Boundary Representation Tolerance Impacts on Mesh Generation and Adaptation

Michael A. Park*
NASA Langley Research Center, Hampton, VA 23681, USA

Robert Haimes†
Massachusetts Institute of Technology, Cambridge, MA 02139, USA

Nicholas J. Wyman‡ and Patrick A. Baker§
Cadence Design Systems, Fort Worth, TX 76104, USA

Adrien Loseille¶
INRIA Paris-Saclay, Alan Turing Building, 91120 Palaiseau, France

Todd Michal‖ and Joshua A. Krakos**
The Boeing Company, St. Louis, MO 63166, USA

The control of discretization error is critical to obtaining reliable simulation results. Recent progress has matured anisotropic mesh adaptation, which automates discretization error control for complex geometries. However, the meshing process can fail when geometric model Boundary REPresentation (BREP) tolerances are larger than boundary layer surface normal and tangential spacing requirements. Geometry sources are often created in Mechanical Computer-Aided Design (MCAD) systems, which are designed to produce models for manufacturing and not the stricter requirements of viscous flow analysis. BREP tolerances are strained by three factors: the inherent complexity of the model (e.g., large range of scales and complex topology), the numerical difficulties arising from surface/surface intersections, and the omission of crucial data during export and translation. Manual preparation of geometry is commonly employed to enable expert-guided mesh generation, which severely inhibits workflow automation. Accommodation of loose BREP tolerances is required for automated mesh processes, because barrier issues may not be detected until well into the solution process. To raise awareness of this class of geometry issues and their impact on simulation, a survey of these barrier issues is presented with example mitigation techniques. This awareness may also impact geometry creation workflows to prevent the introduction of these artifacts.

I. Introduction

The use of Reynolds-averaged Navier–Stokes (RANS) equations has become a critical tool for the analysis of aerospace vehicles. However, RANS-based Computational Fluid Dynamics (CFD) has not reached the level of automation desired by practitioners as identified in the CFD Vision 2030 Study by Slotnick et al. [1]. Unstructured mesh adaptation is an automation strategy for producing accurate solutions to address the Study's automation concerns. Alauzet and Loseille [2] documented the dramatic progress made in anisotropic solution-adaptive methods to resolve simulations with shocks and boundary layers. Park et al. [3] documented the state of solution-based anisotropic mesh adaptation and motivated further development to impact aerospace analysis and design processes. Significant progress has been made to provide adequate linkage to geometry, which is a bottleneck identified in the CFD Vision 2030

*Research Scientist, Computational AeroSciences Branch, AIAA Associate Fellow.
†Principal Research Engineer, Aerospace Computational Design Laboratory, Department of Aeronautics and Astronautics, AIAA Member.
‡Director, Applied Research, AIAA Associate Fellow.
§Director, Product Development, AIAA Senior Member.
¶Researcher, GAMMA3 Team, AIAA Member.
‖Technical Fellow, AIAA Senior Member.
**Engineer, AIAA Senior Member.

Study and these anisotropic mesh adaptation surveys. For example, Park et al. [4] describes the use of Mechanical Computer-Aided Design (MCAD) models for anisotropic mesh adaptation; Wyman at al. [5] provides a schema and library for the associativity of a surface mesh to the geometry model entities and their evaluation. The mesh adaptation community has matured to enable many common aerospace use cases [6], including AIAA workshop configurations [7–9]. This success has led to the wider application of mesh adaptation in production engineering environments. This routine application has identified a weakness related to an interaction of the mesh adaptation mechanics with the details of the geometry model.

The key concept in adequate linkage to geometry is the *interoperability* [10] of the geometry model with the CFD analysis process, because mesh adaptation is often a consumer of geometry created in an external system like a modern MCAD system. Geometry transfer between creation and analysis systems has historically used neutral file formats (e.g., Standard for the Exchange of Product Data (STEP) [11]). Unfortunately, market forces combine with the desire to vertically integrate proprietary MCAD systems with proprietary Computer Aided Engineering (CAE) analysis packages (e.g., CFD), which dissuades investment in the development of seamless translation capabilities. Vertical integration of MCAD and CAE limits options and is therefore unlikely to yield a best-in-class capability for any particular discipline. Translation can be avoided by direct queries to proprietary MCAD geometry-modeling kernels [12], but direct use is complicated by the proprietary nature of these systems. Furthermore, removing translation from the workflow does not eliminate all sources of geometric difficulty when dealing with RANS-based CFD.

CFD is a small appendix of a much larger ecosystem focused on the entire lifecycle of the product description anchored in MCAD. Taylor and Haimes [13] provide an overview of geometry-modeling concepts and requirements for computational simulation, which was motivated by the 1st AIAA Geometry and Mesh Generation Workshop (GMGW-1) [14]. Here we focus on Boundary REPresentation (BREP) geometry-modeling techniques. While alternative techniques exist, Taylor and Haimes share this focus on BREP because it underpins the geometry-modeling kernels embedded in all industrial MCAD platforms. BREP geometry is also a natural fit to describe a closed region for CFD meshing techniques. As show in the diagram on the right, a BREP is constructed of surface, curve, and point geometry entities organized into face, edge, and node topology. The elements of a discrete mesh are referred to as triangle, segment, and vertex.

	topology	geometry	discrete
·	node	point	vertex
–	edge	curve	segment
△	face	surface	triangle

Details of the BREP can have direct impacts on the reliability, robustness, and repeatability of rapid mesh generation processes as reviewed by Gammon, Bucklow, and Fairey [15] and Pointwise® [16]. The BREP is not closed at machine precision in all locations, where the inexactness of BREP stems from different parameterized curve and surface representations that meet arbitrarily in 3D. This inexactness is an artifact of the difficulty in generating exact intersection curves between general surfaces, where these surfaces are commonly represented with a Basis Spline (BSpline) or Nonuniform rational BSpline (NURBS) [17]. Surface-surface intersection is a common operation when performing constructive solid modeling. Typically for the BREP of a *solid* model, all edges are associated with three entities: the 3D curve and two parameter-space curves (or p-curves), one for each trimmed face. The p-curves can be traversed like the 3D curve providing the (u, v) location in the face associated with the p-curve. The distance between geometry entities underlying the topology defines the tolerance of intersections. For edges, tolerance is locally defined as the difference between evaluating on the 3D curve and face evaluation at the (u, v) provided by the p-curve.

Floating-point precision and loss of significance during geometric computations limits the practical dynamic range of computational geometry kernels. The practical dynamic range is typically 10^8 because the surface intersection curves are approximations of the actual intersection for reasons of computational efficiency, reducing double precision by 10^6 as shown in Section II examples. Dynamic range limits can significantly impact high Reynolds number applications, such as aircraft. Based on a unit wing chord, the geometry extents are nominally 100 and boundary layer mesh initial height is 10^{-6} (corresponding to 25M Reynolds number per wing chord and $y^+ = 1$ normal spacing). The ratio of domain extent and initial boundary mesh height gives rise to a dynamic range requirement of 10^8, which strains the geometry kernel as BREP tolerances should be below two orders of magnitude of the minimum resolved scale to avoid impacting the boundary layer mesh. These intersection curve approximations can be insufficient in certain cases (e.g., at the wing-body juncture) where it is desirable to construct an anisotropic surface mesh orthogonal to the intersection curve and viscous spacing in the volume.

The field of isogeometric analysis provides illustrative examples, implications, and potential remedies to interoperability and inexactness in a quest to directly integrate finite-element analysis into geometry representations [18–20]. Discrete mesh tools must be aware of BREP tolerance because this tolerance can vary by many orders of magnitude across the model or a single topological edge. The trim curves that underlay the edges are projected onto the face's

surface to form the borders of the faces and construct the p-curves. According to Gammon, Bucklow, and Fairey [15], "Because of their sensitive role in performance and stability, p-curve generation is a highly proprietary process and they are rarely exchanged between commercial CAD systems, nor exported to standard geometry exchange file formats such as STEP." Errors in the internal representation of the source MCAD model or created/exacerbated via translation are ubiquitous [21]. These common errors have spawned the necessity for repair of the data that composes the BREP model [22–28] or its discrete representation [29–33]. MCAD repair is a critical technology for addressing translation errors (or errors in the source representation), but a full review of BREP repair is beyond the scope of this paper. BREP repair is almost invariably an interactive process making it inappropriate for automation. Here we focus on models with valid (possibility overconstrained) topology, and edge-face closure that is larger than required for automated mesh adaptation. Ambiguity in the topology can emerge as the tolerances grow larger than the size of topological features. Educating the geometry creation pipeline to avoid the creation of these artifacts and loose BREP tolerances is an attractive alternative to repair.

The components of unstructured mesh adaptation are shown on Fig. 1. Starting with an initial mesh, a flow solution is computed. The information from the flow solution is used to estimate error and specify a new mesh resolution and orientation request via a metric field. The current mesh system is modified by mesh mechanics to adhere to the mesh

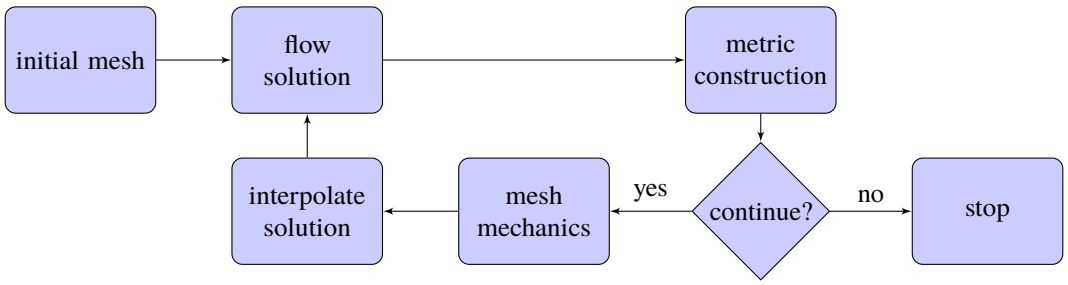


Fig. 1 Solution-based mesh adaptation process.

resolution request. Once the adapted mesh is available, the previous flow solution is interpolated to the new mesh to provide an initial condition for the flow solver that approximates the converged solution. The process is repeated until exit criteria are met (e.g., accuracy requirement, resource limit).

