



Robust Mesh Deformation for Aerodynamic Shape Optimization with Large Geometric Changes

Timo Lahteenmaa-Swerdlyk* and David W. Zingg†

Institute for Aerospace Studies, University of Toronto, Toronto, Ontario, M3H 5T6, Canada

In this paper we present a robust mesh deformation scheme for use in an aerodynamic shape optimization framework with large geometric freedom. For large shape changes, a mesh deformation approach is at risk of potentially producing meshes of reduced quality such that flow solver convergence and accuracy are impacted. This work builds on an existing mesh deformation methodology based on the equations of linear elasticity that integrates a geometry parametrization based on B-spline surfaces with mesh deformation through a B-spline volume control mesh. This deformation scheme is improved by developing spatially-varying formulations for its two parameters: the Young's modulus and the Poisson's ratio. The two parameters work in tandem, controlling local stiffness and compressibility, to maintain element quality near the surface and propagate deformations outwards into the farfield. The presented mesh deformation scheme can be generalized to deformations on most meshes when the linear elasticity approach is used, including structured or unstructured meshes. To demonstrate the performance of the robust deformation scheme, three cases are presented which show its effectiveness on extreme deformations and optimization cases with large shape changes, notably on blended-wing-body aircraft. Presented results first show that the updated scheme is able to maintain mesh quality for a large suite of geometric changes, including extreme wing sweep adjustments and large blending region, centerbody, and winglet changes on a blended-wing-body. Second, this performance gain increases the robustness of presented optimization cases, showing the potential of the updated deformation scheme on optimizations with complex geometries demanding large shape changes.

I. Introduction

With the growing concern of climate change prompting the aviation industry to shift away from fossil fuels, there is an increasing need to develop more fuel-efficient aircraft designs. The industry has remained reliant on conventional tube-and-wing designs since the introduction of the de Havilland Comet in 1952 [1]. While tube-and-wing designs have undergone a steady increase in fuel efficiency of around one to two percent per year [2], a major step forward beyond this rate is required to have a noticeable impact on the aviation industry's climate change impact. One way of addressing these concerns is through novel aircraft configurations. Several different configurations have been proposed, including strut-braced wings [3–5], joined-wing systems [6–8], and blended-wing-bodies [9–11]. These aircraft designs must be appropriately assessed and de-risked. A method of approaching these tasks is through the use of multidisciplinary optimization based on computational fluid dynamics and specifically aerodynamic shape optimization. A high-fidelity flow solver combined with geometric control and numerical optimization creates a powerful design tool to develop and evaluate aircraft configurations.

During the optimization procedure, a new mesh has to be created at each iteration which is fitted to the deformed geometry. To accomplish this, the mesh may either be regenerated or the existing mesh may be allowed to deform. Particularly for structured meshes, allowing the mesh to deform is more efficient than generating a new mesh, which may be time consuming and difficult to automate. In addition, re-meshing also requires a projection of the solution to the new mesh, as the number of nodes and their connectivity may not be the same. This may introduce additional errors into the flow solution and lead to improper convergence during the optimization process [12]. However, for cases with large geometric changes, a mesh deformation approach may produce highly-skewed elements or crossed element nodes, which can prevent an optimization from converging to an optimum geometry [13]. Thus, while a mesh deformation

*PhD Student, University of Toronto, timo.swerdlyk@mail.utoronto.ca

†University of Toronto Distinguished Professor of Computational Aerodynamics and Sustainable Aviation, Associate Fellow AIAA, david.zingg@utoronto.ca

approach is preferred, if large geometric changes are permitted, the mesh deformation method must be sufficiently robust to ensure a mesh of suitable quality is maintained.

A compilation of commonly used mesh deformation approaches may be found in [14]. The strategies used for mesh deformation can be divided into two classes: interpolation or physical analogy. Interpolation schemes use the displacements of the surface nodes to interpolate the displacements to the other points within the volume mesh. Liu et al. [15] proposed a fast algebraic mesh deformation scheme based on mapping the mesh to a Delaunay graph. This method was determined to be effective for large shape changes; however, it was noted that the scheme produced discontinuous objective functions across multiple iterations during an optimization. This may limit the approach to gradient-free optimization algorithms. Jakobsson and Amignon [16] and de Boer et al. [17] each proposed deforming the mesh based on radial basis functions (RBFs). In this scheme, a large system of equations is solved to find the interpolation coefficients, which may then be used to find the new node coordinates through sparse matrix-vector multiplication. The scheme was robust for large deformations and was able to preserve cell orthogonality near the boundary; however, the required inversion of a large, dense system made it computationally expensive and not feasible for large 3D meshes. To address this, Rendall and Allen [18] modified the RBF scheme by mapping the deformation on a coarse control mesh which was a subset of the computational surface mesh. Wang et al. [19] made further improvements to the RBF scheme by using a double-edged greedy algorithm. It was noted that the use of a coarse control mesh with an RBF scheme introduced errors in surface node locations as the computational mesh was interpolated, which propagated into the volume mesh after a deformation. These errors may accumulate throughout an optimization and prevent convergence to a valid solution. This was addressed by Poirier and Nadarajah [20] by performing a polynomial interpolation to correct the surface node locations and propagate the corrections to the volume mesh. Luke et al. [21] proposed an inverse-distance-weighting method which interpolates the volume mesh nodes from a weighted average of the surface node displacements. This method was able to maintain cell orthogonality near the boundary and has a relatively low computation time, making it scalable to large 3D meshes [13].

Physical analogy schemes model the mesh deformation according to a physical process. A popular branch of physical analogies view the mesh as a network of linear springs, first introduced by Batina [22]. Farhat et al. [23] augmented the approach on 2D meshes by including torsional springs to maintain cell orthogonality. Degand and Farhat [24], Matsushima et al. [25] and Burg [26] each extended the scheme to 3D meshes. A spring analogy was determined to be simple and easy to implement; however for large deformations the scheme resulted in crossed vertex nodes and overlapping elements, limiting its use to small deformations [23]. Yang et al. [27] improved the scheme by adopting a cell-centred spring analogy approach, which aided in preventing crossed mesh nodes and preserved cell quality, allowing for accurate flow computations on viscous meshes. Another approach is to deform the mesh based on the Laplace equations. An advantage to this approach is that from the maximum principle, the values of the interior displacements are bounded by the values on the boundary surfaces, which ensures that the interior nodes will not cross the boundaries [14]. To increase robustness of the scheme, the Laplace equations were augmented with a diffusion coefficient to reduce element skewness. Jasak and Tuković [28] proposed setting the diffusion coefficient proportional to either the element off-wall distance or element skewness, while Crumpton and Giles [29] proposed setting the coefficient inversely proportional to the cell volume.

The mesh can also be modelled as a continuum of elastic solid governed by the equations of linear elasticity, first proposed by Lynch and O’Neil [30]. In this approach, the deformation scheme is controlled using two physical constants: the Young’s modulus (E) and the Poisson’s ratio (ν). While this method has proven to be robust for large shape changes while maintaining an accurate boundary profile, its biggest challenge is choosing E and ν such that cell orthogonality is preserved while allowing for large deformation freedom. Tezduyar et al. [31] proposed setting E inversely proportional to the cell volume, so that smaller elements closer to the boundary are more rigid and resistant to deformation. Similar schemes have been proposed by setting E as a function of the cell off-wall distance [32–34]. Stein et al. [35], Bar-Yoseph et al. [36] and Truong et al. [12] each augmented the E formulation with additional mechanisms that further stiffen highly-skewed elements, so the mesh quality does not deteriorate. Similarly, Nielsen and Anderson [37] and Karman and Anderson [38] spatially varied ν based on the cell aspect ratio, which when applied to 2D meshes led to the same effect as the previous approach. Brown et al. [39] augmented the linear elasticity scheme by including an additional mechanism which performs a backtracking line search on the node displacements should an invalid mesh be produced. This reverts the mesh to a valid state, allowing the spatially-varying terms to appropriately stiffen the at-risk cells and allow the deformation to continue. A linear elastic mesh deformation scheme generally has a higher computation time than most algebraic or inverse-distance-weighting algorithms since it requires the solution of a linear system of equations [32], although with appropriate parallelization the computation time only takes up a small fraction of the total optimization time. Furthermore, when undergoing large deformations, it is common to encounter highly-skewed

elements or negative volumes, particularly if simple formulations for E and ν are used [34, 40]. However, complicated spatially-varying formulations for the two variables can result in an ill-conditioned system of equations [21], which can increase the computation time. Thus it becomes a challenging balancing act to select optimal formulations for E and ν .

The purpose of this work is to extend and improve a mesh deformation scheme based on the equations of linear elasticity to handle a broader range of geometries with large shape changes. Special interest is taken in the development of an effective mesh deformation methodology suitable for the aerodynamic shape optimization of complex and unconventional aircraft geometries with high geometric freedom. A highlighted geometry throughout this paper is the blended-wing-body configuration (BWB): Its tightly-integrated nature, regions of high curvature, and additional features such as winglets and control surfaces make the BWB a good candidate to test the robustness of a mesh deformation scheme.

The rest of this paper is organized as follows. Section II gives an outline of the optimization framework within which the mesh deformation scheme is used. Section III investigates encountered challenges and issues with mesh deformation in optimization cases with high geometric freedom, and Section IV proposes an improved deformation scheme to address these challenges. Finally, Section V details select case studies to investigate the robustness of the new mesh deformation scheme, assess its performance on extreme deformations, and show its potential applicability to optimization problems demanding large shape changes.

II. Optimization Framework

The robust deformation scheme will be developed and investigated using a high-fidelity aerodynamic shape optimization framework called Jetstream [41], which consists of four major components: 1) a B-spline geometry parametrization with integrated mesh deformation based on the equations of linear elasticity, 2) a geometry control module using free-form deformation (FFD) volumes and axial curves, 3) a multiblock Newton–Krylov–Schur flow solver for the Reynolds-averaged Navier Stokes (RANS) equations, and 4) a gradient-based optimizer SNOPT [42] with gradients calculated using the discrete adjoint method. Jetstream also has the capability to be augmented with low- and medium-fidelity structural, weight, balance, and propulsion models to allow it to serve as a mixed-fidelity multidisciplinary aircraft optimization code [11].

A. B-Spline Geometry Parametrization with Integrated Linear Elastic Mesh Deformation

The integrated B-spline geometry parametrization and mesh deformation algorithms were developed by Truong et al. [12] and Hicken and Zingg [43]. Each computational mesh block is fitted to a B-spline volume, of fourth order, whose control points form a coarse control mesh which itself is deformed. The fitted B-spline control mesh is usually at least one to two orders of magnitude smaller than the computational mesh. The geometry parametrization in Jetstream is tightly coupled to its mesh deformation algorithm: Jetstream uses a B-spline geometry parametrization by extracting the fitted control points on the surface to form surface B-spline patches, which in turn define the geometric surface analytically. Following this geometric parametrization, the surface can be deformed by adjusting the surface B-spline control points. The resulting deformations of the surface control points are propagated throughout the control mesh using the mesh deformation scheme. By applying the deformation to the control mesh, its B-spline parametrization allows the computational mesh to maintain a high element quality, particularly near the surface, while greatly reducing the computation time [43].

When applied to computational meshes for use on turbulent flows, the above fitting procedure may fail due to the presence of high-aspect ratio cells near the surface [44]. This can be addressed by applying the fitting procedure to a modified computational grid with a higher off-wall spacing. After fitting, the RANS grid is recovered with a finer off-wall spacing through the B-spline mapping.

To deform the control mesh in response to geometric changes, it is modelled as a linear elastic object. To allow for consistency of the mesh across all possible geometric configurations, each deformation is started from the same inputted initial mesh whose elements are assumed to be adequately orthogonal. The new control mesh can be computed by solving the equilibrium equation:

$$\frac{\partial \sigma_{jk}}{\partial x_k} + f_j = 0, \quad (1)$$

where σ_{jk} is the stress tensor and f_j is the vector of external forces defined implicitly from the known surface node displacements. Equation 1 is formed cell-wise using the cell's eight vertices. Each cell is assumed to be an isotropic, homogeneous, linear elastic solid under small strains. The stress tensor σ_{jk} is related linearly to the Cauchy strain

tensor, ε_{jk} through the Young's modulus (E) and the Poisson's ratio (ν):

$$\sigma_{jk} = \frac{E}{1+\nu} \left(\varepsilon_{jk} + \frac{\nu \varepsilon_{mm} \delta_{jk}}{1-2\nu} \right), \quad (2)$$

$$\varepsilon_{jk} = \frac{1}{2} \left(\frac{\partial u_j}{\partial x_k} + \frac{\partial u_k}{\partial x_j} \right), \quad (3)$$

where u_j is the j^{th} component of the displacement.

