

A PANEL-FREE AEROELASTIC SOLVER WITH ADJOINT SENSITIVITIES

Hollis A. Smith¹ and Joshua D. Deaton¹

¹Multidisciplinary Science and Technology Center
Air Force Research Laboratory, Wright-Patterson AFB, OH, 45433, USA
hollis.smith.ctr@us.af.mil
joshua.deaton@us.af.mil

Keywords: computational, aeroelasticity, feature mapping, topology, adjoint sensitivity

Abstract: This paper presents the theoretical development of a novel approach to solving potential-based lifting pressures for aeroelastic analysis that circumvents the need to perturb or remesh the analysis domain. Inspired by feature-mapping approaches used in structural topology optimization, the method maps lifting surfaces and their wakes to a fixed, non-conforming grid wherein an efficient multi-grid solver resolves the lifting pressures. The coupled aeroelastic state is resolved by a consistent and conservative transfer of loads and displacements between disciplines. To facilitate efficient gradient-based optimization, all of the employed mappings are differentiable, thus enabling efficient adjoint computation of design-sensitivities for quantities dependent on the coupled aeroelastic state. The computational efficiency gained by circumventing the need to rebuild or perturb the mesh for each analysis, the topological design flexibility achieved by mapping to an implicit representation, and the efficient adjoint computation of design sensitivities make the proposed aeroelastic solver an attractive alternative to the conventional approach.

1 INTRODUCTION

The conventional approach to medium/high-fidelity aeroelastic analysis involves constructing a surface or fluid-domain mesh to estimate lifting pressures, constructing a finite-element mesh of the substructure to estimate deformations, and performing a consistent and conservative interpolation scheme to simultaneously resolve the coupled state equations [1–4]. Both the structural and the aero meshes are generally constructed from an explicit geometric representation that does not readily accommodate topological changes. When used in design optimization, the conventional approach requires the costly re-evaluation of topological operations like geometric intersections, unions and subtractions to re-build the geometry, followed by an expensive re-meshing step for each analysis.

For decades, topology optimization methods [5] have circumvented the need to remesh upon design changes when solving structural displacements on a fixed mesh by employing a field representation of the geometry. In more recent years, feature-mapping methods [6] have enabled topology optimization with high-level, parameterized geometry by mapping the design to an implicit field representation and solving the displacements on a fixed mesh. This implicit representation naturally and robustly transforms topological operations into inexpensive arithmetic operations. In this work, we leverage the feature-mapping paradigm to circumvent the the need to rebuild or perturb analysis meshes in an aeroelastic analysis.

Panel methods are some of the earliest examples of boundary element methods [7]. Although their main attraction is to restrict the solution space to the the boundary of the domain, boundary element methods are, for many problems, significantly less efficient than volumetric approaches (e.g., the finite element method). This is because the boundary element method employs singular Green's functions (fundamental solutions for given boundary conditions) that generally do not have local support, instead requiring each singularity to be integrated over the entire discretization. This gives rise to dense, fully-populated matrices that have storage and computational costs that grow asymptotically as $O(n^2)$.

The asymptotic time complexity of boundary element methods can be improved by using multipole expansions of the truncated Taylor series approximations of the Green's function. For example, the tree algorithm of Barnes and Hut [8] results in $O(n \log n)$, and the fast-multipole method [9] in $O(n)$ time complexity. However, the trade-off between truncation error and solution speed is often significant.

This work forgoes the conventional boundary element approach of panel methods, instead using the extended finite element method (XFEM). XFEM originated to minimize the re-meshing of cracks in fracture mechanics [10]. In this work, XFEM enables using a fixed, non-conforming grid to solve a perturbation potential that is discontinuous across the immersed boundary and its wake. The analysis grid is invariant to design changes and elastic deformations that would normally require mesh perturbation or even complete re-meshing.

The local support of the finite-element basis functions results in $O(n)$ computational and storage requirements, where n is the number of unknowns. Although the number of unknowns in the volumetric discretization is larger than that of the surface discretization, the sparsity of the resulting equations results in better scaling. Moreover, an efficient multigrid-preconditioned solver enables the iterative solution of the problem in a fixed number of iterations regardless of grid resolution. Although outside the scope of this work, we also mention that n can be significantly reduced by employing adaptive octree meshes [11].

2 METHODOLOGY

2.1 Geometric representations

The design variables are high level geometric parameters that map to a different geometric view for each discipline.

For the aero discipline, the solid-fluid interface surface $\Gamma_{\hat{B}}$ that bounds the interior region $\Omega_{\hat{B}}$ of the reference configuration is deformed by the structural displacements \mathbf{u} to the deformed interface surface $\Gamma_B = \{\hat{\mathbf{y}} + \mathbf{u}(\hat{\mathbf{y}}) : \hat{\mathbf{y}} \in \Gamma_{\hat{B}}\}$ that bounds the deformed interior region Ω_B .