Michal, Krakos, and Kamenetskiy [34] compare impacts of BREP tolerance on expert-crafted and adaptive meshing in the context of GMGW-1. For expert-guided processes, the geometry is only used during the mesh generation phase, where geometry tolerance issues can be identified and corrected based on the mesh resolution choices the user controls. With adaptive meshing, geometry is queried throughout the solution process. As local mesh resolution approaches the resolution of model tolerance between geometry surfaces, large angles or folds can develop in the surface mesh. Figure 2 shows an example from Ref. [3]. Geometry tolerance requirements become coupled to the flow solution and mesh

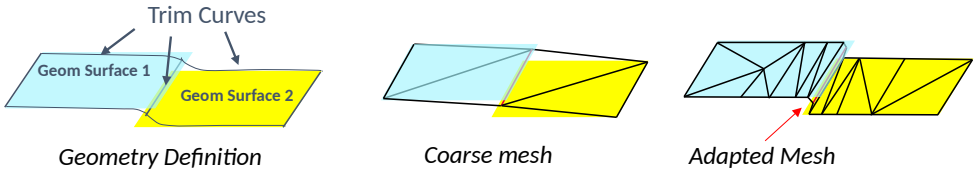


Fig. 2 Spurious adaptive refinement due to face geometry mismatch (from Ref. [3]).

resolution specified by the error estimate, both of which are constantly changing throughout the analysis. Localized BREP curvature, tangency errors, or face-edge gaps may become an attractor for error estimates resulting in pockets of excessive refinement. Assessing the adequacy of a geometry model before mesh adaptation becomes as challenging as predicting future mesh requirements. Problems with the geometry may not be discovered until the adapted mesh is sufficiently resolved to highlight the problem area. Techniques described in Section II can provide guidance for manual intervention early in the process, and these geometry assessment tools may play a critical supporting role until accommodations are fully implemented for mesh automation. Eccles, Steinbrenner, and Abelanet [35] provide a similar BREP tolerance visualization technique for edges. Geometry creation workflows should also be examined for strategies to prevent the introduction of these artifacts and compliment accommodation strategies.

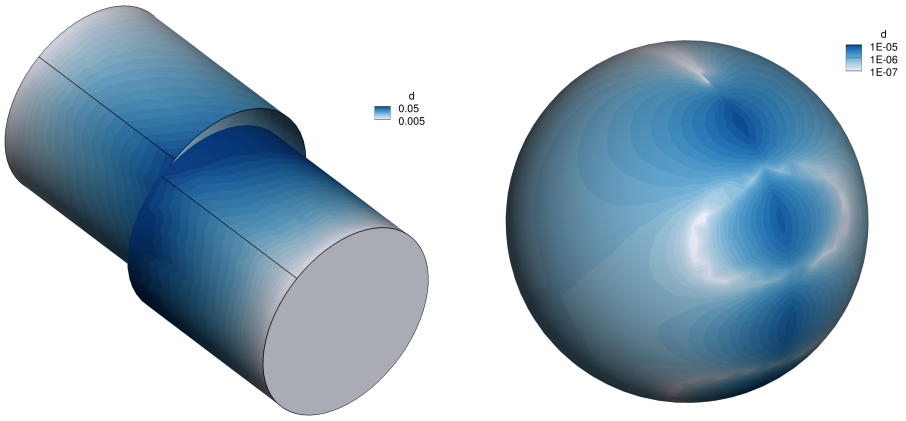
The mesh adaptation process must accept typical BREP tolerances and MCAD construction artifacts, because the tolerances in the geometry definition are rarely tight enough to satisfy the smallest mesh size that the adaptive mesh may request and these artifacts are ubiquitous. For example, models are examined in Section II to show the ubiquity of BREP tolerances larger than typical viscous CFD meshing requirements. When using mesh adaptation in a production environment, accommodating typical intended and unintended MCAD artifacts is critical to avoiding workflow stoppage and rework. Recent experience at Boeing has found that when working on full vehicle geometry prepared for a standard expert-guided process, over half of adaptive meshing applications eventually encounter MCAD artifact issues that need to be addressed. As a result, it is recommended that some form of mitigation or accommodation be considered when developing an adaptive meshing process for production use. Candidate mitigation and accommodation approaches include discrete surrogates [36–42] and virtual topology [29, 43–48], which often incorporates surrogates. In Section III, mitigation approaches are detailed and exercised on the examples from Section II.

II. Illustrative Examples

A series of simple to complex examples are provided to graphically illustrate the effect of BREP tolerances. They range from a simple, contrived example to models obtained from CFD prediction workshops. These models are publicly available to facilitate broad community testing and development of mitigation strategies for the issues described in the Introduction. These publicly-available models are representative of models with limited availability such as the NASA Space Launch System (SLS) [49] with large scales and fine details that are further complicated by including the model support hardware of a wind tunnel test [50].

A. Simple Examples

Two simple examples are provided in Fig. 3. The BREP tolerance of the union of two intentionally offset cylinders is shown in Fig. 3a to illustrate an extreme case of a gap opening in a BREP. The cylinders have a diameter of one, a length of one, and offset of one tenth. The colormap indicates the distance between face p-curves and edge curves smoothed in the faces using a pseudo-Laplacian operator. The edge (black) between the two cylinders is closer to the



(a) Union of two offset cylinders. **(b) Sphere meridian.**

Fig. 3 Smoothed BREP tolerance, measured in diameters, for two simple geometries.

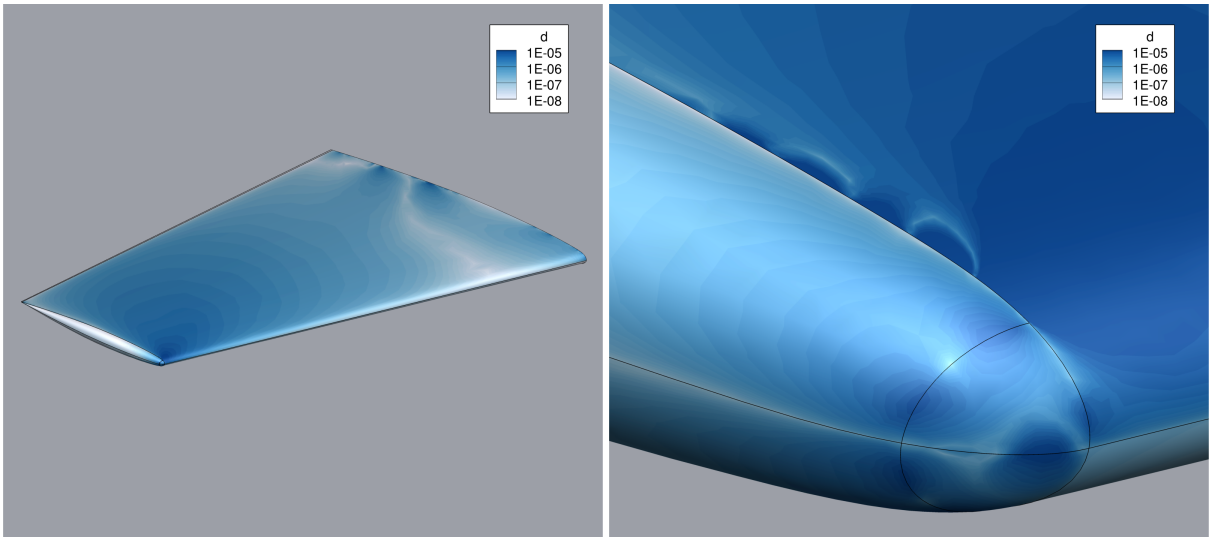
upper left cylinder, which results in discontinuous BREP tolerances across the gap resulting from the offset. The gap is a combination of slip and shear displacement that vary periodically around the circumference. The BREP tolerance of the upper and lower end caps (light) is tighter than the union with the offset (dark). This exaggerated case may seem unreasonable, but its intent is to show the perspective of a much smaller gap with a tight viscous spacing request.

The meridian, or seam, of a sphere is shown in Fig. 3b. The surface of the sphere wraps around and joins along a line with osculating BREP tolerance. The oscillation (i.e. alternating light and dark patterns along the meridian) is due to a different parameterization for the edge, the underlying surface, and the p-curve. The largest BREP tolerance is on the order of 10^{-5} diameters, which would become visible to mesh adaptation if an application requires elements of this size (e.g., for a viscous boundary layer). The surface is continuous on the back side of the sphere, which is indicated by

the low BREP tolerance extending to the rear from the pole (i.e. the white stripe).

B. ONERA M6

The ONERA M6 wing is originally described in an AGARD report [51]. The geometry is modified from this original description to have a sharp trailing edge and a well-defined wingtip shape by Mayeur et al. [52, 53]. The upper wing surface is shown in Fig. 4a. which appears to be a simple geometry model when observing the face and edge topology denoted with black lines. The colormap indicates the distance between face p-curves and edge curves



(a) Upper wing surface and symmetry plane. **(b) Detail of forward wingtip.**

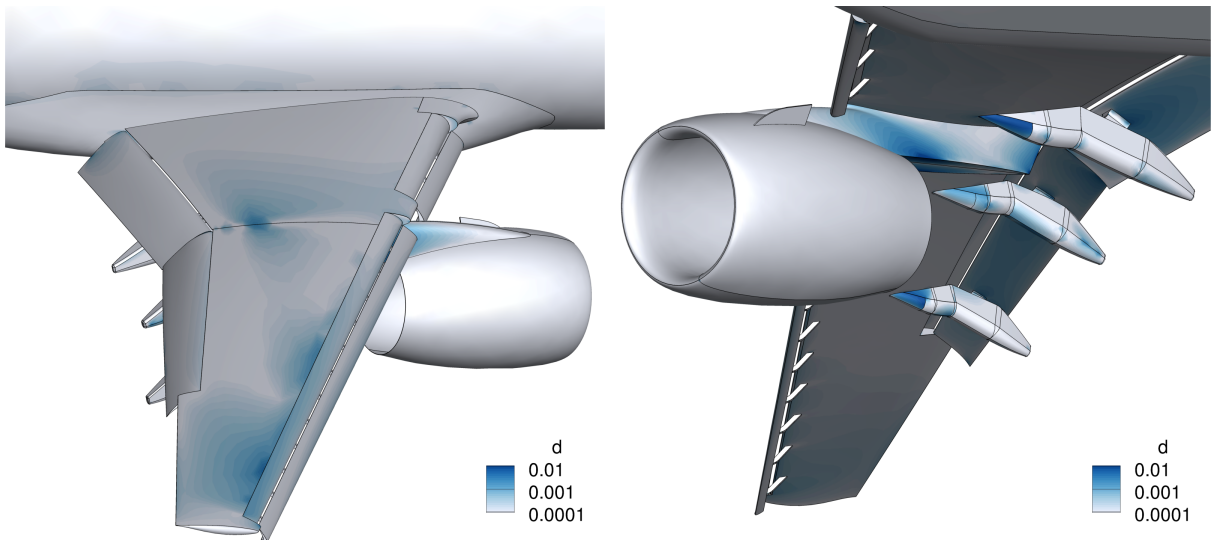
Fig. 4 ONERA M6 Wing topology and smoothed BREP tolerances measured in root chord lengths.

smoothed in the faces by using a pseudo-Laplacian operator. As seen in Fig. 4a, the intersection of the upper wing upper surface and the symmetry plane has multiple regions of large BREP tolerance. A detail of the forward wingtip is shown in Fig. 4b. Viscous spacing requirements for this model approach 10^{-6} root chord for adapted meshes with 10^7 vertices. The BREP tolerances near the symmetry plane and the three-sided faces on the wingtip leading edge exceed this tolerance where the dark regions meet the black edges. The periodic behavior of the BREP tolerance is evident where there are local areas of low tolerance where the edges and faces meet. The BREP tolerances can be different on each side of an edge (e.g., the edge joining the wing and the wingtip, the upper wing and the symmetry plane).

