Overset Structured Mesh Generation on Complex Configurations Using the EPOGS Toolset

William M. Chan*, Andrew M. Chuen†, James C. Jensen‡, and Daniel G. Schauerhamer§
NASA Ames Research Center, Moffett Field, CA 94035

Recent advances in automation of overset structured mesh generation have been implemented in the EPOGS software toolset. This toolset can be used to perform almost all of the pre-processing steps needed to begin flow solution computation starting from a Boundary Representation (BRep) solid definition of the geometry. These include generation of surface meshes, near- and off-body volume meshes, and domain connectivity data files. Additionally, input files containing boundary conditions for the flow solver, and inputs for component-based aerodynamic loads computation are also created. Several recent developments in the automation scheme are presented in this paper. Treatment of node meshes are improved at junctions with two closely-spaced nodes, and at junctions with a cusp or trident topology. Additionally, an automated volume mesh boundary splay adjustment scheme is introduced that improves adjacent mesh overlap away from the wall. Also, an automated assembly scheme is developed to create a grid system for multiple disjoint bodies with multiplicity in relative motion from independently generated individual body systems. The capabilities of EPOGS are demonstrated in various configurations from the AIAA Drag Prediction Workshop 8 and Aeroelastic Prediction Workshop 4, the Tiltwing Test Rig Experiment, and the VAERA fire-fighting drone.

I. Introduction

For many years, overset structured mesh technology has been successfully applied in CFD analyses for a large variety of aerospace applications [1–13]. This method is particularly suited for problems with components in relative motion such as Vertical Take-Off and Landing (VTOL) vehicles and rocket stage separation [14–19]. Body-fitted curvilinear near-body meshes only need to be generated once. Minor mesh spacing adjustments may be needed if additional geometric components in close proximity are introduced at a later time as in a component build-up study. Since arbitrary inter-grid overlap is allowed, these meshes are free to translate and rotate relative to each other during three-dimensional motion. The high-quality curvilinear near-body meshes with excellent orthogonality and viscous clustering properties are not perturbed during rigid-body motion, and only inter-grid connectivity needs to be recomputed. While compressible viscous flow solvers for overset structured mesh technology, such as Overflow [20, 21] and LAVA (Launch Ascent and Vehicle Aerodynamics) [22], are highly efficient and more easily extensible to high order methods compared to unstructured grid approaches, overset structured mesh generation has been a significacnt bottleneck for many users since it requires extensive expertise and human effort.

To help overcome this bottleneck, a long-term effort to automate overset structured mesh generation began in 2018 [23–26]. The developed method starts with a Boundary Representation (BRep) solid definition of the geometry, then sequentially creates surface meshes, surface mesh connectivity, near- and off-body volume meshes, volume mesh connectivity, flow solver boundary conditions for Overflow and LAVA, and inputs for component-based aerodynamic loads computation using the FOMOCO Utilities in the Chimera Grid Tools software toolset [27]. The Engineering Sketch Pad (ESP) open source library [28] is used as the interface to the BRep solid geometry in a STEP, IGES, or EGADS file.

This paper is organized as follows. A short review of the automated meshing process is given in Section II. The meshing workflow using the EPOGS software toolset, consisting of the EGADS2SRF (EGADS to Surfaces) tool and the POGS (Pre-processor for Overset Grid Simulations) tool, is presented in Section III. New developments in node mesh treatment, mesh overlap improvement and assembly of auto-generated meshes in configurations involving disjoint bodies with multiplicity in relative motion are described in Section IV. Recent applications of EPOGS to complex configurations are presented in Section V. Summary and conclusions are given in Section VI.

*Computer Scientist, Computational Aerosciences Branch, M/S 258-2, AIAA Associate Fellow.
†Aerospace Engineer, Computational Aerosciences Branch, M/S 258-2, AIAA Member.
‡Aerospace Engineer, Computational Aerosciences Branch, M/S 258-2, AIAA Member.
§Aerospace Engineer, CFD Research Corp., AIAA Member.

II. Review of Automated Overset Meshing Process

Overset structured grid generation is divided into three main steps: surface mesh generation, volume mesh generation and domain connectivity. The manual meshing process on complex geometries typically requires extensive expertise, effort and time. In order to complete the three steps above, a variety of tools are needed. Typical mesh generation tools include the Chimera Grid Tools (CGT) software suite [27] and Cadence Fidelity Pointwise, while typical domain connectivity tools include PEGASUS5 [29], Overflow/DCF [30], SUGGAR [31], and PUNDIT [32].

The automation scheme introduced in [23–26] begins with a geometry definition in BRep solid format. A geometric components definition file called the *gcomp* file is first manually constructed which lists the BRep faces that belong to each component [26]. This data is used to (1) scale the local maximum grid spacing based on component size, (2) construct inputs for component aerodynamic loads computation, (3) guide construction of X-ray hole-cut maps and instructions if the X-ray hole-cut domain connectivity option is selected.

Using the Engineering Sketch Pad (ESP) open source library [28] as the interface, the topological relationship between the BRep faces, edges, and nodes is extracted. Surface meshes are created over each face, edge and node. The process begins by placing appropriate grid point distributions on the BRep edges with consideration of local geometry convexity and concavity. Surface meshes are generated around the nodes first (Fig. 1a), followed by around the edges (Fig. 1b). The node and edge meshes are generated using algebraic and hyperbolic methods, where proper overlap between the initial curves of these meshes are guaranteed. These meshes form a band around the boundaries of all BRep faces, and as a result, all surface geometric features are also covered by the node and edge meshes. The marching scheme types (algebraic or hyperbolic) and marching distances for the node and edge meshes are automatically selected. A loose coupling scheme [24] is then used to ensure proper surface overlap between adjacent node and edge meshes.

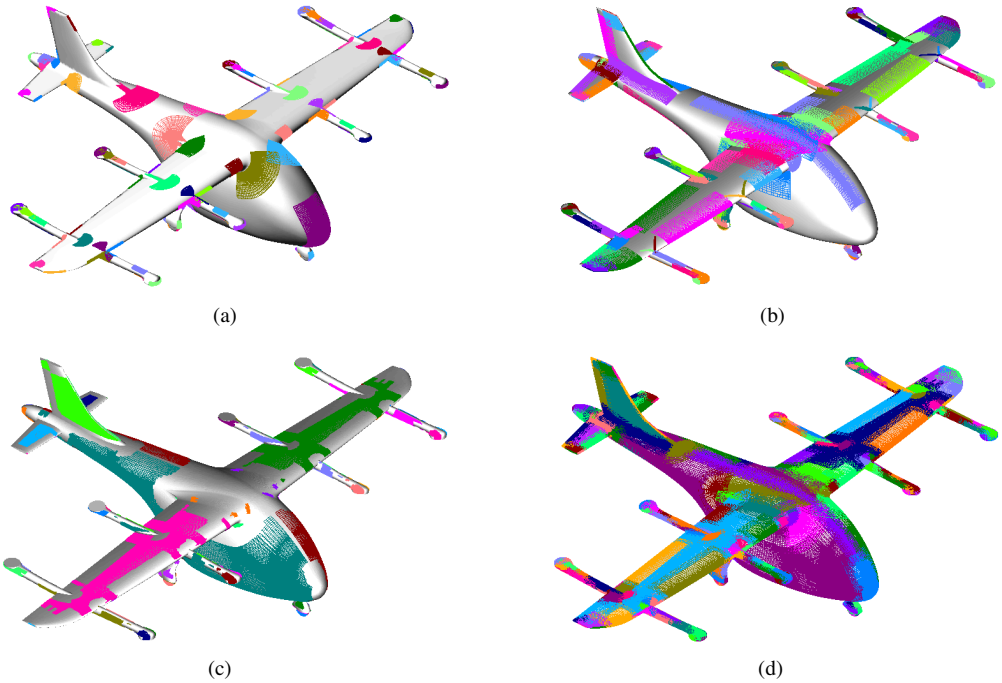
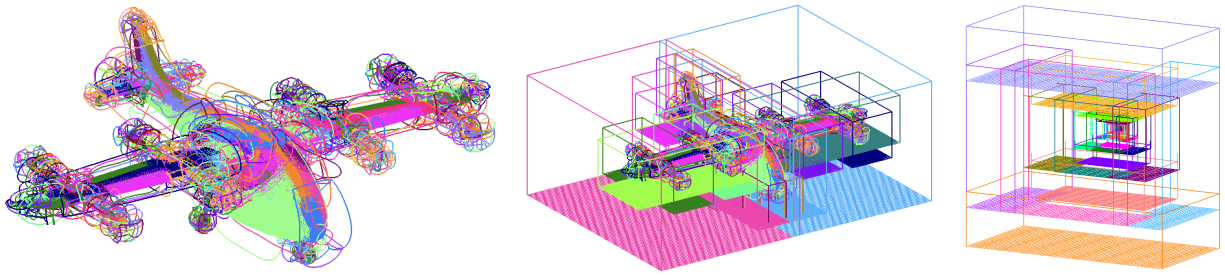


Fig. 1 Auto-generated surface meshes for the Lift+Cruise concept vehicle. (a) Node meshes. (b) Edge meshes. (c) Retracted face meshes. (d) Final combined overset surface mesh system.

Appropriate grid point distributions are placed on the BRep faces where an integer mask (iblank array) is used to tag grid points that are on or off the geometry. Since the boundaries of the face meshes are covered by the node and edges meshes, grid points on the face meshes are retracted (tagged as unused) from their boundaries by adjustment of the iblank values (Fig. 1c). The collection of node, edge, and face meshes forms the overset surface mesh system (Fig. 1d). Note that the iblank retraction step involves finding donor stencils for all fringe points on the surface meshes. This is equivalent to the standard hole-cut procedure on the overset surface grids without requiring manual hole-cut instructions.

This meshing strategy results in a total number of mesh zones (mesh count) that is approximately equal to the sum of the number of BRep faces, edges and nodes. Although the resulting grid system contains a lot more zones compared to typical manual meshing approaches, it has been shown in computed RANS solutions that there is no significant penalty on the convergence behavior, computational cost, or accuracy of the integrated aerodynamic loads [33].



(a) (b) (c)

Fig. 2 Auto-generated volume meshes for L+C. (a) Near-body curvilinear grids. (b) Near-body curvilinear and off-body component Cartesian box grids. (c) Multi-layered off-body Cartesian box grids with telescoping resolution.

Near-body volume grids are generated over all the surface grids using hyperbolic methods (Fig. 2a). The point distribution in the normal direction is determined by geometric stretching based on the prescribed wall-normal grid spacing and a maximum stretching ratio. The marching distance for each mesh is computed by finding the distance needed to reach a grid spacing equal to or just less than the local maximum grid spacing. This local maximum grid spacing is scaled relative to the global maximum grid spacing based on the data from the *gcomp* file, and is dependent on the size of the component that the mesh belongs to. As a result, meshes from smaller components are grown to a smaller distance.

