\delta_{\text{FD}}}, \quad (12)$$

$$-r_{\text{scl},S} \frac{\partial \mathbf{f}_S^{(n)}}{\partial [\mathbf{q}, \mathbf{b}_\Delta]^{(n)}} \begin{bmatrix} \hat{\mathbf{z}}_A \\ \hat{\mathbf{z}}_M \end{bmatrix} \approx \frac{-r_{\text{scl},S} \left[ \mathbf{f}_S(\mathbf{q}^{(n)} + \delta_{\text{FD}} \hat{\mathbf{z}}_A, \mathbf{b}_\Delta^{(n)} + \delta_{\text{FD}} \hat{\mathbf{z}}_M) - \mathbf{f}_S(\mathbf{q}^{(n)}, \mathbf{b}_\Delta^{(n)}) \right]}{\delta_{\text{FD}}}, \quad (13)$$

where  $n$  is the Newton iteration index. To minimize both truncation and subtractive cancellation error in (12) and (13),  $\delta_{\text{FD}}$  is determined by<sup>38,47</sup>

$$\delta_{\text{FD}} = \sqrt{\frac{(n_A + n_M) \epsilon_{\text{Machine}}}{\hat{\mathbf{z}}_A^T \hat{\mathbf{z}}_A + \hat{\mathbf{z}}_M^T \hat{\mathbf{z}}_M}} = \frac{\sqrt{\epsilon_{\text{Machine}}}}{\text{RMS}([\hat{\mathbf{z}}_A, \hat{\mathbf{z}}_M]^T)}, \quad (14)$$

where  $\epsilon_{\text{Machine}} = 10^{-13}$  is the expected accuracy of floating point operations. This choice of  $\delta_{\text{FD}}$  leads to perturbations,  $\delta_{\text{FD}} \hat{\mathbf{z}}_A$  and  $\delta_{\text{FD}} \hat{\mathbf{z}}_M$ , of the state variables that are roughly on the order of  $\sqrt{\epsilon_{\text{Machine}}}$ , where the RMS value of the entries in  $\hat{\mathbf{z}}_A$  and  $\hat{\mathbf{z}}_M$  provides an estimate to the average magnitude of  $[\hat{\mathbf{z}}_A, \hat{\mathbf{z}}_M]$ . The Newton convergence using (12) and (13) is very similar to that using potentially more accurate alternatives, such as a second-order finite-difference approximation combined with an exact evaluation of  $(\partial \hat{\mathbf{R}}_A / \partial \mathbf{q}) \hat{\mathbf{z}}_A$ . Moreover, (12) and (13) require only a single evaluation of  $\hat{\mathbf{R}}_A$  and  $\mathbf{f}_S$ , respectively. In contrast, evaluating the same terms exactly requires differentiation with respect to both  $\mathbf{q}$  and  $\mathbf{b}_\Delta$ , doubling the potential memory requirement and cost per matrix-vector product evaluation. A first-order approximation to  $\partial \hat{\mathbf{R}}_A / \partial \mathbf{q}$  is still needed for the flow block preconditioner (see Section III.C). However, the time and memory requirement

associated with assembling the flow Jacobian approximation is much smaller than that required for the exact Jacobian.

Matrix-vector products with the remaining terms in  $\mathbb{A}$  are evaluated using a matrix-explicit approach. For the structural Jacobian with respect to  $\mathbf{u}$ ,  $r_{\text{scl},\text{S}}(\partial\mathbf{R}_{\text{S}}/\partial\mathbf{u})\hat{\mathbf{z}}_{\text{S}}$  is simply  $r_{\text{scl},\text{S}}\mathbf{K}_{\text{S}}\hat{\mathbf{z}}_{\text{S}}$  in a three-field formulation, where  $\mathbf{K}_{\text{S}}$  is already available in memory. The Jacobian matrix-vector products related to the mesh equation are given by

$$r_{\text{scl},\text{M}} \left( \frac{\partial\hat{\mathbf{R}}_{\text{M}\Delta}^{(n)}}{\partial[\mathbf{b}_{\Delta}, \mathbf{u}]^{(n)}} \begin{bmatrix} \hat{\mathbf{z}}_{\text{M}} \\ \hat{\mathbf{z}}_{\text{S}} \end{bmatrix} \right) = r_{\text{scl},\text{M}} \left( \frac{\partial\hat{\mathbf{R}}_{\text{M}\Delta}^{(n)}}{\partial\mathbf{b}_{\Delta}} \hat{\mathbf{z}}_{\text{M}} + \frac{\partial\hat{\mathbf{R}}_{\text{M}\Delta}^{(n)}}{\partial\mathbf{f}_{\text{M}\Delta}} \frac{\partial\mathbf{f}_{\text{M}\Delta}}{\partial\mathbf{u}} \hat{\mathbf{z}}_{\text{S}} \right). \quad (15)$$

The mesh residual vector  $\hat{\mathbf{R}}_{\text{M}\Delta}$  consists of  $m_{\Delta}$  linear elasticity mesh movement equations for the  $m_{\Delta}$  increments used during aerostructural analysis. The first term on the right-hand-side (RHS) of (15) is expanded to

$$\frac{\partial\hat{\mathbf{R}}_{\text{M}\Delta}}{\partial\mathbf{b}_{\Delta}} \hat{\mathbf{z}}_{\text{M}} = \begin{bmatrix} \hat{\mathbf{K}}_{\text{M}\Delta}^{(1,n)} & 0 & \cdots & 0 & 0 \\ \frac{\partial\hat{\mathbf{R}}_{\text{M}\Delta}^{(2,n)}}{\partial\mathbf{b}_{\Delta}^{(1,n)}} & \hat{\mathbf{K}}_{\text{M}\Delta}^{(2,n)} & \cdots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & \frac{\partial\hat{\mathbf{R}}_{\text{M}\Delta}^{(m_{\Delta},n)}}{\partial\mathbf{b}_{\Delta}^{(m_{\Delta}-1,n)}} & \hat{\mathbf{K}}_{\text{M}\Delta}^{(m_{\Delta},n)} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{z}}_{\text{M}}^{(1)} \\ \hat{\mathbf{z}}_{\text{M}}^{(2)} \\ \vdots \\ \hat{\mathbf{z}}_{\text{M}}^{(m_{\Delta})} \end{bmatrix}. \quad (16)$$

The entries in (16) can be precomputed and stored at the start of each Newton iteration, but each finite-difference approximation of (15) requires updating the scaled mesh stiffness matrix,  $\hat{\mathbf{K}}_{\text{M}\Delta}^{(i)}$ , for all increments according to the perturbed mesh state. The matrix-explicit approach is advantageous because it is much faster to multiply (16) with  $\hat{\mathbf{v}}_{\text{M}}$  than it is to assemble  $\mathbf{K}_{\text{M}\Delta}^{(i)}$  repeatedly. Furthermore, storing the entries in (16) also accelerates the block preconditioning calculations associated with the mesh equation (see Section III.C). For the second term on the RHS of (15), the mapping from the structural state,  $\mathbf{u}$ , to the implicit force vector,  $\mathbf{f}_{\text{M}\Delta}$ , is linear. As a result, this term can be calculated with analytical accuracy by performing a displacement transfer and implicit force vector evaluation using  $\hat{\mathbf{z}}_{\text{S}}$  in place of  $\mathbf{u}$ .

### III.C. Preconditioner

The objective of this section is to identify an effective preconditioner that maximizes the performance of the Newton-Krylov solution algorithm. In domain decomposition, a subdomain usually refers to a part of the physical domain assigned to a processor. A local preconditioner on each processor can be gathered via a Schwarz or Schur method to precondition the global linear system.<sup>48</sup> The application of such a technique to the present methodology is not trivial due to the differences in discretization between the flow, mesh, and structural subproblems. Furthermore, it would not be able to take advantage of the direct factorization of

$\mathbf{K}_S$  which has been designed to address the ill-conditioning from the use of shell elements.<sup>39</sup> It is therefore beneficial to introduce modularity into the preconditioning procedure by treating each of the flow, mesh, and structural subproblems as a subdomain, whereby the local preconditioner can be tailored to each subproblem. The subdomains have no overlap in this case. Hence an additive or multiplicative Schwarz preconditioner defined in this manner is equivalent to a block Jacobi or block Gauss-Seidel preconditioner, respectively, as adopted by a number of authors in the literature.<sup>6,7,16,30</sup> The performance of the block Jacobi and block Gauss-Seidel preconditioners with different local preconditioning parameters will be investigated here in the context of high-fidelity aerostructural analysis and optimization.

### III.C.1. Block Jacobi Preconditioner

Given an input vector  $\mathbf{z} = [\mathbf{z}_A, \mathbf{z}_M, \mathbf{z}_S]^T$ , a block Jacobi preconditioner returns

$$\mathbf{w} = \begin{bmatrix} \mathbf{w}_A \\ \mathbf{w}_M \\ \mathbf{w}_S \end{bmatrix} = \begin{bmatrix} c_{\text{scl},A}^{-1} \mathbf{M}_A^{-1} r_{\text{scl},A}^{-1} \mathbf{z}_A \\ c_{\text{scl},M}^{-1} \mathbf{M}_M^{-1} r_{\text{scl},M}^{-1} \mathbf{z}_M \\ c_{\text{scl},S}^{-1} \mathbf{M}_S^{-1} r_{\text{scl},S}^{-1} \mathbf{z}_S \end{bmatrix} = \begin{bmatrix} c_{\text{scl},A}^{-1} \mathbf{M}_A^{-1} \hat{\mathbf{z}}_A \\ c_{\text{scl},M}^{-1} \mathbf{M}_M^{-1} \hat{\mathbf{z}}_M \\ c_{\text{scl},S}^{-1} \mathbf{M}_S^{-1} \hat{\mathbf{z}}_S \end{bmatrix}, \quad (17)$$

where

$$\mathbf{M}_A^{-1} \approx \left( \frac{\partial \hat{\mathbf{R}}_A^{(n)}}{\partial \mathbf{q}^{(n)}} \right)^{-1}, \quad \mathbf{M}_M^{-1} \approx \left( \frac{\partial \hat{\mathbf{R}}_{M\Delta}^{(n)}}{\partial \hat{\mathbf{b}}_\Delta^{(n)}} \right)^{-1}, \quad \mathbf{M}_S^{-1} \approx \left( \frac{\partial \mathbf{R}_S^{(n)}}{\partial \mathbf{u}^{(n)}} \right)^{-1}, \quad (18)$$

are the preconditioner matrices for the diagonal blocks in the coupled aerostructural Jacobian matrix,  $\hat{\mathbf{A}}$ . The preconditioned vectors,  $\mathbf{w}_A$ ,  $\mathbf{w}_M$ , and  $\mathbf{w}_S$ , can be computed without any communication between subproblems. However, the subproblems are not currently preconditioned in parallel because the flow and mesh modules share the same group of processors, although it is possible in theory. The potential performance of the block Jacobi preconditioner without this constraint will be quantified later. The Newton iteration index  $n$  is dropped from (18) in the subsequent discussions.