C. High Lift version of the Common Research Model (CRM-HL)

Smoothed BREP tolerances of the the high lift version of the Common Research Model (CRM-HL) are shown in Fig. 5a for the upper wing surface and Fig. 5b for the lower wing surface. The mean aerodynamic chord is 275.8 inches. The CRM-HL geometry had a circuitous route to the 4$^{\text{th}}$ High Lift Prediction Workshop (HLPW-4). Lacy and Sclafani [54] describe the initial development of the high lift system with lofts, not MCAD. Further details of geometry development are provided by Lacy and Clark [55], where a wind tunnel model is defined. Taylor and Gammon [56] examine the wind tunnel model MCAD used to create the workshop geometry. The CRM-HL model started as an extremely complex MCAD model to help define wind tunnel model fabrication. It has been simplified for CFD analysis, vetted, and improved by experts in the AIAA High Lift Prediction Workshop organizing committee. It is typical and superior to many examples of this level of topological complexity seen in practice. The initial wall spacing meshing guidelines[a] for HLPW-4 are 0.00239 inches for the coarsest mesh and 0.00035 inches for the finest mesh, which are exceeded by the darker color regions of Fig. 5. Adaptive mesh spacing requests are locally tighter than these mesh generation guidelines. The upper wing surface has hot spots of large BREP tolerance at the inboard flap end, between the main upper wing faces, and the Wing Under Slat Surface (WUSS). The lower wing surface has large BREP tolerance at the forward flap brackets and nacelle pylon.

[a] `https://hiliftpw.larc.nasa.gov/Workshop4/Mesh-Generation-Guidelines-V11-3-Final.pdf` [Accessed 21-JAN-2021].

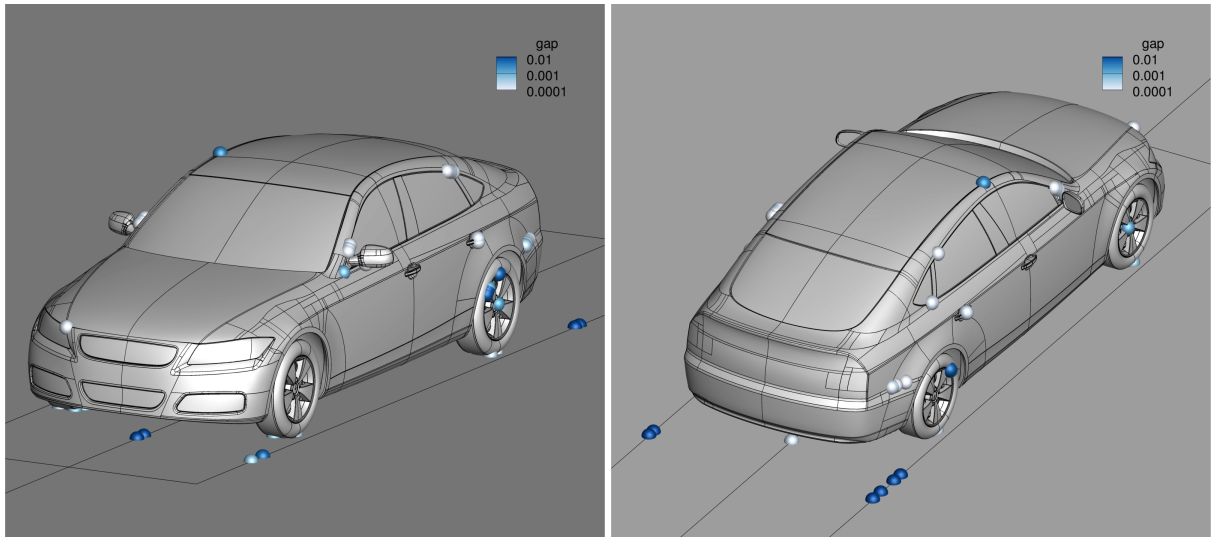


(a) Upper wing surface. **(b) Lower wing surface.**

Fig. 5 CRM-HL smoothed BREP tolerances (inches).

D. DrivAer Smooth Underbody Fastback

The DrivAer model is typical of a medium-sized car [57–59]. The smooth underbody version of the fastback model is shown as an example of geometry intended for CFD simulation. The topology of the model is shown in Fig. 6 with spheres marking locations where the BREP tolerance is larger than 0.1 mm. The meshes provided of this model for the 1st Automotive CFD Prediction Workshop have an initial cell height of 0.018 mm[b], or one-fifth of this minimum threshold. The ground plane and thin spacers connecting to the tires are included in the model, which have the largest BREP tolerances of over a centimeter.



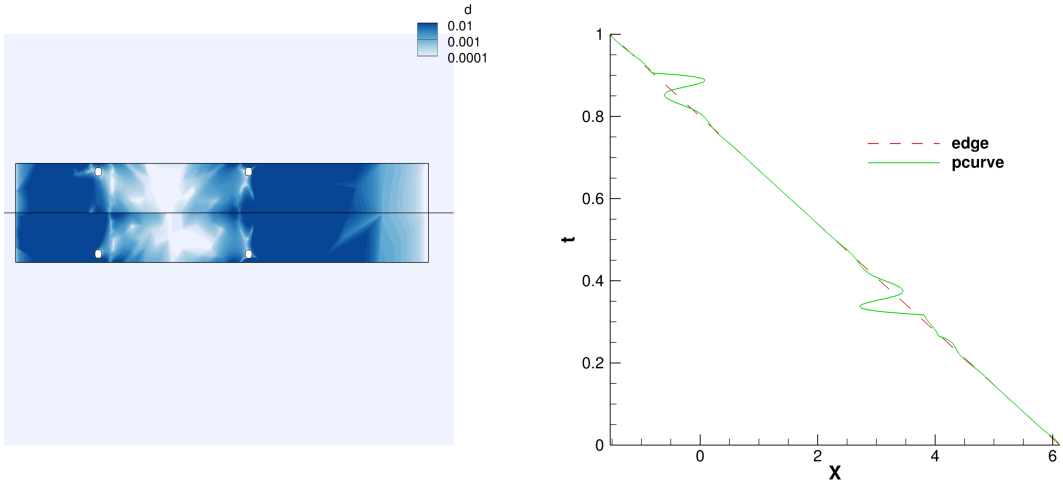
(a) Front view. **(b) Rear view.**

Fig. 6 DrivAer with spheres marking large BREP tolerances (meters).

While the ground plane surface is flat and edges are linear, the ground plane surface has a discontinuous parameterization. As discussed in the Introduction, the p-curves of the model must be reconstructed during geometry

[b] `http://autocfd-transfer.eng.ox.ac.uk/Case2a/Case2-descriptionv1p9.pdf` [Accessed 07-MAY-2021].

import. In this example, Open CASCADE is used to import the STEP model. The p-curve evaluated on the surface is shown with the edge curve in Fig. 7b, where t is the parameter and X is the position along the ground plane. The gap is pure slip displacement due to overshoots in the p-curve as it fits the discontinuous surface parameterization. The parameterization of the edge is linear. This mismatch could be repaired by aligning the p-curve knots with the surface knots or replacing the discontinuously-parameterized surface with a linearly-parameterized surface and refitting the p-curve. However, this scenario can also happen in cases with more complex surfaces or typologies that would make this simple class of fixes impossible to execute.



(a) Floor. **(b) Centerline p-curve.**

Fig. 7 DrivAer floor smoothed BREP tolerances and p-curve (meters).

III. Mitigations

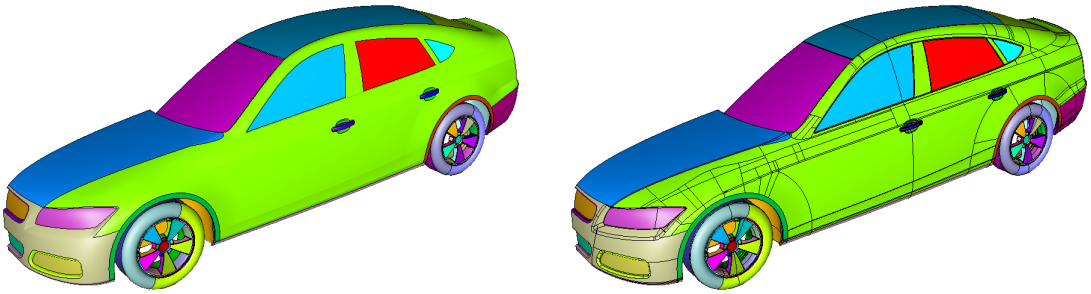
Multiple implementations of mitigation techniques are described. These include variations of surrogate and virtual topology. The CRM-HL and DrivAer models are used in these descriptions.

A. Pointwise®

The Pointwise® meshing software supports manifold and nonmanifold boundary representation solid models. The software uses quilts (virtual topology) for organization and control of the surface mesh at the conceptual component level. Assembly of trimmed surfaces from parametric space bounding curves and of solid models from enclosing trimmed surfaces requires allowance for finite precision during entity construction. The Pointwise® software performs a number of foreign entity processing functions to ensure consistency and validity of the BREP model. For example, the curves defining a closed loop for trimming a surface must be oriented properly and joined within suitable tolerance at shared vertices. Model consistency, even in the presence of large BREP tolerances, is an absolute prerequisite to mesh automation.

The use of virtual topology avoids issues with ill-shaped and small faces with respect to mesh scale surfaces (e.g., slivers) and enables automation of meshes suitable for engineering analysis using complex models such as the DrivAer sedan model from the Technical University of Munich, see Fig. 8. However, local varying tolerances are required to topologically close the model, which reduces the robustness of mesh generation with smaller elements than these local tolerances.