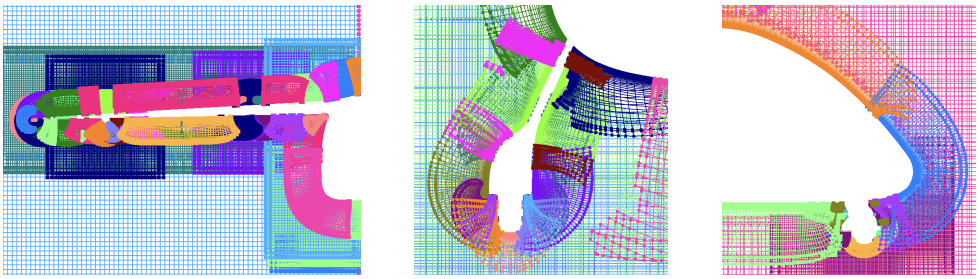
Several near-field off-body Cartesian mesh generation schemes are provided in EPOGS. While the simplest option is to use a single Cartesian grid with a uniform core spacing that matches the near-body grids plus stretched layers extending to the far field, the standard practice is to create a component Cartesian box grid around each component's volume grids (Fig. 2b). Each component box grid contains a core region that encloses all the near-body volume meshes of the component, and with uniform spacing that matches the average outer boundary spacing of the component volume meshes. Stretched layers are added in the $\pm x, \pm y, \pm z$ directions until the grid spacing reaches the global maximum grid spacing. A uniform Cartesian mesh is then used to enclose all the component box grids with a uniform spacing equal to the global maximum grid spacing.

Two options are available to extend the off-body grids to the far field. The first is a mixed-mode approach where the grids going to the far-field are created in the flow solver (Overflow and LAVA) and not in EPOGS. In this approach, the uniform outer box grid created in EPOGS is discarded before sending the grids to the flow solver. In the second approach, multiple layers of telescoping uniform Cartesian grids are added in the $\pm x, \pm y, \pm z$ directions until the layers reach the prescribed far-field distance (Fig. 2c). An expansion factor (> 1.0) is used to specify the grid spacing from one layer to the next.

Volume mesh domain connectivity consists of two steps: hole-cutting and donor stencil search for the fringe points. In the automation scheme described above, surface mesh hole-cutting is already incorporated into the surface mesh generation method. Let J and K be the surface mesh index directions, and L be the wall-normal direction in the hyperbolically grown near-body meshes. Hole-cutting in the near-body volume meshes consists of three steps. (1) Blanked points on a surface mesh are propagated into the volume mesh in the L-direction. (2) Unblanked volume mesh strands in the L-direction are intersected against the surface meshes to form a minimum hole. For each surface grid point (J,K), a distance function in the L direction is computed which stores the distance from the wall to the last unblanked point in the L direction. (3) A clearance test is used to blank further points to establish a variable clearance from the closest surface mesh cell where the clearance is dependent on the L-strand distance function of the closest surface mesh cell computed in step 2. Donor stencil search for the fringe points uses bounding boxes and Structured Auxiliary Mesh (SAM) bins [34] to speed up the search process.

III. Automatic Meshing Software Toolset (EPOGS)

The meshing automation scheme described in Section II has been implemented into the EPOGS software toolset. Starting with a clean BRep solid geometry, the goal is to enable the user to rapidly create all the files needed to begin flow solution computation. These include the volume grid, domain connectivity data, boundary conditions for each grid described in the flow solver input files, and inputs for component aerodynamic loads computation. EPOGS is the first and only software of its kind that attempts to automate all the steps needed prior to running the flow solver for overset structured grid CFD analysis.



(a) (b) (c)

Fig. 3 Auto-generated domain connectivity for L+C grid system. (a) Constant-x cut around right wing. (b) Constant-x cut around landing gear. (c) Constant-y cut over fuselage front.

EPOGS consists of the following two codes:

1) The EGADS2SRF (EGADS to Surfaces) tool creates discretized BRep faces and edges from the geometry file via the EGADS library from ESP. Accepted geometry file formats include STEP, IGES and EGADS (.stp, .igs, .iges, or .egads file name extensions) with at least one BRep solid.
2) The POGS (Pre-processor for Overset Grid Simulations) tool generates the overset surface and volume meshes, domain connectivity data, boundary conditions data in input files for the flow solvers Overflow [20, 21] and LAVA [22], and surface index data in an input file for the aerodynamic loads computation tool FOMOCO Utilities [35].

EPOGS allows the following three types of symmetry in the geometry [26].

1) No symmetry - grids are created on the entire input geometry.
2) A half-body - input geometry is defined on a half-body bounded by the $x = 0$, $y = 0$, or $z = 0$ plane.
3) A symmetric full-body - input geometry is defined on a full-body where the intent is for a symmetric body about the $x = 0$, $y = 0$, or $z = 0$ plane, but the input geometry does not necessarily contain symmetric properties about the symmetry plane. Although this is not best practice, the input geometry may have different face and edge representations on the right and left sides of the symmetry plane. For such cases, a symmetric grid system is created by generating grids on the $+x$, $+y$, or $+z$ side, followed by reflection and concatenation of appropriate grids to form the symmetric grid system.

A. Meshing Workflow

A typical workflow for using EPOGS is illustrated in Fig. 4. There are two possible starting points: A and B. When a BRep solid representation of the geometry is available in the form of a STEP, IGES, or EGADS file, and after performing appropriate geometry clean-ups as described in Section III.C, starting point A is selected. A geometric components data file is first constructed which contains the list of BRep faces (index numbers) belonging to each component. This is typically accomplished by running EGADS2SRF up to the creation of discretized faces only and then building the components data file in an editor by viewing the faces in a graphical interface such as OVERGRID [36]. After running EGADS2SRF, the discretized faces and edges and the BRep edge/face topological relationship are written to the files listed in Table 1. The output structured surface u/v tessellation file generated by EGADS2SRF will contain iblanks with 0 and 1 values if a face is trimmed in the representation of the geometry. If no trimming is used on a face, then the iblank tag has value 1 everywhere.

Table 1 Output files generated by EGADS2SRF from starting point A. Same as input files needed by POGS using starting point B.

File	Format
(1) Trimmed unstructured surface triangulation of the geometry	Cart3d
(2) Structured surface u/v tessellation: untrimmed faces with iblanks, or trimmed faces without iblanks	Plot3d
(3) Curve grids: discretized edges	Plot3d
(4) Topology data file: face index and face normal sense on each side of each edge	ASCII
(5) Input parameters file for POGS	ASCII

At this point, the normal procedure for running EPOGS continues to the POGS step which starts from the output files created by EGADS2SRF and builds the overset grid system. However, when a BRep solid geometry is not available,

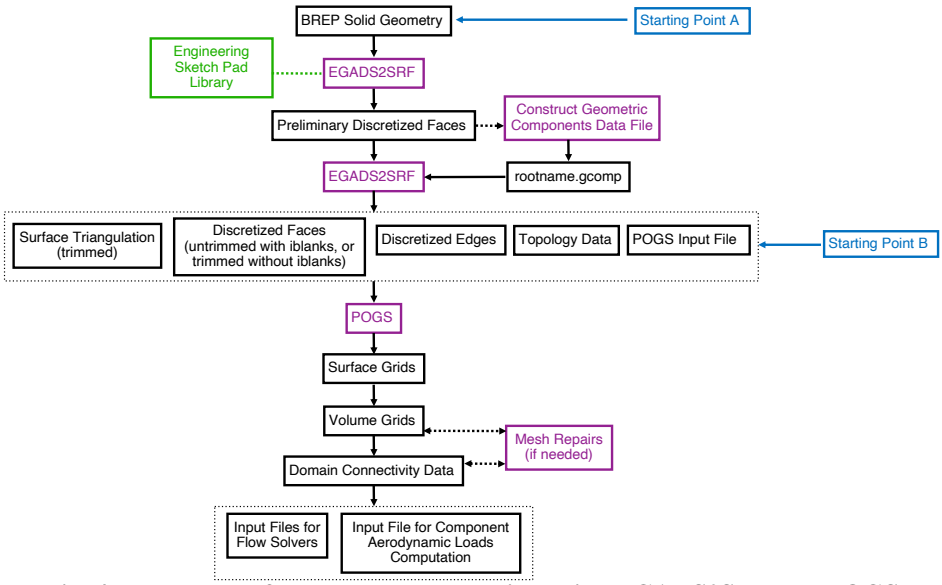


Fig. 4 Flow chart for overset auto-meshing using EGADS2SRF and POGS.

it is possible to begin the EPOGS workflow from starting point B if the list of files in Table 1 is provided from another geometry source. Most CAD software can write an unstructured surface triangulation of the geometry and the discretized edges. Some work will be needed to create the file with the discretized faces tessellated along constant u,v parameters lines. This file contains the untrimmed faces with an iblank array that marks vertices that are on or off the geometry, or the trimmed faces where no iblanks are needed. By definition, the trimmed faces are abutting and any boundary of such faces is a constant u or v parameter line. Similarly, some work will be needed to create the text file containing the topological relationship between each edge and its neighboring faces. The POGS input parameters text file contains just three lines of input and is very easy to construct. Automation of the workflow from starting point B is in the future work plan for EPOGS.

Irrespective of starting from point A or B, the POGS step generates the overset structured grid system and input files needed to run the flow solver and component-based aerodynamic loads computation tool. Check points are implemented at various surface and volume meshing steps in the POGS process where the user can stop, inspect the results, make appropriate adjustments if needed, and restart the process. While a significant reduction of manual effort has been accomplished in the overset meshing process using these tools, a relatively small amount of manual repairs are typically needed to improve local surface and volume mesh overlap (typically 0% - 10% of meshes) and remove negative Jacobians and self-intersecting spiral regions in near-body volume meshes (typically 0% - 1% of meshes). These repairs are usually performed at the end of the near-body volume meshing step. At the same time, the user is also free to substitute any grid(s) in the near-body system and restart the process from there.

At the end of running POGS, the following files are created.

1) Overset surface grid system with adjusted hole-cuts between grids and surface domain connectivity data.
2) Near- and off-body volume grid system.
3) Domain connectivity data for volume grid system.
4) Input files for Overflow and LAVA flow solvers with boundary conditions for the volume grid system.
5) Input file for FOMOCO Utilities in Chimera Grid Tools for component-based aerodynamic loads computation.

B. Global Control Parameters

A small number of global parameters are used to control the entire meshing process. A summary of these is given in Table 2. These parameters are specified on the command line input and/or via an ASCII text input file.

C. Current Limitations on Clean BRep Geometry

The chance of success on overset meshing automation using the EPOGS toolset is currently dependent on the quality of the input BRep solid geometry. A BRep solid consists of a collection of faces, edges and nodes with a topological relationship: each face is bounded by two or more edges; each edge has one face on each side; each node resides at the junction of two or more edges. The mesh repair effort needed is minimized or eliminated if the geometry satisfies the

Table 2 EPOGS global grid control parameters (s=surface mesh parameter, v=volume mesh parameter, d=domain connectivity parameter).