For large shape changes where large strains are expected, the mesh deformation scheme is broken into equal increments,

$$\mathbf{b}_s^{(i)} = \mathbf{b}_s^{(0)} + \frac{i}{m} \left(\mathbf{b}_s^{(m)} - \mathbf{b}_s^{(0)} \right), \quad (4)$$

where $\mathbf{b}_s^{(i)}$ is a vector containing the perturbed surface nodes for the i^{th} increment with m total increments, $\mathbf{b}_s^{(0)}$ are the initial coordinates, and $\mathbf{b}_s^{(m)}$ are the final coordinates corresponding to the fully deformed mesh. To apply the linear elastic formulation to each discrete mesh element, Equation 1 is discretized as trilinear finite elements, which when combined with the incremental approach in Equation 4 results in the following system of algebraic equations for each increment i :

$$\mathbf{K}^{(i)} \left(\mathbf{b}^{(i)} - \mathbf{b}^{(i-1)} \right) = \mathbf{f}^{(i)}, \quad (5)$$

where \mathbf{K} is the global stiffness matrix for all internal nodes in the control mesh, \mathbf{b} are the internal control mesh coordinates, and \mathbf{f} is the vector of external forces acting on each of the nodes, equal to zero except at the boundary where the displacement is known. The stiffness matrix for each element is summed over overlapping entries surrounding each node to form \mathbf{K} .

For each increment, element stiffness is controlled by spatially varying the Young's modulus: First, E is set to be inversely proportional to the element volume at each increment ($V_e^{(i)}$) to stiffen small elements and propagate the mesh deformation away from the surface where elements are larger. Second, for highly skewed or distorted elements, an additional stiffening mechanism is added to further stiffen poor-quality elements and propagate the mesh deformation to higher-quality ones. This is added using Φ_e , an element distortion measure defined by

$$\Phi_e = \left(\prod_{j=1}^8 \frac{1}{(\mathbf{u}_j \times \mathbf{v}_j) \cdot \mathbf{w}_j} \right)^2, \quad (6)$$

where j is looped over all 8 element vertices, and \mathbf{u}_j , \mathbf{v}_j , and \mathbf{w}_j are outgoing unit vectors parallel to the element edges meeting at the j^{th} node. For a cube (orthogonal unit vectors), the distortion measure is equal to one. As the element becomes more skewed, the triple product in the denominator goes to zero, increasing the Young's modulus of the element and its stiffness. Combining the two mechanisms, the Young's modulus for each element is

$$E_e^{(i)} = \frac{1}{V_e^{(i)}} \frac{\Phi_e^{(i)}}{\Phi_e^{(0)}}, \quad (7)$$

where $\Phi_e^{(0)}$ is the initial distortion and $\Phi_e^{(i)}$ is the distortion evaluated at the i^{th} increment. The linear elastic system given by Equation 5 is solved using the preconditioned conjugate gradient (PCG) method to a tight relative tolerance on the order of 10^{-12} . The additive-Schwarz preconditioner with a default overlap of one is used, with Jacobi preconditioning/diagonal scaling applied to each block. To decrease the computation time, the PCG algorithm is parallelized using the Portable, Extensible Toolkit for Scientific Computation (PETSc) [45]. From the new deformed coordinates of the control mesh, which are B-spline volume control points, the deformed computational mesh is recovered algebraically.

Table 1 Errors in initial B-spline surface mesh approximation for a sample rectangular wing

RMS _{error}	$ \Delta x _\infty$	$ \Delta y _\infty$	$ \Delta z _\infty$
4.79×10^{-5}	6.76×10^{-4}	4.4×10^{-4}	1.82×10^{-4}

B. Mesh Fitting and Deformation Example

A sample deformation is completed starting from an extruded NACA0012 rectangular wing with a unit chord and a semi-span of three. Two meshes are inputted: first a computational “fitting” mesh with an increased off-wall spacing, which is used to fit the B-spline control points of the control mesh, and second, the true initial computational mesh with the desired off-wall spacing. The computational mesh consists of 3.9M nodes over 128 blocks, which is fitted with a control mesh consisting of 118k nodes. The fitting procedure used to generate the B-spline control mesh produces an approximation to the initial computational mesh. To assess the quality of the fit, Table 1 gives the root-mean-square (RMS) and L^∞ norm errors for each coordinate for the surface of the rectangular wing. The RMS error is on the order of 10^{-5} and the maximum absolute error in each coordinate is on the order of 10^{-4} . If tighter tolerances are required, more control points can be used.

The wing is deformed to a sweep angle of 20 degrees, a dihedral of 10 degrees, and a wing tip chord reduction by 50%, linearly interpolated over the wing from the root. To limit small strain violations, the deformation is completed using 20 increments, and $\nu = -0.2$. Figure 1 shows a cross-section of the mesh as the deformation takes place. In Figure 1a, the B-spline control mesh is obtained using the fitting mesh with an increased off-wall spacing of 6.0×10^{-4} chord lengths. The fitting process also generates the B-spline approximation to the computational mesh, shown in Figure 1b. The B-spline computational mesh has a reduced off-wall spacing averaging 2.0×10^{-6} chord lengths, suitable for viscous flow evaluations. The control mesh is deformed to the final geometry by perturbing its surface nodes, which, through the linear elastic mesh deformation scheme, leads to the final control mesh in Figure 1c. Using the B-spline fitting, the final computational mesh is obtained algebraically through the B-spline mapping, shown in Figure 1d. By using the B-spline fitting procedure, the deformed computational mesh is able to maintain high quality around the geometry surface.

C. Free-Form Deformation Geometry Control Module

Geometry control in Jetstream uses the two-level FFD approach developed by Gagnon and Zingg [46]. Conceptually, this can be visualized as embedding a piece of rubber inside a transparent material with the same properties. As the larger volume deforms, so will the embedded rubber in a consistent manner. In this implementation, the B-spline control points (\mathbf{X}_p) corresponding to the surface parametrization of the geometry are embedded within FFD volumes. The FFD volumes are also specified using B-splines with their own set of control point coordinates (\mathbf{X}_c), which serve as design variables. The geometry and FFD control points are coupled explicitly, $\mathbf{X}_p = f(\mathbf{X}_c)$ so that the geometry is deformed as FFD control points are adjusted. This two-level deformation approach is used since the setup is independent of the geometry complexity, which is particularly useful for complex geometries. Furthermore, decoupling the FFD control points directly from the surface parametrization allows the number of B-spline control points to remain flexible, allowing for a high-quality fit to the geometry surface, without increasing the number of design parameters. To streamline large-scale geometry adjustments, axial curves can be embedded along the geometry through several FFD volumes, parametrized as a separate B-spline curve. This curve controls high-level span, sweep, and dihedral changes, while embedded FFD volumes control local geometry adjustments such as local twist, chord, section shape and thickness.

D. RANS Flow Solver and Spatial Discretization

The flow solver, Diablo, is a structured multiblock three-dimensional parallel implicit flow solver for the RANS equations closed with the negative variant of the one-equation Spalart–Allmaras turbulence model [47]. Second-order centered difference summation-by-parts operators with simultaneous approximation terms are used for spatial discretization. The resulting system of non-linear equations is solved using Newton’s method in two phases: an approximate-Newton phase followed by an inexact-Newton phase. In the approximate-Newton phase, an approximate first-order flow Jacobian is used with the implicit Euler method and a slowly-increasing time step. The system is solved to a relative residual drop on the order of 10^{-4} . The resulting solution is then used as an initial iterate for the inexact-Newton phase, where the full second-order accurate flow Jacobian is formed. The system is then solved to

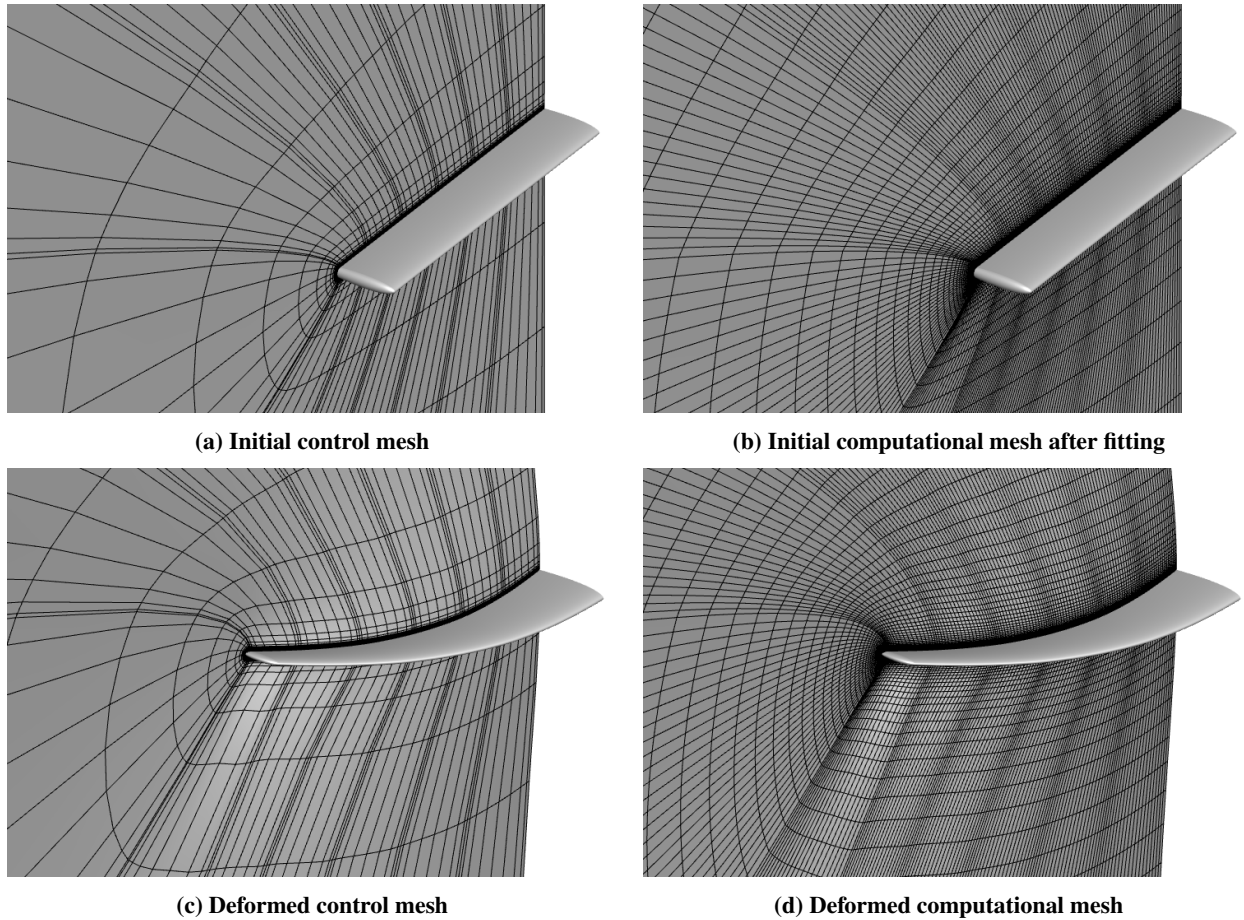


Fig. 1 Initial and deformed control and computational meshes for a sample rectangular wing deformation

steady-state by rapidly increasing the time step. At each outer iteration the system is solved using the generalized minimum residual (GMRES) Krylov subspace method with approximate Schur preconditioning [48]. Flow solutions are converged to a relative tolerance on the order of 10^{-10} . The flow solver has been validated at the Fifth AIAA Drag Prediction Workshop, where the drag coefficient of the NASA Common Research Model (CRM) was predicted to within one count of the median across all participants [49].