Existing linear system solution routines serve as effective block preconditioners. The nonlinear flow analysis capability in the flow module includes a sophisticated iterative solution algorithm for

$$\frac{\partial \hat{\mathbf{R}}_A}{\partial \mathbf{q}} \Delta \mathbf{q} = -\hat{\mathbf{R}}_A. \quad (19)$$

The same routine can be used to compute  $\mathbf{M}_A^{-1} \hat{\mathbf{z}}_A$  by replacing  $\Delta \mathbf{q}$  and  $-\hat{\mathbf{R}}_A$  in the above equation with  $c_{\text{scl},A} \mathbf{w}_A$  and  $\hat{\mathbf{z}}_A$ , respectively. The mesh Jacobian block on the diagonal of  $\hat{\mathbf{A}}$ , which  $\mathbf{M}_M$  should approximate, expands to (16) with  $\mathbf{b}_\Delta$  replaced by  $\hat{\mathbf{b}}_\Delta$ . Given  $\hat{\mathbf{z}}_M$ , the preconditioned vector,  $\mathbf{w}_M$ , can be obtained using block forward-substitution and the solution routines implemented for (2). The direct factorization of the structural stiffness matrix,  $\mathbf{K}_S$ , can be used to compute  $\mathbf{M}_S^{-1} \hat{\mathbf{z}}_S$ .

### III.C.2. Block Gauss-Seidel Preconditioner

A multiplicative Schwarz or block Gauss-Seidel preconditioner accounts for additional coupling between the subproblems compared to a block Jacobi preconditioner. This can result in a much smaller number of iterations. However, preconditioning of each subproblem must be carried out in serial unlike a block Jacobi preconditioner. This may not be a disadvantage in this case because the block Gauss-Seidel preconditioner is applied to three subproblems instead of hundreds of processors. Given that the local preconditioners are more expensive to evaluate, it could be beneficial to minimize the number of coupled FGMRES iterations. Application of the block Gauss-Seidel preconditioner involves solving the following equations:

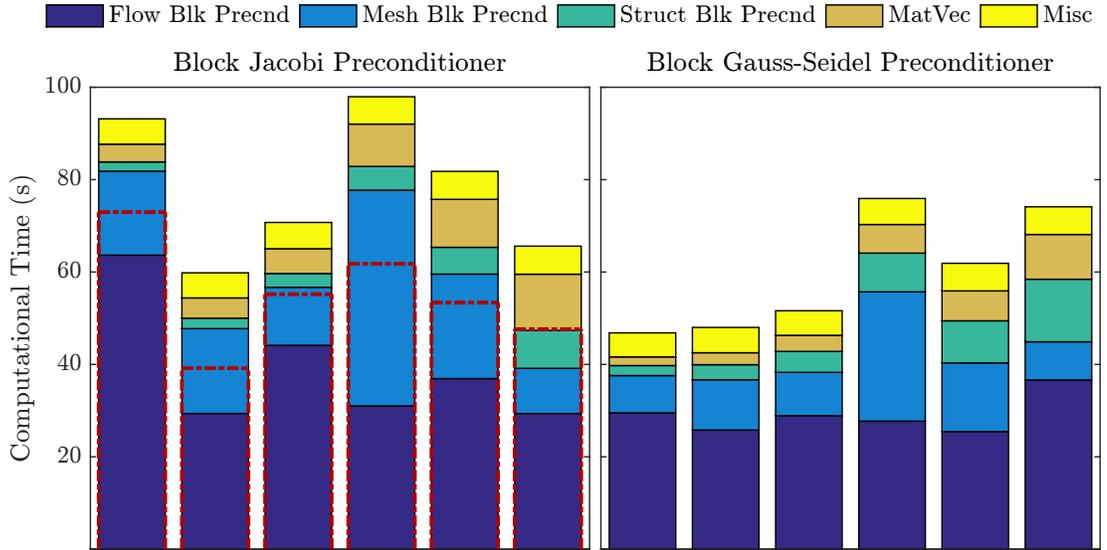
$$\begin{aligned}
 \mathbf{w}_M &= c_{\text{scl},M}^{-1} \mathbf{M}_M^{-1} \hat{\mathbf{z}}_M = c_{\text{scl},M}^{-1} \mathbf{M}_M^{-1} \underbrace{\left( r_{\text{scl},M}^{-1} \mathbf{z}_M \right)}_{\hat{\mathbf{z}}_M}, \\
 \mathbf{w}_A &= c_{\text{scl},A}^{-1} \mathbf{M}_A^{-1} \hat{\mathbf{z}}_A = c_{\text{scl},A}^{-1} \mathbf{M}_A^{-1} \underbrace{\left( r_{\text{scl},A}^{-1} \mathbf{z}_A - \frac{\partial \hat{\mathbf{R}}_A}{\partial \hat{\mathbf{b}}_\Delta} \left( r_{\text{scl},M}^{-1} \mathbf{w}_M \right) \right)}_{\hat{\mathbf{z}}_A}, \\
 \mathbf{w}_S &= c_{\text{scl},S}^{-1} \mathbf{M}_S^{-1} \hat{\mathbf{z}}_S = c_{\text{scl},S}^{-1} \mathbf{M}_S^{-1} \underbrace{\left( r_{\text{scl},S}^{-1} \mathbf{z}_S - \frac{\partial \mathbf{R}_S}{\partial \mathbf{q}} \left( r_{\text{scl},A}^{-1} \mathbf{w}_A \right) - \frac{\partial \mathbf{R}_S}{\partial \hat{\mathbf{b}}_\Delta} \left( r_{\text{scl},M}^{-1} \mathbf{w}_M \right) \right)}_{\hat{\mathbf{z}}_S},
 \end{aligned} \tag{20}$$

where  $\mathbf{M}_A^{-1} \hat{\mathbf{z}}_A$ ,  $\mathbf{M}_M^{-1} \hat{\mathbf{z}}_M$ , and  $\mathbf{M}_S^{-1} \hat{\mathbf{z}}_S$  are computed in the same manner as for the block Jacobi preconditioner. Evaluation of the extra matrix-vector product terms in  $\hat{\mathbf{z}}_A$ ,  $\hat{\mathbf{z}}_M$ , and  $\hat{\mathbf{z}}_S$  is discussed in Section III.B.

### III.C.3. Preconditioner Comparison

The preconditioning operations with  $\mathbf{M}_A$  and  $\mathbf{M}_M$  both invoke iterative solution routines in the flow and mesh modules. The respective tolerance used in each case can be a useful parameter and is investigated for both the block Jacobi and block Gauss-Seidel preconditioners in Figure 2 using the test problem from Appendix A. All tolerances considered are greater than 0.01, which is the coupled linear system solution tolerance specified on page 8. The time to calculate  $\hat{\mathbf{z}}_A$ ,  $\hat{\mathbf{z}}_M$ , and  $\hat{\mathbf{z}}_S$  for the block Gauss-Seidel preconditioner is included in the corresponding block preconditioning time in the bar graphs. In all cases, the coupled problem converges after 7 Newton iterations.

For the block Jacobi preconditioner, an iterative solution tolerance of 0.1 for  $\mathbf{M}_A^{-1} \hat{\mathbf{z}}_A$  and 0.01 for  $\mathbf{M}_M^{-1} \hat{\mathbf{z}}_M$  is the most efficient. In the ideal case assuming fully parallel execution, it leads to the best Newton-Krylov solution time of 39.2s in Figure 2. The second best Newton-Krylov solution time is 46.8s using the block Gauss-Seidel preconditioner with a tolerance of 0.01 for both  $\mathbf{M}_A^{-1} \hat{\mathbf{z}}_A$  and  $\mathbf{M}_M^{-1} \hat{\mathbf{z}}_M$ . The coupled linear system at each Newton step is solved in at most 4 Krylov iterations, which is less than half of the minimum number of iterations needed using the block Jacobi preconditioner. Using a tolerance of 0.1 for  $\mathbf{M}_A^{-1} \hat{\mathbf{z}}_A$  and a tolerance



$\mathbf{M}_A^{-1}$ Tolerance	0.01	0.1	0.1	0.5	0.5	0.5	0.01	0.1	0.1	0.5	0.5	0.5
$\mathbf{M}_M^{-1}$ Tolerance	0.01	0.01	0.1	0.01	0.1	0.5	0.01	0.01	0.1	0.01	0.1	0.5
Total Iterations	57	65	84	147	166	199	25	37	52	96	105	153
Max. Iterations	11	12	17	31	37	38	4	7	10	18	22	28

Figure 2: Computational time breakdown for the block Jacobi and block Gauss-Seidel preconditioners. The different flow ( $\mathbf{M}_A^{-1}$ ) and mesh ( $\mathbf{M}_M^{-1}$ ) block preconditioning tolerances investigated are listed in the first two rows in the table. The last two rows describe the total and the maximum number of coupled FGMRES iterations during the Newton-Krylov solution process. The dashed lines for the block Jacobi preconditioner results show an estimated ideal time if all block preconditioning operations can be performed in parallel.

of 0.01 for  $\mathbf{M}_M^{-1}\hat{\mathbf{z}}_M$  with the block Gauss-Seidel preconditioner also appears to be very efficient. It is found to be a more suitable combination of preconditioning parameters when the size of the flow subproblem is much larger than the other subproblems. However, relaxing the flow tolerance beyond 0.5 is not recommended, as it results in many coupled FGMRES iterations for both types of preconditioners. This increases the memory required by FGMRES and compromises the robustness of the monolithic solution algorithm, especially with an increase in problem size or in the degree of fluid-structure coupling. It can additionally be observed that the block Gauss-Seidel preconditioner becomes less effective relative to the block Jacobi preconditioner with the use of weaker flow and mesh block preconditioners. Although the preconditioners for the solution to (19) and (2) can also be used as the flow and mesh block preconditioners, it is found that doing so provides no benefits in terms of computational time and memory requirements. This is not surprising given the above observations from Figure 2.