Parameter	Explanation
$a_{\max}$	Maximum dihedral angle (s)
$\Delta s_{\max}$	Maximum grid spacing (s,v)
$c_{\max}$	Maximum chordal deviation (s)
$sr_{\max}$	Maximum stretching ratio (s,v)
$np_{\min}$	Minimum number of grid points on a surface feature (s)
Δs_{wall}	Wall normal grid spacing (based on freestream Reynolds number) (v)
$klayer$	Number of constant-spacing layers in normal direction at the wall (v)
d_{ff}	Distance to far-field boundary (v)
N_f	Number of fringe point layers, typically governed by the size of flow solver differencing stencil (d)
$q_{\min}$	Fringe point with donor stencil quality < $q_{\min}$ are tagged as an orphan point ($0.0 < q_{\min} \leq 1.0$) (d)
f_{cut}	Off-body Cartesian grid points are blanked if they are within a distance of $f_{\text{cut}} \times d_v$ from the surface where d_v is the automatically-determined local near-body mesh normal marching distance ($0.0 < f_{\text{cut}} < 1.0$) (d)

following "clean" properties:

1) Poor choice of tolerance settings can sometimes result in the introduction of a tiny edge where four edges meet. Such tiny edges should be removed by adjusting to a proper tolerance (Fig. 5a).
2) A spurious node is a node where only two edges meet and both edges have the same neighboring faces. Spurious nodes should be eliminated by merging the two neighboring edges (Fig. 5b).
3) The number of faces is as small as practically possible. This is achieved by merging adjacent faces where there is no slope discontinuity at the dividing edge (Fig. 5b).
4) Sliver/tiny faces are avoided by merging with adjacent faces (Fig. 5c).
5) A face with large trimmed regions is reconstructed using its bounding edges so that no trimming is needed (Fig. 5d).
6) The constant u/v parameter lines on a face are aligned with local surface curvature, and to its bounding edges as much as possible.

Satisfying properties 1, 2, 3, and 4 would reduce the number of zones in a grid system and avoid tiny meshes. Grid points exist on a face mesh even in the trimmed region of the geometry but they are tagged with an iblank value of zero. Removing trimmed regions from the faces (property 5) as much as possible would reduce the size of the final grid system. Grid points on face meshes are created along constant u/v parameter lines. Aligning the constant parameter lines along surface curvature directions (property 6) would enhance resolution of the geometry. Removal of some the above restrictions is on the near-term development plan for future versions of EPOGS. Currently, investing the effort to achieve the above clean properties on the geometry significantly speeds up the meshing process. The clean up step is typically performed using the CAD software on which the geometry was originally created, or any other CAD software that can handle BRep solids such as Rhino, Creo, SolidWorks, CATIA, and others.

IV. Recent Developments

Several recent developments in the automation scheme are presented in this section. In the surface mesh generation step described in Section II, a node mesh is used to cover a junction between two or more edges where a hyperbolic or algebraic marching scheme is employed to generate the different parts. For most cases, the initial curve for the marching scheme is selected from concatenating clipped segments from the two most appropriate edges meeting at the node [23]. However, a different scheme for constructing the initial curve is needed under certain conditions. Treatment of nodes with cusp or trident edge topologies is presented in Section IV.A. Special treatments to create a single node mesh in the region covering two nodes with sharp connected edges in close proximity are described in Ref. [23]. When the number of connected convex and concave sharp edges is different between the two nodes, a new treatment is needed and is presented in Section IV.B. A method to improve adjacent near-body volume mesh overlap by automatic boundary

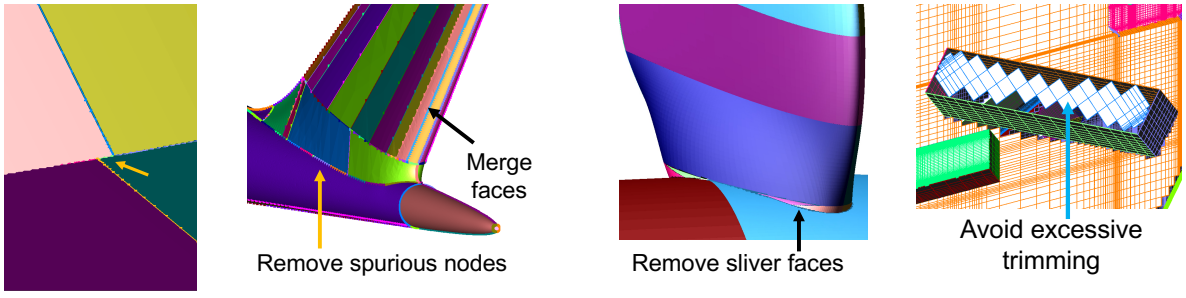


Fig. 5 Common geometry problems to be avoided. (a) Tiny edge due to poor tolerance. (b) Spurious nodes and fractured faces. (c) Sliver/tiny faces. (d) Excessive trim.

splay adjustment is described in Section IV.C. An option in EPOGS to enable rapid assembly of grid and input files for configurations involving multiple disjoint bodies in relative motion is presented in Section IV.D. One or more of the disjoint bodies may be duplicated as in the case of a multi-bladed propeller.

A. Node Meshes at Cusp or Trident Junctions

For a node of degree three (junction where three edges meet), there are two situations when the initial curve for a hyperbolic or algebraic marching scheme should not be selected from two segments of the meeting edges. The first case, called a cusp node, is when the edges meet at a cusp at the node (Fig. 6a). The second case occurs when the dot product between any two of $\vec{e}_1, \vec{e}_2, \vec{e}_3$ is positive and the maximum angle between any pair of these vectors is less than 45°, where $\vec{e}_i$ with $i = 1, 2, 3$ is the unit vector along the i^{th} edge pointing away from the node (Fig. 6b). Such a node is called a trident node from here on.

For both of these node types, selection of the two outer edges in Figs. 6a and 6b to be concatenated to form the initial curve would result in a very difficult concave corner for marching. An alternative to achieve better mesh quality is to select the middle edge and extrapolate it to the opposite side (Figs. 6c and 6d) to form the initial curve, and relax the requirement to enforce a grid line along the other two edges. An example of the initial curve created for a cusp and a trident node are shown in Fig 6c and 6d by the red dotted line.

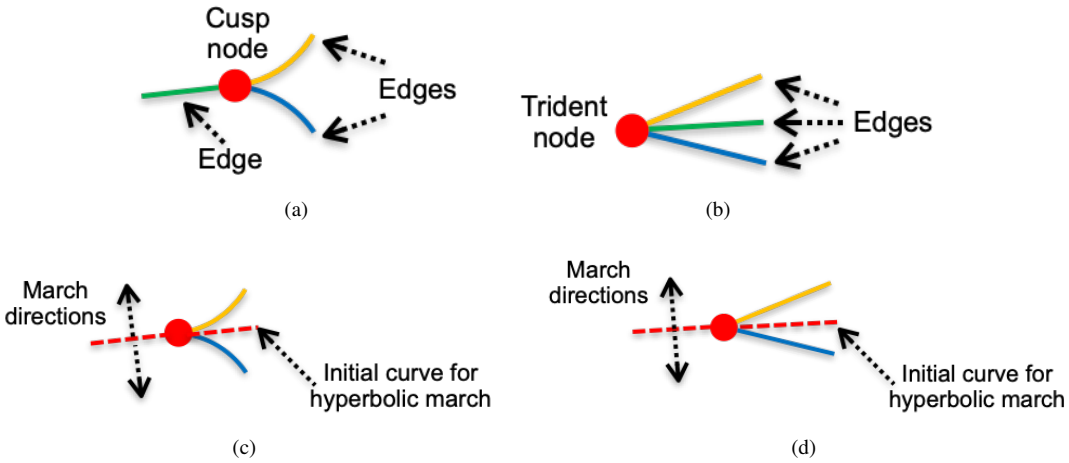
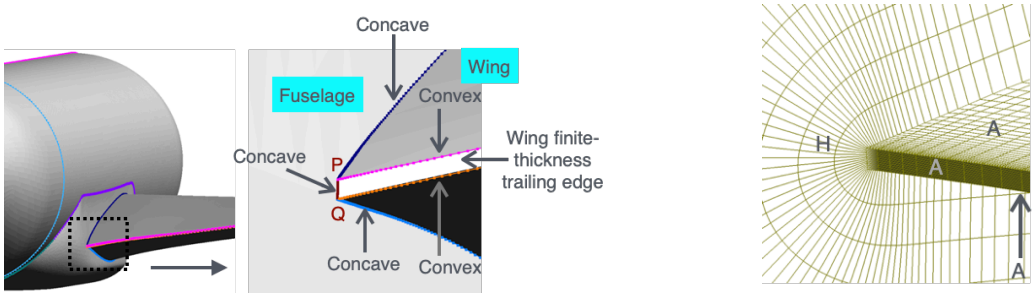


Fig. 6 Cusp and trident nodes. (a) Cusp node and edges. (b) Trident node and edges. (c) Constructed initial curve and marching directions for cusp node. (d) Constructed initial curve and marching directions for trident node.

B. Node Meshes for Two Nodes in Close Proximity

A thin face is defined to be a face bounded by four edges where two opposite edges have very small arc lengths (short edges). This situation typically occurs on finite thickness trailing edges of components with airfoil shapes such as wings, tails, slats, flaps, pylons, nacelles, fins and canards. At the end of a thin face, two nodes lie in close proximity to each other connected by a short edge. Instead of creating two node meshes here, a single node mesh that covers both

nodes is preferred. Various topological scenarios for these thin face node meshes are described in Ref.[25]. An example of a thin face node mesh at the junction between the wing root trailing edge and fuselage is shown in Fig. 7. Nodes P and Q are both located at a junction between two sharp concave edges and one sharp convex edge (Fig. 7a). A single surface mesh covering both nodes is constructed by concatenating various algebraic and hyperbolic parts (Fig. 7b).

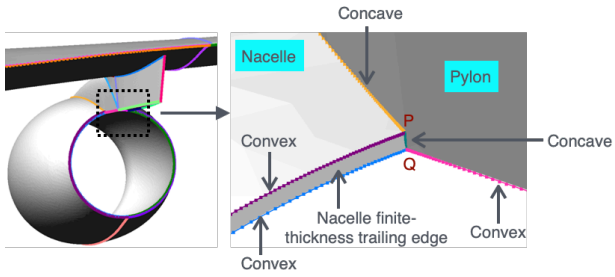


(a) (b)

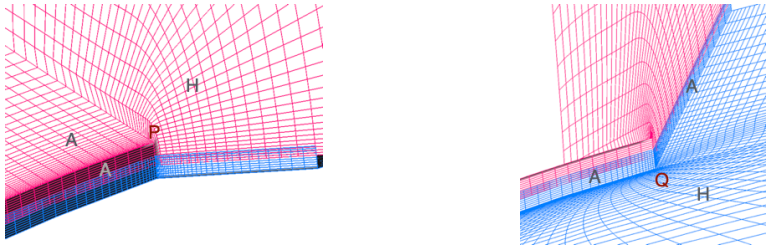
Fig. 7 Junction with two nodes in close proximity where each node has the same number of connected sharp convex and concave edges. (a) Edges connected to node P and node Q. (b) Surface mesh constructed to cover both nodes P and Q (A = algebraic part, H = hyperbolic part).

Let $\Delta\theta_e$ be the angle between the face unit normals on each side of an edge. The edge is considered sharp if $\Delta\theta_e > 30$ degrees. It is convex if the normals are pointing away from each other, and concave if the normals are pointing towards each other. A single node mesh can be used to cover two nodes in close proximity if both nodes have the same number of convex and concave sharp edges. Otherwise, if the number of sharp convex and concave edges connected to the two nodes are different, two specially constructed node meshes are still needed.

Fig. 8a shows a junction between the finite-thickness trailing edge of a nacelle and a pylon where node P has one convex and two concave sharp edges, while node Q has one concave and two convex sharp edges. The two surface meshes constructed using various algebraic and hyperbolic parts to cover the two nodes are shown in Fig. 8b and 8c.



(a)



(b) (c)

Fig. 8 Junction with two nodes in close proximity where each node has a different number of connected sharp convex and concave edges. (a) Edges connected to node P and node Q. (b) Surface mesh constructed to cover node P. (c) Surface mesh constructed to cover node Q (A = algebraic part, H = hyperbolic part).