E. Gradient-Based Optimization

Jetstream uses a gradient-based local optimization algorithm called SNOPT [42], which solves large-scale non-linear constrained optimization problems using a sequential quadratic programming framework. The gradient of the objective function at each optimization iteration is evaluated using the discrete-adjoint method [43, 44, 50, 51]. Of particular note is that the adjoint equations are augmented to include the sensitivity of the objective function to the mesh deformation algorithm to ensure gradient accuracy [12]. The necessary partial derivatives for the adjoint equations are evaluated either through analytical differentiation or by using a complex-step [52]. The flow adjoint equation is solved to a relative tolerance on the order of 10^{-8} using a simplified and flexible version of the generalized conjugate residual with orthogonalization and truncation (GCROT) method [53]. The linear system for the mesh adjoint equations remains symmetric and positive-definite, and thus is solved using the PCG method to a relative tolerance on the order of 10^{-12} .

III. Investigating the Mesh Deformation Scheme

For a linear elastic mesh deformation scheme, one must develop formulations for the Young’s modulus (E) and the Poisson’s ratio (ν) that strike a balance between adequate element orthogonality and allowing large deformations, while preventing ill-conditioning. An E formulation controls how easily elements are able to deform (stiffness), whereas a ν formulation controls how the elements physically deform (compressibility) [54]. Therefore, by selectively building up appropriate formulations for both variables, we should be able to form a robust linear-elastic mesh deformation scheme that works for a variety of shape changes and meshes. It should be noted that there is naturally an upper limit of the magnitude of geometric changes above which all mesh deformation schemes which do not alter the mesh topology will fail. There exists a point where a geometry becomes so removed from its initial shape that re-meshing is necessary. We look to improve the current scheme in Jetstream to expand the range of geometric changes that can be handled without encountering difficulties associated with mesh deformation, moving closer to this upper limit.

A. Diagnosing Issues Encountered During Mesh Deformation

During an optimization, mesh deformation failures may occur when the deformation scheme fails to deform to a suitable computational mesh. Four types of failures may occur during a mesh deformation: 1) crossed mesh nodes, 2) degenerate mesh elements, 3) unsuccessful PCG convergence, and 4) unsuccessful flow evaluation after deformation due to an insufficient mesh quality. The first two failures yield invalid mesh elements; the third failure prevents a suitable deformed mesh from being obtained. A flow evaluation can be considered a “test” of the deformed mesh if the other three failures are not encountered, for which a failure may be caused by low-quality elements after deformation.

To diagnose mesh deformation failures on optimization cases, a procedural checklist was developed to properly determine the cause of the failure. When a deformation failure occurs, the first step is to check the modified geometry. Is the geometry feasible? Geometric constraints may be improperly inputted into the optimization problem which leads to invalid, e.g. crossed-over, geometries. Furthermore, does the geometry surface near the problematic control mesh elements look realistic? Are there unusual kinks in the geometry surface or control surfaces which may be causing an invalid deformation? An improper geometry may also point to another issue with the problem formulation, which must be separately diagnosed. Next, was the control mesh properly fitted to the computational mesh? If the initial control mesh is invalid, containing crossed nodes or degenerate or poor-quality elements, this may be solved by adjusting the number of B-spline control points or the order of the B-spline volumes. If the control mesh is valid, did the PCG solution of the linear system converge successfully? During complex deformations, the residual may stall and fail to converge. This may be addressed by either modifying the preconditioner or adjusting deformation scheme parameters to aid convergence. If the PCG solve is valid, were enough increments used to break up the deformation? A low number of increments will violate the small strain assumption for large deformations and cannot properly stiffen at-risk elements, leading to a poor quality mesh. Finally, we check the problematic elements: Are they adequately orthogonal, or are they highly skewed, degenerate, or contain crossed mesh nodes? The subsequent sections describe a few issues which were encountered during optimization runs which may trigger mesh deformation failures.

B. Increasing Element Quality During Deformation

If the initial mesh has adequately orthogonal elements and the mesh does not undergo a large deformation, the spatially-varying E formulation described in Section II is robust for a large variety of shape changes. However for more complex geometries, such as BWBs, the initial mesh may contain skewed elements, particularly near highly curved blending regions or extruded elements such as winglets. We can visualize this by plotting a quality measure similar to the element distortion used to stiffen elements in the linear elasticity algorithm [43]. Specifically, the quality for an element (Q_e) is given by

$$Q_e = \sqrt{\frac{1}{\Phi_e}} = \prod_{j=1}^8 (\mathbf{u}_j \times \mathbf{v}_j) \cdot \mathbf{w}_j, \quad (8)$$

where Φ_e is the element distortion given in Equation 6. From this definition, orthogonal elements have $Q_e = 1$ while highly skewed elements have $Q_e \rightarrow 0$. Figure 2 shows the quality distribution of the control meshes for a NACA0012 extruded rectangular wing and a narrow-body BWB from Gray and Zingg [11]. Both meshes use an O-O topology and are shown in Figure 3. We can see that the BWB mesh, which contains more complex regions of high curvature, has a large number of highly skewed elements relative to the mesh around the rectangular wing. Generally, a more complex geometry will contain more lower-quality elements.

For large shape changes on these complex meshes, some elements, particularly those near regions of high curvature,

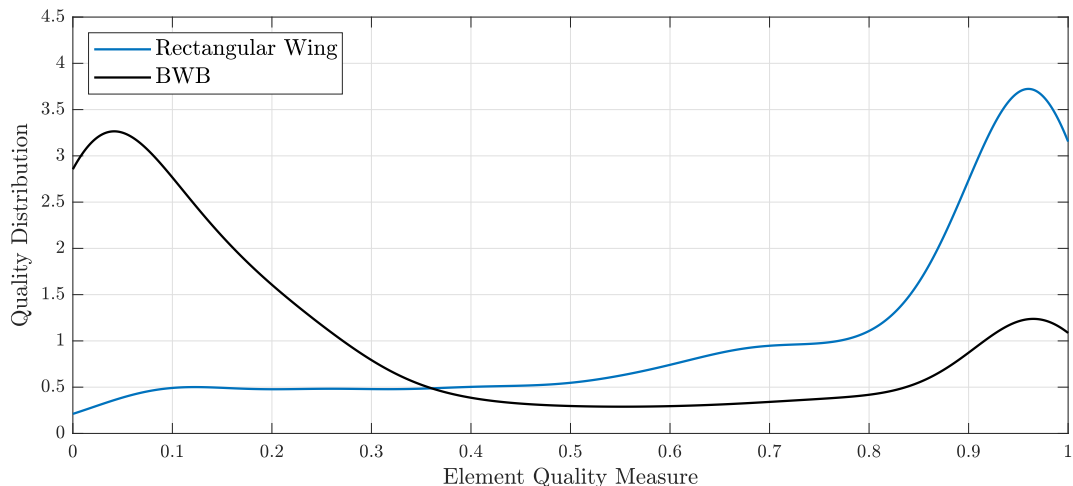


Fig. 2 Quality distribution of the initial control meshes for a NACA0012 extruded rectangular wing from Koo and Zingg [55] and a narrow-body BWB from Gray and Zingg [11]. Both control meshes consist of 118k nodes

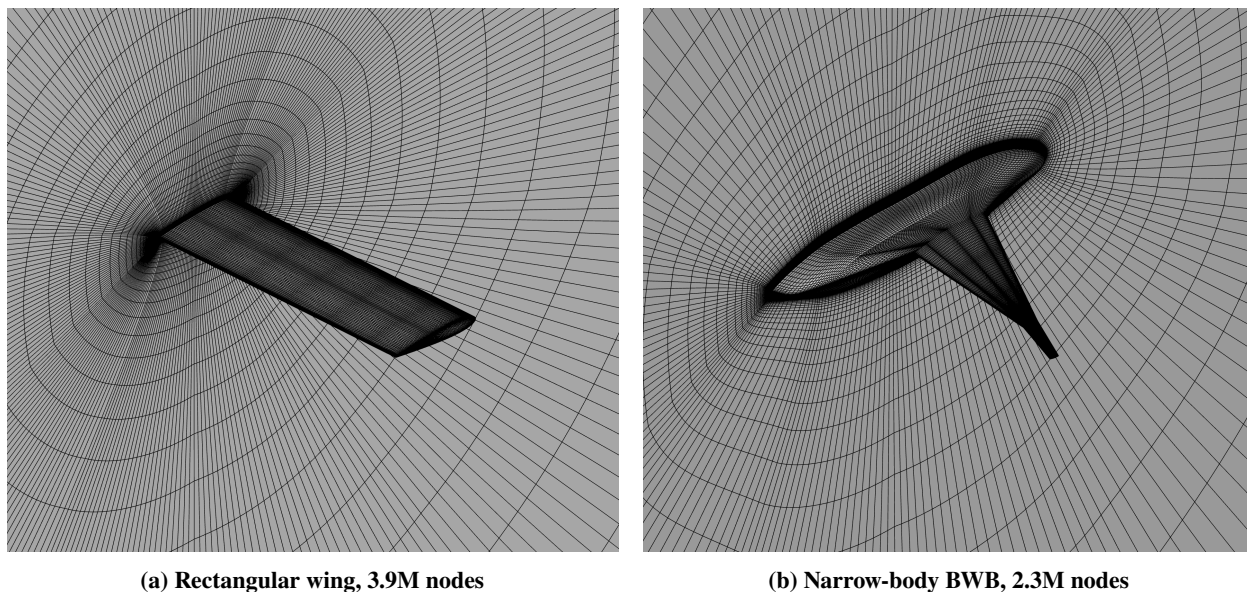


Fig. 3 Computational meshes for the geometries in Figure 2, showing their O-O topology

may (uncharacteristically) un-skew and get better in quality throughout the mesh deformation. This causes these elements to have low distortions $\Phi^{(i)}$ relative to their initial values $\Phi^{(0)}$, lowering their Young's modulus in usually higher-stiffness regions. The value of E can become up to eight or nine orders of magnitude lower than neighbouring elements. The large spread of stiffness values in \mathbf{K} can result in crossed mesh nodes post-deformation, leading to negative volumes.

C. Mesh Dependencies

For some multipoint optimization cases, multiple meshes may exist to appropriately model a geometry at different operating conditions. For example, a BWB optimization problem formulation from Gray and Zingg [11] contains both a hemispherical free-air mesh as well as a mesh which contains a ground plane to model the ground effect at take-off, both shown in Figure 4. Both these meshes deform to the same geometry at each optimization iteration. For the free-air mesh,

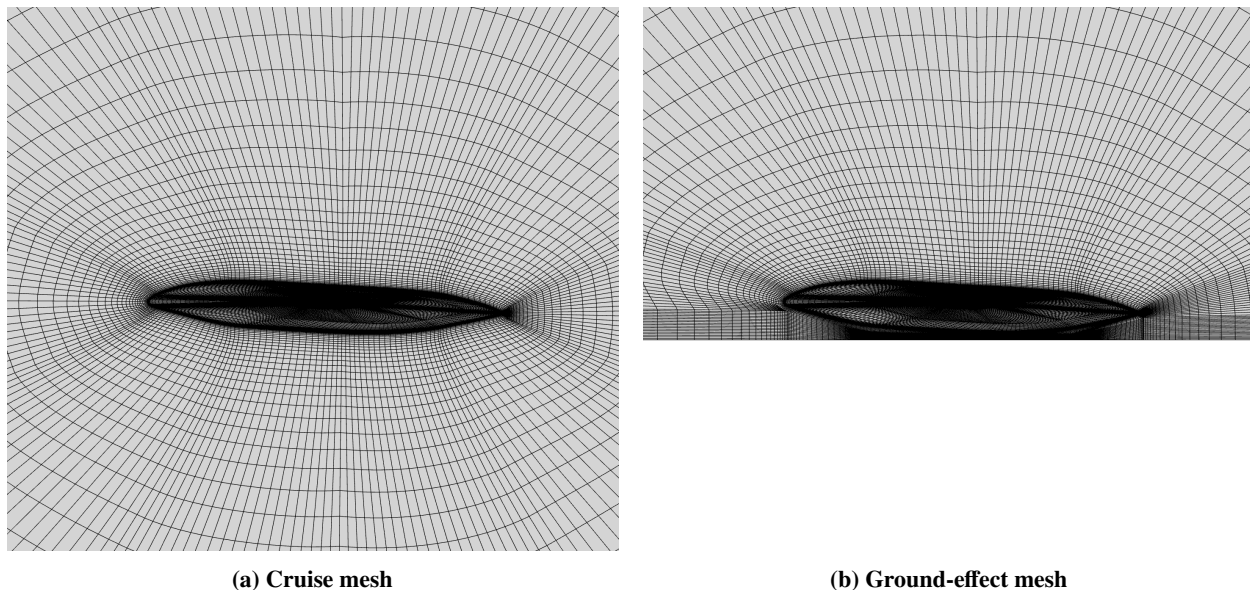


Fig. 4 Two computational meshes used in Gray and Zingg [11] for multipoint optimizations, showing their symmetry plane

it was observed that using a Poisson's ratio of $\nu = [0.45, 0.5]^{\ddagger}$ allows for the highest geometric freedom, increasing robustness of most optimization runs. This mimics a near-incompressible material such as rubber where the element volume is preserved during deformation. However, with the ground-plane mesh, some deformations on the lower surface of the geometry were not successful with large ν values, resulting in degenerate elements or an unsuccessful PCG convergence with Jacobi preconditioning. This was addressed by using $\nu \approx -0.2$ on the ground-plane mesh [11].