There are a number of arguments favoring the use of the block Gauss-Seidel preconditioner. For the recommended range of tolerances between 0.1 and 0.01 for the flow and mesh block preconditioners, the block Gauss-Seidel preconditioner is more efficient than the block Jacobi preconditioner in two out of the three cases, including the ideal block Jacobi time. The performance of the block Gauss-Seidel preconditioner

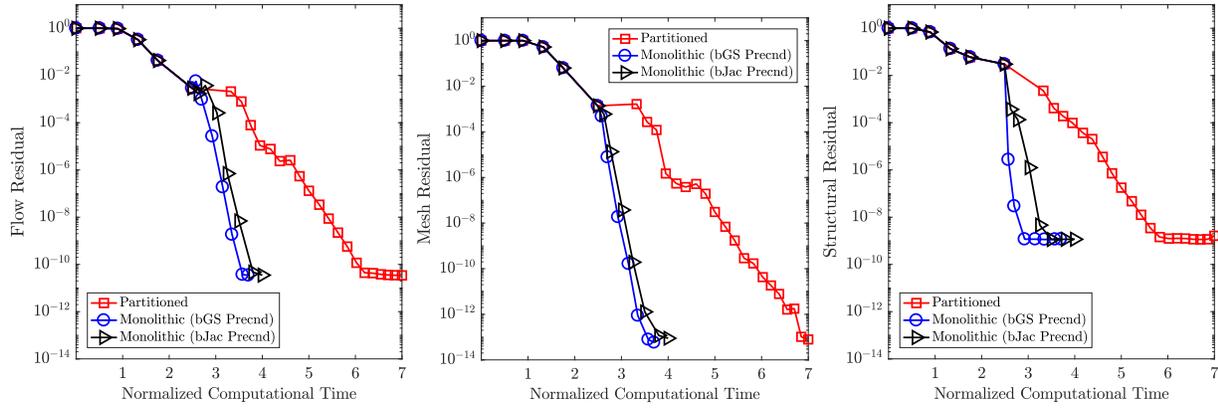


Figure 3: Convergence of  $\|\mathbf{R}_A\|_2$ ,  $\|\mathbf{R}_{M\Delta}\|_2$ , and  $\|\mathbf{R}_S\|_2$  using the partitioned method and the monolithic method where the Newton-Krylov solution uses a block Jacobi (bJac) and a block Gauss-Seidel (bGS) preconditioner. Computational time on the  $x$ -axis is normalized by the time required for a rigid flow analysis, which is approximately 40s.

is thus more consistent. By keeping the number of iterations low, the block Gauss-Seidel preconditioner can improve the robustness of the solution algorithm with increased problem size and complexity. The memory requirement is reduced due to a smaller Krylov subspace size. Finally, the estimated ideal block Jacobi time equivalently assumes that an additional 112 processors are allocated for the mesh calculations. It is unclear whether the ideal block Jacobi preconditioner would remain the most efficient if the total number of processors were to remain unchanged from the present implementation. For the above reasons, the block Gauss-Seidel preconditioner is a more effective way to precondition the Newton-Krylov solution algorithm.

### III.D. Detailed Cost Comparison between Monolithic and Partitioned Analysis

Figure 3 plots the full analysis convergence for the test problem described in Appendix A using the monolithic method and the partitioned method. Both the block Jacobi and block Gauss-Seidel preconditioners for the monolithic analysis use a tolerance of 0.1 for  $\mathbf{M}_A^{-1}\hat{\mathbf{z}}_A$  and 0.01 for  $\mathbf{M}_M^{-1}\hat{\mathbf{z}}_M$ . The partitioned method requires 25 iterations. The first 5 overlap with the monolithic analysis before it switches to the Newton-Krylov solution algorithm and converges in 7 additional iterations. Figure 3 shows that the monolithic solution method improves the efficiency of the coupled analysis by almost 50% relative to the partitioned method. Most of the efficiency gain are due to the much smaller number of coupled nonlinear iterations, as the average time per iteration does not differ substantially between the two methods.

A detailed cost comparison between the monolithic and partitioned analysis reveals a number of insights. In the present methodology, each partitioned iteration requires one and  $m_\Delta$  linear system solutions for the structural and mesh subproblems, respectively. During a monolithic inexact Newton iteration, the numbers of mesh and structural linear system solutions are scaled by the number of coupled Krylov iterations needed to solve (8). This increases the costs associated with the mesh and structural subproblems for an inexact

Newton iteration. The flow analysis, on the other hand, is nonlinear, so multiple linear system solutions are required during a partitioned iteration. The flow calculation time per monolithic iteration can be smaller than a partitioned iteration if the number of coupled FGMRES iterations is kept low. For the above reasons, using the monolithic method for analysis can potentially be more beneficial if the flow subproblem is much larger in size than the mesh and structural subproblems, and with the use of a nonlinear structural analysis or mesh movement strategy. A higher level performance comparison between the monolithic and partitioned methods for a wider range of test conditions will be presented in Section V.

## IV. Preconditioned Krylov Solution to the Adjoint Problem

If the partitioned solution procedure to the coupled adjoint problem is already in place, implementing the monolithic solution strategy using a preconditioned Krylov subspace method is more straightforward than in the case of the analysis problem. Matrix-vector products with  $\mathbb{A}^T$  can be evaluated using the routines implemented for (5–6). It will also be shown that the block Gauss-Seidel iteration outlined in (5–6) provides an effective block iterative preconditioner, but a flexible Krylov subspace method is needed. Equation (4) additionally needs to be solved to a tight tolerance. The present methodology uses GCROT( $m, k$ )<sup>42</sup> to maintain a fixed memory usage even though the required number of Krylov iterations is not known in general. GCROT( $m, k$ ) is a simplified and flexible variant of GCROT (Generalized Conjugate Residual with Inner Orthogonalization and Outer Truncation), while  $m$  and  $k$  are the inner and outer subspace sizes, respectively. FGMRES is used as the inner method and the first outer iteration of GCROT( $m, k$ ) is equivalent to FGMRES with an inner subspace size of  $m + k$ . If the solution tolerance is not reached after  $m + k$  iterations, GCROT( $m, k$ ) offers a robust way to continue the iterative solution process without significantly hindering the convergence. GCROT( $m, k$ ) with  $m = 20$  and  $k = 1$  is found to be suitable for the present methodology. Investigations related to the scaling and preconditioning procedures are presented in the remainder of this section.

### IV.A. Scaling

The coupled adjoint problem can be scaled in a similar manner as (10). The mesh subproblem is first scaled followed by the scaling of the Jacobian blocks. The relative scaling between the Jacobian blocks can be

$$\begin{bmatrix} \mathcal{O}(10^{-1}) & 0 & \mathcal{O}(10^{-2}) \\ \mathcal{O}(10^{-3}) & \mathcal{O}(10^{-3}) & \mathcal{O}(10^{-1}) \\ 0 & \mathcal{O}(10^{-3}) & \mathcal{O}(10^7) \end{bmatrix}$$

(a) Before

$$\begin{bmatrix} \mathcal{O}(1) & 0 & \mathcal{O}(10^{-3}) \\ \mathcal{O}(10^{-3}) & \mathcal{O}(10) & \mathcal{O}(10^{-3}) \\ 0 & \mathcal{O}(10^{-5}) & \mathcal{O}(1) \end{bmatrix}$$

(b) After

Figure 4: Estimated scaling in  $\mathbb{A}^T$  before and after Jacobian block scaling is applied.

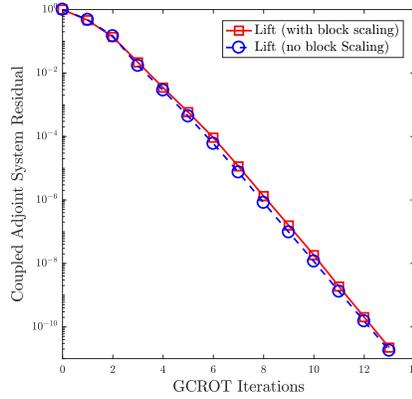


Figure 5: Coupled adjoint convergence for the lift functional.

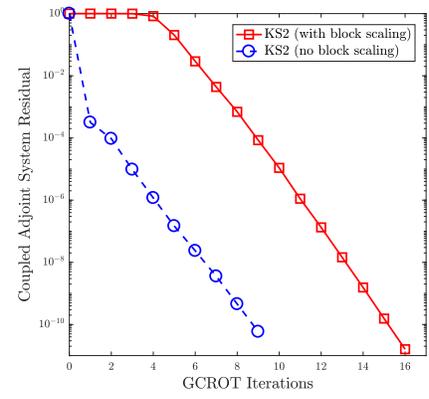


Figure 6: Coupled adjoint convergence for the KS functional.

estimated as follows without explicit access to individual entries in the transposed Jacobian,  $\mathbb{A}^T$ :

$$\mathbb{S} = \begin{bmatrix} r_{\text{scl},A} \frac{1}{n_A} \left\| \left( \frac{\partial \mathbf{R}_A}{\partial \mathbf{q}} \right)^T \mathbf{1}_A \right\|_2 c_{\text{scl},A} & 0 & r_{\text{scl},A} \frac{1}{n_A} \left\| \left( \frac{\partial \mathbf{R}_S}{\partial \mathbf{q}} \right)^T \mathbf{1}_S \right\|_2 c_{\text{scl},S} \\ r_{\text{scl},M} \frac{1}{n_M} \left\| \left( \frac{\partial \mathbf{R}_A}{\partial \hat{\mathbf{b}}_\Delta} \right)^T \mathbf{1}_A \right\|_2 c_{\text{scl},A} & r_{\text{scl},M} \frac{1}{n_M} \left\| \left( \frac{\partial \hat{\mathbf{R}}_{M\Delta}}{\partial \hat{\mathbf{b}}_\Delta} \right)^T \mathbf{1}_M \right\|_2 c_{\text{scl},M} & r_{\text{scl},M} \frac{1}{n_M} \left\| \left( \frac{\partial \mathbf{R}_S}{\partial \hat{\mathbf{b}}_\Delta} \right)^T \mathbf{1}_S \right\|_2 c_{\text{scl},S} \\ 0 & r_{\text{scl},S} \frac{1}{n_S} \left\| \left( \frac{\partial \hat{\mathbf{R}}_{M\Delta}}{\partial \mathbf{u}} \right)^T \mathbf{1}_M \right\|_2 c_{\text{scl},M} & r_{\text{scl},S} \frac{1}{n_S} \left\| \left( \frac{\partial \mathbf{R}_S}{\partial \mathbf{u}} \right)^T \mathbf{1}_S \right\|_2 c_{\text{scl},S} \end{bmatrix}. \quad (21)$$

The block row and column scaling values,  $r_{\text{scl},\star}$  and  $c_{\text{scl},\star}$ , are determined automatically such that  $\log_{10}(\mathbb{S}_{ij})$  for all  $\mathbb{S}_{ij} \neq 0$  are as close to unity as possible in a least-squares sense.