C. Volume Mesh Overlap Enhancement

All near-body volume meshes in the auto-meshing scheme are grown hyperbolically in the normal direction to the surface independently of each other. For J and K index boundaries that do not lie along a symmetry plane, a splay boundary condition is applied in the hyperbolic marching scheme. Unfortunately, this does not guarantee proper overlap

between neighboring volume meshes. Proper overlap means neighboring meshes have $2 \times N_f$ cells overlap in order to avoid orphan points where N_f is the number of fringe point layers. Suppose proper overlap has been established between neighboring surface grids, a new scheme is introduced to automatically enhance proper volume mesh overlap between neighboring meshes in the normal direction.

The splay boundary condition has a splay control parameter $0 \leq \lambda < 1$ that correlates to the amount of outward bending from the interior. For each grid point along the mesh boundary, a local in/out dividing plane can be constructed with unit normal determined by the cross product of the local unit surface normal and the unit vector along the mesh boundary at the point (Fig. 9a). If all grid points on the L-direction ray from the red surface grid point in Fig. 9a stays on this plane or on the side of the plane opposite to the mesh interior, overlap with a neighboring mesh is guaranteed along a current mesh boundary if all points along the current mesh boundary possess this property. The effect of using different mesh boundary splay values on the L-direction ray from the red grid point is illustrated in Fig. 9b. In this case, a splay value of 0.04 results in all grid points on the L-direction ray from the red grid point to be located on the outside of the local dividing plane. Note that for a different grid point on the same mesh boundary, a different splay value may be needed to achieve the same outcome.

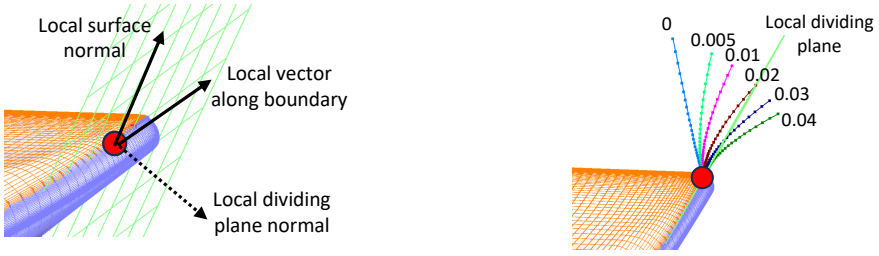


Fig. 9 Splay boundary condition at surface mesh boundary. (a) Local normal dividing plane at grid point (red circle) on surface mesh boundary. (b) Normal L-direction ray for different splay values.

For each L-direction ray from a surface mesh boundary (constant J or K), the fraction F_o of grid points that lie on or outside of the local dividing plane is computed. The splay parameter λ on this boundary is gradually increased until $F_o = 1$ or if negative Jacobians are found. Currently, a single λ value is used to control all points on the boundary. Most grid boundaries show benefits on improving volume mesh overlap using this scheme. If $F_o = 1$ is achieved for all boundaries of all meshes, adjacent volume mesh overlap is guaranteed everywhere. In practice, $F_o = 1$ cannot be achieved without compromising grid quality and resulting in negative Jacobians/cell-volumes.

Orphan points without and with the automatic splay adjustment scheme for two cases are shown in Fig. 10. For both cases, a significant reduction in the number of orphan points is achieved when the automatic splay adjustment scheme is activated.

D. Bodies in Relative Motion

CFD analyses of configurations with multiple bodies in relative motion, where some bodies have multiplicity greater than one, are becoming more common. Examples include simulations of VTOL vehicles with rotors and propellers, and rocket stage separation. These problems require significant pre-processing time prior to starting a flow solution. Typically, the grid system as well as inputs for the flow solver and aerodynamic loads computation are generated independently for each body. The contents of these files then have to be assembled into corresponding files for the final configuration accounting for component multiplicity and grid index shifting.

A new capability called "combine-bodies" has been developed in EPOGS to enable rapid assembly of input files needed to begin Overflow flow solution computation for such problems using the mixed-mode approach. Suppose there are N unique disjoint bodies in a configuration where each body has multiplicity μ_i with $i = 1, .., N$. For each individual body, the following files are first created using EPOGS or any other tool: volume grid system file, domain connectivity data file, X-ray map file containing an X-ray map for the entire body plus X-ray maps for off-body component box grids, flow solver input file with boundary conditions, and component aerodynamic loads computation FOMOCO input file. For a body with $\mu_i > 1$, data for only one instance of the body are contained in these files.

The combine-bodies function takes the above files, multiplicity and name for each body, and creates the corresponding files for the total configuration. With body multiplicity accounted for, operations automated include combining the grids, shifting the donor stencil indices, X-ray map body indices, shifting of grid number indices in the FOMOCO input

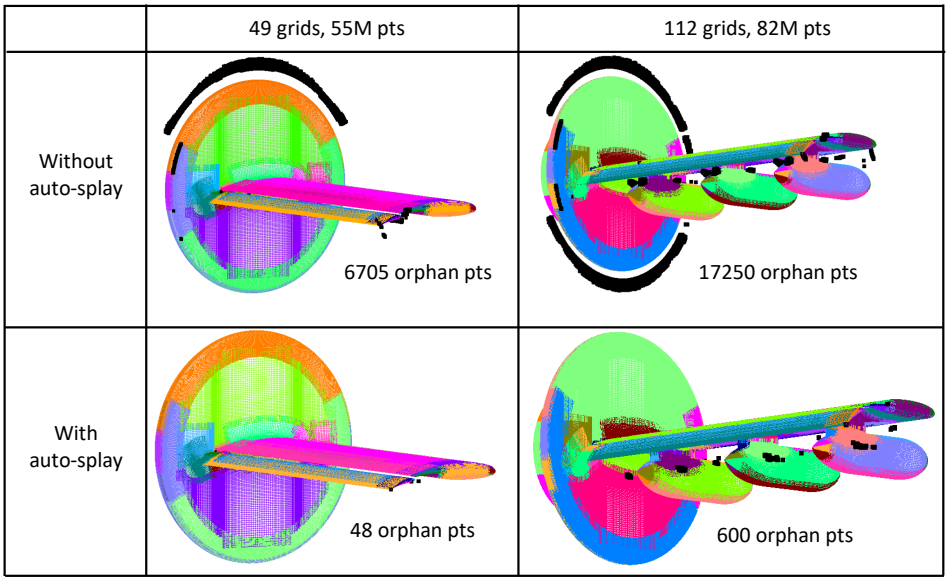


Fig. 10 Orphan points for two cases where volume meshes are grown without and with the automatic splay adjustment scheme (black symbols indicate orphan points).

file, and automatic generation of meaningful grid and component names in the Overflow and FOMOCO input files, respectively. An example using this capability is illustrated in Section V.D.

V. Recent Applications

The EPOGS software toolset has been tested on a variety of aerospace configurations. In this section, a sample of these test cases is presented: various geometries from the AIAA Drag Prediction Workshop 8 (DPW8) and Aeroelastic Prediction Workshop 4 (AePW4), a DPW8 wind tunnel sting effects study, the Tiltwing Test Rig (TWTR) experiment [37], and the VAERA fire-fighting drone [38, 39]. These cases serve to demonstrate the software capability on a variety of geometries with different topological features, as well as the capability to create meshes at different resolutions under a large range of global parameter settings. The first three cases contain multiple related configurations. For all cases shown, significant reductions in turn-around time on CFD analyses are demonstrated using the EPOGS workflow compared to a completely manual meshing approach.

A. AIAA Drag Prediction Workshop 8 (DPW8) & Aeroelastic Prediction Workshop 4 (AePW4)

The AIAA Drag Prediction Workshop 8 and Aeroelastic Prediction Workshop 4 will be held at the AIAA Aviation Forum 2026. There are five different configurations shown by the list below. The first two configurations include different wing-loaded geometries under different angles of attack (α).

1) The Common Research Model (CRM) wing-body with the wing loaded under 8 different angles of attack (8 different geometries).
2) The Japan Aerospace Exploration Agency (JAXA) wing-body-tail at 4 different angles of attack (4 different geometries).
3) The JIG wing-body.
4) The JIG wing-body-tail.
5) The JIG wing-body-nacelle-pylon.

A summary of the different cases and configurations required by the workshop is given in Table 3 where each row represents a different geometry and 'x' indicates where a grid system is needed. The EPOGS toolset is used to create all 40 overset grid systems. For configuration 1 (at $\alpha = 2.5°$ and $3.0°$), configuration 2 (at $\alpha = 4.84°$), configurations 3 and 5, a 6-level mesh family (L1 - L6) is created. For all other geometries, a Level 3 (L3) grid system is created.

A summary of the EPOGS global mesh control parameters used to create the mesh families is given in Table 4. For the JIG wing/body mesh family, these parameters are selected to target a total volume mesh point count of about 10 million (M) for Level 1 and 1 billion (B) for Level 6. This translates to a factor of about 2.5 to 3.0 $\times$ point count increase from one level to the next. Surface grids for three representative grid systems are shown in Fig. 11.

Table 3 Configurations and overset grid systems created by EPOGS for AIAA DPW8/AePW4 (x=grid systems needed by the workshop).

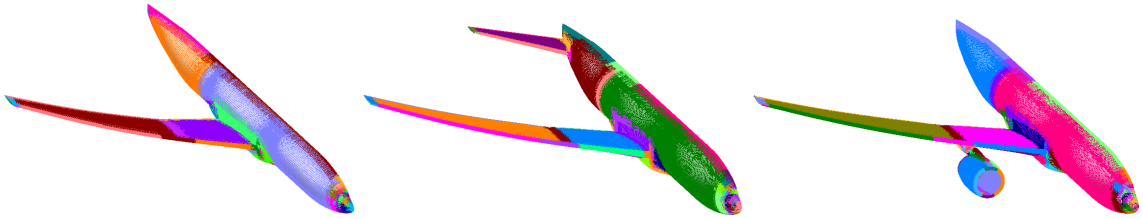
Geometry	α (deg.)	L1	L2	L3	L4	L5	L6
CRM wing/body	2.50	x	x	x	x	x	x
	2.75			x			
	3.00	x	x	x	x	x	x
	3.25			x			
	3.50			x			
	3.75			x			
	4.00			x			
	4.25			x			
JAXA wing/body/tail	1.22			x			
	2.29			x			
	4.84	x	x	x	x	x	x
	5.89			x			
JIG wing/body	0.0	x	x	x	x	x	x
JIG wing/body/tail	0.0			x			
JIG wing/body/nacelle/pylon	0.0	x	x	x	x	x	x

Table 4 EPOGS global mesh control parameters for creating DPW8/AePW4 mesh families.