Automated mesh topology construction involves agglomeration of quilt shared boundary curves into chains of curves based on turning angle. The quilt boundary, a 1D analog to the quilt concept, constrains the location of mesh edges along the boundary and avoids “sliver edges” similar to avoidance of sliver surfaces. The quilt boundary is a collection of model space curves that are an approximation of the parametric space curves used in the definition of trimmed surfaces contained in the quilt or quilts sharing the boundary. The model space curve approximation is bounded by the



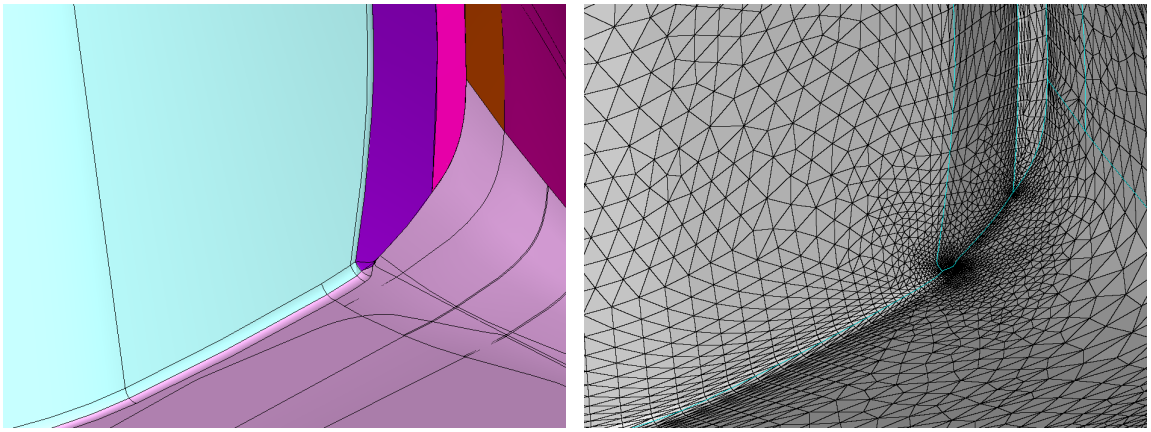
(a) Colored to show quilt topology. **(b) Quilt component surface boundaries.**

Fig. 8 DrivAer BREP.

local BREP tolerance. Furthermore, note that there are tolerance issues with the joining of model space curves within the quilt boundary as trimming curves are only approximately coincident at shared vertices.

For most of the model, the local model tolerances are significantly smaller than the local mesh edge length allowing meshing algorithms to remain robust as long as handling for inequalities between model space and parametric space is included. That is to say, a mesh point on a shared quilt boundary will have parametric coordinates corresponding to the shared curves and surfaces in the model at that location, but no forward evaluation of those coordinates will exactly match the model space coordinates. Despite this disparity, maintaining the parametric linkage to the model components is essential so the meshing algorithm can query local model properties (e.g., surface curvature).

Meshing robustness issues arise when the local BREP tolerance is of the same order of magnitude (or larger) as the local mesh edge length. If the model geometry is roughly planar, ad hoc solutions such as locally reverting to meshing in model space (3D meshing) are sufficient to generate a mesh meeting the analyst's intent. Model-space meshing typically happens for a small percentage of points when meshing a complex real-world model, for example the rear wheel well of the DrivAer model, see Fig. 9. Pointwise® is able to automatically detect local conditions that require model-space



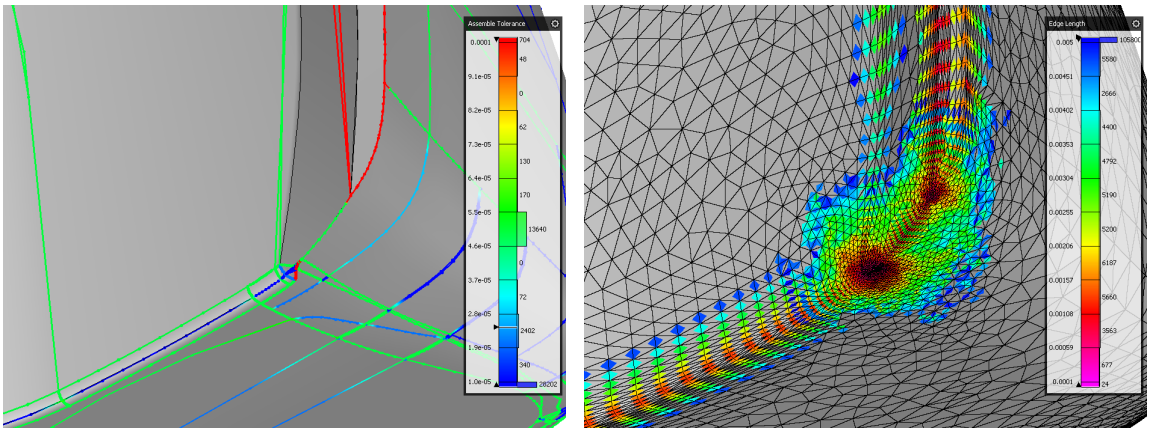
(a) Colored quilt topology on high curvature surfaces with lines denoting original topology. **(b) Anisotropic mesh topology matching quilt topology to resolve curvature and sharp edges.**

Fig. 9 DrivAer rear wheel well.

meshing and preserve robustness (validity) of the mesh through a heuristic assessment of target mesh edge length, local BREP tolerance, and other measures such as surface normal. These detection methods are also required to smoothly and robustly mesh the quilt boundary curve.

The BREP tolerance can be larger than the local mesh spacing at the quilt boundary. A quilt boundary designated

for anisotropic semistructured meshing to more efficiently resolve a geometry feature orthogonal to the boundary such as curvature reduces the mesh sizing request to the mesher, see Fig. 10. In this case, Pointwise® will first construct a coarse isotropic "scaffold" mesh by using a mixture of parametric and 3D meshing as described above. Resolution of the scaffold mesh is adaptive based on factors such as geometry curvature and BREP tolerance. The anisotropic mesh can then be produced using the scaffold mesh for support wherever BREP tolerance prevents robust inverse geometry evaluation (point projection). Throughout the remedial action, mesh association to the parametric geometry is maintained.



(a) Colored local BREP tolerance. **(b) Mesh edge lengths on the order of BREP tolerances.**

Fig. 10 DrivAer rear wheel-well length scales.

The presented methodology has proven effective at meshing complex geometries with BREP tolerances near or above mesh size requests. However, the ad hoc nature of the solution prevents a rigorous assessment of the systematic impact of local deviations of the discrete geometry representation from the analytic B-rep model, particularly when analyst intent is unknown. That is to say, deviation from the analytic model, while normally viewed negatively, may be desirable if mesh smoothness is improved.

B. `EGADS` Effective Topology

Aerospace vehicle design can be described as an evolutionary process of gathering information to make informed decisions. Meticulous application of this process involves numerous simulations covering many disciplines and fidelity levels. A design team needs to be able to easily increase or decrease fidelity as they gather more information and learn about a particular design (and potentially change the design parameterization to better capture the mission). To this end a fully-parametric geometry system that can support multidisciplinary, multifidelity analysis from a single source is required (an Analysis CAD system). The Computational Aircraft Prototype Syntheses [60–62] (CAPS) system, which is a part of the Engineering Sketch Pad [63] (ESP), satisfies the above by combining proven computational geometry, meshing, and analyses model generation techniques into a complete browser-based, client-server environment that is accessible to the entire design team of an aerospace vehicle. ESP/CAPS is accessed from a browser-based user interface, a Python interface and/or Multidisciplinary Analysis and Optimization (MDAO) framework through an Application Programming Interface (API).

The ESP software environment is open source and freely available at `http://acdl.mit.edu/ESP`. ESP is a collection of purpose built APIs orchestrated through the suite or used inside of other applications. OpenCSM [64] is the parametric engine, where `EGADS` [65] is another of these components. `EGADS` is the foundational object-based geometry kernel that supports geometry construction (from either *top-down* or *bottom-up* perspective) and provides a suite of functions that fully support surface meshing. These meshing functions include topological traversal, evaluations (with 1st & 2nd derivatives), inverse evaluations, containment predicates, p-curve handling, computation of arc-length, curvature, and more. *Interoperability* is a design feature where the application of metadata in the form of attributes that can be assigned to objects (and gets tracked through construction operations). `EGADS` is differentiated to support sensitivity analysis in parametric settings. The API is used within `EGADS` to build both triangle and full quadrilateral watertight discrete representations of the BREP (`EGADS` *tessellation objects*). There is also a variant (`EGADSlite`) [66]

designed specifically for use within HPC environments.

1. Virtual Topology

The concept of Virtual Topology [43] has been around for decades. The topology found in the BREP rarely provides the segregation of geometry entities as viewed from the "engineering" perspective and adjusting BREP topology can facilitate analysis of the BREP model [48]. BREP topology is a function of the model construction, and construction operations may result in unnecessary nodes and edges, and *sliver* faces as well as small edges. Because many meshing techniques generate grids that follow the BREP, these artifacts are presented to the solver resulting in step-size stability limitations. In general, Virtual Topology provides the ability to split and merge BREP faces and edges to better approximate the "engineering" intent. For example, see Fig. 8.

2. Effective Topology

The purpose of adding a variation on Virtual Topology to `EGADS` is to support meshing across BREP faces by coalescing them into what is commonly referred to as a "composite" or "quilt". This is useful so that sliver faces (and the small elements filling these faces) are not exposed in the mesh. If one is attempting to *drape* 4-sided patches over geometry, composites/quilts (along with scribing in `EGADS`) allow for better control of the bounding surface shapes and therefore, can provide an alternative "engineering" topology of the geometric model free of construction artifacts.

`EGADS` does not create new geometry for the collection of entities (i.e. it is not a "reskinning" of the group of faces). Instead, a mapping is used to generate a global (u, v) for the requisite collection of BREP faces as described by Marcum and Gaither [67]. The mapping comes from an input triangulation in the form of an `EGADS` Tessellation Object. The Tessellation Object (when generated internally by `EGADS`), is watertight and does not insert vertices in regions where geometric closure will be an issue. The issues as depicted in Fig. 2 will not be seen in the resulting triangulation. The mapping is generated by applying a simple biharmonic scheme applied to the triangulation coupled with an orthogonal boundary condition.

The list of faces that make up a composite/quilt must be touching so that their tessellations share at least a single triangle side. The same holds for collections of edges ("chains"), where there is a simple mapping (an extended t range that is the aggregate of the edge ranges), but not new geometry. The mapping is one way (not bijective), as with the composites/quilts. Chains are collected automatically during the construction of composites and left in a minimal state (unless marked by attribution to be *kept*). Any BREP node that only touches two BREP edges in a *solid* will be removed unless it is at a degenerate location or also marked.

In order for this to work *effectively* (hence the term) with BREP tolerances/closure issues, evaluations on *Effective* objects have special treatment so that no gaps and/or overlaps are seen for any internalized entities (BREP edges and nodes that are paved over). In addition, the exposed bounding entities are moved closer together reducing the mismatch. This scheme supports most surface meshing algorithms because there is a global (u, v) for the *Effective Face*, which can be treated the same as an actual BREP face. The collections of BREP faces that make up an *Effective Face* can be driven by attribution found on the topological entities, which allows for a degree of automation.

3. Closure

Properly dealing with geometry closure is important when paving over internal edges (those no longer expressed) so that the *jumps* in displacement are not seen, and regardless of the vertex placement algorithm used, a smooth result is obtained. This is all facilitated by using an input tessellation object to initialize the *Effective Topology*. The triangulation is marked so that the deviations between the BREP edge evaluation and that of the face are reconciled and a local linear ramp of the deviation is applied over the triangles touching BREP edges. This ramp adjusts the *Effective* topology evaluations and inverse evaluations. A similar procedure is used on these edges as they approach BREP nodes.