Overall, multipoint cases that use multiple meshes can require individually tuned ν values for each operating point to successfully deform the control mesh, hindering automation. This motivates the use of a spatially-varying ν formulation, a more universal framework which is applicable to a larger variety of meshes.

D. Degenerate Elements on the Surface

During an optimization, particularly one with high geometric freedom, the surface may be adjusted such that it contains undesirable kinks or irregularities which are gradually smoothed as the optimization progresses. However, the prescribed displacement of the mesh nodes on the surface may cause surrounding elements on the control mesh to either highly skew or become degenerate. Figure 5a shows a partially optimized BWB geometry, focused on the base of its trailing edge. The geometry has an outward kink at the root of the wing. Figure 5b shows a cross-section of the control mesh at the kink; we can see the elements at the surface are degenerate with adjacent edges in-line with each other, forming triangle-like elements.

These degenerate elements can occur at the interfaces between two B-spline patches or two axial curves where the geometry is highly curved or may not be C^2 continuous. These elements are difficult to appropriately deform in a linear-elastic scheme due to the awkward prescribed surface displacement. This may be dealt with by appropriately constraining the geometric movement, although this is problematic for cases with higher intended freedom. Some deformations have been successful by tweaking the Poisson's ratio, although successful ν values were observed to be case-dependent with no noticeable pattern.

A similar issue may occur where the computational mesh becomes degenerate after deformation if the control mesh highly skews. This also typically occurs in regions of high curvature, although the geometry surface may be smooth. Figure 6a shows the planform of a partially optimized BWB, and Figure 6b shows the elements on the control mesh near the winglet after deformation. We see that the block face highly skews such that two adjacent faces are in-line, causing degenerate elements on the computational mesh.

[‡]A Poisson's ratio of $\nu = 0.5$ is not possible in the linear elastic equations since $\sigma \rightarrow \infty$ in Equation 2 with $\nu = 0.5$.

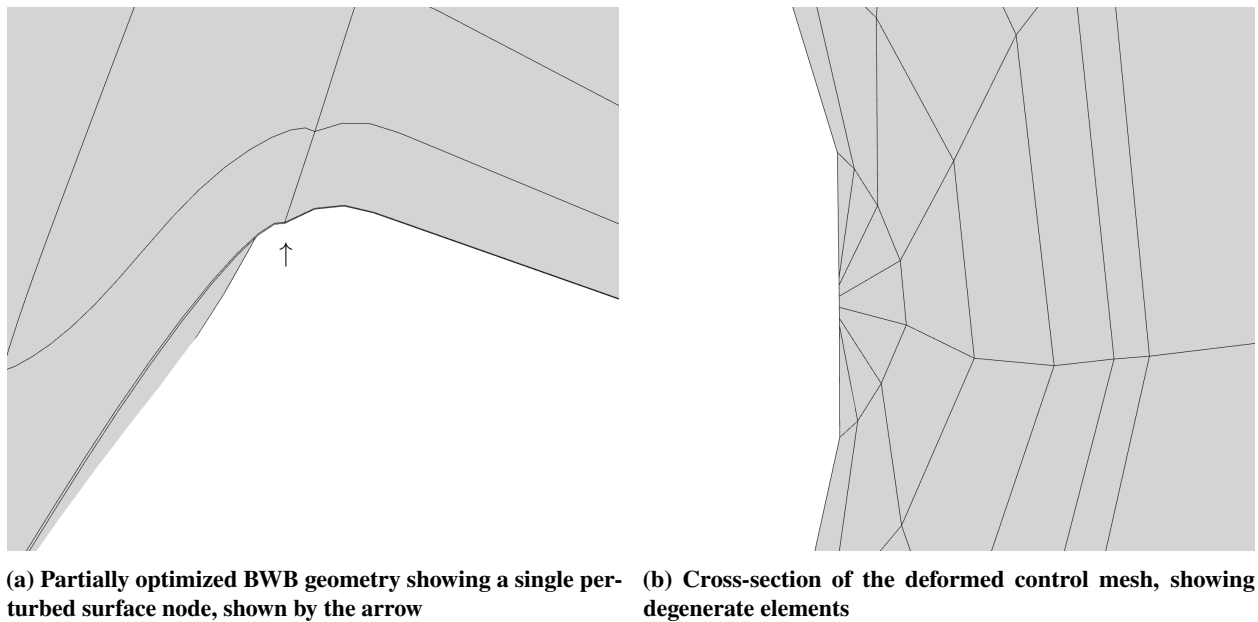


Fig. 5 Example of degenerate elements due to irregular kinks in the geometry

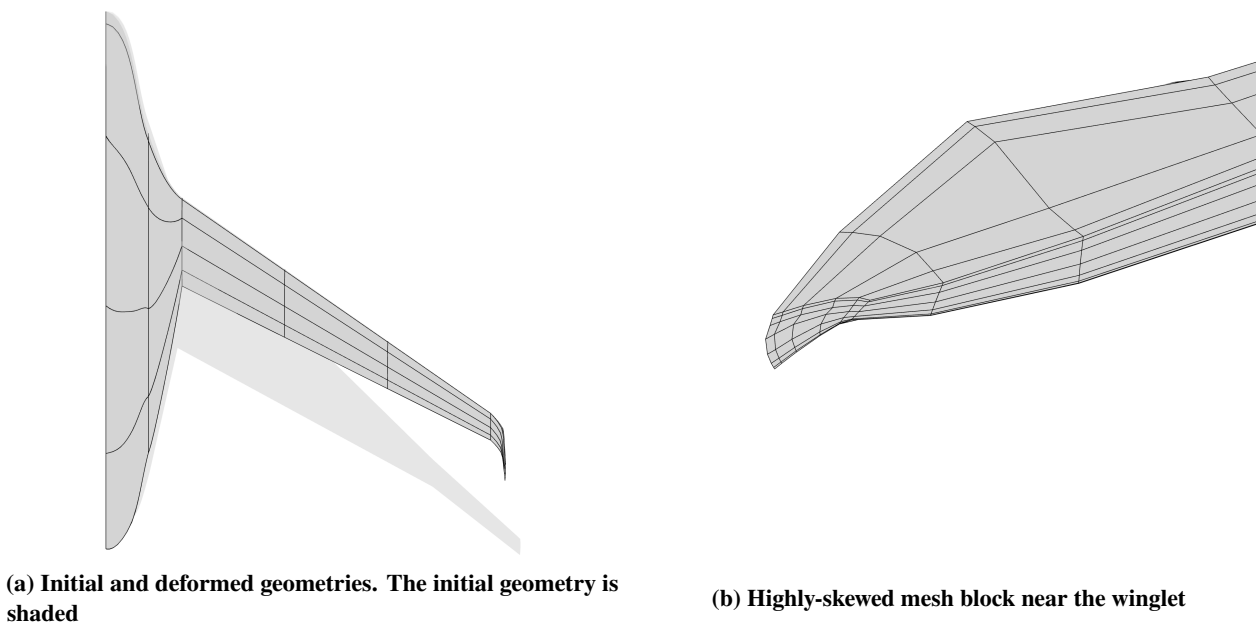


Fig. 6 Partially optimized BWB with degenerate elements near the winglet on the computational mesh. The control mesh consists of 118k nodes

The skewness of these blocks may be reduced by adjusting the Poisson’s ratio or increasing the distortion dependency on E . However, as noted above, a suitable Poisson’s ratio is case-dependent, and increasing the distortion dependency may create an unnecessarily ill-conditioned stiffness matrix \mathbf{K} , inhibiting performance of the mesh deformation scheme on other geometries during an optimization. This further motivates revised E and ν formulations to prevent these highly-skewed blocks from occurring while retaining large geometric freedom.

IV. A Robust Mesh Deformation Scheme

The discussions in Section III.A suggest that a more robust mesh deformation scheme can be developed by augmenting the linear elastic deformation through updated spatially-varying Young's modulus and Poisson's ratio formulations. These updated formulations are described in the subsequent sections. The changes made augment only E and ν , the two inputs to the equations of linear elasticity. Thus, the scheme is not only applicable to a B-spline control mesh framework, but can be generalized to deformations on most computational meshes or control mesh parametrizations, including structured or unstructured meshes, when the linear elasticity approach is used.

A. A Revised Young's Modulus Formulation

To address unrealistically low Young's modulus values during deformation, one approach is to truncate the allowed E values at a lower bound [33]. For a given initial control mesh, it is assumed that it is adequately orthogonal to serve as a starting point for all mesh deformations. Therefore, it is unnecessary to lower the E value beyond the initial value if an element becomes more orthogonal. Consequently, we truncate the distortion ratio to a minimum value of unity, leading to the following revised formulation:

$$E_e^{(i)} = \frac{1}{V^{(i)}} \max \left(1, \frac{\Phi_e^{(i)}}{\Phi_e^{(0)}} \right). \quad (9)$$

Thus if the quality of an element increases during mesh deformation (distortion decreases below its initial value), its stiffness is not allowed to decrease. To enable differentiability to calculate the mesh adjoints during an optimization, the max function is smoothed with a smooth-maximum unit operator [56]:

$$\max \left(1, \frac{\Phi_e^{(i)}}{\Phi_e^{(0)}} \right) \approx \max_p \left(1, \frac{\Phi_e^{(i)}}{\Phi_e^{(0)}} \right) = \frac{1 + \frac{\Phi_e^{(i)}}{\Phi_e^{(0)}} + \left[\left(1 - \frac{\Phi_e^{(i)}}{\Phi_e^{(0)}} \right)^2 + p \right]^{1/2}}{2}, \quad (10)$$

where setting p , which controls how gradually the two functions in the max function cross over, to $p = 5 \times 10^{-4}$ has been successful. To make the formulation suitable for a larger set of shape changes, we can also adjust the relative volume and distortion contributions to E . We can accomplish this by generalizing Equation 7 by adding an exponent to the distortion term:

$$E_e^{(i)} = \left(\frac{1}{V^{(i)}} \right) \left(\frac{\Phi_e^{(i)}}{\Phi_e^{(0)}} \right)^\beta. \quad (11)$$

The parameter $\beta \in [0, \infty)$ adjusts the distortion contribution to E relative to the volume contribution. As β decreases, the spread of stiffness values in the global \mathbf{K} matrix decreases, which was observed to lead to an easier system to solve. However, lowering β also decreases the stiffening of highly skewed elements, which may lead to a poorer-quality mesh unsuitable for a flow evaluation.