L	$a_{\max}$ (deg.)	$\Delta s_{\max}$ (in.)	$c_{\max}$	$np_{\min}$	$sr_{\max}$	$\Delta s_{\text{wall}} \times 10^{-4}$ (in.)	$klayer$
1	5.0	15.0	0.5	9	1.19	5.423	2
2	4.5	10.0	0.5	13	1.135	3.615	3
3	4.0	6.667	0.5	17	1.10	2.410	4
4	3.5	4.444	0.5	21	1.07	1.607	5
5	3.0	3.0	0.1	25	1.05	1.071	6
6	2.5	2.0	0.1	29	1.04	0.704	7

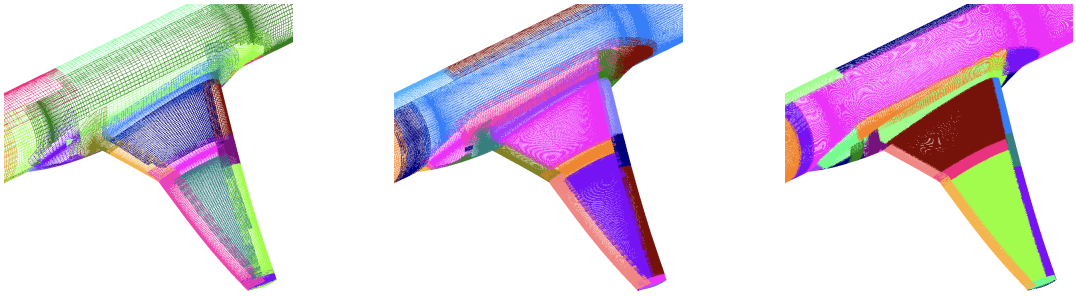
A summary of the mesh statistics for the CRM wing/body $\alpha = 3.0^\circ$ family is given in Table 5. Surface meshes for three levels of the CRM wing/body $\alpha = 3.0^\circ$ family are shown in Fig. 12. Run times in the tables are given for code executions on a Linux workstation with an AMD Ryzen Threadripper PRO 7985WX processor using 64 OpenMP threads. Repair times in the tables refer to the wall clock time needed for mesh overlap adjustments to remove orphan points (about 0% to 10% of meshes), and for removal of negative Jacobians through adjustments of hyperbolic volume grid generation smoothing parameters (about 0% to 1% of meshes). It took about one to two hours of wall clock time to create a Level 1 system, while it took under one day for a Level 6 system. The number of volume grid points varies from 11.8M points for a Level 1 system to 1.1B points for a Level 6 system.

To verify that these meshes are appropriate, flow solutions for the Level 1, 3 and 6 grid systems of the CRM wing/body $\alpha = 3.0^\circ$ case are computed using Overflow at a freestream Mach number of 0.85, Reynolds number per inch of 18129.08, and a freestream temperature of $508^\circ R$. For all three grid levels, steady-state convergence was achieved, with at least five-orders of magnitude residual drop and variations in the loads of less than 1 drag count or 0.0001. A comparison of the convergence of the three cases for the residual, lift, drag and pitching moment coefficients is shown in Fig. 13. As expected, convergence of the residual takes longer as the mesh is refined (note that the L1 and L3 scales are on the lower side, and the L6 scale is on the upper side of the plots). Also shown in the figure are the lower and upper bounds from eight datasets for a level 3 mesh using different grid and solution technologies from DPW7. The level 3 and 6 results from the EPOGS meshes sit nicely within those bounds.



(a) (b) (c)

Fig. 11 Sample EPOGS-created surface grid systems from DPW8/AePW4. (a) CRM wing/body $\alpha = 3.0°$. (b) JAXA wing/body/tail $\alpha = 4.84°$. (c) JIG wing/body/nacelle/pylon $\alpha = 0.0°$.

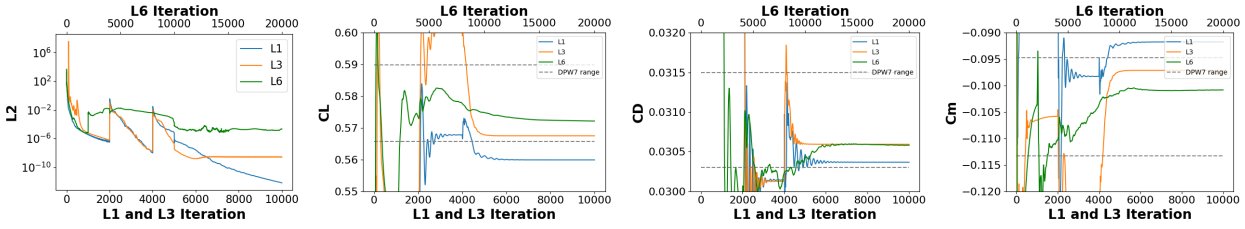


(a) (b) (c)

Fig. 12 Sample levels from mesh family for CRM wing/body ($\alpha = 3.0°$). (a) Level 1. (b) Level 3. (c) Level 6.

Table 5 Grid system statistics for CRM wing/body $\alpha = 3.0°$ mesh family.

L	# Near-body Grids	# Vol Pts (M)	Run Time (min.)	Repair Time (hr.)
1	76	11.8	0.6	1-2
2	70	26.7	1.2	1-2
3	71	66.4	2.7	1
4	74	174.9	11.5	2-3
5	74	469.7	29.2	2-3
6	75	1117.6	84.8	5



(a) (b) (c) (d)

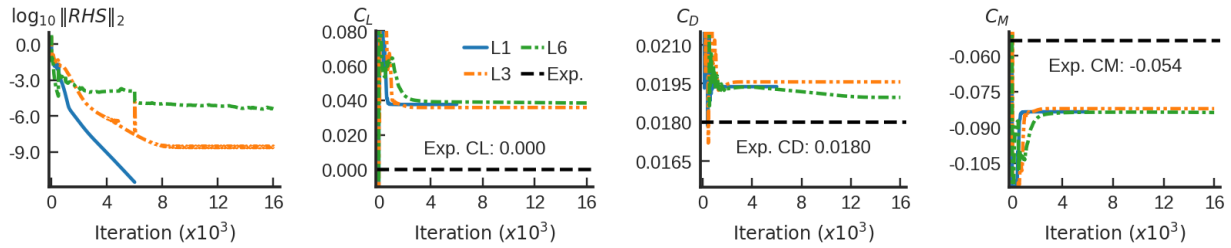
Fig. 13 Comparison of convergence for mesh levels 1, 3 and 6 for CRM wing/body at $\alpha = 3.0°$. Lower and upper bounds of level 3 results from several different data sets computed using different mesh and solver schemes from DPW7 are shown by the dotted grey lines. (a) Residuals. (b) Lift coefficient. (c) Drag coefficient. (d) Pitching moment coefficient.

Table 6 Grid system statistics for JIG wing/body/nacelle/pylon mesh family.

L	# Near-body Grids	# Vol Pts (M)	Run Time (min.)	Repair Time (hr.)
1	135	20.1	0.9	4
2	128	47.7	1.7	4
3	129	112.4	5.3	4
4	133	306.8	14.5	8
5	132	825.7	44.0	6
6	133	2041.0	146.6	8

Mesh statistics for the JIG wing/body/nacelle/pylon family is given in Table 6. A Level 1 system for this case takes 20M points while a Level 6 system takes 2B points. To verify that these meshes are appropriate, flow solutions are computed on the grids using Overflow using the same freestream conditions as the CRM wing/body case above. The solutions were computed using the third-order HLLE++ (IRHS=6) scheme, and advanced using improved SSOR (ILHS=26) with Global Linear Solve option. CFL ramping was also introduced to assist in accelerating the convergence for the Level 1 and 3 cases, but it was not used for Level 6 due to the overly aggressive ramping. The turbulence was modeled using RANS with SA-RC-QCR2000. Fig. 14 shows the residual and integrated loads convergence for the L1, L3, and L6 grids at $\alpha = -1.0°$. This angle-of-attack was chosen because the specific geometry in question corresponds to no loading on the wing, and the experimental data for this angle of attack is closest to being unloaded with effectively zero lift.

For all three grid levels, steady-state convergence was achieved, with at least five-orders of magnitude residual drop and variations in the loads of less than 1 drag count or 1.0E-4. All loads appear to initially show poor agreement with the experimental data. However, from the sting effect results in Section V.B, it is expected that including the sting in the CFD computations (at low α) has the effect of reducing the lift and drag coefficients, and raising the pitching moment coefficient. This would move the current results to be in better agreement with the experimental data.



(a) (b) (c) (d)

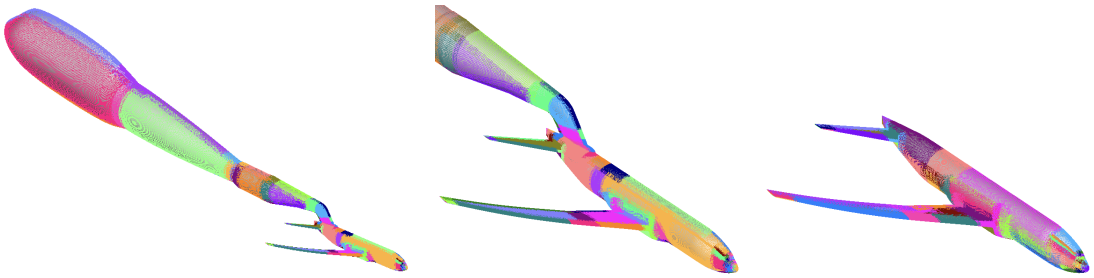
Fig. 14 Comparison of convergence for mesh levels 1, 3 and 6 for JIG wing/body/nacelle/pylon case. Experimental data at $\alpha = -1.053°$ is shown by black dotted line. (a) Residuals. (b) Lift coefficient. (c) Drag coefficient. (d) Pitching moment coefficient.

B. DPW8 Wind Tunnel Sting Effects Study

The workshop mentioned in the previous section also includes a study on the effects of the wind-tunnel sting. Attempting the complete study is beyond the scope of the current paper, but two cases are chosen here to demonstrate the capabilities of EPOGS. Both cases are in wind tunnel 2.7% model scale with the wing deflected at $\alpha = 2.5°$. The first case contains the wing, body, tail and sting (WBTS) while the second case contains just the wing, body and tail (WBT). EPOGS grids are created using Level 3 grid parameters (Fig. 15). There are 65 million grid points in the WBT system and 98 million grid points in the WBTS system. EPOGS run time for the WBTS case is about 17.5 min while that for the WBT is about 11.5 minutes on a Mac M2 laptop. Each grid system requires about 2 hours of repair time.

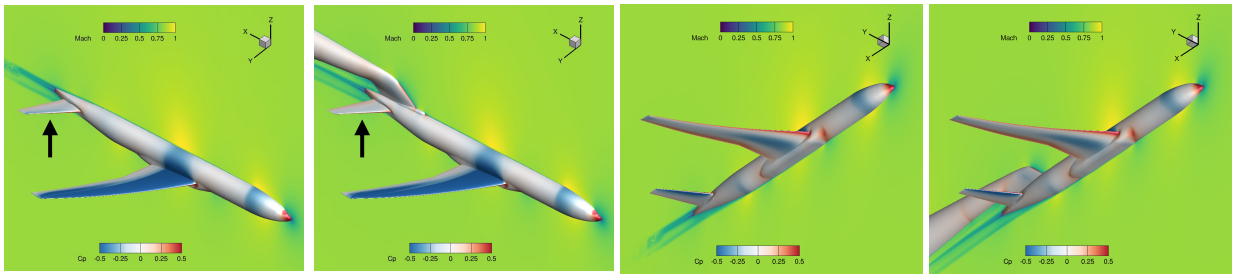
To quantify the sting effects, Overflow solutions are completed using SSOR, HLLC, and the SA turbulence model with the rotational correction and QCR. Convergence is accelerated with full-multigrid, global linear solve, and CFL ramping.

Fig. 16 shows the top-down isometric views of both WBT and WBTS surface pressure coefficient. Effects of the sting on the surface pressure on the upper surface of the tail can be seen by the lower pressure region (blue) in the



(a) (b) (c)

Fig. 15 Surface grids for wind tunnel sting effects study for wing at $\alpha = 2.5°$. (a) Wing/body/tail/sting (far view). (b) Wing/body/tail/sting (near view). (c) Wing/body/tail.