Figure 4 shows the entries in  $\mathbb{S}$  for the test problem described in Appendix A with and without the Jacobian block scaling. The corresponding coupled adjoint convergence is additionally shown in Figures 5 and 6 for the lift and KS functionals, respectively, where KS refers to the Kreisselmeier-Steinhauser stress constraint aggregation.<sup>35,49,50</sup> The KS function here is for the elements in the bottom skin of the structural model. Figures 5 and 6 exhibit noticeably different convergence behaviors because the partial derivative of an aerodynamic functional is zero with respect to  $\mathbf{u}$ , whereas the partial derivative of a structural functional is zero with respect to  $\mathbf{q}$  and  $\mathbf{b}_\Delta$ . The use of Jacobian block scaling has a more noticeable effect on the convergence of the KS functional than that of the lift functional. By comparison with the partitioned adjoint calculations, it can be shown that the unscaled Krylov iterations produce inaccurate gradients for the KS functional even though the coupled adjoint problem appears to converge in fewer iterations. In the unscaled case, both the mesh and flow adjoint equations are in fact undersolved relative to the structural adjoint equation, the value of which dominates the coupled linear system residual. The above observations are found to be true for other structural functionals of interest. This problem can be addressed by checking

the linear system residual of the individual equations for convergence. However, scaling the Jacobian blocks in  $\mathbb{A}^T$  is still useful for ensuring the accuracy of the monolithic coupled adjoint solution, as it reduces the impact of round-off errors.<sup>46</sup>

#### IV.B. Preconditioner

A block Gauss-Seidel iteration involving (5–6) can be used as a preconditioner with the following modifications:

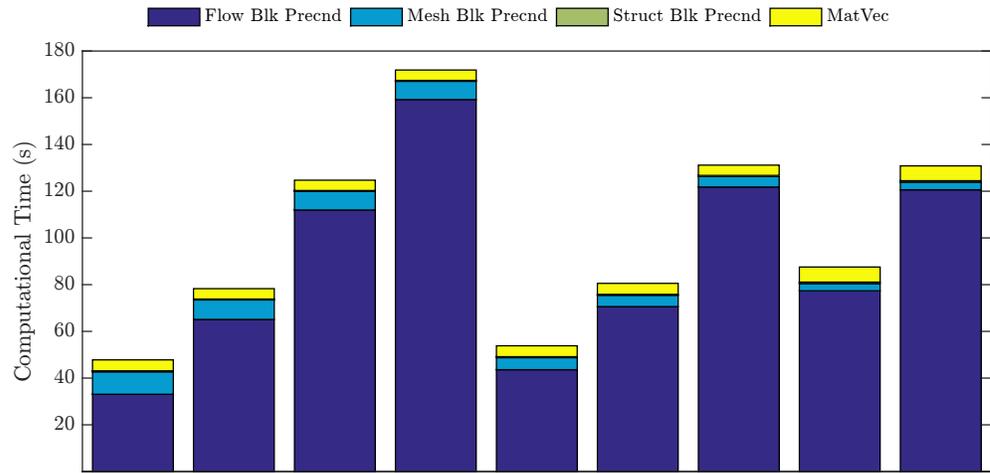
$$\frac{\partial \mathbf{R}_A}{\partial \mathbf{q}}^T (c_{\text{scl},A} \mathbf{w}_A) = r_{\text{scl},A}^{-1} \mathbf{z}_A = \hat{\mathbf{z}}_A \quad (22)$$

$$\frac{\partial \hat{\mathbf{R}}_{M\Delta}}{\partial \hat{\mathbf{b}}_\Delta}^T (c_{\text{scl},M} \mathbf{w}_M) = r_{\text{scl},M}^{-1} \mathbf{z}_M - \frac{\partial \mathbf{R}_A}{\partial \hat{\mathbf{b}}_\Delta}^T (c_{\text{scl},A} \mathbf{w}_A) = \hat{\mathbf{z}}_M \quad (23)$$

$$\mathbf{M}_S (c_{\text{scl},S} \mathbf{w}_S) = r_{\text{scl},S}^{-1} \mathbf{z}_S - \frac{\partial \hat{\mathbf{R}}_{M\Delta}}{\partial \mathbf{u}}^T (c_{\text{scl},M} \mathbf{w}_M) = \hat{\mathbf{z}}_S. \quad (24)$$

Here  $[\mathbf{w}_A, \mathbf{w}_M, \mathbf{w}_S]$  is the preconditioned vector given  $[\mathbf{z}_A, \mathbf{z}_M, \mathbf{z}_S]$  as an input, and  $\mathbf{M}_S$  is the direct factorization of  $\mathbf{K}_S$ . The approximate solutions to (22) and (23), which use existing iterative adjoint solvers in the flow and mesh modules, respectively, serve the same purpose as  $\mathbf{M}_A^{-1}$  and  $\mathbf{M}_M^{-1}$  during the monolithic analysis. The use of different tolerances for the evaluation of  $\mathbf{M}_A^{-1} \hat{\mathbf{z}}_A$  and  $\mathbf{M}_M^{-1} \hat{\mathbf{z}}_M$  is investigated in Figure 7 for the lift functional. The coupled adjoint problem is solved to a tolerance of  $10^{-10}$ . The bar graphs show a similar breakdown of the computational time as in Figure 2. The results for other functionals are very similar; hence they are not presented here.

Figure 7 shows that the flow block preconditioning calculation involving the solution to (22) is consistently much more expensive than the rest of the coupled adjoint calculations. A block Jacobi preconditioner is hence not considered because it will increase the number of flow block preconditioner evaluations. The results in Figure 7 further indicate that it is optimal to use a tolerance of  $10^{-1}$  for  $\mathbf{M}_A^{-1} \hat{\mathbf{z}}_A$  and a tolerance between  $10^{-4}$  and  $10^{-6}$  for  $\mathbf{M}_M^{-1} \hat{\mathbf{z}}_M$ . This leads to the fastest flow and mesh block preconditioners but still allows the block Gauss-Seidel preconditioner to be effective, thereby maintaining a small number of Krylov iterations. Not surprisingly, the same tolerances for the flow and mesh adjoint equations also result in the best partitioned adjoint solution performance. The last rows in the table of Figure 7 show the monolithic adjoint solution time normalized by the partitioned solution time with different relaxation parameters,  $\theta$ . The choice of  $\theta$  for a particular problem strongly influences the efficiency and robustness of the partitioned calculations. This will be more clearly illustrated in Sections V and VI. The fastest monolithic coupled adjoint solution leads to a 72% reduction in computational time relative to the partitioned method with  $\theta = 0.5$ , and a reduction of 61% with  $\theta = 0.75$ . Not all options considered in Figure 7 result in an efficiency improvement relative to the partitioned method. This highlights the importance of choosing an appropriate preconditioner for the



Tolerance for $\mathbf{M}_A^{-1}\hat{\mathbf{z}}_A$	1E-1	1E-2	1E-4	1E-6	1E-1	1E-2	1E-4	1E-1	1E-2
Tolerance for $\mathbf{M}_M^{-1}\hat{\mathbf{z}}_M$	1E-6	1E-6	1E-6	1E-6	1E-4	1E-4	1E-4	1E-2	1E-2
Total GCROT Iterations	13	12	12	12	13	12	12	20	20
Normalized Time ( $\theta = 0.5$ )	0.28	0.46	0.74	1.02	0.32	0.48	0.78	0.52	0.77
Normalized Time ( $\theta = 0.75$ )	0.39	0.63	1.01	1.39	0.44	0.65	1.06	0.71	1.06

Figure 7: Comparison of different solution tolerances for the flow block preconditioning calculation ( $\mathbf{M}_A^{-1}\hat{\mathbf{z}}_A$ ) and the mesh block preconditioning calculation ( $\mathbf{M}_M^{-1}\hat{\mathbf{z}}_M$ ). The coupled adjoint problem considered is for the lift functional. The monolithic solution time listed in the last two rows is normalized by the partitioned coupled adjoint solution time, where  $\theta$  is the relaxation parameter used for the partitioned method.

monolithic coupled adjoint solution. As in the case of the analysis problem, the efficiency gain from using the monolithic method is mainly from reducing the number of coupled adjoint iterations from 48 ( $\theta = 0.5$ ) and 29 ( $\theta = 0.75$ ) to 13.

## V. Influence of the Degree of Fluid-Structure Coupling on the Relative Performance of Monolithic and Partitioned Methods

This section compares the performance of the monolithic and partitioned methods with varying degrees of aerostructural coupling. The magnitude of the aerodynamic forces acting on the wing is proportional to the freestream dynamic pressure,  $q_\infty$ , and the stiffness of the structure is proportional to the Young's modulus,  $E$ . The ratio given by  $q_\infty/E$  therefore provides an indication of the degree of coupling between the flow and the structures. Ten equally spaced  $q_\infty/E$  values are selected for this study, as summarized in Table 1. The  $q_\infty$  value for data point 3 in Table 1 is from a 1g cruise flight condition, with a freestream Mach number of 0.785 and an altitude of 35,000 feet, or equivalently a freestream pressure of 23.843 kPa. Similarly, the  $q_\infty$  value for data point 7 is taken from a 2.5g maneuver flight condition, which has a freestream Mach number of 0.798 and is assumed to be at an altitude of 12,000 feet, translating to a freestream pressure of

Data Point	1	2	3 <sup>†</sup>	4	5	6	7 <sup>‡</sup>	8	9	10
$q_\infty/E (1 \times 10^{-8})$	1.46	7.76	14.1	20.4	26.7	33.0	39.3	45.6	51.9	58.2
Tip Deflection (% Semi-span)	0.86	3.81	5.82	7.2	8.15	8.79	9.18	9.39	9.46	9.41

Table 1: A sample of ten  $q_\infty/E$  values used to analyze the performance of the monolithic and partitioned solution methods. The superscripts <sup>†</sup> and <sup>‡</sup> indicate that the corresponding  $q_\infty/E$  ratio is obtained from the cruise and maneuver flight conditions, respectively. The bottom row indicates the normalized tip deflections computed for the present study.