The linear ramps that are used to provide a closed BREP are best understood by looking at a BREP edge in the original geometric representation. The edge usually has 2 BREP nodes that trim (or limit the extent of) the underlying curve. There is a discontinuity between evaluating the curve at the t representing the node and the node's location, which specifies the local deviation. The discretization of the edge is used, where the first vertex away from the node and the node itself represent the ramp. The deviation is linearly applied starting at the inner vertex and terminating at the node with the full adjustment, providing closure. This is unlike what is displayed in Figs. 3, 4 and 5 where the deviation has been smoothed over a broad region; here the deviations are local, where the locality is determined by the initiating `EGADS` tessellation.

The result is that C^0 (value is continuous) is achieved and the model is closed at the expense of an adjustment in C^1 (derivative is continuous) at the segments touching nodes due to the linear ramp. Therefore, *Effective* Topology avoids the situations seen in Fig. 2 (for internalized geometry) and can be freely used in adaptive meshing where the algorithm no longer needs to avoid vertex placement in these locations due to tolerance issues. This closure scheme is applied to all bounding entities (not just the paved-over edges/nodes). If the exposed BREP edge is linear, then closure at machine precision is also achieved, otherwise the gap is reduced but cannot be eliminated because the triangle side is only an approximation of the curve locally and the closure adjustments at endpoints does not guarantee closure along the segment.

C. FEFLO.A Surrogates

FEFLO.A is a 2D, 3D, and surface mesh adaptation tool. It uses a combination of generalized standard operators (e.g., insertion, collapse, element swap). The generalized operators are based on recasting the standard operators in a cavity framework [68, 69]. The cavity operator allows a simultaneous application of multiple standard operator combinations. Quality improvements are attained with the cavity operator that are not possible through a sequential application of standard operators. FEFLO.A can support virtual topology indirectly through a discrete surface surrogate.

Allowing the discrete grid topology to be different than the BREP topology has the possibility of alleviating the challenges of problematic BREP features discussed in the Introduction. There are also situations where a geometry model may not be available (e.g., adaptation based on an existing mesh without underlying geometry support). The input mesh may have been constructed with an external tool that made BREP topology modifications internally without providing a consistent geometry model that matches the surface mesh. For instance, quilting is a common approach to reduce the number of BREP patches where the link to the initial geometry is often lost or difficult to recover from the generated mesh.

For these situations, FEFLO.A generates a high-order (cubic) discrete surface grid as a surrogate geometry model. This mesh is either derived from the BREP directly or from a linear discrete mesh. The choice of (at least) a cubic mesh is motivated by the constraint of ensuring C^1 continuity at vertices. However, using rational or higher-order interpolation are possible. When a geometry is provided, FEFLO.A relies on EGADS and EGADSlite to perform standard BREP forward evaluations, inverse evaluations, and topology queries.

1. Anisotropic mesh generation

From a conceptual point of view, meshing a parametric surface consists of meshing a 2D domain in the parametric space, where a robust surface mesh generation is not as naive as it seems. Additional difficulties need to be taken into account beyond the issues in the Introduction:

- The mapping function is not bijective, i.e., an infinite number of parameter values may have the same value in $\mathbb{R}^3$;
- A valid mesh in the parametric space may be invalid when mapped to 3D as σ (parameterization function) is not necessarily monotone (e.g., Fig. 7b);
- Having a uniform mesh in $\mathbb{R}^3$ creates a highly anisotropic adapted mesh in parametric $\mathbb{R}^2$ due to the length distortion imposed by σ;
- The typical BREP queries (normal, tangent planes, principal curvatures) are based on the derivatives of σ that may have undefined behaviors especially near the boundaries of the parametric space.

To accurately capture the features of the geometry, an anisotropic framework is mandatory. We quickly recall here some metric-based error estimates to control the surface approximation. These metrics are then used to drive the adaptation process described in the next paragraph. Based on the Frénet frame, and on the curvature, a 3D metric tensor can be deduced via:

$$\mathcal{M}_1 = \left({}^t\mathbf{T}\,{}^t\mathbf{N}\,{}^t\mathbf{B}\right)\begin{pmatrix} \frac{1}{(2\sqrt{\epsilon(2-\epsilon)}\rho(t))^2} & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{pmatrix}\begin{pmatrix} \mathbf{T} \\ \mathbf{N} \\ \mathbf{B} \end{pmatrix}$$

where $\lambda \in \mathbb{R}$ is an arbitrary constant[c], $\rho(t) = \frac{1}{\kappa(t)}$ is the radius of curvature, and $2\sqrt{\epsilon(2-\epsilon)}$ is a scaling coefficient that guarantees, for a second-order approximation of the curve, to maintain a deviation gap between the mesh elements and the curve geometry of ϵ [70]. As the metric relies on only intrinsic data, it is independent of the parameterization. The

[c]It sets the size in the normal plane to the curve.

metric can be mapped back to the parameter space via the following formula:

$$\tilde{\mathcal{M}}_1 = {}^t\gamma'(t)\mathcal{M}_1\gamma'(t).$$

In this case, the formula simplifies to $\tilde{\mathcal{M}}_1 = \frac{1}{h_1^2} = \frac{||\gamma'(t)||^2}{(2\sqrt{\epsilon(2-\epsilon)}\rho(t))^2}$. Once the metrics for curves are set, the next step is to define metrics for the surfaces. The meshing process of parametric surfaces is a bit more complex. It relies on some differential geometry notions [71]. For this purpose, let us consider a parametric surface $(u, v) \to \sigma(u, v) \in \mathbb{R}^3$ that we will assume smooth enough. In this case, the *first fundamental form* $I(du, dv)$ is defined as follows:

$$I(du, dv) = (du\ dv)\begin{pmatrix} ||\sigma_u||^2 & (\sigma_u, \sigma_v) \\ (\sigma_u, \sigma_v) & ||\sigma_v||^2 \end{pmatrix}\begin{pmatrix} du \\ dv \end{pmatrix},$$

where (du, dv) is an elementary displacement, σ_u is the partial derivative of σ with respect to u, and σ_v is the partial derivative of σ with respect to v.

The first fundamental form explains how the 3D distances are perceived in the two-dimensional space. In particular, it provides a 2D Riemannian structure to the surface with a metric tensor defined as:

$$\mathcal{M}_I = \begin{pmatrix} ||\sigma_u||^2 & (\sigma_u, \sigma_v) \\ (\sigma_u, \sigma_v) & ||\sigma_v||^2 \end{pmatrix}.$$

In the same framework, the *second fundamental form* $II(du, dv)$ is defined as follows:

$$II(du, dv) = (du\ dv)\begin{pmatrix} (\sigma_{uu}, \mathbf{N}) & (\sigma_{uv}, \mathbf{N}) \\ (\sigma_{uv}, \mathbf{N}) & (\sigma_{vv}, \mathbf{N}) \end{pmatrix}\begin{pmatrix} du \\ dv \end{pmatrix},$$

where $(\sigma_{uu}, \sigma_{uv}, \sigma_{vv})$ are the second derivatives of σ with respect to (u, v) and $\mathbf{N} = \frac{\sigma_u \times \sigma_v}{||\sigma_u \times \sigma_v||}$ is the normal vector to the surface. The second fundamental form expresses a second-order estimate of the gap between a surface to its tangent plane.

Based on these two quadratic forms and their matrices, we are able, for a given point of the surface, to define the principal curvatures $(\kappa_i)_{i=1,2}$ and principal directions $(\mathbf{V}_i)_{i=1,2}$ (in 3D) as solution of the generalized eigenvalue problem:

$$\begin{cases} \mathcal{M}_{II}\mathbf{v}_i = \kappa_i \mathcal{M}_I \mathbf{v}_i \\ \qquad \mathbf{V}_i = \dfrac{(\sigma_u\ \sigma_v)\,\mathbf{v}_i}{||\,(\sigma_u\ \sigma_v)\,\mathbf{v}_i||} \end{cases} i = 1, 2$$

with $\mathcal{M}_{II}$ being the symmetric matrix associated to the second fundamental form. These quantities are independent of the parameterization, and when $\kappa_1 \neq \kappa_2$, $(\mathbf{V}_1, \mathbf{V}_2)$, forms an orthonormal basis of the tangent plane. If we complete the basis with $\mathbf{N}$, they form a local basis $(\mathbf{V}_1, \mathbf{V}_2, \mathbf{N})$ of $\mathbb{R}^3$. Now, thanks to this basis and on the curvatures, we can define the following 3D metric tensor:

$$\mathcal{M}_2 = \left({}^t\mathbf{V}_1\,{}^t\mathbf{V}_2\,{}^t\mathbf{N}\right) \times \begin{pmatrix} \frac{1}{(c_1\rho_1(u,v))^2} & 0 & 0 \\ 0 & \frac{1}{(c_2\rho_2(u,v))^2} & 0 \\ 0 & 0 & \lambda \end{pmatrix}\begin{pmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \mathbf{N} \end{pmatrix},$$

where $\lambda \in \mathbb{R}$ is an arbitrary constant, $\rho_i(u, v) = \frac{1}{\kappa_i(u,v)}$ for $i = 1, 2$ are the radii of curvature, with the convention $\rho_1(u, v) \leq \rho_2(u, v)$ and $c_1$and c_2 are scaling coefficients. For the direction of greater curvature (e.g., the direction given by $\mathbf{V}_1$), we want to control the deviation under a threshold ϵ, where c_1 is set to the value of $2\sqrt{\epsilon(2-\epsilon)}$. Now, as we want the same threshold in all the directions in the tangent plane, the coefficient c_2 is set to $2\sqrt{\epsilon\frac{\rho_1}{\rho_2}(2-\epsilon\frac{\rho_1}{\rho_2})}$ [72]. Similarly to curves, the metric relies only on intrinsic data and is therefore independent of the parameterization. Now, the metric $\mathcal{M}_2$ can be mapped back to the parameter space by applying the first fundamental form:

$$\tilde{\mathcal{M}}_2 = \left({}^t\sigma_u\ {}^t\sigma_v\right)\mathcal{M}_2\begin{pmatrix} \sigma_u \\ \sigma_v \end{pmatrix}.$$

This is the metric that is used as an anisotropic metric for the mesh adaptation in the parameter space. Recently, an extension of the previous metrics for high-order interpolation has been derived [39].

2. P_3 Surrogate Mesh Generation from BREP

When a geometry is provided, the surrogate P_3 mesh is generated from the BREP directly by using a standard classical parametric-surface mesh generation process. The process implies the following steps and with the previous metrics:

1. For each edge:
   1.1 Generate a 3D adaptive mesh using $\mathcal{M}_1$.
2. For each face:
   2.1 Generate a fast (u, v)-aligned tessellation,
   2.2 Compute high-order metric $\tilde{\mathcal{M}}_2$ on the tessellation,
   2.3 Project 3D edges of loops as parametric curves and generate a 2D (u, v) mesh forming the boundary of the patch,
   2.4 Check for closure and self-intersection,
   2.5 Insert points from the (u, v)-aligned tessellation onto the current mesh,
   2.6 Convert $\tilde{\mathcal{M}}_2$ to $\mathcal{M}_2$ (transform to 3D),
   2.7 Generate high-order mesh in 3D.