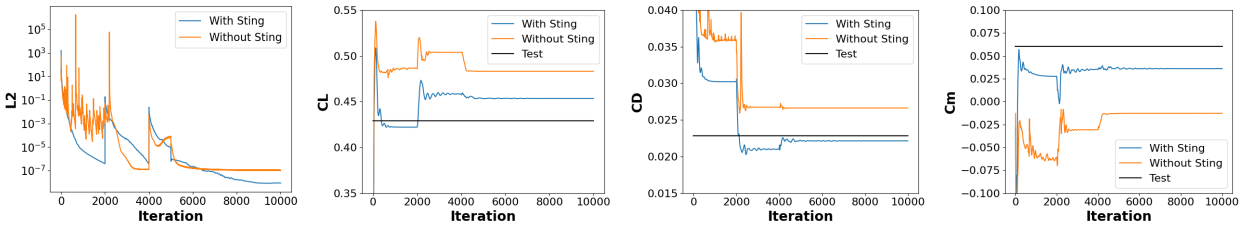


(a) (b) (c) (d)

Fig. 16 Surface pressure coefficient for sting effects study for wing/body/tail at $\alpha = 2.5°$. (a) Sting-off top iso view. (b) Sting-on top iso view. (c) Sting-off bottom iso view. (d) Sting-on bottom iso view.

no-sting case (Fig. 16a) compared to grey for the sting case (Fig. 16b) . The convergence histories of the residuals and aerodynamics loads in the simulations are compared to test data from the NTF run 92 results, interpolated to 2.5° angle of attack (Fig. 17). It is found that the solution with the sting is consistently closer to the wind tunnel test data. Quantified sting effects are given in Table 7, showing the significant changes to the force and moment coefficients due to the presence of this large sting when compared to the no-sting model.

An Overflow simulation for the case without the sting is also computed with the Overflow 2.5.1 GPU version. The force and moment coefficients are consistent with the CPU version results. The CPU version is computed on a single Turin node and takes three hours wall time to complete. The GPU version is computed on 2 Skylake V100 nodes and takes one hour wall time to complete.



(a) (b) (c) (d)

Fig. 17 Convergence plots for wing/body/tail wind tunnel model sting effect study. (a) Residuals. (b) Lift coefficient. (c) Drag coefficient. (d) Pitching moment coefficient.

Table 7 Comparison of converged aerodynamic loads for sting-off and sting-on wing/body/tail case at $\alpha = 2.5°$.

	Sting-off	Sting-on	Experiment	Sting-on - Sting-off
C_L	0.48316	0.45347	0.41556	-0.02969
C_D	0.02663	0.02214	0.02226	-0.00449
Cm	-0.01278	0.03604	0.06442	0.04882

C. Tiltwing Test Rig (TWTR)

In support of NASA's Tiltwing Test Rig experiment [37], pre-test [19] and post-test CFD analyses of various configurations are performed. The full geometry consists of a wing and a flap mounted on top of a splitter plate sitting on the ground plane. For the pre-test analyses, components connected to the wing and flap include five brackets, three sets of pylon and nacelle, and three rotor-blade sets with five blades on each rotor. In the actual experiment, only two sets of pylon, nacelle and rotor-blades were tested which are matched by the post-test analyses grid systems.

Grid systems are generated using EPOGS to support the following pre-test studies while flow solutions are computed using Overflow [19].

1) A rotor modeling study using a clean wing (no flap, no brackets, no pylons, no nacelles) comparing solutions from blade-resolved runs versus rotor actuator disk model runs.
2) A component build-up study by comparing solutions from five different configurations (Fig. 18):
    a) clean-wing,
    b) wing and flap,
    c) wing, flap and five brackets,
    d) wing, flap, five brackets, three sets of pylon and nacelle,
    e) wing, flap, five brackets, three sets of pylon, nacelle and rotor blades with each set containing five blades

    A disk model is used to represent each rotor for cases (a)-(d).
3) A flap deflection study by comparing solutions from three different configurations with flap deflection angle set to 20°, 40°, and 60° (Fig. 19). Each configuration consists of brackets, pylons and nacelles. A disk model is used to represent each rotor.

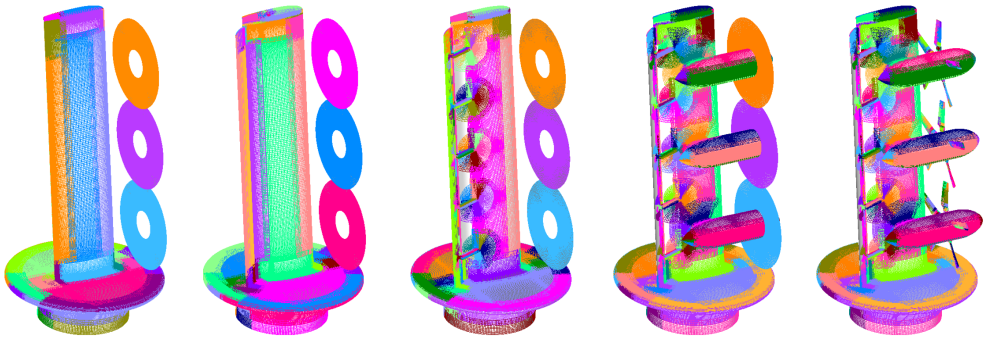


Fig. 18 EPOGS-generated meshes for TWTR component build-up study. Rotors modeled using actuator disks in first four cases. (a) Clean wing and splitter plate. (b) Wing, flap and splitter plate. (c) Wing, flap, brackets and splitter plate. (d) Wing, flap, brackets, pylons, nacelles and splitter plate. (e) Wing, flap, brackets, pylons, nacelles, rotor blades and splitter plate.

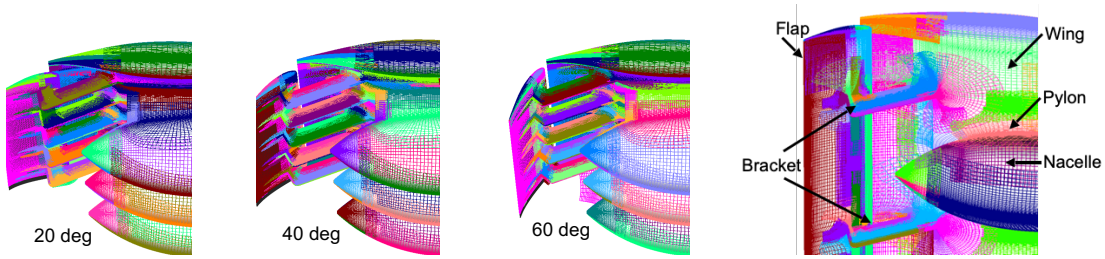
A summary of the mesh statistics for the component build-up study is given in Table 8. The total meshing time, including run and repair time on an M2 Mac laptop, varies from about 15 minutes for the baseline case with just the wing and splitter plate, to about 10 hours for the all-components case with baseline, flap, brackets, pylons, nacelles, and rotor blades. Typical repair tasks needed include mesh overlap improvements (from 0% to about 10% of meshes) and removal of negative Jacobians (from 0% to 1% of meshes). Meshing time needed for the entire component build-up study is accomplished in under one week. An estimate for the manual meshing effort needed to complete the same study would be about one month.

Subsequent to the pre-test CFD analyses, the Tiltwing geometry was tested in NASA Langley's 14- by 22-Foot Subsonic Tunnel. The goal of the test was to assess the aerodynamic performance of the tiltwing concept, with special interest in the performance characteristics of the wing as it transitioned between forward flight and hover. A series of simulations was performed using the Overflow flow solver that used the same geometry and run conditions as the experiment to verify that meshes generated with EPOGS (Fig. 20) would produce reasonable results. Simulations shown in this paper were performed with a freestream velocity of 20 knots, rotor RPM of 4000, disk loading of 14 psf, and a freestream temperature of 84.64 °F. The geometry consisted of the clean wing, two nacelles, the splitter plate, and the rotor blades. In the Overflow simulations, the rotor blades were approximated using the Rotor Disk boundary condition within the solver. The simulations were also run using the steady-state RANS mode within Overflow.

The first comparisons are the lift and drag versus the tilt angle (Fig. 21). The tilt angle is the same as the angle of

Table 8 Mesh statistics for TWTR component build-up study. Baseline contains wing and splitter-plate on ground plane (M=million, m=minutes, h=hours).

Components	# Grids	# Volume points (M)	Run Time	Repair Time
Baseline	40	21.4	10 m	5 m
Baseline, flap	66	37.4	15 m	2.5 h
Baseline, flap, brackets &	166	69.5	22 m	6 h
Baseline, flap, brackets, pylons, nacelles	232	98.6	30 m	10 h
Baseline, flap, brackets, pylons, nacelles, blades	511	156	35 m	10 h

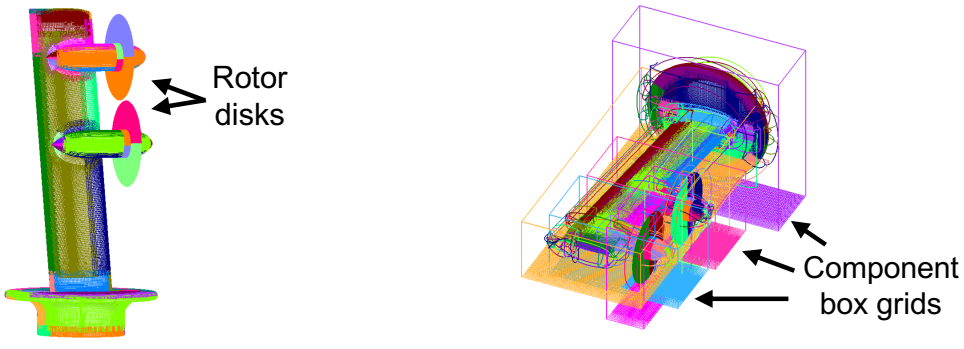


(a) (b) (c) (d)

Fig. 19 Meshes for TWTR flap deflection study. (a) 20°. (b) 40°. (c) 60°. (d) Close-up view of bracket intersections with wing and flap.

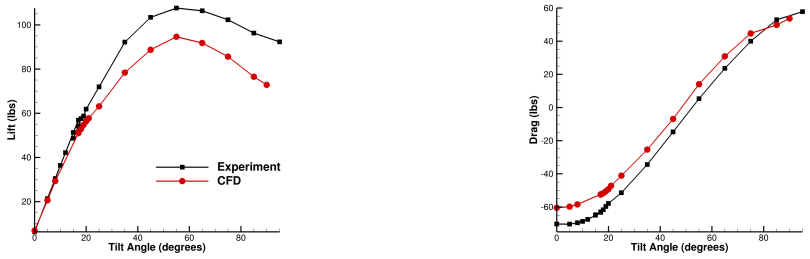
attack of the wing. This means a 0° tilt angle is the cruise configuration like a typical aircraft and a 90° tilt angle is in hover like a helicopter. The lift versus tilt angle plot shows that at lower tilt angles, roughly those below 20°, the CFD is matching the experiment closely. After that point, the CFD significantly under predicts the lift. This is not unexpected since previous work with high lift geometries showed that RANS is not able to accurately predict the physics when there are large regions of separation [40]. A positive aspect of this is that the trend in the lift curve from the experiment is still followed by the CFD, but with a large offset in the value of the predicted lift at higher tilt angles. Interestingly, the drag versus tilt angle plot has an inverse trend to the lift, where the drag is over predicted at the lower tilt angles and moves closer to matching the experiment as the tilt angle increases.