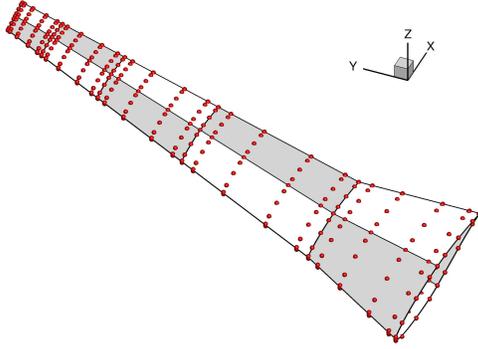


Figure 8: Surface patches on the geometry used for the analysis and adjoint calculations.

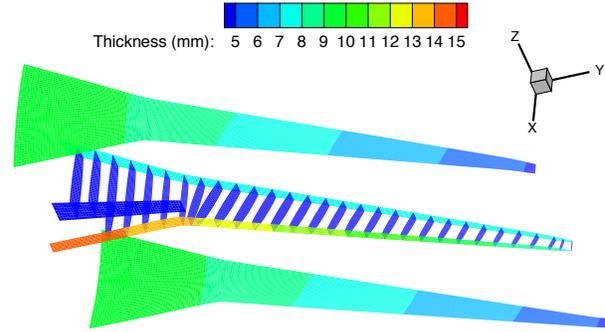


Figure 9: The component thickness distribution in the structural model.

64.442 kPa. The structures are assumed to be Aluminum, which has a Young's modulus of 73.1GPa. The rest of the material properties are provided in Appendix B. This allows  $q_\infty/E$  for data points 3 and 7 to be determined, from which the rest of the data points are then extrapolated. This ensures that the typical flight conditions considered in an optimization are included in this study, along with additional scenarios where the degree of coupling is either between or beyond those corresponding to the cruise and maneuver conditions.

This study is performed on a geometry with a planform based on the Boeing 737-900 wing, a half span of 16.13m, and the RAE 2822 airfoil section. An angle of attack of  $2^\circ$  and a freestream Mach number of 0.785 are used in all cases. Surface patches and control points describing the geometry of interest are shown in Figure 8. The flow grid has 458,752 nodes and is divided into 112 blocks. Each block is parameterized by  $6 \times 6 \times 6$  B-spline control points. The internal structure is modeled with 30,473 MITC shell elements and is illustrated in Figure 9. The component thickness values shown in Figure 9 follow a linear variation along the span. Three mesh movement increments are used to obtain the deformed flow grid due to structural deflections. The aerostructural analyses for this study are converged to a tolerance of  $10^{-8}$ , and the coupled adjoint problems are converged to a tolerance of  $10^{-9}$ .

Figure 10 shows an increase in computational time for the coupled analysis with increasing  $q_\infty/E$ . For the smallest  $q_\infty/E$  value considered, the partitioned method and the monolithic solution method require roughly the same amount of computing time. The benefit of using the monolithic strategy becomes more

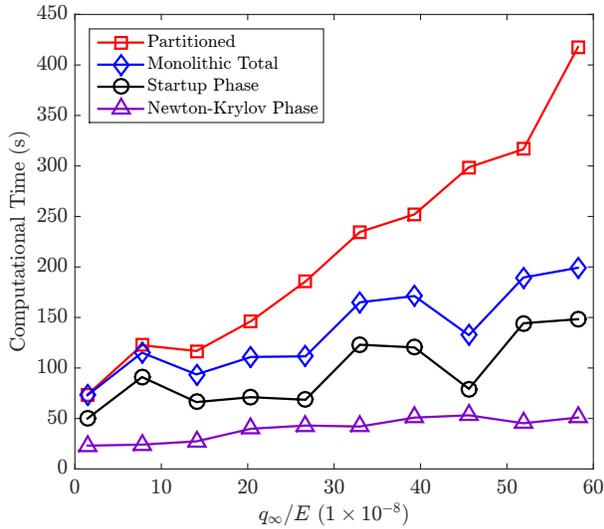


Figure 10: Computational time for analysis plotted with respect to the different  $q_\infty/E$  values at  $M_\infty = 0.785$ .

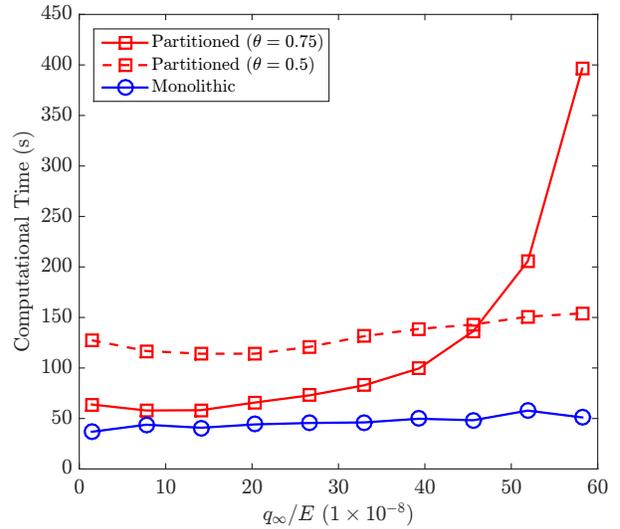


Figure 11: Time for coupled adjoint solution plotted with respect to the different  $q_\infty/E$  values at  $M_\infty = 0.785$ .

apparent as the problem becomes more tightly coupled. At the last data point where  $q_\infty/E$  is the highest, the monolithic solution method results in an overall saving of about 52% in computational time relative to the partitioned method. It is also worth pointing out that the start-up phase of the monolithic solution using partitioned iterations exhibits a more rapid increase in computational time with increasing  $q_\infty/E$  than the Newton-Krylov portion of the algorithm. While this is not surprising, it serves as the motivation for a more robust globalization strategy as part of future work. When the flow grid resolution is doubled in all three coordinate directions keeping the same number of processors, the computational savings become 8% and 54%, respectively, for the lowest and highest  $q_\infty/E$  ratios considered.

Figure 11 shows a similar comparison between the monolithic and the partitioned adjoint calculations. Due to the small differences in the monolithic adjoint convergence between an aerodynamic and a structural functional, as discussed in Section IV.A, the adjoint solution time is averaged between the two types of functionals. For the coupled adjoint solution, the monolithic method consistently outperforms the partitioned method for all values of  $q_\infty/E$ . For smaller  $q_\infty/E$ , the monolithic method is about 42% more efficient than the partitioned method with a relaxation factor of 0.75. However, it is evident that the use of such large relaxation factor becomes inefficient as  $q_\infty/E$  increases. In fact, for the largest  $q_\infty/E$  value, the partitioned method requires over 100 coupled iterations to converge to the specified tolerance with  $\theta = 0.75$ . This is undesirable during an optimization in terms of both robustness and efficiency. A more conservative choice of  $\theta = 0.5$  is likely needed during an optimization where the level of aerostructural interaction is difficult to predict in advance. A relaxation factor of 0.5 is more robust than  $\theta = 0.75$ , but the convergence rate has also been reduced unnecessarily for less challenging cases. In comparison, the monolithic method offers the same

$q_\infty/E$ ( $1 \times 10^{-8}$ )	Tip Deflection (% Semi-Span)	Monolithic Analysis (s)	Monolithic Adjoint (s)	Partitioned Analysis (s)	Partitioned Adjoint (s)
153	4.48	187	73.6	983	694 ( $\theta = 0.25$ )
184	2.72	256	81.4	9060	2922 ( $\theta = 0.1$ )
247	0.24	356	116	Failed	Failed ( $\theta = 0.1$ )

Table 2: The monolithic and partitioned solution time in seconds for a number of challenging scenarios with large  $q_\infty/E$  ratios.

level of robustness, if not more, as the partitioned method with  $\theta = 0.5$ , while being about 60–70% more efficient in terms of the computational time required. When the flow grid resolution is doubled in all three coordinate directions, the monolithic coupled adjoint solution leads to a minimum computational saving of 59% and 68%, respectively, for the lowest and highest  $q_\infty/E$  ratios considered.

To identify the limits of the monolithic and partitioned solution methods,  $q_\infty/E$  values up to four times as large as the last data point listed in Table 1 have also been investigated. The results are summarized in Table 2.<sup>a</sup> Although the partitioned method is able to converge for some of the cases, the required computational time becomes excessive. It is also necessary to lower the initial relaxation factor for Aitken acceleration during analysis, and to choose an increasingly small  $\theta$  for the adjoint calculations. With a  $q_\infty/E$  value of  $2.47 \times 10^{-6}$ , neither the partitioned analysis nor the partitioned adjoint calculations converge successfully despite the use of very conservative relaxation parameters. In contrast, performance of the monolithic solution method is much less sensitive to the choice of solution and physical parameters. Although the partitioned start-up phase still requires adjustments to the initial relaxation factor for Aitken acceleration, the use of an inexact Newton method allows the aerostructural analysis to converge successfully and efficiently in all cases. The slight increase in monolithic solution time with large  $q_\infty/E$  ratios is much more manageable than its partitioned counterpart. These results suggest that the monolithic method is advantageous in terms of efficiency and robustness.

## VI. An Aerostructural Optimization Example

The objective of this section is to investigate the performance of the monolithic solution method during a high-fidelity aerostructural optimization. Intermediate geometries and structural thickness distributions can provide a stiff test of robustness. An optimization where the wing span is allowed to vary can be especially challenging computationally because wings with a light structure or high aspect ratio as a result of optimization can both increase the fluid-structure coupling. The geometry used for this study, along with

<sup>a</sup>The tip deflection decreases with increasing  $q_\infty/E$  in this case because the lift produced by the deflected wing is decreasing, especially near the wing tip which twists downward with deflections. The total lift at  $q_\infty/E = 2.47 \times 10^{-6}$  is only 32% of the lift produced at Data Point 10 from Table 1.