Combining both changes, the modified E formulation is expressed as:

$$E_e^{(i)} = \frac{1}{V_e^{(i)}} \max_p \left[1, \left(\frac{\Phi_e^{(i)}}{\Phi_e^{(0)}} \right)^\beta \right] \quad (12)$$

Setting $\beta = [0.5, 1.0]$ has led to improved robustness of the mesh deformation scheme on a variety of large deformations. However, to maximize mesh quality it is recommended to default to $\beta = 1$ and then adjust the value downwards as required if problems arise. As a quick note, for application to unstructured and other non-hexahedral-based meshes, an alternative distortion metric to Equation 6 may be substituted.

B. A Spatially-Varying Poisson's Ratio

To increase the robustness of the mesh deformation scheme across a variety of meshes and shape changes, as well as remove the task of selecting a suitable ν value, we augment the scheme using a spatially-varying Poisson's ratio. To aid in building a suitable scheme, particularly since scarce literature exists on suitable ν formulations for a linear-elastic deformation scheme [33], it is necessary to define its critical requirements. Most of the requirements fall in line with previously developed formulations by Nielsen and Anderson [37] and Karman and Anderson [38]. First, for simplicity,

ν will be restricted to positive values. Second, elements near the surface should have $\nu \rightarrow 0.5$, mimicking a highly incompressible material, so their volume conservation combined with the forced surface node displacement preserves element quality. Furthermore, elements away from the surface should have $\nu \rightarrow 0$, mimicking a material such as cork, so the lack of tangential movement in response to displacements allows these elements to deform more easily to accommodate shape changes. The formulation must be able to integrate well and accommodate the functionality of the spatially-varying E formulation, whose purpose is to stiffen poor quality elements and propagate deformations away from the surface. It is also necessary for the formulation to be able to work with a large number of meshes, and it should not depend on the non-dimensional extent of the mesh. Hence, ν itself should be non-dimensional. Finally, the formulation should be able to integrate well with PCG for a wide variety of meshes and shape changes.

From the requirements above, we set ν based on the normalized off-wall distance:

$$\frac{1}{1 - 2\nu} = \left(\frac{d_{\max}}{d_e} \right)^\gamma, \quad (13)$$

where d_e is the cell-centered off-wall distance of the element on the control mesh, d_{\max} is the maximum off-wall distance within the domain, and γ is a tuning parameter. This normalizes all element off-wall distances to between zero and one independent of the non-dimensional extent of the mesh. An advantage of this method is that it guarantees elements near the surface will have $\nu \approx 0.5$ and elements away from the surface will have $\nu \approx 0$. Furthermore, the smoothness of the off-wall distance across the mesh domain aids with ensuring suitable PCG performance, minimizing the additional computation time. For this formulation, setting $\gamma = 0.5$ has been successful.

V. Case Studies

In this section, the robust mesh deformation scheme in Section III is applied to three case studies to assess its effectiveness for cases demanding large shape changes and extreme deformations. The three case studies are: 1) a rectangular wing to BWB deformation, 2) a single-aisle-size BWB optimization with chord freedom across its planform, and 3) a twin-aisle-size BWB optimization with large geometric freedom.

A. Wing to BWB Deformation

The initial shape is an extruded NACA0012 rectangular wing with a unit chord and a semi-span of three. The computational mesh consists of 3.9M nodes over 128 blocks, which is fitted with a control mesh consisting of 118k nodes. The initial off-wall spacing averages 2.0×10^{-6} chord lengths. The wing is deformed using Jetstream's FFD control module by fitting the wing to one axial curve, of fourth order, with nine control points. For local control, 36 FFD volumes are embedded along the axial curve. The BWB shape was made by hand to mimic a narrow-body BWB, including a narrow winglet, rounded at its tip, curved upwards, a high negative trailing-edge sweep angle in the blending region, and a wing root twist and dihedral of five degrees each. Figure 7 shows the initial and final geometries for the deformation along with the location of the axial control points. The cross-section over the BWB remains a NACA0012 airfoil. To limit small strain violations, 50 increments are used, with $\beta = 0.9$.

Figure 8 shows the control and computational meshes for the deformation. We see that by using B-spline control points to form the control mesh, the computational mesh is able to maintain a high quality near the surface despite the extreme deformation. It should be noted that the average off-wall spacing increases to 5.4×10^{-6} on the deformed mesh. The majority of off-wall increase occurs at the winglet, although an increase is seen at the wing root where there is also high distortion due to the deliberately squared-off trailing edge. We can observe this on the control meshes in Figure 8, where the blocks near the winglet are of a greater width in the off-wall direction than the blocks along the wing. Due to the high Poisson's ratio at the surface, we can observe a general correlation between the relative area change and the off-wall spacing: Near the winglet as the chord decreases from its initial length, the mesh elements must squeeze together, forcing an increase in the off-wall direction due to the high Poisson's ratio. This may be fixed by manually adjusting the off-wall spacing of the initial computational mesh through the B-spline fitting procedure, although this must be monitored throughout an optimization to ensure accurate flow results are obtained. Another fix may be to reduce γ , which controls how quickly ν decreases away from the surface. Setting $\gamma = 0.25$, fewer control mesh elements have a high Poisson's ratio near the surface, lowering the average off-wall spacing to 4.6×10^{-6} . However, further lowering γ results in too many distorted elements, causing crossed mesh nodes post-deformation.

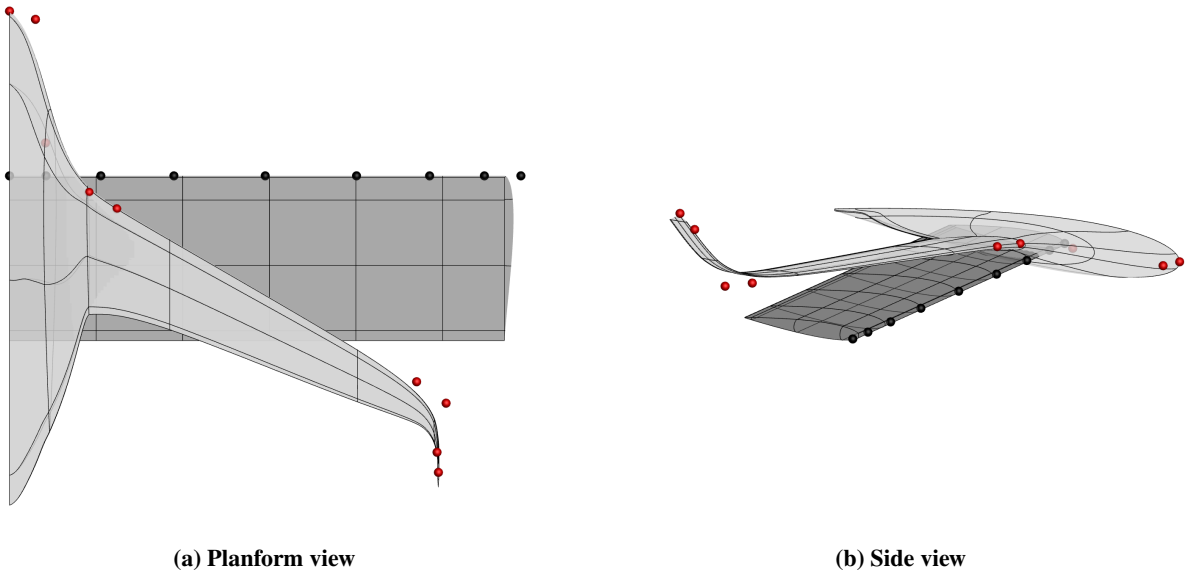


Fig. 7 Overlay of initial rectangular wing and deformed BWB shape. The axial points are shown in black (initial) and red (deformed)

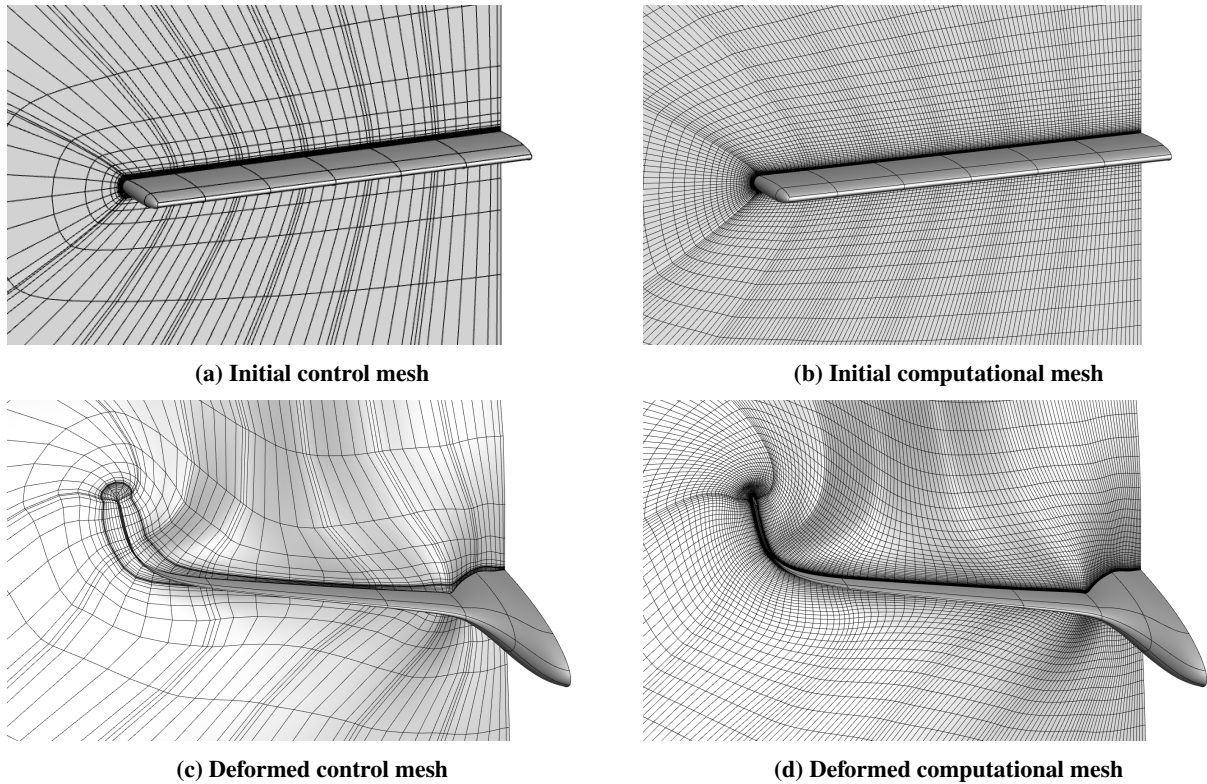


Fig. 8 Initial (top) and deformed (bottom) control and computational meshes for a rectangular wing to BWB deformation

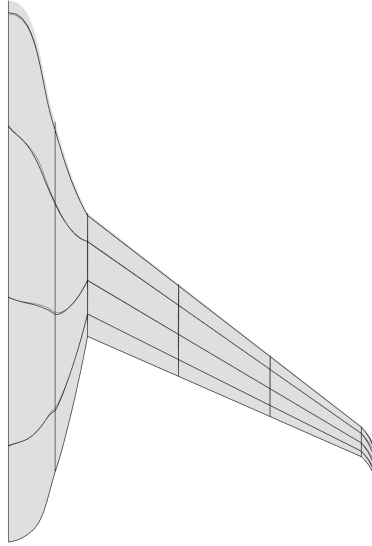


Fig. 9 Sample initial BWB geometry from Gray and Zingg [11].

Table 2 Summary of operating conditions for the BWB optimization problem formulation from Gray and Zingg [11]

	Mach #	Alt. (ft)	Re	Initial AoA	K_n	Thrust	Mesh
Cruise	0.78	36000	76.55×10^6	1.92°	No	Full	Free-air
Take-off	0.15	0	44.23×10^6	0° (constant)	No	1 engine	Ground-plane
Approach	0.20	0	60.20×10^6	9.88°	Yes ($\geq -4\%$)	None	Free-air

B. BWB Optimization with Expanded Geometric Freedom

This case is a shape optimization problem which minimizes block-fuel-burn (BFB) for a single-aisle-size BWB over a design mission of 3100 nmi with a payload of 22 880 lbs. The initial geometry and geometry control system are taken from Gray and Zingg [11]. Global geometry control consists of 4 joined axial curves with 10 control points, corresponding to the centerbody, blending region, wing and winglet. Local control is achieved using 25 FFD cross-sections. Figure 9 shows the initial geometry.