To further examine where the differences between the simulation and the experiment originate, the pressure coefficient Cp through a cross section of the wing was examined (Fig. 22). In this paper, only a single cross section, located halfway between both nacelles, is shown at just two different tilt angles for brevity. At the lower tilt angle, the pressure along the upper surface of the wing is matching closely between the experiment and the CFD. Along the lower surface, the CFD is vastly under-predicting the pressure. This may be partially caused by slight differences between the thrust prescribed in the Rotor Disk model and the thrust reported in the wind tunnel data. An improved explanation of the lower surface behavior may be achievable when the blade-resolved computations are completed in planned future runs. At the higher tilt angle, the current CFD results match more closely with the experiment along the lower surface



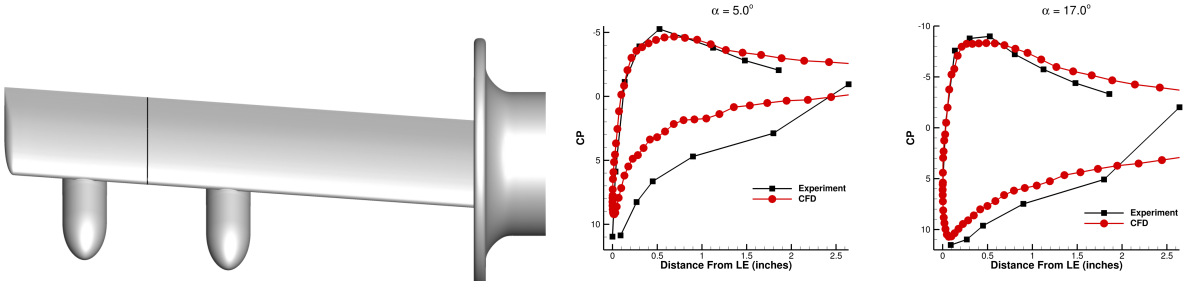
(a) (b)

Fig. 20 TWTR grids for post-test analyses using the Rotor Disk boundary condition to model the rotor blades. (a) Surface grids. (b) Volume grids.



(a) (b)

Fig. 21 TWTR tilt angle study. (a) Lift comparison. (b) Drag comparison.



(a) (b) (c)

Fig. 22 TWTR surface Cp along constant span cut. (a) Cut location. (b) $\alpha = 5°$. (c) $\alpha = 17°$.

of the wing while still under-predict the pressure on the upper surface. This would partially explain the trends seen in the drag versus tilt angle plot. While these results are still preliminary, it is encouraging to see that the automatically generated meshes are producing results that are comparing reasonably well to the experiment. Complete results from the post-test CFD analyses will be shown in a future paper.

D. VAERA Drone

The VAERA (VTOL Analysis for Emergency Response Applications) fire-fighting drone [38, 39] is a test case consisting of a central body with four booms where each boom ends in a hub and a rotor with two blades. Of the four rotors, two contain counter-clockwise (CCW) rotating blades, while the other two contain clockwise (CW) rotating blades. This problem is treated as one with three disjoint bodies in relative motion where first body is the central body with multiplicity 1; the second body is a single CCW blade with multiplicity 4; while the third body is a single CW blade with multiplicity 4 (Fig. 23).

Grids and input files for the central body are created automatically using EPOGS in about 10 minutes where no mesh repairs are needed. Using the default inputs, there are no volume grids with negative Jacobians, and volume mesh domain connectivity resulted in only 6 isolated orphans points. Grids and input files for the CCW and CW blades are created using the Bladegen tool [41] and a simple script based on the Chimera Grid Tools Script Library [27] generated all necessary files for the blades in about 10 minutes. Assembly of the input files for the three-body configuration is accomplished using the combine-bodies option in EPOGS in about 2 minutes. Construction of the initial transforms input using GMP (Geometry Manipulation Protocol) [42] to position the blades, and subsequent mixed-mode domain connectivity between the near-body and off-body grids using Overflow X-ray hole cuts takes about 30 minutes. The total wall-clock time to prepare the final three-body relative motion configuration for flow solution computation takes just under one hour. This test case marks the first time that EPOGS is able to complete the entire process with no repairs in mesh quality or overlap needed. The combined grid system is shown in Fig. 24. This problem would have taken one to two days to generate all the files needed to run the flow solver using a manual approach.

VI. Summary and Conclusions

This paper summarizes the overset structured mesh generation automation scheme under development since 2018. Starting with a BRep solid representation of the geometry, all files needed to begin flow solution computation are auto-generated. These include the volume mesh system, domain connectivity data, boundary conditions in flow solver input files, and grid indices data in the input file for component-based aerodynamic loads computation.

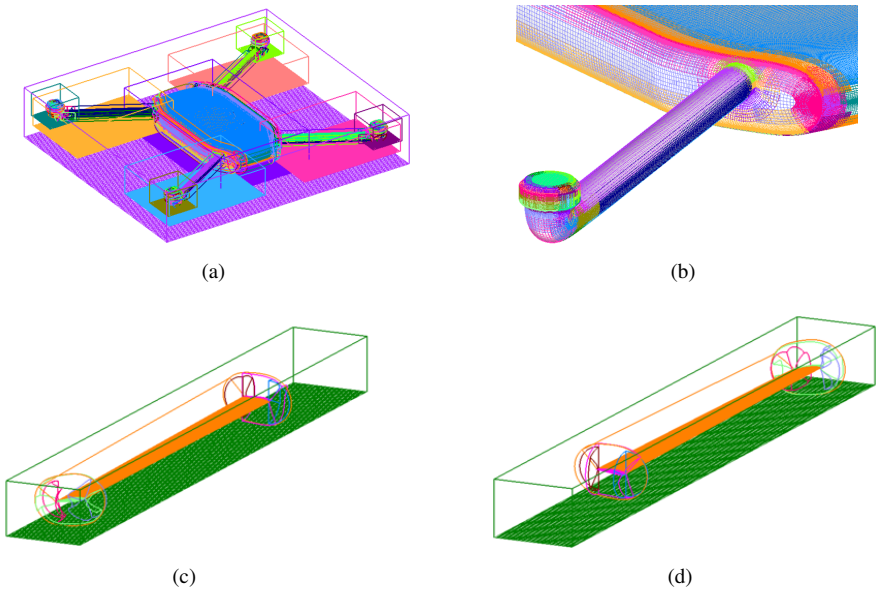


Fig. 23 VAERA drone three-body relative motion test case. (a) Main body volume meshes. (b) Close-up view of boom and rotor hub surface meshes. (c) Counter-clockwise (CCW) blade volume meshes. (d) Clockwise (CW) blade volume meshes.

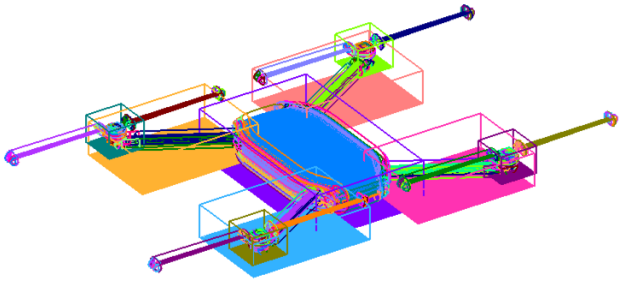


Fig. 24 Combined volume grid system for VAERA drone with three disjoint bodies in relative motion: main body (multiplicity 1), CCW blade (multiplicity 4), CW blade (multiplicity 4).

The automation scheme has been implemented in the EPOGS software toolset (EGADS2SRF and POGS) which is the first and only software of its kind to attempt automation of the steps needed prior to running the flow solver for overset structured mesh analysis. A small number of global parameters, some with automatic scaling to local component sizes, are used to control various mesh characteristics such as maximum grid spacings and maximum stretching ratios. Stop and restart check points are available in the software at key steps in the workflow: surface mesh system, near-body volume mesh system, off-body volume mesh system, and domain connectivity. The automation scheme significantly reduces the time required to create the grid system, from weeks to days, and sometimes even down to hours. However, manual repairs on mesh overlap and negative Jacobians on a small number of grids are still typically needed. This is accomplished by substituting the defective grids with manually created ones at one of the restart check points.

Several recent advances in the automation scheme are also presented. Special treatments are developed for node meshes at cusp and trident nodes, as well as at junctions involving two nodes in close proximity (around finite-thickness trailing edges). Mesh overlap between adjacent near-body hyperbolic volume grids is enhanced by an automatic splay adjustment scheme based on a local normal dividing plane. Rapid assembly of input files for a configuration involving multiple disjoint bodies with multiplicity in relative motion is developed as a software option.

Multiple test cases are presented to illustrate the capability of the software under a variety of geometries as well as a variety of global control parameters. For the DPW8/AePW4 cases, 40 mesh systems were generated for 15 different geometries. For four of the geometries, a mesh family with six levels of refinement is also created. A Level-1 system with 11.8M - 20M points is created in 1 to 2 hours (including run and repair time), while a Level-6 system with 1.1B - 2.0B points is created in about one day or less. A manual approach to generate one mesh family would have taken several weeks.

Two grid systems are tested in the sting effects study on the wing/body/tail geometry at $\alpha = 2.5°$. Flow solutions

obtained from the EPOGS grids show that aerodynamic loads obtained from computations that include the sting are closer to the wind tunnel data compared to results from computations without the sting.

For the TWTR pre-test analyses, five mesh systems are created for a component build-up study and three mesh systems are created for a flap deflection study. Using the EPOGS procedure, the grid systems needed for each of these two studies is created in under one week, compared to an estimate of about one month for a completely manual meshing approach. For the TWTR post-test analyses, similar grid systems are created using EPOGS for two rotors instead of three. Preliminary RANS flow solutions show good agreements in aerodynamic loads with wind tunnel data for low tilt angles. At higher tilt angles, RANS computations are inadequate in modeling the flow. The deviations from wind tunnel data are higher but the trends are captured as the tilt angle is increased.

The VAERA Drone test case marks the first time that a complete grid system is created using EPOGS where no manual repairs are needed and the entire process for preparing all input files for the multi-body relative motion problem is completed in under one hour.

Success of the current automation scheme is currently dependent on the quality of the input BRep geometry. Fractured, excessively trimmed, or tiny faces, spurious nodes, and poorly designed u/v parameter lines would degrade the quality and/or prevent completion of the final grid system. Future work is planned to remove many of these limitations by automatic reconstruction of the input BRep face/edge topology. The original topology is frequently not created with considerations for best practices that lead to high quality grids when using the automated EPOGS strategy. Algorithms will be explored for automatic reconstruction of the face/edge topology that is more aligned with surface features and geometry best practices. This new capability would enable EPOGS to be applied to a wider range of geometries with higher quality outcomes, with removal of some of the restrictions discussed in Section III.C.

Acknowledgements

This work is partially funded by the following NASA Projects: Subsonic Vehicle Technologies and Tools (SVTT), Revolutionary Vertical Lift Technology (RVLT), Transformational Tools, and Technologies (TTT), and Subsonic Flight Demonstrator (SFD). Computational resources supporting this work were provided by the NASA High-End Computing (HEC) Program through the NASA Advanced Supercomputing (NAS) Division at Ames Research Center. The authors would like to thank Ben Rider from the Boeing Company for guidance and data provision on the DPW8/AePW4 test cases, Eduardo Solis from Analytical Mechanics Associates (AMA) for help in converting various geometry representations used in this paper into BRep solids, and Jeff Housman and James Koch from NASA Ames Research Center for reviewing the manuscript.