Once the face mesh is transformed to 3D (in step 2.6), only 3D operations are used (e.g., point insertion, smoothing), where those operations are similar to a classical mesh adaptation procedures.

3. P_3 Surrogate Mesh Generation from a Linear Mesh

When a linear mesh is provided, the P_3 surrogate mesh (i.e. the cubic reconstruction [41]) relies on the Bézier representation of a curve and surface. A preprocessing step first consists in estimating lines or points of (normal) discontinuity, ridges, directly from the discrete mesh. On ridges, a high-order cubic line is recovered. On corners (normal discontinuity), different normals are used to compute the curved triangles. High-order cubic triangles are recovered on surface interiors from the discrete normals. For triangles sharing a ridge edge, the high-order cubic line becomes a constraint to the curving algorithm. Let us give a normal at the each of the vertices A and B; the two Bézier control points P and Q are determined by:

- P as the orthogonal projection of the point X such that $X = \frac{1}{3}A + \frac{2}{3}B$ on the tangent vector/plane associated with the first point.
- Q as the orthogonal projection of the point Y such that $Y = \frac{2}{3}A + \frac{1}{3}B$ on the tangent vector/plane associated with the second point.

The normal at a vertex is computed as the weighted sum of the normals of all the boundary elements containing this vertex. Note that in the case of corners and ridges a special procedure is required. This way a P_3 curve is obtained. For the computation of the inner Bézier control points of the P_3 triangle, a *Serendipity* model [73] is used. The central Bézier control point is computed as a weighted sum of the vertex and the edge control points of the triangle ABC:

$$P^{\text{central}} = -\frac{1}{6}(A + B + C) + \frac{1}{4}\sum_{i=1}^{3}(P_1^{\text{edge}_i} + P_2^{\text{edge}_i}).$$

For low regularity discrete meshes, the gap across edges is controlled and the curvature is automatically reduced to remove artifacts.

4. Using the P_3 Surrogate within an Adaptive Process

During an adaptation process, FEFLO.A uses a reference background surface mesh. This surface mesh is always a cubic mesh based in the geometry or a discrete mesh. The mesh is used to hold the associativity to the original mesh if available. Recovering the BREP-mesh associativity is based on a fast inverse projection process. For each entity on edge curves or face surfaces, given a point P, a nonlinear search is performed to find $\sigma(u, v)$ minimizing $\|P - \sigma(u, v)\|$, where σ is the parametric function. This problem consists of finding roots of the following (f, g) functions:

$$\begin{cases} g(u, v) = (\sigma(u, v) - P) \cdot \sigma_u(u, v) = 0 \\ f(u, v) = (\sigma(u, v) - P) \cdot \sigma_v(u, v) = 0. \end{cases}$$

As for any nonlinear problem, the initial guess is critical to ensure validity and fast convergence. Octree and neighboring vertex associations are used for robust initial guesses. The result of this optimization is a match between each surface

vertex and the underlying geometry model. The association recovery process can be applied to any initial surface mesh, where most of the topology, gap, and tolerance issues have been resolved. Initializing the mesh adaptation process with an association to a valid surface mesh implicitly recovers these geometry repairs with the association. To increase evaluation robustness, a hierarchical approach is used to project newly created points to the surface. The point is first projected onto the surrogate P_3 mesh before projection onto the BREP (if available). The P_3 mesh is then used as an intermediate level between the linear discrete mesh and the fully continuous geometry model. When a new surface vertex is inserted, a hierarchical projection is used. The vertex is projected to the BREP (through geometry model interaction) and the P_3 mesh. P_3 projection can replace a BREP evaluation failure, provide a guess to BREP inverse evaluation, or help detect BREP construction artifacts in the case of a mismatch.

D. *refine* BREP Closures

refine accesses BREP model topology and evaluates geometry via EGADS. High-performance computing applications are supported by EGADSlite [66], a lightweight ANSI-C version of the EGADS functions required by the initial and adaptive mesh process. While work is underway to leverage EGADS Effective Topology in *refine*, the description here will be focused on EGADS and EGADSlite without Effective Topology.

Two strategies are described for reducing the BREP tolerances exposed to the adaptive mesh mechanics in *refine*. The first computes a smooth displacement field defined for edges and faces, which reduces the distance between face p-curves and edge curves (same for edge curve endpoints and points). The second replaces BREP surfaces and curves with a discrete surrogate, which can be constructed as watertight to machine precision. While the discrete surrogate appears to be the most robust option in production use, both methods are described to illustrate their relative strengths and weaknesses.

1. refine Displacement Map

A linear surface mesh is constructed with EGADS as described in Park et al. [4] to capture curvature and topology. The displacement map is constructed on edge curves by computing the displacement required to move the edge curve endpoints to the BREP node points. The displacement is smoothed in the interior of a discrete edge curve mesh with a pseudo-Laplacian operator (the displacement at a vertex is the average of is neighbors). The displacement field is persisted on a cached initial mesh to enable subsequent mesh operations. The displacement is linearly interpolated inside each segment using barycentric coordinates in segment t and added to the position evaluated at the underlying curve t. A neartree [74] is used to prune the localization in t and decrease execution time.

The displacement vector is computed at each vertex between the displaced edge curve and the face surface evaluated at the p-curve. These displacements stored at discrete face vertices on the face boundaries are smoothed in the interior with a pseudo-Laplacian operator (the displacement at a vertex is the average of is neighbors). The displacement field is persisted on a cached initial mesh to enable subsequent mesh operations. The displacement is linearly interpolated inside each triangle using barycentric coordinates in triangle and added to the position evaluated at the underlying surface (u, v). A neartree is used to prune the localization in (u, v) and decrease execution time.

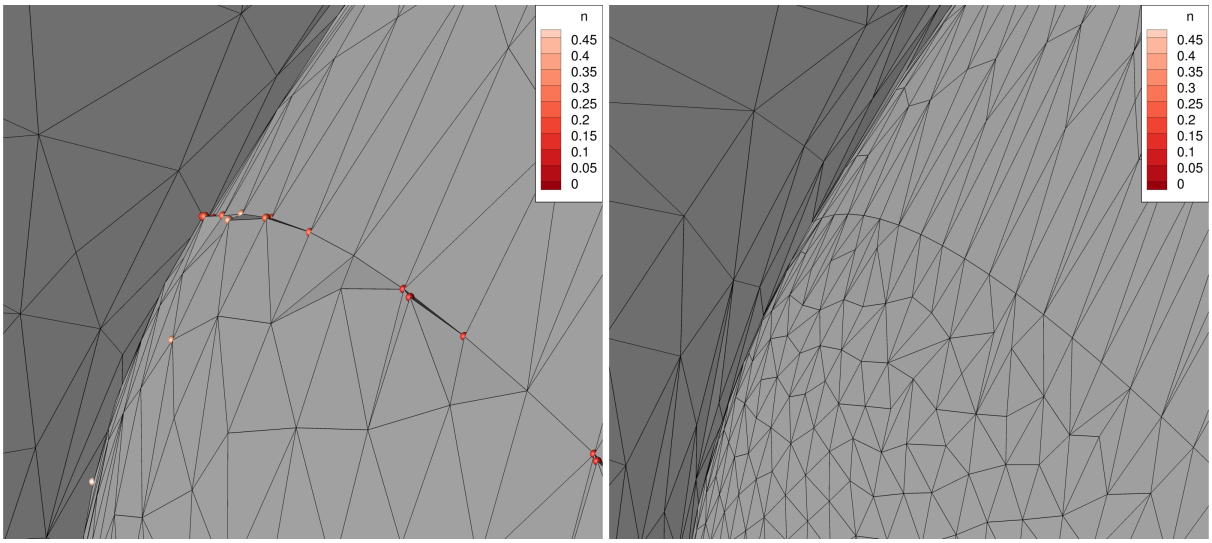
The current implementation uses the underlying EGADS surface/curve curvature and inverse evaluation (accounting for the inverse (opposite sign) of the displacement). While the vertices of the mesh used to construct the displacement are now closed to machine precision, there can be gaps that open when the linearly interpolated displacements do not match the nonlinear gap between vertices. Higher-order interpolation can be used in the displacement to reduce the gaps in between vertices, but it can not eliminate the gaps.

2. refine Surrogate

The same procedure as the displacement map is used to create a linear surface mesh. The linear (P_1) mesh can be promoted to quadratic (P_2) or cubic (P_3) Lagrange basis by linearly interpolating segment t or triangle (u, v) to evaluate midvertices with EGADS. This allows the surrogate to maintain a direct link to the underlying surfaces and curves. An (x, y, z) is shared by vertices and midvertices between triangles and segments, which creates a watertight surrogate by construction. The choice of linear parameter space for the discrete curved elements in (x, y, z) makes (u, v, t) localization and containment (i.e. vertex is inside the face/edge) efficient, but can cause distortion for nonsmooth parameterizations. The surrogate could be nonlinear in parameter space, thereby requiring a Newton solve to find barycentric coordinates for interpolation. Curved elements in parameter space would allow for smoothing of the surrogate to improve the fidelity of the curved element surrogate to the underlying surface or curve. The Lagrange basis could be replaced with another basis (e.g. Chebyshev, Bézier) to yield different properties of the surrogate, which is a topic for future work.

The current implementation uses the underlying EGADS surface/curve curvature and inverse evaluation. An alternative is to walk the surrogate for localization, which would remove the need to persist (u, v, t) and allow for a discrete mesh (created externally) to become a geometry source.

A detail of the CRM-HL inboard flap bracket fairing is shown in Fig. 11. In Fig. 5b, the inboard flap bracket has a



(a) Original definition. **(b) P_3 surrogate.**

Fig. 11 CRM-HL inboard flap bracket fairing adapted triangle normal deviation (n).

large tolerance (dark color) forward of the edge and low tolerance (light color) aft of the edge. The spheres are shown in Fig. 11 when the dot product between the underlying surface normal and the discrete triangle normal is lower than 0.5. A triangle oriented exactly with the surface normal would have a value of $n = 1$ and an inverted triangle will have a value of $n = -1$. Darker red indicates a larger deviation in normal directions (low dot product between normals). The P_3 surrogate has identical parmeterization of the curve and adjoining surfaces where they meet. There is a small slope discontinuity, which is smaller than the slope discontinuity created by triangles that span the C^0 discontinuity between faces and the edge.