The case takes the baseline optimization problem formulation from Gray and Zingg [11] and expands the allowed geometric freedom to permit free chord adjustments along the entire centerbody and wing. The three operating conditions considered are listed in Table 2. These are a cruise condition, a take-off condition with one engine inoperable in ground effect at a constant angle-of-attack of zero, and a low-speed high-lift condition near landing with no thrust. Reynolds numbers used are based on a mean aerodynamic chord of 42.4 ft. The static margin (K_n) is constrained to be greater than or equal to -4% of the mean aerodynamic chord at the low-speed operating point. To aid with trim, winglet rudders are included at take-off and three flap-elevons and a centerbody elevator are included at low-speed approach. The winglet remains constrained to the original geometric bounds of Gray and Zingg [11]; no additional freedom is given. Furthermore, the mesh for the take-off condition contains a ground plane to model the ground-effect, compared to the hemispherical mesh used at cruise and approach. At cruise and approach, a coarse computational mesh is used consisting of 1.0M nodes over 128 blocks, fitted to a control mesh with 118k nodes. On this grid, the average y^+ is 1.18. At take-off, the computational mesh consists of 1.1M nodes over 144 blocks fitted to a control mesh of 110k nodes. Both meshes are shown in Figure 4. Deformations are broken into 20 increments and use $\beta = 0.7$. Several low- and medium-fidelity models and constraints are included in the problem formulation, including a representative internal cabin which must fit inside the outer mold line of the aircraft and a wing volume constraint based on the required fuel volume for the design mission. To aid convergence, the optimization is warm-started from the optimized benchmark geometry from Gray and Zingg [11].

This optimization could not successfully converge with the original mesh deformation scheme, exiting early in

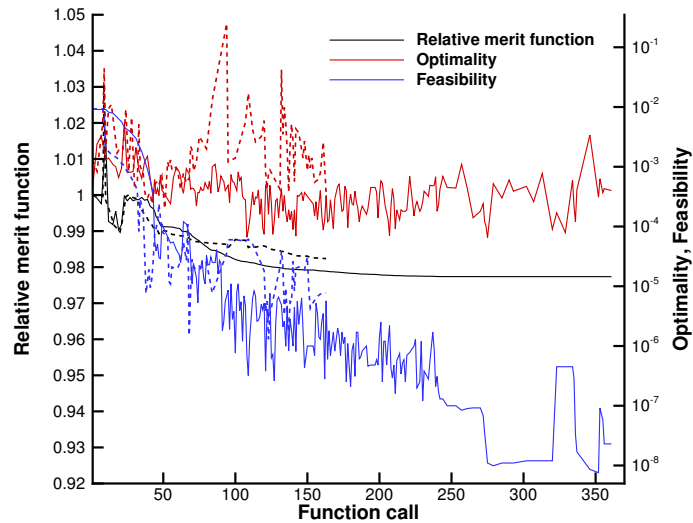


Fig. 10 Optimization histories for BWB geometries with expanded geometric freedom with the original (dashed) and updated (solid) mesh deformation schemes

the optimization due to crossed mesh nodes during deformation. Figure 10 shows the optimization histories with and without the updated deformation scheme. The original deformation scheme uses $\nu = 0.45$ and $\nu = -0.2$ on the cruise and take-off grids respectively. From the optimization histories, the optimization does not progress as smoothly as with the new deformation scheme: With the old deformation scheme the optimization encountered 23 mesh deformation failures over 164 function evaluations (a 14% failure rate), the majority occurring in the later iterations due to crossed mesh nodes. Any small geometric perturbation from a working geometry resulted in a deformation failure, preventing the optimization from adequately exploring the design space near the minimum. In contrast, the optimization with the updated deformation scheme encountered eight mesh deformation failures over 369 function evaluations (a 2% failure rate). The majority of these failures occur directly after four Hessian resets which were performed by SNOPT throughout the optimization, which generated unrealistic geometries as improper steps were taken in the design space. With the updated deformation scheme, the optimization was able to adequately explore the design space around the minimum, showing its robustness and reliability.

Figure 11 shows the optimized geometry obtained using the updated deformation scheme compared to the benchmark optimized geometry from Gray and Zingg [11]. Table 3 shows performance statistics for the optimized geometries. Unlike Gray and Zingg [11], the computational mesh is not refined on optimized geometries. Thus, the performance statistics vary from the reported values in Gray and Zingg [11] for the benchmark optimization. From Figure 11, the aircraft develops a wing crank as well as a curved trailing-edge interface with the blending region; generally, the aircraft appears more blended. There is also a slight centerbody length increase, and the trailing-edge along the centerbody and blending region is less monotonic. The majority of the wing volume has shifted inwards, with the wing chord sharply decreasing to the fixed winglet chord length for nearly two-thirds of the wing. It should be noted that the FFD control points are selectively located to form the flat control surfaces along the wing. This leads to the appearance of three distinct straight regions along the wing. From Table 3, there is a decrease in the wetted area (S_{wet}), leading to an increase in the lift-to-drag ratio (L/D) at cruise. There is also an increase in span to aid with directional trim at take-off, which results in an increase in the maximum take-off weight (MTOW) and operating empty weight (OEW). Overall, there is a decrease in BFB of 1.6% from the optimized benchmark with less geometric freedom.

Finally, we recognize that the optimized geometry which is obtained does not satisfy all important design requirements, and additional constraints are required to make the optimization problem more suitable to a practical application. However, this optimization shows the potential of the updated deformation scheme and the expanded design space which is now able to be robustly and reliably explored.

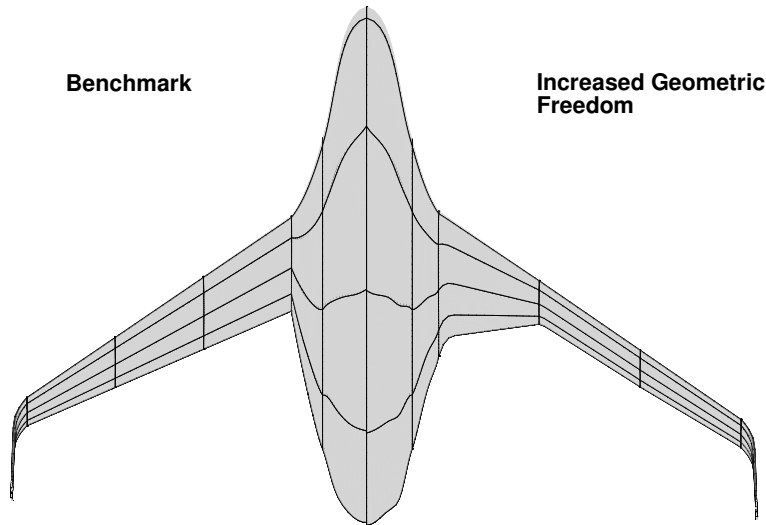


Fig. 11 Optimized BWB geometries with and without expanded geometric freedom

Table 3 Performance statistics for optimized BWB geometries with and without expanded geometric freedom. The lift-to-drag ratio (L/D) and drag coefficient (C_D) are reported for the cruise condition

	BFB (lbs)	MTOW (lbs)	OEW (lbs)	C_D (cnts.)	L/D	Span (ft)	S_{proj} (ft ²)	S_{wet} (ft ²)	AR_{wet}	Sweep (°)
Benchmark	20179	111430	64498	106.7	21.32	125.1	2672	6298	2.48	34.3
High Freedom	19860	112560	66003	104.4	22.02	137.4	2665	6218	3.04	34.2
$\Delta\%$	-1.6	+1.0	+2.3	-2.2	+3.3	+9.8	+0.3	-1.3	+22.6	-0.3

Table 4 Summary of operating conditions for a twin-aisle-size BWB optimization problem formulation from Yazdi et al. [57]

	Mach #	Alt. (ft)	Re	Initial AoA	K_n	Thrust	Mesh
Cruise	0.85	36000	135.5×10^6	2.0°	No	Full	Free-air
Take-off	0.15	0	72.1×10^6	0° (constant)	No	1 engine	Ground-plane
Approach	0.21	0	97.9×10^6	6.57°	Yes ($\geq -4\%$)	None	Free-air

C. Twin-Aisle-Size BWB Optimization

This optimization takes the problem formulation from Gray and Zingg [11] and applies it to a twin-aisle-size BWB to match the specifications of an Airbus A350-900. The updated problem formulation is taken from Yazdi et al. [57]. The objective function which is minimized is BFB over a design mission of 6000 nmi with a payload of 114 641 lbs. As in the case in Section V.B, three operating conditions are considered, as shown in Table 4, where the Reynolds numbers used are based on a mean aerodynamic chord of 75.5 ft. The static margin (K_n) is constrained to be greater than or equal to -4% of the mean aerodynamic chord at the low-speed operating point. An identical FFD and axial control scheme to Section V.B is used, consisting of 4 joined axial curves with 10 control points and 25 FFD cross-sections for global and local control respectively. The geometric freedom is consistent with Gray and Zingg [11], without the expanded chord freedom used in Section V.B. The initial geometry is shown in Figure 12. At cruise and approach, the computational mesh consists of 1.0M nodes over 128 blocks, fitted to a control mesh with 103k nodes. At take-off, the computational mesh consists of 1.1M nodes over 144 blocks fitted to a control mesh of 102k nodes, and contains a ground plane to model the ground-effect. Deformations are broken into 30 increments and use $\beta = 0.7$.

Figure 13 shows a planform view of the optimized geometry obtained with the original and updated deformation schemes. Figure 14 shows their optimization histories. This case could not converge deeply with the original deformation scheme, exiting early due to degenerate elements in the blending region along the trailing edge as a result of a sharp

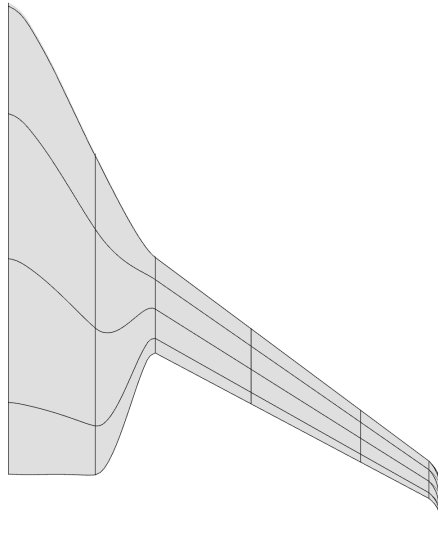


Fig. 12 Initial BWB geometry from Yazdi et al. [57].

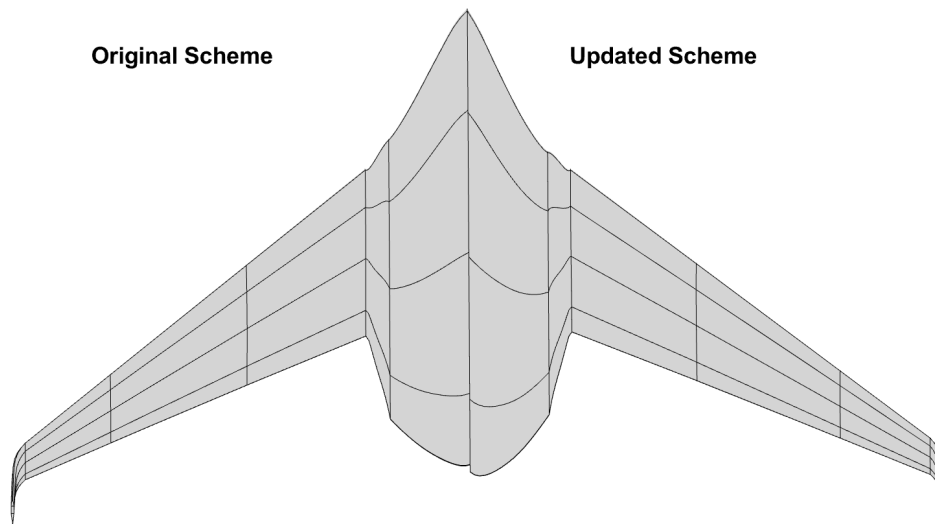


Fig. 13 Optimized twin-aisle-size BWB geometries with and without the updated deformation scheme

decrease in the blending region span throughout the optimization. There is also a centerbody length increase and rounded-off trailing edge with the updated scheme, which the original scheme was unable to fully form.