A surface discretization may be obtained by tessellating the trimmed parametric surfaces in a boundary representation (B-rep), or by solving the zero level set of an implicit representation such as the field obtained by deforming the signed distance to the reference configuration (negative inside Ω_B) as defined in equation (1b).

$$s_{\Omega_{\hat{B}}}(\mathbf{x}) = \text{sgn}_{\Omega_{\hat{B}}}(\mathbf{x}) \min_{\hat{\mathbf{y}} \in \partial\Omega_{\hat{B}}} \|\mathbf{x} - \hat{\mathbf{y}}\|, \text{ for } \mathbf{x} \in \Omega \supset \Omega_{\hat{B}} \quad (1a)$$

$$\tilde{s}_{\Omega_B}(\mathbf{x}) = s_{\Omega_{\hat{B}}}(\mathbf{x} - \mathbf{u}(\mathbf{x})), \text{ for } \mathbf{x} \in \Omega \supset \Omega_B \quad (1b)$$

After deformation, \tilde{s}_{Ω_B} no longer has a unit gradient, but its zero level set corresponds to $\partial\Omega_B \equiv \Gamma_B$, and thus the direction of its gradient at the boundary is the outward surface normal of the deformed configuration.

Although a surface discretization tracks the deformed interface, it is independent of the grid used in each solver; it merely serves as a common interface to compute surface integrals, to transfer displacements loads, and to map design sensitivities. Thus, its resolution should match or exceed the grid resolution of each discipline, but element quality is of little concern.

The geometric view for the structural discipline is given by unioning a set of components $\{\Omega_c\}_{c=1}^{n_c}$, intersecting the union with the interior region $\Omega_{\hat{B}}$ and subsequently unioning with the skin Ω_S . In set notation, the set of points in the elastic structural volume is defined as

$$\Omega_E = \Omega_S \cup (\Omega_{\hat{B}} \cap (\cup_c \Omega_c)) \quad (2)$$

In a typical B-rep geometry kernel, these Boolean operations are expensive to compute and can fail to represent multiple disconnected regions. Implicit functions can represent open sets like the elastic structural region as $\Omega_E = \{\mathbf{x} : \chi_{\Omega_E}(\mathbf{x}) > 0\}$, where the Boolean operations of equation (2) become the robust and inexpensive min/max operations of equation (3)

$$\chi_{\Omega_E}(\mathbf{x}) = \max(\chi_{\Omega_S}(\mathbf{x}), \min(\chi_{\Omega_{\hat{B}}}(\mathbf{x}), \max_c \chi_{\Omega_c}(\mathbf{x}))) \quad (3)$$

In the geometry projection method [12], the implicit functions $\chi_{\Omega_c}(\mathbf{x}) \in [0, 1]$ are volume fractions (densities). The use of volume-fraction fields enables fractional membership of complete geometric components in the design, facilitating topological changes as in density based topology optimization. When used in gradient-based optimization, the characteristic functions are smooth across the zero level set, and the min/max functions are replaced with smooth approximations.

The geometry is passed to the structural solver implicitly as the volume-fraction (density) function of equation (3) where, as in [13], the density function for each component is computed as the smooth Heaviside of the signed distance function as

$$\chi_{\Omega_c}(\mathbf{x}) = \tilde{H}(-s_{\Omega_c}(\mathbf{x})/r) \quad (4)$$

where r is the radius of the region where χ_{Ω_c} smoothly transitions from 0 to 1.

Although the signed distance to certain reference shapes can be expressed analytically, this becomes impossible for general shapes. An approximate solution of equation (1a) is given by discretizing the reference configuration and computing the signed distance to the discretization. The closest-point search on the surface discretization is accelerated by building a bounding volume hierarchy of the surface elements [14].

2.2 Aero load computation

As mentioned in the introduction, this work forgoes the conventional boundary-element approach of panel methods, instead using the volume-based extended finite element method (XFEM). XFEM enables modeling fields with discontinuities across immersed boundaries without remeshing the analysis domain. In the case of lifting perturbation potential, XFEM naturally models jump discontinuities across the boundary of the deformed configuration and/or its wake. Discontinuities are modeled by extending the standard finite element approximation with enrichment functions. Figure 1 illustrates an XFEM approximation of a jump discontinuity across the boundary of an airfoil on a 2d grid.

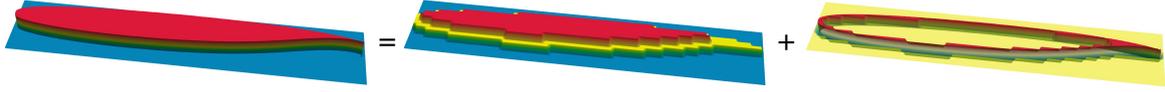


Figure 1: XFEM model of a jump discontinuity (left) represented by extending the standard shape functions (middle) with enrichment shape functions (right).

2.2.1 The strong form

To simplify the presentation, we discuss only incompressible potential-flow governed by the Laplace equation, noting that linearized compressible potential flow may also be considered by applying the Prandtl–Glauert transformation [15] and Göthert’s inverse transformation [16] as additional pre- and post-processing steps.