IV. Conclusions

Background on Boundary REPresentation (BREP) models commonly manipulated through Mechanical Computer-Aided Design (MCAD) systems is presented. The BREP is used in Computational Fluid Dynamics (CFD) to define the domain or volume for mesh generation and mesh adaptation. The fit of the faces and edges that compose the BREP is the BREP tolerance, where a model that is adequately closed for a particular application is refereed to as "watertight." Typical examples of "CFD-ready" geometry with local BREP tolerances that exceed the surface normal and tangent mesh element size requirements for expert-crafted and solution-adaptive boundary layer meshing are shown. These examples are not unique and issues induced by loose BREP tolerances are common in production CFD application. Unstructured mesh adaptation aims to automate the labor-intensive expert-guided mesh generation process. To attain this automation goal, the adaptive mesh modification mechanics must accommodate typical BREP tolerances unless upstream geometry creation processes can be improved. Poorly shaped elements or failure of the meshing process can occur when solution-adaptive error estimates create mesh sizing requests at or below local BREP tolerances. Application of mesh adaptation to "CFD-ready" geometry requires accommodation in a majority of these applications.

The realities of surface-surface intersection inexactness and geometry model interoperability are presented. Potential approaches are surveyed to mitigate these robustness bottlenecks. They included "repair" of the BREP, surrogate geometry, and a combination of these approaches. Multiple implementations of these techniques are presented and applied to illustrative examples, which are typical of CFD analysis tasks. Virtual topology is enabled by most of these approaches, which reduces topology constraints on meshes and mitigates situations where loose BREP tolerances create topological ambiguities.

These mitigations show significant progress toward accommodation of typical issues with minimal intervention. The maturation of these mitigation techniques will continue the trend from manual repair of input MCAD to an automatic accommodation of MCAD source geometries. Until this transition is complete, the methods shown to identify BREP tolerance issues in Section II can be used to quickly focus manual intervention or identify systemic process weaknesses that can be improved. These BREP interrogation techniques provide rapid targeted guidance to MCAD technicians, and reduce expensive trial-and-error probing.

The mitigation and accommodation techniques described in this paper provide a significant advancement toward automation and the reduction of manual intervention. However, a completely automated solution to meshing these classes of MCAD input geometries is not universally available. Documentation of the issues related to BREP tolerance will continue to attract ideas and resources to address the BREP tolerance bottleneck for automated meshing. Realization of the CFD Vision 2030 Study goals [1] includes supporting geometry linkage to "make mesh generation and adaptation less burdensome and, ultimately, invisible to the CFD process."

Acknowledgments

This work was partially supported by the Transformational Tools and Technologies (TTT) Project of the NASA Transformative Aeronautics Concepts Program (TACP) under the Aeronautics Research Mission Directorate. In addition, this material is partially based on research sponsored by the U.S. Air Force under agreement number FA8650-20-2-2002, where Ryan Durscher AFRL/RQVC is the Technical Monitor. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Air Force or the U.S. Government.

References

[1] Slotnick, J., Khodadoust, A., Alonso, J., Darmofal, D., Gropp, W., Lurie, E., and Mavriplis, D., "CFD Vision 2030 Study: A Path to Revolutionary Computational Aerosciences," NASA CR-2014-218178, Langley Research Center, Mar. 2014. doi:2060/20140003093.

[2] Alauzet, F., and Loseille, A., "A Decade of Progress on Anisotropic Mesh Adaptation for Computational Fluid Dynamics," *Computer-Aided Design*, Vol. 72, 2016, pp. 13–39. doi:10.1016/j.cad.2015.09.005, 23rd International Meshing Roundtable Special Issue: Advances in Mesh Generation.

[3] Park, M. A., Krakos, J. A., Michal, T., Loseille, A., and Alonso, J. J., "Unstructured Grid Adaptation: Status, Potential Impacts, and Recommended Investments Toward CFD Vision 2030," AIAA Paper 2016–3323, 2016. doi:10.2514/6.2016-3323.

[4] Park, M. A., Kleb, B., Jones, W. T., Krakos, J. A., Michal, T., Loseille, A., Haimes, R., and Dannenhoffer, J. F., III, "Geometry Modeling for Unstructured Mesh Adaptation," AIAA Paper 2019–2946, 2019. doi:10.2514/6.2019-2946.

[5] Wyman, N. J., Park, M. A., Baker, P. A., and Chawner, J. R., "A Framework for Mesh-Geometry Associativity during Mesh Adaptation," AIAA Paper 2021–1900, 2021. doi:10.2514/6.2021-1900.

[6] Kleb, B., Park, M. A., Wood, W. A., Bibb, K. L., Thompson, K. B., Gomez, R. J., III, and Tesch, S. H., "Sketch-to-Solution: An Exploration of Viscous CFD with Automatic Grids," AIAA Paper 2019–2948, 2019. doi:10.2514/6.2019-2948.

[7] Michal, T., Kamenetskiy, D., and Krakos, J., "Anisotropic Adaptive Mesh Results for the Third High Lift Prediction Workshop (HiLiftPW-3)," AIAA Paper 2018–1257, 2018. doi:10.2514/6.2018-1257.

[8] Michal, T. R., Kamenetskiy, D. S., Krakos, J., Mani, M., Glasby, R. S., Erwin, T., and Stefanski, D., "Comparison of Fixed and Adaptive Unstructured Grid Results for Drag Prediction Workshop 6," *AIAA Journal of Aircraft*, Vol. 55, No. 4, 2018, pp. 1420–1432. doi:10.2514/1.C034491.

[9] Vanharen, J., Loseille, A., Alauzet, F., and Park, M. A., "Nearfield Anisotropic Mesh Adaptation for the Third AIAA Sonic Boom Workshop," AIAA Paper 2021–347, 2021. doi:10.2514/6.2021-0347.

[10] Bianconi, F., Conti, P., and Di Angelo, L., "Interoperability among CAD/CAM/CAE Systems: A Review of Current Research Trends," *Geometric Modeling and Imaging–New Trends (GMAI'06)*, 2006, pp. 82–89. doi:10.1109/GMAI.2006.30.

[11] Pratt, M. J., "Geometric Modelling: Lessons Learned From The 'Step' Standard," *Geometric Modelling: Theoretical and Computational Basis towards Advanced CAD Applications. IFIP TC5/WG5.2 Sixth International Workshop on Geometric Modelling December 7–9, 1998, Tokyo, Japan*, edited by F. Kimura, Springer US, Boston, MA, 2001, pp. 130–146. doi:10.1007/978-0-387-35490-3_9.

[12] Haimes, R., and Crawford, C., "Unified Geometry Access for Analysis and Design," *Proceedings of the 12th International Meshing Roundtable*, Sandia National Laboratories, 2003, pp. 21–31.

[13] Taylor, N. J., and Haimes, R., "Geometry Modelling: Underlying Concepts and Requirements for Computational Simulation," AIAA Paper 2018–3402, 2018. doi:10.2514/6.2018-3402.

[14] Chawner, J. R., Michal, T., Slotnick, J. P., and Rumsey, C. L., "Summary of the 1st AIAA Geometry and Mesh Generation Workshop (GMGW-1) and Future Plans," AIAA Paper 2018–128, 2018. doi:10.2514/6.2018-0128.

[15] Gammon, M., Bucklow, H., and Fairey, R., "A Review of Common Geometry Issues Affecting Mesh Generation," AIAA Paper 2018–1402, 2018. doi:10.2514/6.2018-1402.

[16] Pointwise, "Preparation of Geometry Models for Mesh Generation and CFD [Whitepaper]," , 2021. `https://info.pointwise.com/en-us/preparation-geometry-models-for-mesh-generation-cfd`.

[17] Piegl, L., and Tiller, W., *The NURBS Book*, 2nd ed., 1995.

[18] Hughes, T. J. R., Cottrell, J. A., and Bazilevs, Y., "Isogeometric Analysis: CAD, Finite Elements, NURBS, Exact Geometry and Mesh Refinement," *Computer Methods in Applied Mechanics and Engineering*, Vol. 194, No. 39, 2005, pp. 4135–4195. doi:10.1016/j.cma.2004.10.008.

[19] Marussig, B., and Hughes, T. J. R., "A Review of Trimming in Isogeometric Analysis: Challenges, Data Exchange and Simulation Aspects," *Archives of Computational Methods in Engineering*, Vol. 25, 2018, pp. 1059–1127. doi:10.1007/s11831-017-9220-9.

[20] Marussig, B., "Advances in the Treatment of Trimmed CAD Models due to Isogeometric Analysis," *13th World Congress on Computational Mechanics (WCCM XIII) and 2nd Pan American Congress on Computational Mechanics (PANACM II), July 22-27, 2018, New York City, NY, USA*, 2018, pp. 2082–2093.

[21] González-Lluch, C., Company, P., Contero, M., Camba, J. D., and Plumed, R., "A Survey on 3D CAD Model Quality Assurance and Testing Tools," *Computer-Aided Design*, Vol. 83, 2017, pp. 64–79. doi:10.1016/j.cad.2016.10.003.

[22] Assadi, A. D., "CAD Model Robustness Assessment and Repair," Ph.D. thesis, Iowa State University, Jun. 2003. doi:10.31274/rtd-180813-14289.

[23] Butlin, G., and Stops, C., "CAD Data Repair," *5th International Meshing Roundtable*, 1996, pp. 7–12.

[24] Mezentsev, A. A., and Woehler, T., "Methods and Algorithms of Automated CAD Repair For Incremental Surface Meshing," *8th International Meshing Roundtable*, 1999, pp. 299–309.

[25] Petersson, N. A., and Chand, K. K., "Detecting Translation Errors in CAD Surfaces and Preparing Geometries for Mesh Generation," *10th International Meshing Roundtable*, 2001, pp. 363–371.

[26] Alleaume, A., "Automatic Non-manifold Topology Recovery and Geometry Noise Removal," *Proceedings of the 18th International Meshing Roundtable*, edited by B. W. Clark, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 267–279. doi:10.1007/978-3-642-04319-2_16.

[27] Bronsart, R., Edessa, D. M., and Kleinsorge, L., "Automatic Pre-Mesh CAD Data Repairing," *International Journal of Mechanical Engineering and Applications*, Vol. 1, No. 1, 2013, pp. 1–9. doi:10.11648/j.ijmea.20130101.11.

[28] Szymczak, W. G., Dey, S., Mestreau, E., Aubrey, R., and Williamschen, M., "Geometry Repair and Construction using NURBS Refitting in Capstone," AIAA Paper 2020–899, 2020. doi:10.2514/6.2020-0899.

[29] Steinbrenner, J. P., Wyman, N. J., and Chawner, J. R., "Procedural CAD Model Edge Tolerance Negotiation for Surface Meshing," *Engineering with Computers*, Vol. 17, 2001, pp. 315–325. doi:10.1007/PL00013392.

[30] Marchandise, E., Piret, C., and Remacle, J.-F., "CAD and Mesh Repair with Radial Basis Functions," *Journal of Computational Physics*, Vol. 231, No. 5, 2012, pp. 2376–2387. doi:10.1016/j.jcp.2011.11.033.

[31] Barequet, G., and Kumar, S., "Repairing CAD Models," *Proceedings of the 8th Conference on Visualization '97*, IEEE Computer Society Press, Washington, DC, USA, 1997, pp. 363–371. doi:10.5555/266989.267105.