Table 5 shows performance statistics for the optimized geometries with the original and updated deformation scheme. The updated scheme yields 3.3%, 1.1%, and 4.5% decreases in BFB, the MTOW, and the C_D respectively from the original scheme, as well as a 2.9% increase in the L/D ratio. These show the potential of the new deformation scheme when applied to this class of aircraft, notably the deeper convergence which can be obtained.

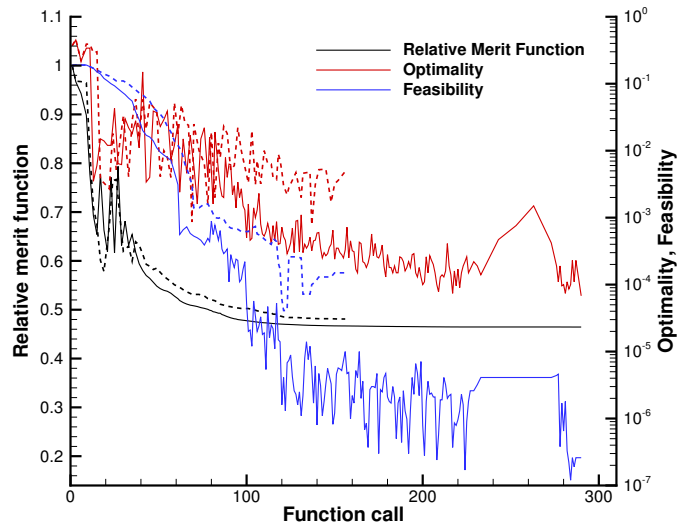


Fig. 14 Optimization histories for a twin-aisle-size BWB with the original (dashed) and updated (solid) mesh deformation schemes

Table 5 Performance statistics for optimized twin-aisle-size BWB geometries with and without the updated deformation scheme. The lift-to-drag ratio (L/D) and drag coefficient(C_D) are reported for the cruise condition

	BFB (lbs)	MTOW (lbs)	OEW (lbs)	C_D (cnts.)	L/D	Span (ft)	S_{proj} (ft ²)	S_{wet} (ft ²)	AR_{wet}	Sweep (°)
Original Scheme	131694	625308	362757	116.9	26.34	242.1	9952	21788	2.69	39.4
Updated Scheme	127406	618543	360755	111.6	27.16	249.2	9947	21718	2.86	38.1
$\Delta\%$	-3.3	-1.1	-0.6	-4.5	+3.1	+2.9	-	-	+6.3	-3.3

VI. Conclusions

In this paper, we present a framework for a robust mesh deformation scheme based on the equations of linear elasticity within the context of an aerodynamic shape optimization algorithm. We spatially vary both the Young's modulus and the Poisson's ratio to work in tandem, controlling element stiffness and compressibility respectively, to maintain element quality near the surface and propagate deformations outwards into the farfield. In particular, E is set using the element volume and a truncated element distortion measure, and ν is set using the element off-wall distance. The scheme is designed to work on a variety of grids and geometries, allowing for large shape changes.

To demonstrate the performance of the updated deformation scheme, three cases were presented: a wing to narrow-body BWB deformation, a single-aisle-size BWB optimization with high geometric freedom, and a twin-aisle-size BWB optimization. Results for the wing to BWB deformation show the updated deformation scheme is able to handle a large suite of extreme deformations, including large wing sweeps and twists, winglet formations, and large wing-centerbody shape changes. Presented optimizations exploit this performance improvement to produce converged optimized aircraft for cases with large shape changes. For a single-aisle-size BWB optimization with expanded geometric freedom, deep convergence could not be achieved with the original mesh deformation scheme. Convergence was achieved with the updated mesh deformation scheme, leading to performance benefits from the larger design space associated with the increased geometric freedom. For a twin-aisle-size BWB optimization, deeper convergence was obtained with the updated deformation scheme, resulting in a more optimized aircraft with more refined geometric features near the blending region. The presented results demonstrate the robustness of the updated deformation scheme on complex geometries and show its potential on cases demanding large shape changes.

Acknowledgments

This research is partially funded by Bombardier, Transport Canada, the Natural Sciences and Engineering Research Council (NSERC), the University of Toronto and the Digital Research Alliance of Canada. All results in this paper were computed on the Niagara supercomputer at the SciNet HPC Consortium, a part of Compute Canada.

References

- [1] Reist, T. A., and Zingg, D. W., "Aerodynamic Design of Blended Wing-Body and Lifting-Fuselage Aircraft," *34th AIAA Applied Aerodynamics Conference, AIAA Paper 2016-3874*, 2016. <https://doi.org/10.2514/6.2016-3874>, URL <https://arc.aiaa.org/doi/abs/10.2514/6.2016-3874>.
- [2] Torenbeek, E., *Advanced Aircraft Design: Conceptual Design, Analysis and Optimization of Subsonic Civil Airplanes*, Aerospace Series, Wiley, 2013. URL <https://onlinelibrary.wiley.com/doi/book/10.1002/9781118568101>.
- [3] Gern, F. H., Ko, A., Sulaeman, E., Gundlach, J. F., Kapania, R. K., and Haftka, R. T., "Multidisciplinary Design Optimization of a Transonic Commercial Transport with Strut-Braced Wing," *Journal of Aircraft*, Vol. 38, No. 6, 2001, pp. 1006–1014. <https://doi.org/10.2514/2.2887>, URL <https://doi.org/10.2514/2.2887>.
- [4] Gagnon, H., and Zingg, D. W., "Euler-Equation-Based Drag Minimization of Unconventional Aircraft Configurations," *Journal of Aircraft*, Vol. 53, 2016, pp. 1361–1371. URL <https://api.semanticscholar.org/CorpusID:49318539>.
- [5] Chau, T., and Zingg, D. W., "Aerodynamic Optimization and Fuel Burn Evaluation of a Transonic Strut-Braced-Wing Single-Aisle Aircraft," *Journal of Aircraft*, Vol. 60, No. 5, 2023, pp. 1638–1658. <https://doi.org/10.2514/1.C037158>, URL <https://doi.org/10.2514/1.C037158>.
- [6] Gallman, J. W., Smith, S. C., and Kroo, I. M., "Optimization of Joined-Wing Aircraft," *Journal of Aircraft*, Vol. 30, No. 6, 1993, pp. 897–905. <https://doi.org/10.2514/3.46432>, URL <https://doi.org/10.2514/3.46432>.
- [7] Kroo, I., Gallman, J., and Smith, S., "Aerodynamic and Structural Studies of Joined-Wing Aircraft," *Journal of Aircraft*, Vol. 28, No. 1, 1991, pp. 74–81. <https://doi.org/10.2514/3.45994>, URL <https://doi.org/10.2514/3.45994>.
- [8] Gagnon, H., and Zingg, D. W., "Aerodynamic Optimization Trade Study of a Box-Wing Aircraft Configuration," *Journal of Aircraft*, Vol. 53, No. 4, 2016, pp. 971–981. <https://doi.org/10.2514/1.C033592>, URL <https://doi.org/10.2514/1.C033592>.
- [9] Liebeck, R. H., "Design of the Blended Wing Body Subsonic Transport," *Journal of Aircraft*, Vol. 41, No. 1, 2004, pp. 10–25. <https://doi.org/10.2514/1.9084>, URL <https://doi.org/10.2514/1.9084>.
- [10] Kuntawala, N., Hicken, J., and Zingg, D. W., "Preliminary Aerodynamic Shape Optimization of a Blended-Wing-Body Aircraft Configuration," *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, AIAA Paper 2011-642*, 2012. <https://doi.org/10.2514/6.2011-642>, URL <https://arc.aiaa.org/doi/abs/10.2514/6.2011-642>.
- [11] Gray, A. L., and Zingg, D. W., "Blended-Wing-Body Regional Aircraft Optimization with High-Fidelity Aerodynamics and Critical Design Requirements," *Journal of Aircraft*, Vol. 61, No. 6, 2024, pp. 1775–1792. <https://doi.org/10.2514/1.C037876>, URL <https://doi.org/10.2514/1.C037876>.
- [12] Truong, A. H., Oldfield, C. A., and Zingg, D. W., "Mesh Movement for a Discrete-Adjoint Newton–Krylov Algorithm for Aerodynamic Optimization," *AIAA Journal*, Vol. 46, No. 7, 2008, pp. 1695–1704. <https://doi.org/10.2514/1.33836>, URL <https://doi.org/10.2514/1.33836>.
- [13] Secco, N. R., Kenway, G. K. W., He, P., Mader, C., and Martins, J. R. R. A., "Efficient Mesh Generation and Deformation for Aerodynamic Shape Optimization," *AIAA Journal*, Vol. 59, No. 4, 2021, pp. 1151–1168. <https://doi.org/10.2514/1.J059491>, URL <https://doi.org/10.2514/1.J059491>.
- [14] Selim, M., and Koomullil, R., "Mesh Deformation Approaches – A Survey," *Journal of Physical Mathematics*, Vol. 7, No. 2, 2016, pp. 1–9. <https://doi.org/10.4172/2090-0902.1000181>, URL https://www.researchgate.net/publication/304998611_Mesh_Deformation_Approaches_-_A_Survey.
- [15] Liu, X., Qin, N., and Xia, H., "Fast Dynamic Grid Deformation Based on Delaunay Graph Mapping," *Journal of Computational Physics*, Vol. 211, No. 2, 2006, pp. 405–423. <https://doi.org/https://doi.org/10.1016/j.jcp.2005.05.025>.
- [16] Jakobsson, S., and Amoignon, O., "Mesh Deformation Using Radial Basis Functions for Gradient-Based Aerodynamic Shape Optimization," *Computers and Fluids*, Vol. 36, No. 6, 2007, pp. 1119–1136. <https://doi.org/https://doi.org/10.1016/j.compfluid.2006.11.002>, URL <https://www.sciencedirect.com/science/article/pii/S0045793006001320>.