The perturbation potential about an immersed body with boundary Γ_B and a wake surface Γ_W extending from a sharp trailing edge γ_E is governed on a domain $\Omega \supset (\Gamma_B \cup \Gamma_W)$ with a boundary $\partial\Omega$ at infinity by equations (5a–f).

$$-\Delta\phi = 0 \quad \text{on } \Omega \setminus (\Gamma_B \cup \Gamma_W) \quad (5a)$$

$$\phi = 0 \quad \text{on } \partial\Omega \quad (5b)$$

$$\partial_n\phi = -V_n^\infty \quad \text{on } \Gamma_B^+ \quad (5c)$$

$$\phi = 0 \quad \text{on } \Gamma_B^- \quad (5d)$$

$$\partial_n\phi = -V_n^\infty \quad \text{on } \gamma_E^+ \quad (5e)$$

$$\phi|_{\Gamma_W^-} - \phi|_{\Gamma_W^+} = w(\mathbf{y}(\mathbf{x})) \quad \text{where } \mathbf{y}(\mathbf{x}) \text{ maps } \mathbf{x} \in \Gamma_W \text{ to } \mathbf{y}(\mathbf{x}) \in \gamma_E \quad (5f)$$

The unknowns are the perturbation potential ϕ and its jump w across the wake at the trailing edge. The symbol Δ denotes the Laplacian, ∂_n denotes the directional derivative along the normal direction, V_n^∞ is the normal component of the free stream velocity, the superscripts $^+$ and $^-$ applied to a surface indicate the upper and lower sides respectively, with the surface normal pointing from the lower to the upper side. The $^+$ applied to the trailing edge γ_E indicates the constraint applies outside the body (in the flow) near γ_E .

Equation (5c) corresponds to the tangential flow condition, and equation (5d) corresponds to an arbitrary value of zero perturbation potential everywhere inside the closed body. Equation (5e) corresponds to a linear (velocity) Kutta condition where the normal at the trailing edge is determined by the geometry and the flow direction such that the flow does not recirculate about the sharp trailing edge. We choose the normal on the trailing edge γ_E to be perpendicular to both γ_E and the adjacent surface that is more aligned with the flow direction. Notably this choice is made for simplicity. The physically-based nonlinear (pressure) Kutta condition could alternatively be imposed, but Hess [7] admits that a simpler linear condition works well to estimate lift. We defer consideration of the pressure-based Kutta condition to future work. In equation (5f), the wake jump is extended from points on the trailing edge γ_E to points on the wake Γ_W .

2.2.2 Isolating the wake potential

Instead of meshing a large finite volumetric domain and integrating over a large finite wake, we analytically integrate the fundamental solutions over a semi-infinite wake and compute a wake potential ϕ_W for a given jump distribution w at the trailing edge. Thus, the perturbation potential is given by the superposition $\phi = \phi_W + \psi$ where ψ is a potential that is continuous across the wake and is added to the wake potential to obtain the perturbation potential. By construction,

the Laplacian of the wake potential vanishes everywhere outside the wake, thus we only need to evaluate its directional derivative to impose the immersed boundary conditions. By isolating the wake potential, we can approximate the equations governing the curl-free ψ on a much smaller finite domain $\Omega \supset \Gamma_B$. We approximate an infinite domain with the Robin boundary condition in equation (6b) that averages the reflective Neumann ($\partial_n \psi = 0$) and an absorptive Dirichlet ($\psi = 0$) conditions. With these changes, the governing equations become

$$-\Delta\psi = 0 \quad \text{on } \Omega \setminus (\Gamma_B \cup \Gamma_W) \quad (6a)$$

$$\partial_n \psi + \psi = 0 \quad \text{on } \partial\Omega \quad (6b)$$

$$\partial_n \psi = -V_n^\infty - \partial_n \phi_W \quad \text{on } \Gamma_B^+ \quad (6c)$$

$$\psi = 0 \quad \text{on } \Gamma_B^- \quad (6d)$$

$$\partial_n \psi = -V_n^\infty - \partial_n \phi_W \quad \text{on } \gamma_E^+ \quad (6e)$$

$$\phi_W(\mathbf{x}) = \int_{\gamma_E} w(\tilde{\mathbf{x}}(s)) \overline{G}_y(\mathbf{x} - \tilde{\mathbf{x}}(s)) ds \quad \forall \mathbf{x} \quad (6f)$$

where \overline{G}_y , the integral of a unit dipole along an infinite ray, is defined by equation (22) in the appendix. In 2d, the integral over the trailing edge simplifies to the identity. In 3d, the integral can be computed by a weighted sum of values at quadrature points. However, the singular integrand renders standard Gauss quadrature insufficient, instead requiring specialized quadrature rules designed to deal with the singularities. Since \overline{G}_y can be integrated analytically over a rectangle of finite width, cf. equation (23), we can take a simpler approach described in the following paragraph.

We discretize trailing edges with linear (line segment) elements in 3d and with a finite set of points in 2d. Then, at the center \mathbf{x}_e of the e th element, we linearly interpolate the edge normal, project it into the plane orthogonal to the free stream direction $\hat{\mathbf{u}}_\infty$ and normalize it to obtain a unit vector $\hat{\mathbf{n}}_e$. In 3d, we also project the segment into the plane orthogonal to the free stream direction to compute a width h_e . Then we compute a local coordinate system for each line segment as

$$x_e(\mathbf{x}) = \hat{\mathbf{u}}_\infty \cdot (\mathbf{x} - \mathbf{x}_e) \quad (7a)$$

$$y_e(\mathbf{x}) = \hat{\mathbf{n}}_e \cdot (\mathbf{x} - \mathbf{x}_e) \quad (7b)$$