[32] Emelyanov, A., Astakhov, Y., and Klimenko, S., "General Concept of Repairing CAD-Models," *2009 International Conference on CyberWorlds*, 2009, pp. 108–113. doi:10.1109/CW.2009.53.

[33] Bischoff, S., and Kobbelt, L., "Structure Preserving CAD Model Repair," *Computer Graphics Forum*, Vol. 24, No. 3, 2005, pp. 527–536. doi:10.1111/j.1467-8659.2005.00878.x.

[34] Michal, T., Krakos, J., and Kamenetskiy, D., "Generation of Anisotropic Adaptive Meshes for the First AIAA Geometry and Mesh Generation Workshop," AIAA Paper 2018–658, 2018. doi:10.2514/6.2018-0658.

[35] Eccles, N. C., Steinbrenner, J. P., and Abelanet, J. P., "Solid Modeling and Fault Tolerant Meshing — Two Complementary Strategies," AIAA Paper 2005–5237, 2005. doi:10.2514/6.2005-5237.

[36] Löhner, R., "Regridding Surface Triangulations," *Journal of Computational Physics*, Vol. 126, No. 1, 1996, pp. 1–10. doi:10.1006/jcph.1996.0115.

[37] Persson, P.-O., Aftosmis, M. J., and Haimes, R., "On The Use of Loop Subdivision Surfaces for Surrogate Geometry," *15th International Meshing Roundtable*, 2006, pp. 375–392. doi:10.1007/978-3-540-34958-7_22.

[38] Laug, P., and Borouchaki, H., "Discrete CAD Model for Visualization and Meshing," *Procedia Engineering*, Vol. 163, 2016, pp. 149–161. doi:10.1016/j.proeng.2016.11.039, 25th International Meshing Roundtable.

[39] Feuillet, R., Coulaud, O., and Loseille, A., "Anisotropic Error Estimate for High-Order Parametric Surface Mesh Generation," *28th International Meshing Roundtable*, 2019.

[40] Ruiz-Gironés, E., and Roca, X., "Imposing Boundary Conditions to Match a CAD Virtual Geometry for the Mesh Curving Problem," *27th International Meshing Roundtable*, edited by X. Roca and A. Loseille, Springer International Publishing, 2019, pp. 343–361. doi:10.1007/978-3-030-13992-6_19.

[41] Vlachos, A., Peters, J., Boyd, C., and Mitchell, J. L., "Curved PN Triangles," *I3D '01 Proceedings of the 2001 symposium on Interactive 3D graphics*, 2001, pp. 159–166. doi:10.1145/364338.364387.

[42] Kosinka, J., and Cashman, T. J., "Watertight Conversion of Trimmed CAD Surfaces to Clough–Tocher Splines," *Computer Aided Geometric Design*, Vol. 37, 2015, pp. 25–41. doi:10.1016/j.cagd.2015.06.001.

[43] Sheffer, A., Bercovier, M., Blacker, T., and Clements, J., "Virtual Topology Operators for Meshing," *International Journal of Computational Geometry & Applications*, Vol. 10, No. 03, 2000, pp. 309–331. doi:10.1142/S0218195900000188.

[44] Gammon, M. R., Fellows, C. D., and Whyman, S. H., "Enabling Virtual Topology for High Quality CFD Surface Meshing of Complex CAD Geometry," AIAA Paper 2018–3721, 2018. doi:10.2514/6.2018-3721.

[45] Sun, L., Yao, W., Robinson, T. T., Armstrong, C. G., and Marques, S., "Automated Mesh Deformation for Computer-Aided Design Models that Exhibit Boundary Topology Changes," *AIAA Journal*, Vol. 58, No. 9, 2020. doi:10.2514/1.J058760.

[46] Dannenhoffer, J. F., III, and Haimes, R., "Quilts: A Technique for Improving Boundary Representations for CFD," AIAA Paper 2003–4132, 2003. doi:10.2514/6.2003-4131.

[47] Dannenhoffer, J. F., III, and Haimes, R., "Robust Algorithms for Generating Quilts and Chains," AIAA Paper 2006–943, 2006. doi:10.2514/6.2006-943.

[48] Tierney, C. M., Sun, L., Robinson, T. T., and Armstrong, C. G., "Using Virtual Topology Operations to Generate Analysis Topology," *Computer-Aided Design*, Vol. 85, 2017, pp. 154–167. doi:10.1016/j.cad.2016.07.015, 24th International Meshing Roundtable Special Issue: Advances in Mesh Generation.

[49] Rogers, S. E., Dalle, D. J., and Chan, W. M., "CFD Simulations of the Space Launch System Ascent Aerodynamics and Booster Separation," AIAA Paper 2015–778, 2015. doi:10.2514/6.2015-0778.

[50] Wilcox, F. J., Jr., Pinier, J. T., Chan, D. T., and Crosby, W. A., "Space Launch System Booster Separation Aerodynamic Testing in the NASA Langley Unitary Plan Wind Tunnel," AIAA Paper 2016–796, 2016. doi:10.2514/6.2015-0796.

[51] Schmitt, V., and Charpin, F., "Pressure Distributions on the ONERA-M6-Wing at Transonic Mach Numbers," *Experimental Data Base for Computer Program Assessment: Report of the Fluid Dynamics Panel Working Group 04*, AR-138, NATO Research and Technology Organisation AGARD, 1979, pp. B1:1–B1:44.

[52] Mayeur, J., Dumont, A., Destarac, D., and Gleize, V., "Reynolds-Averaged Navier–Stokes Simulations on NACA0012 and ONERA-M6 Wing with the ONERA elsA Solver," *AIAA Journal*, Vol. 54, No. 9, 2016, pp. 2671–2687. doi:10.2514/1.J054512.

[53] Mayeur, J., Dumont, A., Destarac, D., and Gleize, V., "RANS Simulations on TMR 3D Test Cases with the ONERA elsA Flow Solver," AIAA Paper 2016–1357, 2016. doi:10.2514/6.2016-1357.

[54] Lacy, D. S., and Sclafani, A. J., "Development of the High Lift Common Research Model (HL-CRM): A Representative High Lift Configuration for Transonic Transports," AIAA Paper 2016–308, 2016. doi:10.2514/6.2016-0308.

[55] Lacy, D. S., and Clark, A. M., "Definition of Initial Landing and Takeoff Reference Configurations for the High Lift Common Research Model (CRM-HL)," AIAA Paper 2020–2771, 2020. doi:10.2514/6.2020-2771.

[56] Taylor, N. J., and Gammon, M., "A Wind Tunnel Model of the NASA High Lift Common Research Model: A Geometry-Handling Perspective," AIAA Paper 2019–3460, 2019. doi:10.2514/6.2016-1357.

[57] Heft, A. I., Indinger, T., and Adams, N. A., "Introduction of a New Realistic Generic Car Model for Aerodynamic Investigations," SAE Technical Paper 2012-01-0168, 2012. doi:10.4271/2012-01-0168.

[58] Heft, A. I., Indinger, T., and Adams, N. A., "Experimental and Numerical Investigation of the DrivAer Model," *Fluids Engineering Division Summer Meeting*, Vol. 1, 2012, pp. 41–51. doi:10.1115/FEDSM2012-72272.

[59] Heft, A. I., Indinger, T., and Adams, N. A., "Investigation of Unsteady Flow Structures in the Wake of a Realistic Generic Car Model," AIAA Paper 2011–3669, 2011. doi:10.2514/6.2011-3669.

[60] Edward Alyanak, R. D., Haimes, R., Dannenhoffer, J. F., III, Bhagat, N., and Allison, D., "Multi-fidelity Geometric-centric Multi-disciplinary Analysis for Design," AIAA Paper 2016–4007, 2016. doi:10.2514/6.2016-4007.

[61] Donovan, A. B., and Allison, D. L., "Multidisciplinary Design Including Geometry-Enabled Wind Tunnel Fidelity Level Analysis," AIAA Paper 2017–3662, 2017.

[62] Bryson, D. E., Haimes, R., and Dannenhoffer, J. F., III, "Toward the Realization of a Highly Integrated, Multidisciplinary, Multifidelity Design Environment," AIAA Paper 2019–2225, 2019. doi:10.2514/6.2019-2225.

[63] Haimes, R., and Dannenhoffer, J. F., III, "The Engineering Sketch Pad: A Solid-Modeling, Feature-Based, Web-Enabled System for Building Parametric Geometry," AIAA Paper 2013–3073, 2013. doi:10.2514/6.2013-3073.

[64] Dannenhoffer, J. F., III, "OpenCSM: An Open-Source Constructive Solid Modeler for MDAO," AIAA Paper 2013–701, 2013. doi:10.2514/6.2013-701.

[65] Haimes, R., and Drela, M., "On The Construction of Aircraft Conceptual Geometry for High-Fidelity Analysis and Design," AIAA Paper 2012–683, 2012. doi:10.2514/6.2012-683.

[66] Haimes, R., and Dannenhoffer, J. F., III, "EGADSlite: A Lightweight Geometry Kernel for HPC," AIAA Paper 2018–1401, 2018. doi:10.2514/6.2018-1401.

[67] Marcum, D. L., and Gaither, J. A., "Unstructured Surface Grid Generation Using Global Mapping and Physical Space Approximation," *8th International Meshing Roundtable*, 1999, pp. 397–406.

[68] Loseille, A., and Löhner, R., "Cavity-Based Operators for Mesh Adaptation," AIAA Paper 2013–152, 2013.

[69] Loseille, A., Alauzet, F., and Menier, V., "Unique Cavity-Based Operator and Hierarchical Domain Partitioning for Fast Parallel Generation of Anisotropic Meshes," *Computer-Aided Design*, Vol. 85, 2017, pp. 53–67. doi:10.1016/j.cad.2016.09.008, 24th International Meshing Roundtable Special Issue: Advances in Mesh Generation.

[70] Frey, P. J., "About Surface Remeshing," *9th International Meshing Roundtable*, 2000, pp. 123–136.

[71] Perdigão do Carmo, M., *Differential Geometry of Curves and Surfaces*, Prentice-Hall, Inc., 1976.

[72] George, P.-L., Borouchaki, H., Frédéric, Laug, P., Loseille, A., and Maréchal, L., *Meshing, Geometric Modeling and Numerical Simulation, Volume 2: Metrics, Meshes and Mesh Adaptation*, John Wiley & Sons, 2019.

[73] George, P.-L., and Borouchaki, H., "Validity of Lagrange (Bézier) and Rational Bézier Quads of Degree 2," *International Journal for Numerical Methods in Engineering*, Vol. 99, No. 8, 2014, pp. 611–632. doi:10.1002/nme.4696.

[74] Andrews, L., "A Template for the Nearest Neighbor Problem," *C/C++ Users Journal*, Vol. 19, No. 11, 2001, pp. 40–49.