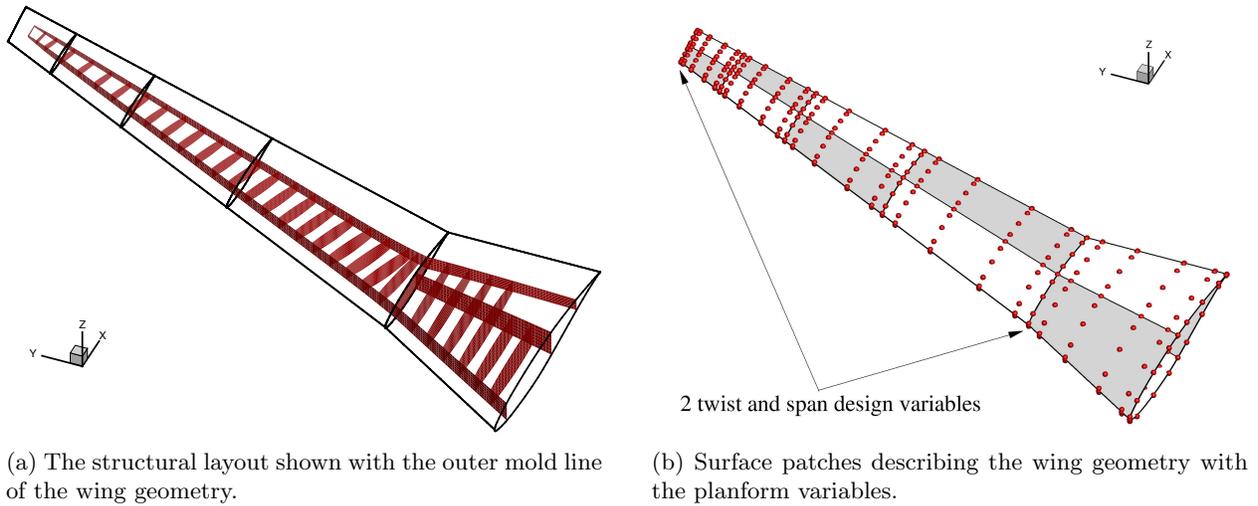


Figure 12: The structural layout and geometry parameterization used for the span optimization.

the structural layout, is shown in Figure 12a. The wing planform is based on the Boeing 737-900 wing, with an initial half span of 16.13m and the RAE 2822 airfoil section. The objective function is given by the following expression:

$$\mathcal{J} = \beta \frac{D}{D_0} + (1 - \beta) \frac{W}{W_0}, \quad (25)$$

where  $\beta$  is a parameter between zero and unity,  $D$  is the inviscid drag produced by the wing of interest, and  $W$  is the weight of the wingbox structural model. Both the drag and weight values are normalized by their respective initial values,  $D_0$  and  $W_0$ , at the start of an optimization. Increasing the wing span has a substantial benefit in reducing the induced drag, which is inversely proportional to the wing span squared. From a structural perspective, however, a higher span incurs a higher root-bending moment, necessitating a heavier structure to ensure feasibility. The optimal span for the wing of interest therefore depends on the relative emphasis on weight and drag in the objective function. Three  $\beta$  values of 0.25, 0.5, and 0.75 are considered, where a larger  $\beta$  places more emphasis on drag reduction.

For each optimization, the value of  $D$  in (25) is evaluated at the cruise condition. A 2.5g maneuver condition is added to size the structures, which determines the value of  $W$ . The cruise condition assumes a Mach number of 0.785 and an altitude of 35,000 feet. The maneuver condition assumes a Mach number of 0.789 and an altitude of 12,000 feet. The lift at each load condition is constrained to the estimated weight of the aircraft scaled by the appropriate load factor. The aircraft weight is in turn obtained by adding the weight of the wing predicted by the finite-element model to a fixed weight of 785,000N. The weight of the wing incorporates a mark-up factor of 1.5 to account for any components not included in the structural model.<sup>7</sup> Structural constraints are imposed at the maneuver condition to ensure that the von Mises stresses in the structures do not exceed the yield stress of the material. Three KS functions are used to aggregate the

failure constraints on the ribs and spars, the top skin, and the bottom skin of the wingbox, respectively. The structural sizing in practical wing design considers many additional load conditions. For the purpose of this study, however, it suffices to capture the correct trends in structural sizing by using a single 2.5g maneuver load condition. A more conservative safety factor of 2 is used as a result. While buckling constraints are not included for the same reason, this does not affect the comparison between the monolithic and partitioned methods.

There are 2 angle of attack design variables, one for each load condition. The thickness of the structural components in Figure 12a is controlled by 156 design variables, which are initialized to a uniform thickness of 10mm. This is only intended to be a starting point for the optimization, which likely produces unrealistic deflections and internal load distributions. The optimization will ensure a more realistic final structural sizing upon satisfying all lift and structural constraints. Figure 12b illustrates the geometry parameterization of the wing using B-spline surfaces, where the surface control points are shown as red spheres. There are two span and two twist design variables as indicated. The optimizer can additionally modify the airfoil sections along the span. This results in a total of 228 geometric design variables.

The flow grid used for the analysis at each design iteration has 112 blocks. Each block is parameterized with  $6 \times 6 \times 6$  control points. The optimization is first performed on a coarse flow grid with 193,536 nodes. Upon satisfying all nonlinear constraints on the coarse grid, the optimization is continued on a flow grid with 458,752 nodes. Due to the grid dependence of the flow solution, a post-optimality analysis using higher flow grid resolution is usually needed to predict the lift and drag values with sufficient accuracy. However, the main focus of this section is on performing the optimization, and past experience indicates that the chosen flow grid resolutions are sufficient for this purpose.<sup>9,51</sup> The structural mesh resolution is fixed during both stages of the optimization with 30,473 MITC shell elements, or 174,204 degrees of freedom.

Table 3 summarizes the results of the optimization in terms of the wing span, estimated weight of the aircraft each wing must carry (half the total weight), and the inviscid drag of the wing, for the three  $\beta$  values. It can be observed that when more emphasis is placed on drag by increasing  $\beta$ , the optimized span has indeed increased at the cost of a higher weight. The resulting increase in weight can be explained by examining the optimized thickness distribution on the upper and lower skins in Figure 13 for the three  $\beta$  values. It is clear that the increase in the span has led to an increase in the thickness of the skin panels in order to maintain structural integrity. The optimizer has also increased the thickness inboard for all  $\beta$  values, which is in agreement with the results reported by other authors for similar studies.<sup>52</sup>

The last design iteration on the fine flow grid as well as the optimization on the coarse flow grid are repeated using the partitioned method. The partitioned adjoint solution with  $\theta = 0.75$  fails to converge at the maneuver condition for  $\beta = 0.5$  and  $\beta = 0.75$ . A more conservative relaxation parameter of  $\theta = 0.5$

	Half Span (m)	Weight (N)	Drag (N)
$\beta = 0.25$	17.3	$4.15 \times 10^5$	$1.57 \times 10^4$
$\beta = 0.50$	21.4	$4.28 \times 10^5$	$1.09 \times 10^4$
$\beta = 0.75$	24.8	$4.48 \times 10^5$	$8.54 \times 10^3$

Table 3: Optimized half span, weight of the aircraft (half), and drag (one wing) for the three  $\beta$  values.

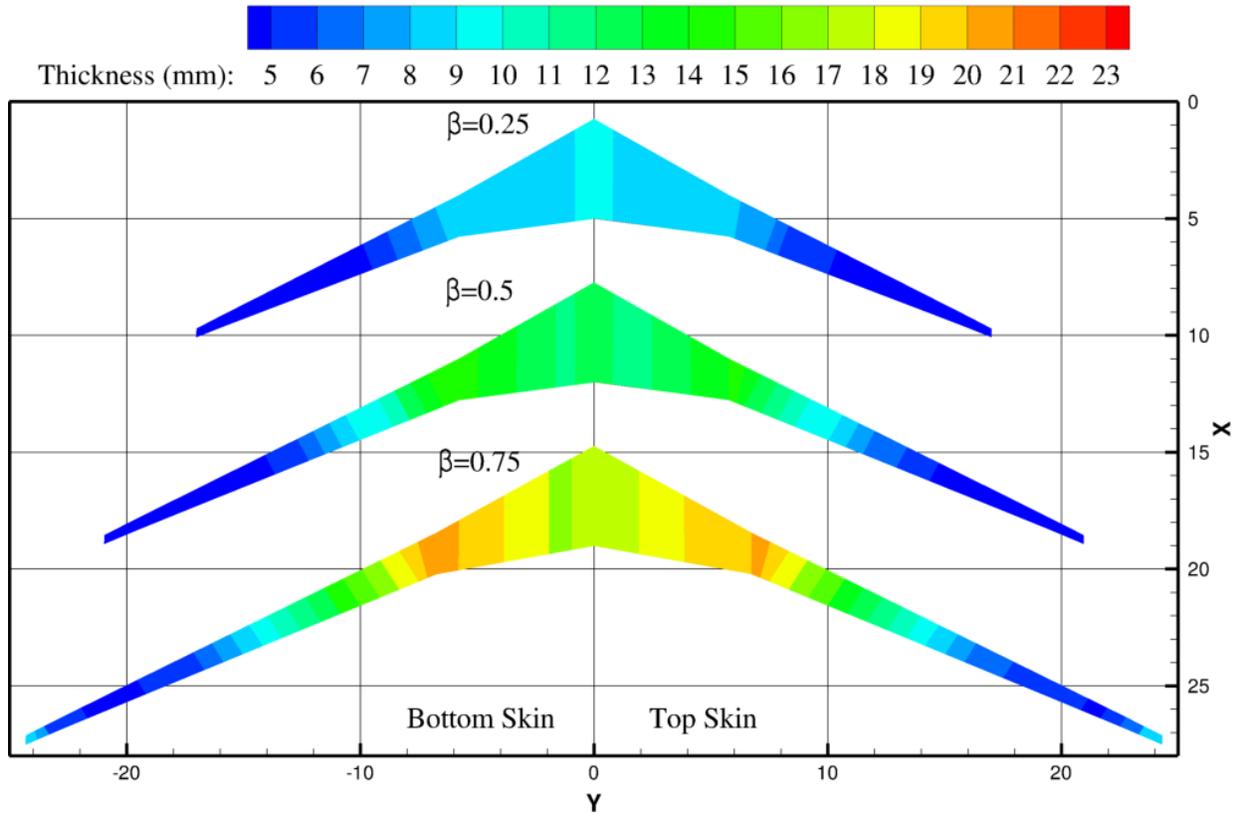


Figure 13: Optimized thickness distribution on the upper and lower skins for different  $\beta$  values.

		Time per design iteration (min)		Percentage reduction in computational time
		Monolithic	Partitioned	
Fine	$\beta = 0.25$	7.55	20.30	62.8%
	$\beta = 0.5$	8.18	18.77	56.4%
	$\beta = 0.75$	10.77	27.00	60.1%
Coarse	$\beta = 0.25$	3.75	7.64	50.9%
	$\beta = 0.5$	3.53	7.78	54.6%
	$\beta = 0.75$	3.53	9.93	64.5%

Table 4: The computational cost of one design iteration using the monolithic and partitioned methods at different  $\beta$  values and flow grid resolutions. The fine grid values correspond to the final design iteration, whereas those reported for the coarse grid are averaged over the course of the optimization.

is thus necessary. The computational times of one design iteration using the two methods are compared in Table 4 for different  $\beta$  values. The calculations are performed using a total of 240 processors and the results on the coarse and fine flow grids are both included. In all cases, the use of the monolithic solution method leads to an efficiency improvement of 50% to over 60% compared to the partitioned method, allowing the same number of design iterations to be performed in less than half the time.