References

[1] Gomez, R. J., Vicker, D., Rogers, S. E., Aftosmis, M. J., Chan, W. M., Meakin, R. L. and Murman, S., "STS-107 Investigation Ascent CFD Support," AIAA Paper 2004–2226, 2004. https://doi.org/10.2514/6.2004-2226.

[2] Ahmad, J. U., Pandya, S. A., Chan, W. M. and Chaderjian, N. M., "Navier-Stokes Simulation of Air-Conditioning Facility of a Large Modern Computer Room," FEDSM 2005-77225, Proceedings of the 2005 ASME Fluids Engineering Division Summer Meeting and Exhibition, Houston, Texas, 2005. https://doi.org/10.1115/FEDSM2005-77225.

[3] Pandya, S., Onufer, J., Chan, W. and Klopfer, G., "Capsule Abort Recontact Simulation," AIAA Paper 2006–3324, 2006. https://doi.org/10.2514/6.2006-3324.

[4] Kiris, C. C., Kwak, D., Chan, W. M., Housman, J. A., "High-Fidelity Simulations of Unsteady Flow Through Turbopumps and Flowliners," *Computers & Fluids*, Vol. 37, pp. 536-546, 2008. https://doi.org/10.1016/j.compfluid.2007.07.010.

[5] Bhagwat, M., Dimanlig, A., Saberi H., Meadowcroft, E., Panda, B. and Strawn, R., "CFD/CSD Coupled Trim Solution of the Dual-Rotor CH-47 Helicopter Including Fuselage Modeling," Proceedings of the American Helicopter Society Aeromechanics Specialist's Conference, San Francisco, 2008.

[6] Kiris, C., Housman, J., Gusman, M., Chan, W. and Kwak, D. , "Time-Accurate Computational Analysis of Ignition Overpressure in the Flame Trench," *Computational Fluid Dynamics Review*, Eds. Hafez, Oshima, Kwak, Publisher: World Scientific, 2010. https://doi.org/10.1142/9789814313377_0015.

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

[8] Housman, J. A., Sozer, E., Moini-Yekta, S. and Kiris, C. C., "LAVA Simulations for the AIAA Sonic Boom Prediction Workshop," AIAA Paper 2014-2008, 2014. https://doi.org/10.2514/6.2014-2008.

[9] Pandya, S. A., Huang, A., Espitia, A., Uranga, A., "Computational Assessment of the Boundary Layer Ingesting Nacelle Design of the D8 Aircraft," AIAA Paper 2014-0907, 2014. https://doi.org/10.2514/6.2014-0907.

[10] Housman, J. A. and Kiris, C. C., "Numerical Simulations of Shock/Plume Interaction Using Structured Overset Grids," AIAA Paper 2015-2262, 2015. https://doi.org/10.2514/6.2015-2262.

[11] Housman, J. A. and Kiris, C. C., "Structured Overset Grid Simulations of Contra-Rotating Open Rotor Noise," AIAA Paper 2016-0814, 2016. https://doi.org/10.2514/6.2016-0814.

[12] Koch, J. R., Housman, J. A., and Hosseini, S. S., "Microjet Angle Sensitivity for Active Flow Control on Multi-Element High-Lift Systems," AIAA Paper 2023–4307, 2023. https://doi.org/10.2514/6.2023-4307.

[13] Meeroff, J. G., Dalle, D. J., Rogers, S. E., Burkhead, A. C., Schauerhamer, D. G., and Diaz, J. F., "Advances in Space Launch System Booster Separation Computational Fluid Dynamics," *J. Spacecraft & Rockets*, Vol. 5, 2024. https://doi.org/10.2514/1.A35807@jsr.2025.Virtual_Collection.issue-5.

[14] Ventura Diaz, P., and Yoon, S., "High-Fidelity Simulations of a Quadrotor Vehicle for Urban Air Mobility," AIAA Paper 2022–0152, Jan. 2022. https://doi.org/10.2514/6.2022-0152.

[15] Ventura Diaz, P., Perez, D. G., and Yoon, S., "Multi-Fidelity Computational Analysis of a Quiet Single-Main Rotor Helicopter for Air Taxi Operations," AIAA Paper 2023–3669, 2023. https://doi.org/10.2514/6.2023-3669.

[16] Perez, D. G., Ventura Diaz, P., and Yoon, S., "High-Fidelity Simiulations of a Tiltwing Vehicle for Urban Air Mobility," AIAA Paper 2023–2282, Jan. 2023. https://doi.org/10.2514/6.2023-2282.

[17] Dumlupinar, E., Housman, J. A., and Duensing, J. C., "Exploring the Impact of Sideslip Angle on Acoustic Predictions of an Open Rotor System Using Scale-Resolving Simulations," AIAA Paper 2024–3679, 2024. https://doi.org/10.2514/6.2024-3679.

[18] Ventura Diaz, P., Yoon, S. and Edquist K., "High-Fidelity Simulations of a Dragonfly Lander Test Article in the National Full-Scale Aerodynamics Complex," AIAA Paper 2025–3665, 2025. https://doi.org/10.2514/6.2025-3665.

[19] Jensen, J. C., Chuen, A. M., and Chan, W. M., "Pre-Test CFD Analyses for the Tiltwing Test Rig Experiment," AIAA Paper 2025–3384, 2025. https://doi.org/10.2514/6.2025-3384.

[20] Buning, P. G., and Pulliam, T. H., "Cartesian Off-Body Grid Adaption for Viscous Time-Accurate Flow Simulations," AIAA Paper 2011-3693, 2011. https://doi.org/10.2514/6.2011-3693.

[21] Buning, P. G., and Pulliam, T. H., "Near-Body Grid Adaption for Overset Grids," AIAA Paper 2016-3326, 2016. https://doi.org/10.2514/6.2016-3326.

[22] Kiris, C. C., Housman, J. A., Barad, M. F., Brehm, C., Sozer, E., Moini-Yekta, S. , "Computational Framework for Launch, Ascent, and Vehicle Aerodynamics (LAVA)," *Aerospace Science and Technology*, Vol. 55, pp. 189-219, 2016. https://doi.org/10.1016/j.ast.2016.05.008.

[23] Chan, W. M., Pandya, S. A. and Haimes, R., "Automation of Overset Structured Surface Mesh Generation on Complex Geometries," AIAA Paper 2019–3671, 2019. https://doi.org/10.2514/6.2019-3671.

[24] Chuen, A. M. and Chan, W. M., "Overlap Preservation Using Loosely-Coupled Boundary Conditions for Body-Fitted Structured Overset Grids," AIAA Paper 2022-0216, 2022. https://doi.org/10.2514/6.2022-0216.

[25] Chan, W. M., Pandya, S. A. and Chuen, A. M., "Automation of Overset Structured Surface Mesh Generation on Boundary Representation Geometries," AIAA Paper 2022–3607, 2022. https://doi.org/10.2514/6.2022-3607.

[26] Chan, W. M., Chuen, A. M. and Jensen, J. C., "Advances in Automation of Overset Structured Volume Mesh Generation and Domain Connectivity," AIAA Paper 2024–4305, 2024. https://doi.org/10.2514/6.2024-4305.

[27] Chan, W. M., "Developments in Strategies and Software Tools for Overset Structured Grid Generation and Connectivity," AIAA Paper 2011-3051, 2011.

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

[29] Rogers, S. E., Suhs, N. E. and Dietz, W. E., "PEGASUS5 : An Automated Pre-Processor for Overset-Grid CFD," *AIAA J.*, Vol. 41, No. 6, pp. 1037-1045, Dec. 2003.

[30] Meakin, R. L., "Object X-rays for Cutting Holes in Composite Overset Structured Grids," AIAA Paper 2001–2537, 2001.

[31] Noack, R. W., "Suggar++: An Improved General Overset Grid Assembly Capability," AIAA Paper 2009–3992, 2009.

[32] Sitaraman, J., Floros, M., Wissink, A. and Potsdam, M., "Parallel Domain Connectivity Algorithm for Unsteady Flow Computations Using Overlapping and Adaptive Grids," *J. Comp. Phys.*, Vol. 229, pp. 4703-4723, Mar. 2010.

[33] Chuen, A. M., Hosseini, S. S., Jensen, J. C. and Chan, W. M., "Aerodynamic Simulations for Complex Geometries Using Automatically Generated Structured Overset Meshes," AIAA Paper 2023–3602, 20232. https://doi.org/10.2514/6.2023-3602.

[34] Khoshniat, M., Stuhne, G. R. and Steinman, D. A., "Relative Performance of Geometric Search Algorithms for Interpolating Unstructured Mesh Data," Medical Imaging Computing and Computer-Assisted Intervention, Springer-Verlag GmbH, Eds. Ellis and Peters, 2003.

[35] Chan, W. M. and Buning, P. G., "User's Manual for FOMOCO Utilities – Force and Moment Computation Tools for Overset Grids," NASA TM 110408, 1996.

[36] Chan, W. M., "The OVERGRID Interface for Computational Simulations on Overset Grids," AIAA Paper 2002–3188, 2002. https://doi.org/10.2514/6.2002-3188.

[37] Whiteside, K. S., Litherland, B. L., "Tiltwing Validation Test: Conceptual Design and Research Objectives," https://ntrs.nasa.gov/api/citations/20240002905/downloads/20240320-TWTATWG-Whiteside.pdf, 2024.

[38] Conley, S., Peters, N., Aires, J., Wagner, L., Kallstrom, K., Withrow-Maser, S., Wright, S., Radotich, M., Chianello, C., Rivera, G., Ewing, G., Anderson, A., Klokkevold, K. and Kluch, E., "VTOL Analysis for Emergency Response Applications (VAERA) - Identifying Technology Gaps for Wildfire Relief Rotorcraft Missions," Sixth Decennial VFS Aeromechanics Specialists Conference, Santa Clara, CA, Feb. 6-8, 2024 (https://legacy.vtol.org/store/product/vtol-analysis-for-emergency-response-applications-vaera-identifying-technology-gaps-for-wildfire-relief-rotorcraft-missions-18511.cfm), Feb. 2024.

[39] Silva, C. and Solis, E., "Conceptual Design of Uninhabited Aircraft for Support of Wildfire Suppression," VFS Vertical Lift Aircraft Design & Aeromechanics Specialists Conference, San Jose, CA, Jan. 27-29, 2026, Jan. 2026.

[40] Rumsey, C. L., Slotnick, J. P., and Woeber, C., "HLPW-4/GMGW-3: Overview and Workshop Summary," AIAA Paper 2022–3295, 2022. https://doi.org/10.2514/6.2022-3295.

[41] Peters, N., Silva, C., and Ekaterinaris, J., "A Data-Driven Reduced-Order Model for Rotor Optimization," *Wind Energy Science*, Vol. 8, No. 7, pp. 1201–1223, 2023. https://doi.org/10.5194/wes-8-1201-2023.

[42] Murman, S., Chan, W. M., Aftosmis, M. J., and Meakin, R. L., "An interface for Specifying Rigid-Body Motions for CFD Applications," AIAA Paper 2003–1237, Jan. 2003. https://doi.org/10.2514/6.2003-1237.