$$z_e(\mathbf{x}) = \hat{\mathbf{u}}_\infty \times \hat{\mathbf{n}}_e \cdot (\mathbf{x} - \mathbf{x}_e) \quad (7c)$$

so that we can apply equation (23) to each element as

$$B^e(\mathbf{x}) = \overline{\overline{G}}_y(x_e(\mathbf{x}), y_e(\mathbf{x}), z_e(\mathbf{x}); h_e) \quad (8)$$

which renders an elemental basis for a discontinuous wake with constant-strength, semi-infinite rectangular strips extending from each line segment γ_E^e . We obtain a C^0 basis by averaging the elemental basis to the nodes. For node i , this basis is

$$B_i(\mathbf{x}) = \frac{\sum_e B^e(\mathbf{x}) \int_{\gamma_E^e} \tilde{\varphi}_i ds}{\sum_e \int_{\gamma_E^e} \tilde{\varphi}_i ds} \quad (9)$$

where $\tilde{\varphi}_i$ is the basis function for node i in the discretization of γ_E . Let L denote the set of indices for all nodes discretizing γ_E . In what follows, index sets in the place of indices denote arrays of coefficients. With this notation, $\tilde{\varphi}_L = \{\tilde{\varphi}_i\}_{i \in L}$ is a column vector. In the absence of

an index set, we denote matrices and vectors with enclosing square brackets. The vector of coefficients parameterizing the wake jump is w^L , and the wake potential at a point \mathbf{x} is given by the linear combination of equation (10).

$$\phi_W(\mathbf{x}) = B_L(\mathbf{x})^\top w^L \quad (10)$$

Figure 2 shows contours of the wake potential ϕ_W for a jump of $w = 1$ extending from the sharp trailing edges of a 2d airfoil and a 3d forward-swept configuration.

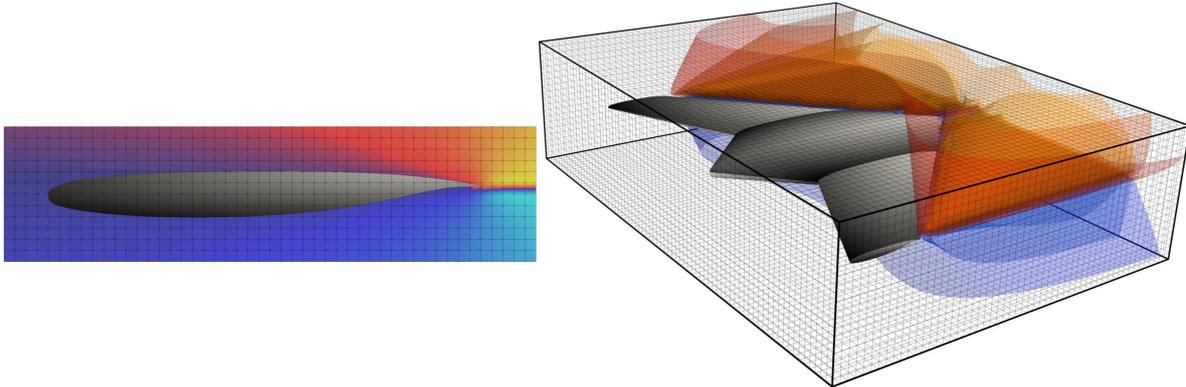


Figure 2: Contours of the wake potential ϕ_W for a unit jump trailing from a 2d airfoil (left) and a 3d forward-swept configuration (right).

The gradient of the wake potential corresponds to the portion of the flow velocity due to the wake. The directional derivative $\partial_n \overline{\overline{G}}_y$, defined by equation (24) in the appendix is used to compute this gradient. Figure 3 depicts streamlines of $\nabla \phi_W$ for $w = 1$ which clearly illustrate vortices trailing from the wing tips.

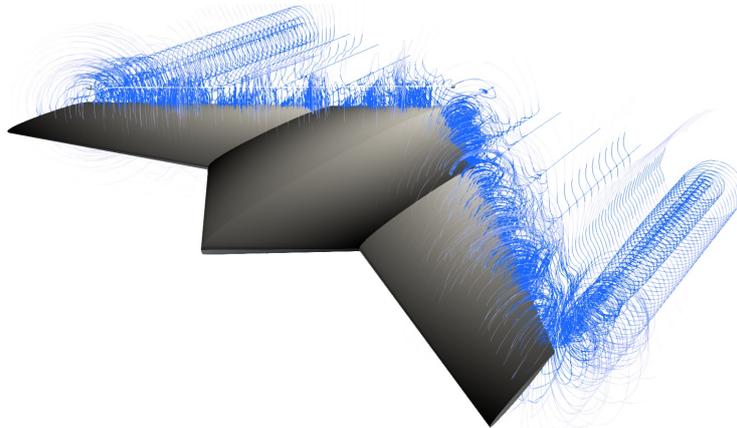


Figure 3: Streamlines of the wake potential gradient $\nabla \phi_W$ for a unit jump across the trailing edge.

2.2.3 Trailing Edge Detection

Sharp trailing edges are detected automatically on the deformed surface configuration. A threshold on the dihedral angle characterizes sharp edges, and a threshold on the angle from the flow direction characterizes sharp edges as trailing. The default values for both angle thresholds are 90° . Figure 4 illustrates the detected sharp and trailing edges, and the infinite wake that implicitly extends from any sharp trailing edges along the free-stream direction.