When the optimization on the coarse flow grid is repeated using the partitioned method, the coupled analysis and adjoint calculations require a noticeably increased number of iterations as the span increases with higher  $\beta$ . The monolithic solution method demonstrates a consistent performance in that all analysis and adjoint calculations converge well within 20 iterations, even in the most challenging scenarios. The partitioned method, on the other hand, requires more than 30 analysis iterations and over 50 coupled adjoint iterations for the maneuver calculations with  $\beta = 0.75$ . The recent advances in composite materials allow for wings with larger spans and lower weights such as the Boeing 787 wing. These wings tend to have larger tip deflections than their conventional counterparts. Using the partitioned method for the design of such wings could require very conservative relaxation factors and a potentially unpredictable amount of computational time. The above arguments thus favor strongly the use of the monolithic solution method for the design of highly flexible wings, for its advantage in efficiency and reliable performance.

## VII. Conclusions

An effective monolithic solution strategy has been proposed for the high-fidelity aerostructural optimization methodology by Zhang *et al.*,<sup>9</sup> which has led to substantial efficiency improvements relative to a partitioned method. A Newton-Krylov solution algorithm was developed for the coupled analysis problem and a preconditioned Krylov subspace method, GCROT( $m, k$ ), was used for the monolithic solution of the coupled adjoint problem. The importance of scaling during the monolithic solution process has been established, and appropriate scaling strategies have been proposed. For the monolithic analysis, the Jacobian matrix-vector product is obtained using a combination of matrix-free approximation and matrix-explicit exact differentiation. A block Gauss-Seidel preconditioner that reuses existing iterative linear system solution routines in the flow, mesh, and structural modules has further been recommended for both analysis and coupled adjoint solutions, along with the optimal choices of preconditioning parameters.

The advantages of the monolithic solution method have been clearly demonstrated. For the test problem chosen to investigate the design of the monolithic solution technique, the monolithic method improves the efficiency of the coupled analysis by 48% relative to the partitioned method. The monolithic adjoint solution is 60% more efficient than the partitioned method. A range of aerostructural analysis and coupled adjoint problems with incremental increases in  $q_\infty/E$  values is presented to study the performance of the monolithic

and partitioned methods with varying degrees of coupling in the physics. For more tightly coupled problems with larger  $q_\infty/E$  values, the monolithic analysis is 50%–60% more efficient than the partitioned method and remains competitive for problems with low  $q_\infty/E$  values. In terms of the coupled adjoint calculations for problems with larger  $q_\infty/E$  values, the monolithic adjoint solution demonstrates superior robustness in comparison to the partitioned method with a relaxation parameter of 0.75, while still being over 40% more efficient for problems with weaker fluid-structure coupling. Compared to the partitioned adjoint calculations with a relaxation parameter of 0.5, the monolithic method is 60%–70% more efficient consistently for the range of  $q_\infty/E$  values considered. The robustness of the solution methods is further investigated for highly challenging problems with large  $q_\infty/E$  values. The advantage of the monolithic method is evident from its ability to converge and maintain efficiency when the partitioned method fails even with the use of very conservative solver parameters. For optimization studies allowing for changes in the wing span, the monolithic solution method is 50% to over 60% faster than the partitioned method in terms of the computational time per design iteration. The efficiency improvements from the use of the monolithic solution method allow, on average, the same optimization to be completed in less than half the time. In addition, the monolithic solution method shows a consistent performance that is not sensitive to the increase in span with higher  $\beta$  values, and the optimal solver parameters are less problem dependent. This further substantiates the suitability of the monolithic solution method for optimization involving highly flexible wings.

The globalization of the inexact Newton method is currently achieved by applying a number of nonlinear block Gauss-Seidel iterations. A more effective globalization strategy is an important next step. A potentially promising approach could involve using the monolithic homotopy continuation algorithm proposed recently by Brown and Zingg,<sup>53</sup> which has been applied successfully to high-fidelity flow analysis. As a possible way to implement this technique for aerostructural analysis, the homotopy could start with the flow solution on the undeflected geometry. The continuation parameter can then be used to gradually increase the structural deflection while converging the flow, mesh, and structural solutions simultaneously until the final deflected geometry is recovered. The monolithic solution techniques discussed here may also be applicable to viscous and unsteady simulations using implicit time-marching, where similar percentage savings reported here can translate to substantial differences in computational time.

## Acknowledgments

The authors would like to acknowledge Prof. J. R. R. A. Martins at the University of Michigan, Ann Arbor for sharing his framework for the purpose of constructing our methodology. The authors are also grateful for the funding provided by Zonta International Amelia Earhart Fellowships and the Ontario Graduate Scholarship. Computations were performed on the GPC supercomputer at the SciNet HPC Consortium.

SciNet is funded by the Canada Foundation for Innovation under the auspices of Compute Canada, the Government of Ontario, Ontario Research fund - Research Excellence, and the University of Toronto.

## A. Test Problem for Sections III and IV

A test problem is introduced to investigate the effects of various solver parameters in Sections III and IV. A wing with planform based on the Boeing 737-900 wing, as shown in Figure 8, and with the RAE 2822 airfoil section is used for this purpose. It is operating at a Mach number of 0.798, an altitude of 12,000 feet, and an angle of attack of  $2^\circ$ . The flow grid has a total of 458,752 nodes and is partitioned into 112 blocks, where each block is controlled by  $6 \times 6 \times 6$  B-spline volume control points. The wingbox model from Figure 12a is used for the aerostructural analysis and it has a total of 171,960 degrees of freedom. Properties for the Aluminum structures are listed in Appendix B. A uniform component thickness distribution of 10mm is assumed because it is often used as a starting point for an aerostructural optimization. Although the resulting deflections may not be realistic, it serves as an appropriate test case for studying the performance of the monolithic solution method. Three mesh movement increments are used for the coupled calculations. The flow and mesh subproblems are distributed over 112 Intel Xeon E5540 processors, whereas the structural subproblem is distributed over 8 processors. Each processor operates at 2.53GHz and has 2GB of memories available.

## B. Material Properties: Aluminum

The Aluminum structures used throughout this paper has a yield stress of 0.324GPa, a Young's modulus of 73.1GPa, a density of 2780kg/m<sup>3</sup>, and a Poisson ratio of 0.33.

## References

<sup>1</sup>Reuther, J. J., Alonso, J. J., Martins, J. R. R. A., and Smith, S. C., "A Coupled Aero-Structural Optimization Method For Complete Aircraft Configurations," *AIAA 37th Aerospace Sciences Meeting*, 1999, pp. AIAA-99-0187.

<sup>2</sup>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.

<sup>3</sup>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. doi:[10.1002/nme.599](https://doi.org/10.1002/nme.599).

<sup>4</sup>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.

<sup>5</sup>Martins, J. R. R. A., Alonso, J. J., and Reuther, J. J., "A Coupled-Adjoint Sensitivity Analysis Method for High-Fidelity Aero-Structural Design," *Optimization and Engineering*, Vol. 6, 2005, pp. 33-62.

<sup>6</sup>Kenway, G. K. W., 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, 2014, pp. 935–951. doi:[10.2514/1.J052255](https://doi.org/10.2514/1.J052255).

<sup>7</sup>Kennedy, G. J. and Martins, J. R., “A parallel aerostructural optimization framework for aircraft design studies,” *Structural and Multidisciplinary Optimization*, Vol. 50, No. 6, 2014, pp. 1079–1101. doi:[10.1007/s00158-014-1108-9](https://doi.org/10.1007/s00158-014-1108-9).

<sup>8</sup>Kenway, G. K. W., Kennedy, G. J., and Martins, J. R. R. A., “Aerostructural optimization of the Common Research Model configuration,” *15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Atlanta, GA, June 2014.

<sup>9</sup>Zhang, Z. J., Khosravi, S., and Zingg, D. W., “High-Fidelity Aerostructural Optimization with Integrated Geometry Parameterization and Mesh Movement,” *Structural and Multidisciplinary Optimization*, 2016. doi:[10.1007/s00158-016-1562-7](https://doi.org/10.1007/s00158-016-1562-7), (Accepted).

<sup>10</sup>Pironneau, O., “On Optimum Design in Fluid Mechanics,” *Journal of Fluid Mechanics*, Vol. 64, No. 1, 1974, pp. 97–110.

<sup>11</sup>Jameson, A., “Aerodynamic Design via Control Theory,” *Journal of Scientific Computing*, Vol. 3, No. 3, 1988, pp. 233–260.

<sup>12</sup>Zingg, D. W., Nemec, M., and Pulliam, T. H., “A Comparative Evaluation of Genetic and Gradient-based Algorithms Applied to Aerodynamic Optimization,” *Revue européenne de mécanique numérique*, Vol. 17, No. 1-2, March 2008, pp. 103–126. doi:[10.3166/remn.17.103-126](https://doi.org/10.3166/remn.17.103-126).

<sup>13</sup>Barcelos, M., Bavestrello, H., and Maute, K., “A Schur-Newton-Krylov solver for steady-state aeroelastic analysis and design sensitivity analysis,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 195, 2006, pp. 2050–2069.

<sup>14</sup>Michler, C., Hulshoff, S., van Brummelen, E., and de Borst, R., “A monolithic approach to fluid-structure interaction,” *Computers & Fluids*, Vol. 33, No. 5-6, 2004, pp. 839 – 848. doi:[10.1016/j.compfluid.2003.06.006](https://doi.org/10.1016/j.compfluid.2003.06.006), Applied Mathematics for Industrial Flow Problems.

<sup>15</sup>Heil, M., Hazel, A. L., and Boyle, J., “Solvers for large-displacement fluid-structure interaction problems: segregated versus monolithic approaches,” *Computational Mechanics*, Vol. 43, No. 1, 2008, pp. 91–101.

<sup>16</sup>Gee, M. W., Küttler, U., and Wall, W. A., “Truly monolithic algebraic multigrid for fluid-structure interaction,” *International Journal for Numerical Methods in Engineering*, Vol. 85, No. 8, 2011, pp. 987–1016. doi:[10.1002/nme.3001](https://doi.org/10.1002/nme.3001).

<sup>17</sup>Wu, Y. and Cai, X.-C., “A fully implicit domain decomposition based {ALE} framework for three-dimensional fluid-structure interaction with application in blood flow computation,” *Journal of Computational Physics*, Vol. 258, No. 0, 2014, pp. 524 – 537. doi:[10.1016/j.jcp.2013.10.046](https://doi.org/10.1016/j.jcp.2013.10.046).