- [17] de Boer, A., van der Schoot, M., and Bijl, H., “Mesh Deformation Based on Radial Basis Function Interpolation,” *Computers and Structures*, Vol. 85, No. 11, 2007, pp. 784–795. <https://doi.org/https://doi.org/10.1016/j.compstruc.2007.01.013>, URL <https://www.sciencedirect.com/science/article/pii/S0045794907000223>, fourth MIT Conference on Computational Fluid and Solid Mechanics.
- [18] Rendall, T., and Allen, C., “Reduced Surface Point Selection Options for Efficient Mesh Deformation Using Radial Basis Functions,” *Journal of Computational Physics*, Vol. 229, No. 8, 2010, pp. 2810–2820. <https://doi.org/https://doi.org/10.1016/j.jcp.2009.12.006>, URL <https://www.sciencedirect.com/science/article/pii/S0021999109006779>.
- [19] Wang, G., Mian, H. H., Ye, Z.-Y., and Lee, J.-D., “Improved Point Selection Method for Hybrid-Unstructured Mesh Deformation Using Radial Basis Functions,” *AIAA Journal*, Vol. 53, No. 4, 2015, pp. 1016–1025. <https://doi.org/10.2514/1.J053304>, URL <https://doi.org/10.2514/1.J053304>.
- [20] Poirier, V., and Nadarajah, S., “Efficient Reduced-Radial Basis Function-Based Mesh Deformation Within an Adjoint-Based Aerodynamic Optimization Framework,” *Journal of Aircraft*, Vol. 53, No. 6, 2016, pp. 1905–1921. <https://doi.org/10.2514/1.C033573>, URL <https://doi.org/10.2514/1.C033573>.
- [21] Luke, E., Collins, E., and Blades, E., “A Fast Mesh Deformation Method Using Explicit Interpolation,” *Journal of Computational Physics*, Vol. 231, No. 2, 2012, pp. 586–601. <https://doi.org/https://doi.org/10.1016/j.jcp.2011.09.021>, URL <https://www.sciencedirect.com/science/article/pii/S0021999111005535>.
- [22] Batina, J. T., “Unsteady Euler Airfoil Solutions Using Unstructured Dynamic Meshes,” *AIAA Journal*, Vol. 28, No. 8, 1990, pp. 1381–1388. <https://doi.org/10.2514/3.25229>, URL <https://doi.org/10.2514/3.25229>.
- [23] Farhat, C., Degand, C., Koobus, B., and Lesoinne, M., “Torsional Springs for Two-Dimensional Dynamic Unstructured Fluid Meshes,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 163, No. 1, 1998, pp. 231–245. [https://doi.org/https://doi.org/10.1016/S0045-7825\(98\)00016-4](https://doi.org/https://doi.org/10.1016/S0045-7825(98)00016-4), URL <https://www.sciencedirect.com/science/article/pii/S0045782598000164>.
- [24] Degand, C., and Farhat, C., “A Three-Dimensional Torsional Spring Analogy Method for Unstructured Dynamic Meshes,” *Computers and Structures*, Vol. 80, No. 3, 2002, pp. 305–316. [https://doi.org/https://doi.org/10.1016/S0045-7949\(02\)00002-0](https://doi.org/https://doi.org/10.1016/S0045-7949(02)00002-0), URL <https://www.sciencedirect.com/science/article/pii/S0045794902000020>.
- [25] Matsushima, K., Murayama, M., and Nakahashi, K., “Unstructured Dynamic Mesh for Large Movement and Deformation,” *40th AIAA Aerospace Sciences Meeting & Exhibit, AIAA Paper 2002-122*, 2002. <https://doi.org/10.2514/6.2002-122>, URL <https://arc.aiaa.org/doi/abs/10.2514/6.2002-122>.
- [26] Burg, C., “A Robust Unstructured Grid Movement Strategy Using Three-Dimensional Torsional Springs,” *34th AIAA Fluid Dynamics Conference and Exhibit, AIAA Paper 2004-2529*, 2004. <https://doi.org/10.2514/6.2004-2529>, URL <https://arc.aiaa.org/doi/abs/10.2514/6.2004-2529>.
- [27] Yang, Y., Özgen, S., and Kim, H., “Improvement in the Spring Analogy Mesh Deformation Method Through the Cell-Center Concept,” *Aerospace Science and Technology*, Vol. 115, 2021, p. 106832. <https://doi.org/https://doi.org/10.1016/j.ast.2021.106832>, URL <https://www.sciencedirect.com/science/article/pii/S1270963821003424>.
- [28] Jasak, H., and Tukovic, Z., “Automatic Mesh Motion for the Unstructured Finite Volume Method,” *Transactions of FAMENA*, Vol. 30, 2006, pp. 1–20.
- [29] Crumpton, P., and Giles, M., “Implicit Time Accurate Solutions on Unstructured Dynamic Grids,” *12th Computational Fluid Dynamics Conference, AIAA Paper 1995-1671*, 1995. <https://doi.org/10.2514/6.1995-1671>, URL <https://arc.aiaa.org/doi/abs/10.2514/6.1995-1671>.
- [30] Lynch, D. R., and O’Neill, K., “Elastic Grid Deformation for Moving Boundary Problems in Two Space Dimensions,” *Finite elements in water resources*, Vol. 2, 1980.
- [31] Tezduyar, T., Behr, M., Mittal, S., and Johnson, A., “Computation of Unsteady Incompressible Flows with the Stabilized Finite Element Methods: Space-Time Formulations, Iterative Strategies and Massively Parallel Implementations,” 1992, pp. 7–24.
- [32] Diosady, L. T., and Murman, S. M., “A Linear-Elasticity Solver for Higher-Order Space-Time Mesh Deformation,” *2018 AIAA Aerospace Sciences Meeting, AIAA Paper 2018-0919*, 2018. <https://doi.org/10.2514/6.2018-0919>, URL <https://arc.aiaa.org/doi/abs/10.2514/6.2018-0919>.
- [33] Marcon, J., Garai, A., Denison, M., and Murman, S. M., “An Adjoint Elasticity Solver for High-Order Mesh Deformation,” *AIAA Scitech 2021 Forum, AIAA Paper 2021-1238*, 2021. <https://doi.org/10.2514/6.2021-1238>, URL <https://arc.aiaa.org/doi/abs/10.2514/6.2021-1238>.

- [34] Zhou, J., Wu, X., Chen, B., Yu, J., Liu, S., and Jia, H., “Gradient-Based Optimization of Linear Elasticity Mesh Deformation Quality,” *International Journal of Aerospace Engineering*, Vol. 2025, No. 1, 2025, p. 4441909. <https://doi.org/https://doi.org/10.1155/ijae/4441909>, URL <https://onlinelibrary.wiley.com/doi/abs/10.1155/ijae/4441909>.
- [35] Stein, K., Tezduyar, T., and Benney, R., “Mesh Moving Techniques for Fluid-Structure Interactions With Large Displacements,” *Journal of Applied Mechanics*, Vol. 70, 2003, pp. 58–63. <https://doi.org/10.1115/1.1530635>.
- [36] Bar-Yoseph, Z., Mereu, S., Chippada, S., and Kalro, V., “Automatic Monitoring of Element Shape Quality in 2D and 3D Computational Mesh Dynamics,” *Computational Mechanics*, Vol. 27, 2001, pp. 378–395. <https://doi.org/10.1007/s004660100250>.
- [37] Nielsen, E. J., and Anderson, W. K., “Recent Improvements in Aerodynamic Design Optimization on Unstructured Meshes,” *AIAA Journal*, Vol. 40, No. 6, 2002, pp. 1155–1163. <https://doi.org/10.2514/2.1765>, URL <https://doi.org/10.2514/2.1765>.
- [38] Karman, S. L., Anderson, W. K., and Sahasrabudhe, M., “Mesh Generation Using Unstructured Computational Meshes and Elliptic Partial Differential Equation Smoothing,” *AIAA Journal*, Vol. 44, No. 6, 2006, pp. 1277–1286. <https://doi.org/10.2514/1.15929>, URL <https://doi.org/10.2514/1.15929>.
- [39] Brown, D. A., Nadarajah, S., Yang, H., Castonguay, P., Raiesi, H., Sermeus, K., and Germain, P., “Quality-Preserving Linear Elasticity Mesh Movement Algorithm for Multi-Element Unstructured Meshes,” *AIAA Journal*, Vol. 57, No. 2, 2019, pp. 521–531. <https://doi.org/10.2514/1.J057463>, URL <https://doi.org/10.2514/1.J057463>.
- [40] Tezduyar, T., Behr, M., Mittal, S., and Liou, J., “A New Strategy for Finite Element Computations Involving Moving Boundaries and Interfaces—The Deforming-Spatial-Domain/Space-Time Procedure: II. Computation of Free-Surface Flows, Two-Liquid Flows, and Flows with Drifting Cylinders,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 94, No. 3, 1992, pp. 353–371. [https://doi.org/https://doi.org/10.1016/0045-7825\(92\)90060-W](https://doi.org/https://doi.org/10.1016/0045-7825(92)90060-W), URL <https://www.sciencedirect.com/science/article/pii/004578259290060W>.
- [41] Reist, T. A., Koo, D., Zingg, D. W., Bochud, P., Castonguay, P., and Leblond, D., “Cross Validation of Aerodynamic Shape Optimization Methodologies for Aircraft Wing-Body Optimization,” *AIAA Journal*, Vol. 58, No. 6, 2020, pp. 2581–2595. <https://doi.org/10.2514/1.J059091>, URL <https://doi.org/10.2514/1.J059091>.
- [42] Gill, P. E., Murray, W., and Saunders, M. A., “SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization,” *SIAM Journal on Optimization*, Vol. 12, No. 4, 2002, pp. 979–1006. <https://doi.org/10.1137/S1052623499350013>, URL <https://doi.org/10.1137/S1052623499350013>.
- [43] 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. <https://doi.org/10.2514/1.44033>, URL <https://doi.org/10.2514/1.44033>.
- [44] Osusky, L., Buckley, H., Reist, T., and Zingg, D. W., “Drag Minimization Based on the Navier–Stokes Equations Using a Newton–Krylov Approach,” *AIAA Journal*, Vol. 53, No. 6, 2015, pp. 1555–1577. <https://doi.org/10.2514/1.J053457>, URL <https://doi.org/10.2514/1.J053457>.
- [45] Balay, S., Gropp, W. D., McInnes, L. C., and Smith, B. F., “Efficient Management of Parallelism in Object Oriented Numerical Software Libraries,” 1997, pp. 163–202.
- [46] Gagnon, H., and Zingg, D. W., “Two-Level Free-Form and Axial Deformation for Exploratory Aerodynamic Shape Optimization,” *AIAA Journal*, Vol. 53, No. 7, 2015, pp. 2015–2026. <https://doi.org/10.2514/1.J053575>, URL <https://doi.org/10.2514/1.J053575>.
- [47] Osusky, M., and Zingg, D. W., “Parallel Newton–Krylov—Schur Flow Solver for the Navier–Stokes Equations,” *AIAA Journal*, Vol. 51, No. 12, 2013, pp. 2833–2851. <https://doi.org/10.2514/1.J052487>, URL <https://doi.org/10.2514/1.J052487>.
- [48] 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. <https://doi.org/10.2514/1.34810>, URL <https://doi.org/10.2514/1.34810>.
- [49] Osusky, M., Boom, P. D., and Zingg, D. W., “Results from the Fifth AIAA Drag Prediction Workshop Obtained with a Parallel Newton–Krylov–Schur Flow Solver Discretized Using Summation-By-Parts Operators,” *31st AIAA Applied Aerodynamics Conference, AIAA Paper 2013-2511*, 2013. <https://doi.org/10.2514/6.2013-2511>, URL <https://arc.aiaa.org/doi/abs/10.2514/6.2013-2511>.
- [50] Jameson, A., “Aerodynamic Design via Control Theory,” *Journal of Scientific Computing*, Vol. 3, 1988. <https://doi.org/10.1007/BF01061285>.

- [51] Jameson, Antony, "Aerodynamic Shape Optimization Using the Adjoint Method," *Lectures at the Von Karman Institute, Brussels*, Vol. 6, 2003. URL <http://aero-comlab.stanford.edu/Papers/jameson.vki03.pdf>.
- [52] Squire, W., and Trapp, G., "Using Complex Variables to Estimate Derivatives of Real Functions," *SIAM Review*, Vol. 40, No. 1, 1998, pp. 110–112. <https://doi.org/10.1137/S003614459631241X>, URL <https://doi.org/10.1137/S003614459631241X>.
- [53] 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, 2010, pp. 1672–1694. <https://doi.org/10.1137/090754674>, URL <https://doi.org/10.1137/090754674>.
- [54] Castrillo, P., Canelas, A., Schillaci, E., Rigola, J., and Oliva, A., "High-Order Finite Volume Method for Linear Elasticity on Unstructured Meshes," *Computers and Structures*, Vol. 268, 2022, p. 106829. <https://doi.org/https://doi.org/10.1016/j.compstruc.2022.106829>, URL <https://www.sciencedirect.com/science/article/pii/S004579492200089X>.
- [55] Koo, D., and Zingg, D. W., "Investigation into Aerodynamic Shape Optimization of Planar and Nonplanar Wings," *AIAA Journal*, Vol. 56, 2017, pp. 1–14. <https://doi.org/10.2514/1.J055978>.
- [56] Biswas, K., Kumar, S., Banerjee, S., and Pandey, A. K., "SMU: Smooth Activation Function for Deep Networks Using Smoothing Maximum Technique," , 2022. URL <https://arxiv.org/abs/2111.04682>.
- [57] Yazdi, R. R., Gray, A. L., and Zingg, D. W., "Optimization of Twin-Aisle-Class Blended-Wing-Body Aircraft Incorporating Critical Requirements and High-Fidelity Aerodynamic Modelling," *AIAA Scitech 2026 Forum*, 2026.