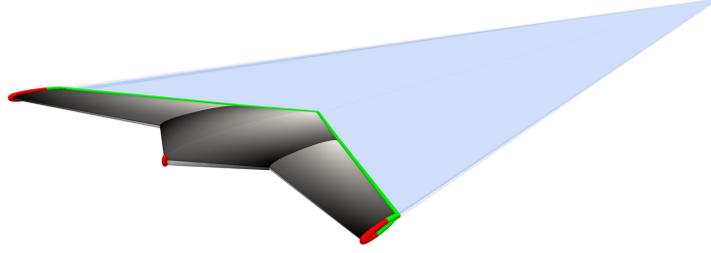


Figure 4: Sharp (red and green) and trailing (green) edges are detected and the wake (light blue) implicitly extends to infinity from the sharp trailing edges.

2.2.4 The weak form

To employ XFEM, we first convert the strong form the the governing equations given in (6a–f) to the weak form. In XFEM, a prescribed jump across a discontinuity becomes a Dirichlet boundary condition. For now, we consider the trailing edge jump distribution w to be known such that $\partial_n \phi_W$ may be directly computed as the directional derivative of equation (6f). Then the Dirichlet boundary is Γ_B^- , the Neumann boundary is Γ_B^+ and the Robin boundary is $\partial\Omega$. Let v denote a virtual function that vanishes on the Dirichlet boundary, then the weak form is

$$\int_{\Omega} \nabla v \cdot \nabla \psi \, d\Omega + \int_{\partial\Omega} v \psi \, d\Gamma = - \int_{\Gamma_B^+} v (V_n^\infty + \partial_n \phi_W) \, d\Gamma \quad (11)$$

The weak form incorporates the Neumann and Robin boundary conditions, while the Dirichlet condition (6d) must still be imposed. For now, we ignore the Kutta condition (6e) since we have assumed a distribution for w . Later, we will resolve w such that the Kutta condition is satisfied.

2.2.5 Extended finite element discretization

We extend the finite element basis to support jump discontinuities across the boundary of the immersed body. The extended finite element approximation is

$$\psi = \sum_{i \in I} \varphi_i \psi^i + \sum_{j \in J} \text{sgn}_B \varphi_{i_j} \psi^j \quad (12)$$

where φ_i is the finite element basis function of grid-node i , I is the set of all grid node numbers, J is the set of extended degrees of freedom numbers supporting the immersed boundary Γ_B , i_j is the grid-node number of extended degree of freedom j , sgn_B is -1 below and $+1$ above Γ_B , and ψ^i is the curl-free perturbation-potential coefficient of the i th degree of freedom.

Let $[\tilde{\varphi}] = [\varphi_I^\top, \text{sgn}_B \varphi_{i_j}^\top]^\top$ be the vector of shape functions for all degrees of freedom. Then the unconstrained discrete equations can be written as $[\tilde{A}][\tilde{\psi}] = [\tilde{b}]$ where the coefficients $[\tilde{\psi}]$ are unknown and

$$[\tilde{A}] = \int_{\Omega} \nabla[\tilde{\varphi}] \cdot \nabla[\tilde{\varphi}]^\top \, d\Omega + \int_{\partial\Omega} [\tilde{\varphi}][\tilde{\varphi}]^\top \, d\Gamma \quad (13a)$$

$$[\tilde{b}] = \int_{\Gamma_B^+} [\tilde{\varphi}] (-V_n^\infty - \partial_n \phi_W) \, d\Gamma \quad (13b)$$

The Dirichlet constraint on Γ_B^- is

$$\psi|_{\Gamma_B^-} = \varphi_I \psi^I - \varphi_J \psi^J = 0 \quad (14)$$

for which ψ^J can be solved by directly assigning the the grid-node coefficients to the corresponding jump coefficients.

$$\psi^J = \psi^{iJ} = C^J_I \psi^I \quad (15)$$

The constraint equation (15) is then condensed into the unconstrained linear system (13a,b) to obtain the constrained system $[A][\psi] = [b]$ in which $[\psi] = \psi^I$ are the unknown coefficients on the grid nodes I :

$$[A] = \tilde{A}_{II} + (C^J_I)^\top \tilde{A}_{JI} + \tilde{A}_{IJ} C^J_I + (C^J_I)^\top \tilde{A}_{JJ} C^J_I \quad (16a)$$

$$[b] = \tilde{b}_I + (C^J_I)^\top \tilde{b}_J \quad (16b)$$

The Kutta condition of equation (6e) can be imposed on the nodes discretizing the trailing edge as

$$0 = \underbrace{(\partial_n \varphi_I(\mathbf{x}^L) + (C^J_I)^\top (\partial_n \varphi_J(\mathbf{x}^L)))^\top}_{E^L_I} \psi^I + \underbrace{(\partial_n B_L(\mathbf{x}^L))^\top}_{E^L_L} w^L + \underbrace{V_n^\infty(\mathbf{x}^L)}_{e^L} \quad (16c)$$

where \mathbf{x}^L are shifted from the edge node positions by a small distance along the projection of \mathbf{u}_∞ orthogonal to the edge normal to avoid the singularity at the trailing edge. This affine equation can be directly solved as

$$w^L = \underbrace{(-E^L_L)^{-1} E^L_I}_{\tilde{E}^L_I} \psi^I + \underbrace{(-E^L_L)^{-1} e^L}_{\tilde{e}^L} \quad (16d)$$

It is possible to solve the matrix \tilde{E}^L_I and condense the Kutta condition into a single linear system in which the grid-node coefficients ψ^I are the only unknowns. Admittedly, the computational and storage costs of this dense matrix scale poorly. An alternative approach that scales better and also easily extends to the nonlinear Kutta condition is to resolve w^L with a Newton iteration as described in algorithm 1.