<sup>18</sup>van Zuijlen, A., Bosscher, S., and Bijl, H., “Two level algorithms for partitioned fluid-structure interaction computations,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 196, No. 8, 2007, pp. 1458 – 1470. doi:[10.1016/j.cma.2006.03.014](https://doi.org/10.1016/j.cma.2006.03.014).

<sup>19</sup>van Zuijlen, A. H. and Bijl, H., “Multi-Level Accelerated Sub-Iterations for Fluid-Structure Interaction,” *Fluid Structure Interaction II: Modelling, Simulation, Optimization*, edited by H.-J. Bungartz, M. Mehl, and M. Schäfer, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 1–25. doi:[10.1007/978-3-642-14206-2\\_1](https://doi.org/10.1007/978-3-642-14206-2_1).

<sup>20</sup>Scholcz, T., van Zuijlen, A., and Bijl, H., “Space-mapping in fluid-structure interaction problems,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 281, 2014, pp. 162 – 183. doi:[10.1016/j.cma.2014.07.028](https://doi.org/10.1016/j.cma.2014.07.028).

<sup>21</sup>Blom, D., van Zuijlen, A., and Bijl, H., “Multi-level acceleration with manifold mapping of strongly coupled partitioned fluid-structure interaction,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 296, 2015, pp. 211 – 231. doi:[10.1016/j.cma.2015.08.004](https://doi.org/10.1016/j.cma.2015.08.004).

<sup>22</sup>Degroote, J., Bathe, K.-J., and Vierendeels, J., “Performance of a new partitioned procedure versus a monolithic procedure in fluid-structure interaction,” *Computers & Structures*, Vol. 87, 2009, pp. 793–801.

<sup>23</sup>van Brummelen, E., Hulshoff, S., and de Borst, R., “Energy conservation under incompatibility for fluid-structure in-

teraction problems,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 192, No. 25, 2003, pp. 2727 – 2748. doi:[10.1016/S0045-7825\(03\)00287-1](https://doi.org/10.1016/S0045-7825(03)00287-1).

<sup>24</sup>Hübner, B., Walhorn, E., and Dinkler, D., “A monolithic approach to fluid-structure interaction using space-time finite elements,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 193, No. 23-26, 2004, pp. 2087–2104. doi:[10.1016/j.cma.2004.01.024](https://doi.org/10.1016/j.cma.2004.01.024).

<sup>25</sup>Ryzhakov, P. B., Rossi, R., Idelsohn, S. R., and Oñate, E., “A monolithic Lagrangian approach for fluid-structure interaction problems,” *Computational Mechanics*, Vol. 46, No. 6, Nov 2010, pp. 883–899. doi:[10.1007/s00466-010-0522-0](https://doi.org/10.1007/s00466-010-0522-0).

<sup>26</sup>Jog, C. S. and Pal, R. K., “A monolithic strategy for fluid-structure interaction problems,” *International Journal for Numerical Methods in Engineering*, Vol. 85, No. 4, Jan 28 2011, pp. 429–460. doi:[10.1002/nme.2976](https://doi.org/10.1002/nme.2976).

<sup>27</sup>Tezduyar, T. E. and Sathe, S., “Modelling of fluid-structure interactions with the space-time finite elements: Solution techniques,” *International Journal for Numerical Methods in Fluids*, Vol. 54, No. 6-8, 2007, pp. 855–900. doi:[10.1002/flid.1430](https://doi.org/10.1002/flid.1430).

<sup>28</sup>Bazilevs, Y., Calo, V. M., Hughes, T. J. R., and Zhang, Y., “Isogeometric fluid-structure interaction: theory, algorithms, and computations,” *Computational Mechanics*, Vol. 43, No. 1, 2008, pp. 3–37. doi:[10.1007/s00466-008-0315-x](https://doi.org/10.1007/s00466-008-0315-x).

<sup>29</sup>Badia, S., Nobile, F., and Vergara, C., “Robin-Robin preconditioned Krylov methods for fluid-structure interaction problems,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 198, No. 33–36, 2009, pp. 2768–2784.

<sup>30</sup>Crosetto, P., Deparis, S., Fourestey, G., and Quarteroni, A., “Parallel Algorithms for Fluid-Structure Interaction Problems in Haemodynamics,” *SIAM Journal on Scientific Computing*, Vol. 33, No. 4, 2011, pp. 1598–1622. doi:[10.1137/090772836](https://doi.org/10.1137/090772836).

<sup>31</sup>Heil, M., “An efficient solver for the fully coupled solution of large-displacement fluid-structure interaction problems,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 193, 2004, pp. 1–23.

<sup>32</sup>Barker, A. T. and Cai, X.-C., “Scalable parallel methods for monolithic coupling in fluid-structure interaction with application to blood flow modeling,” *Journal of Computational Physics*, Vol. 229, No. 3, 2010, pp. 642 – 659. doi:[10.1016/j.jcp.2009.10.001](https://doi.org/10.1016/j.jcp.2009.10.001).

<sup>33</sup>Badia, S., Quaini, A., and Quarteroni, A., “Modular vs. non-modular preconditioners for fluid-structure systems with large added-mass effect,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 197, 2008, pp. 4216–4232.

<sup>34</sup>Elham, A., van Tooren, and L., M. J., “Coupled adjoint aerostructural wing optimization using quasi-three-dimensional aerodynamic analysis,” *Structural and Multidisciplinary Optimization*, 2016, pp. 1–18. doi:[10.1007/s00158-016-1447-9](https://doi.org/10.1007/s00158-016-1447-9).

<sup>35</sup>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*, Fort Worth, Texas, September 2010, AIAA-2010-9308.

<sup>36</sup>Hicken, J. E. and Zingg, D. W., “Aerodynamic Optimization Algorithm with Integrated Geometry Parameterization and Mesh Movement,” *AIAA Journal*, Vol. 48, No. 2, 2010, pp. 400–413.

<sup>37</sup>Zhang, J. Z., *Exploratory High-Fidelity Aerostructural Optimization using an Efficient Monolithic Solution Method*, Ph.D. thesis, University of Toronto, January 2017.

<sup>38</sup>Hicken, J. E. and Zingg, D. W., “Parallel Newton-Krylov Solver for the Euler Equations Discretized Using Simultaneous-Approximation Terms,” *AIAA Journal*, Vol. 46, No. 11, 2008, pp. 2773–2786.

<sup>39</sup>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, September 2014, pp. 56–73. doi:[10.1016/j.finel.2014.04.011](https://doi.org/10.1016/j.finel.2014.04.011).

<sup>40</sup>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. doi:[10.1002/nme.1620010306](https://doi.org/10.1002/nme.1620010306).

<sup>41</sup>Küttler, U. and Wall, W. A., “Fixed-point fluid-structure interaction solvers with dynamic relaxation,” *Computational Mechanics*, Vol. 43, No. 1, 2008, pp. 61–72. doi:[10.1007/s00466-008-0255-5](https://doi.org/10.1007/s00466-008-0255-5).

<sup>42</sup>Hicken, J. E. and Zingg, D. W., “A Simplified and Flexible Variant of GCROT for Solving Nonsymmetric Linear Systems,” *SIAM Journal on Scientific Computing*, Vol. 32, No. 3, June 2010, pp. 1672–1694. doi:[10.1137/090754674](https://doi.org/10.1137/090754674).

<sup>43</sup>Eisenstat, S. C. and Walker, H. F., “Choosing the Forcing Terms in an Inexact Newton Method,” *SIAM Journal on Scientific Computing*, Vol. 17, No. 1, 1996, pp. 16–32. doi:[10.1137/0917003](https://doi.org/10.1137/0917003).

<sup>44</sup>Chisholm, T. T. and Zingg, D. W., “A Jacobian-free Newton-Krylov algorithm for compressible turbulent fluid flows,” *J. Comput. Physics*, Vol. 228, No. 9, 2009, pp. 3490–3507.

<sup>45</sup>Amestoy, P. R., Duff, I. S., Ruiz, D., and Uçar, B., “A Parallel Matrix Scaling Algorithm,” *High Performance Computing for Computational Science - VECPAR 2008: 8th International Conference, Toulouse, France, June 24-27, 2008. Revised Selected Papers*, edited by J. M. L. M. Palma, P. R. Amestoy, M. Daydé, M. Mattoso, and J. C. Lopes, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 301–313. doi:[10.1007/978-3-540-92859-1\\_27](https://doi.org/10.1007/978-3-540-92859-1_27).

<sup>46</sup>Golub, G. H. and Van Loan, C. F., *Matrix Computations (3rd Ed.)*, John Hopkins University Press, Baltimore, MD, USA, 1996.

<sup>47</sup>Nielsen, E. J., Walters, R. W., Anderson, W. K., and Keyes, D. E., “Application of Newton-Krylov methodology to a three-dimensional unstructured Euler code,” *12th AIAA Computational Fluid Dynamics Conference*, San Diego, CA, 1995, AIAA-95-1733.

<sup>48</sup>Saad, Y., *Iterative Methods for Sparse Linear Systems*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2nd ed., 2003.

<sup>49</sup>Wrenn, G. A., “An Indirect Method for Numerical Optimization Using the Kreisselmeier–Steinhauser Function,” Tech. Rep. CR-4220, NASA, 1989.

<sup>50</sup>Akgün, M. A., Haftka, R. T., Wu, K. C., and Walsh, J. L., “Sensitivity of lumped constraints using the adjoint method,” *40th Structures, Structural Dynamics, and Materials Conference and Exhibit*, St. Louis, Missouri, April 1999, AIAA-99-1314.

<sup>51</sup>Khosravi, S. and Zingg, D. W., “An Aerostructural Perspective on Winglets,” *Journal of Aircraft*, 2017, (Accepted).

<sup>52</sup>Kennedy, G. J., Kenway, G. K. W., and Martins, J. R. R. A., “High Aspect Ratio Wing Design: Optimal Aerostructural Tradeoffs for the Next Generation of Materials,” *Proceedings of the AIAA Science and Technology Forum and Exposition (SciTech)*, National Harbor, MD, January 2014, AIAA-2014-0596.

<sup>53</sup>Brown, D. A. and Zingg, D. W., “A monolithic homotopy continuation algorithm with application to computational fluid dynamics,” *Journal of Computational Physics*, Vol. 321, 2016, pp. 55–75. doi:[10.1016/j.jcp.2016.05.031](https://doi.org/10.1016/j.jcp.2016.05.031).