Algorithm 1: Newton's method for the Kutta condition

$w^L \leftarrow 0$

while *not converged* **do**

$[\tilde{b}] \leftarrow$ equation (13b)

$[b] \leftarrow$ equation (16b)

$\psi^I \leftarrow [A]^{-1} [b]$

$w^L \leftarrow (-E^L_L)^{-1} (E^L_I \psi^I + e^L)$

2.2.6 Pressure forces

The pressure coefficient is computed from the gradient of the perturbation potential as

$$C_p = -\frac{2\nabla\phi \cdot V_\infty}{|V_\infty|^2} - \left(\frac{|\nabla\phi|}{|V_\infty|} \right)^2 \quad (17)$$

from which the static pressure is computed as

$$p = p_\infty + \frac{1}{2} \rho_\infty V_\infty^2 C_p \quad (18)$$

The pressure forces are integrated over the surface in the deformed configuration $\Gamma_B \ni \mathbf{x}$ and mapped to the nodes, I , in the structural analysis grid that support the reference configuration $\Gamma_{\hat{B}} \ni \hat{\mathbf{x}}$ as

$$\mathbf{f}_I = \int_{\Gamma_B} \varphi_I(\hat{\mathbf{x}}(\mathbf{x})) p \mathbf{n} d\Gamma \quad (19)$$

where, the displacements deform the reference configuration as

$$\mathbf{x} = \hat{\mathbf{x}} + \mathbf{u}^I \varphi_I(\hat{\mathbf{x}}) \quad (20)$$

Although unlike in [3] the structural geometry in this work contains the interface $\Gamma_{\hat{B}}$, the same arguments hold to show this transfer scheme is consistent and conservative.

2.3 Elastic deformation

We solve the linear (small-strain) elasticity equations with the standard finite element method. The geometry is discretized by an element-uniform density field on a fixed non-conforming grid. The elastic stiffness tensor in each element is interpolated as

$$C_{ijkl}^e = C_{ijkl}^{\text{void}} + \chi(\mathbf{x}^e) (C_{ijkl}^{\text{solid}} - C_{ijkl}^{\text{void}})$$

where \mathbf{x}^e is the centroid of element e , $\chi(\mathbf{x})$ is the element volume fraction function given in equation (3), C_{ijkl}^{void} is a small isotropic elastic stiffness assigned to void regions to avoid an ill-posed analysis and C_{ijkl}^{solid} is the elastic stiffness tensor of the solid material.

The resulting system of equations is

$$[K][u] = [f] \quad (21)$$

where $[K]$ is the stiffness matrix, $[u]$ is the vector of nodal displacements, and $[f]$ is the vector of nodal forces corresponding to equation (19). To obtain a well-posed system, we simply prescribe zero displacement to the root chord and defer consideration of inertia relief to future work.

2.4 Coupled solution

The symmetric positive definite system for each independent discipline can be solved by a highly efficient multigrid-preconditioned conjugate gradient method. However, the coupling is nonlinear and the tangent Jacobian matrix for the coupled residual is not symmetric. We solve the Newton iteration for coupled residual by the block-Jacobi method, resulting in the following iteration to update each Newton step

$$\begin{bmatrix} \delta\psi \\ \delta w \\ \delta u \end{bmatrix} \leftarrow \begin{bmatrix} A^{-1}(b + D_{(\delta w, \delta u)}(b(w, u) - A(u) \cdot \psi)) \\ \bar{A}^{-1}(\bar{b} + D_{(\delta\psi, \delta u)}(\bar{b}(\psi, u) - \bar{A}(u) \cdot w)) \\ K^{-1}(f + D_{\delta\psi}(f(\psi))) \end{bmatrix} - \begin{bmatrix} \psi \\ w \\ u \end{bmatrix}$$

where $D_{(\delta a, \delta b)} = \delta a \cdot \partial_a + \delta b \cdot \partial_b$ denotes a multi-variable directional derivative, and we have dropped the convention of enclosing unindexed matrices in brackets for clarity. Notably, the function $\bar{A}(u) = -E^L_L$ on the configuration perturbed by displacement u , and $\bar{b}(\psi, u) = (E^L_I \psi^I + e^L)$ on the configuration perturbed by u .

2.4.1 Adjoint Design Sensitivities

The design sensitivity of quantities Q that depend on the coupled state S can be efficiently computed by the adjoint method. In general, the design variables X define the geometry, flow conditions and material properties (collectively referred to as the input, I). The chain rule maps input sensitivities, ∂_I , to design sensitivities, ∂_X , as a directional derivative with respect to the design sensitivity of the input, $\bar{\partial}_X = \partial_X I \cdot \partial_I$.

We overload the symbol Q to also represent the map $Q(X) : X \mapsto Q$ that abstracts away the dependence on the coupled state. To avoid a collision of symbols, we denote by \bar{Q} the map $\bar{Q}(S) : S \mapsto Q$ that describes the dependence on the coupled state. With this notation, we have the following identity

$$\begin{aligned} Q(X) &= Q(S(I(X))) \\ \partial_X Q &= \partial_X I \cdot \partial_I S \cdot \partial_S Q = \bar{\partial}_X S \cdot \partial_S Q \end{aligned}$$

In general, $\partial_S Q$ is easily computed, but directly computing $\bar{\partial}_X S$ requires a coupled solution for each design variable X . When Q is smaller than X , the adjoint approach is preferred because a coupled adjoint solution is required for each Q .

In the adjoint approach, we differentiate the state equation to eliminate $\bar{\partial}_X S$. Let $R + S \cdot \partial_S R = 0$ denote the linearized vector residual equation of the coupled state, then

$$\begin{aligned} \bar{\partial}_X (R + S \cdot \partial_S R) &= 0 \\ \bar{\partial}_X R + \bar{\partial}_X (\partial_S R)^\top \cdot S &= \bar{\partial}_X S \cdot (-\partial_S R) \\ [\bar{\partial}_X R + \bar{\partial}_X (\partial_S R)^\top \cdot S] \cdot (-\partial_S R)^{-1} &= \bar{\partial}_X S \end{aligned}$$

Thus, eliminating $\bar{\partial}_X S$ yields

$$\partial_X Q = \underbrace{[\bar{\partial}_X R + \bar{\partial}_X (\partial_S R)^\top \cdot S]}_{X_S} \cdot \underbrace{(-\partial_S R)^{-1}}_{S_Q} \cdot \partial_S Q$$

where S_Q is the adjoint solution and S is the primal solution. Although the primal state S is governed by a nonlinear system, the system of equations governing the adjoint state S_Q is linear. The matrix X_S is the design sensitivity of the linearized residual holding the state S constant. In the case X is small, X_S can be computed quite efficiently via finite differences, and accurately by employing a complex step. However, the sparsity of the input sensitivity of the linearized residual holding S constant, $I_S = \partial_I R + \partial_I (\partial_S R)^\top \cdot S$, often renders the design sensitivity computed as $\partial_X Q = \partial_X I \cdot I_S \cdot S_Q$ for an analytical I_S more efficient than finite differencing the residual. Moreover, a very memory-efficient computation is achieved with a matrix-free implementation which we denote as $(\partial_X I, S_Q)_{I_S} \equiv \partial_X I \cdot I_S \cdot S_Q$.

To be generic, the adjoint solver accepts the state sensitivity of arbitrary quantities $\partial_S Q$ and the design sensitivity of inputs for arbitrary designs $\partial_X I$, internally stores the state variables S from the primal solution, and computes design sensitivities $\partial_X Q$ according to algorithm 2.

3 CONCLUSIONS

The proposed approach to static aeroelastic analysis avoids mesh perturbation and remeshing by solving potential-based lifting pressures coupled to structural displacements by mapping the geometry to a fixed grid for analysis. By avoiding the conventional boundary element method, the

Algorithm 2: Adjoint computation of design sensitivities

inputs: $\partial_X I, \partial_S Q$, output: $\partial_X Q$

$$S_Q \leftarrow (-\partial_S R)^{-1} \cdot \partial_S Q$$

$$\partial_X Q \leftarrow (\partial_X I, S_Q)_{I_S}$$

resulting coupled equations are sparse and can be resolved accurately by an efficient multi-grid preconditioned solver. Since all of the mappings to the analysis grid are differentiable, the method supports the efficient adjoint computation of design-sensitivities for quantities dependent on the coupled aeroelastic state. By circumventing the need to rebuild or deform the analysis mesh, our proposed solver eliminates a costly step required in conventional approaches. Moreover, by mapping geometric features to an implicit representation, topological operations defining the structural geometry are substantially more robust and efficient compared to B-rep Boolean operations. Although presented in the context of static aeroelasticity with a potential-based fluid model and small-strain linear elasticity, the proposed approach could also be extended to higher fidelity flow solvers based on the Euler or Navier-Stokes equations, and to nonlinear (finite-strain) elasticity.

ACKNOWLEDGMENTS

This research was performed while the author held an NRC Research Associateship award at the Air Force Research Laboratory.

4 REFERENCES

- [1] Venkayya, V. and Tischler, V. (2004). Force and displacement transformations for aero-structure interaction analysis. In *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*. p. 4328.
- [2] Alonso, J., Martins, J. R., Reuther, J., et al. (2003). High-fidelity aero-structural design using a parametric cad-based model. In *16th AIAA Computational Fluid Dynamics Conference*. p. 3429.
- [3] Martins, J. R., Alonso, J. J., and Reuther, J. J. (2005). A coupled-adjoint sensitivity analysis method for high-fidelity aero-structural design. *Optimization and Engineering*, 6, 33–62.
- [4] Kenway, G., Kennedy, G., and Martins, J. R. (2010). A cad-free approach to high-fidelity aerostructural optimization. In *13th AIAA/ISSMO multidisciplinary analysis optimization conference*. p. 9231.
- [5] Deaton, J. D. and Grandhi, R. V. (2014). A survey of structural and multidisciplinary continuum topology optimization: post 2000. *Structural and Multidisciplinary Optimization*, 49, 1–38.
- [6] Wein, F., Dunning, P. D., and Norato, J. A. (2020). A review on feature-mapping methods for structural optimization. *Structural and multidisciplinary optimization*, 62, 1597–1638.
- [7] Hess, J. L. (1988). Development and application of panel methods. In *Advanced Boundary Element Methods: Proceedings of the IUTAM Symposium, San Antonio, Texas, April 13–16, 1987*. Springer, pp. 165–177.
- [8] Barnes, J. and Hut, P. (1986). A hierarchical O(N log N) force-calculation algorithm. *nature*, 324(6096), 446–449.

- [9] Rokhlin, V. (1985). Rapid solution of integral equations of classical potential theory. *Journal of computational physics*, 60(2), 187–207.
- [10] Belytschko, T. and Black, T. (1999). Elastic crack growth in finite elements with minimal remeshing. *International journal for numerical methods in engineering*, 45(5), 601–620.
- [11] Burstedde, C., Wilcox, L. C., and Ghattas, O. (2011). p4est: Scalable algorithms for parallel adaptive mesh refinement on forests of octrees. *SIAM Journal on Scientific Computing*, 33(3), 1103–1133. doi:10.1137/100791634.
- [12] Norato, J., Bell, B., and Tortorelli, D. (2015). A geometry projection method for continuum-based topology optimization with discrete elements. *Computer Methods in Applied Mechanics and Engineering*, 293, 306–327.
- [13] Smith, H. A. and Deaton, J. D. (2024). A skin-stabilizing constraint for feature-based topology optimization of a wingbox. In *AIAA SCITECH 2024 Forum*. p. 0565.
- [14] Ericson, C. (2004). *Real-time collision detection*. Crc Press.
- [15] Glauert, H. (1928). The effect of compressibility on the lift of an aerofoil. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 118(779), 113–119.
- [16] Göthert, B. H. (1940). Plane and three-dimensional flow at high subsonic speeds: Extension of the Prandtl rule. Tech. rep.

APPENDIX

Here we derive the integrals of the Green’s functions for a semi-infinite dipole strip in 2d and 3d. Let $\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}$ be orthogonal unit vectors. The ray along $\hat{\mathbf{x}}$ emitting from the origin is

$$U = \{t \hat{\mathbf{x}} : t \in [0, \infty)\}.$$

The Green’s functions that satisfy $\Delta G = \delta$ with $G = 0$ as $r \rightarrow \infty$ where r is the distance from the origin are

$$G(r) = \begin{cases} \frac{\log(r)}{2\pi}, & \text{in 2d} \\ -\frac{1}{4\pi r}, & \text{in 3d.} \end{cases}$$

Integrating the directional derivative of G at a point $x\hat{\mathbf{x}} + y\hat{\mathbf{y}} + z\hat{\mathbf{z}}$ with respect to $\hat{\mathbf{y}}$ over U gives

$$\bar{G}_y(x, y, z) = \int_0^\infty \partial_y(G \circ r)(x - t, y, z) dt = \begin{cases} \frac{\frac{1}{2\pi} \arctan(y, -x)}{\left[\frac{1}{4} + \frac{1}{2\pi} \frac{x}{|x|} \arctan|x/y| \right]}, & \text{in 2d} \\ \frac{1}{4\pi} \frac{y}{y^2+z^2} \left[1 + \frac{x}{\sqrt{x^2+y^2+z^2}} \right], & \text{in 3d.} \end{cases} \quad (22)$$

In 3d, the integral over a semi-infinite rectangle of width h is

$$\begin{aligned}\overline{\overline{G}}_y(x, y, z; h) &= \int_{-h/2}^{h/2} \overline{G}_y(x, y, z - t) dt \\ &= \frac{1}{4\pi} \left(\arctan a^+ + \arctan a^- + \frac{xy}{|xy|} (\arctan b^+ + \arctan b^-) \right),\end{aligned}\quad (23)$$

where

$$a^\pm = \frac{c^\pm}{y}, \quad b^\pm = \frac{|x/y|c^\pm}{\sqrt{d^\pm}}, \quad d^\pm = x^2 + y^2 + (c^\pm)^2, \quad c^\pm = h/2 \pm z.$$

The directional derivative of $\overline{\overline{G}}_y$ along a vector $n_x \hat{\mathbf{x}} + n_y \hat{\mathbf{y}} + n_z \hat{\mathbf{z}}$ is

$$\partial_n \overline{\overline{G}}_y = \frac{1}{4\pi} \sum_{\pm} \left\{ A^\pm \left(-\frac{n_y}{y} \pm \frac{n_z}{c^\pm} \right) + B^\pm \left[\left(\frac{1}{x} - \frac{x}{d^\pm} \right) n_x + \left(-\frac{1}{y} - \frac{y}{d^\pm} \right) n_y \pm \left(\frac{1}{c^\pm} - \frac{c^\pm}{d^\pm} \right) n_z \right] \right\},$$

where

$$A^\pm = \frac{a^\pm}{(a^\pm)^2 + 1}, \quad B^\pm = \frac{b^\pm}{(b^\pm)^2 + 1}.$$
(24)