AFLR4 Analysis Interface Module (AIM) Manual

Ryan Durscher
AFRL/RQVC

June 8, 2022

0.1 Introduction

0.1.1 AFLR4 AIM Overview

A module in the Computational Aircraft Prototype Syntheses (CAPS) has been developed to interact with the unstructured, surface grid generator AFLR4 [2] [1].

The AFLR4 AIM provides the CAPS users with the ability to generate "unstructured, 3D surface grids" using an "↩ Advancing-Front/Local-Reconnection (AFLR) procedure." Only triangular elements may be generated, with planned future support of quadrilateral elements.

An outline of the AIM's inputs, outputs and attributes are provided in AIM Inputs and AIM Outputs and AIM Attributes, respectively. The complete AFLR documentation is available at the `SimCenter`.

Example surface meshes:

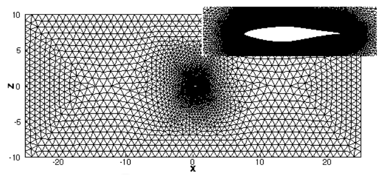

Figure 1 AFLR4 meshing example - 2D Airfoil

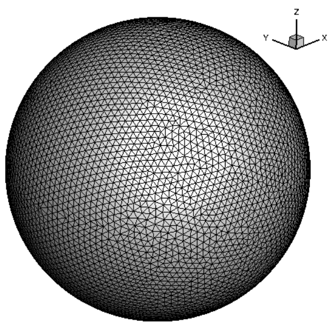

Figure 2 AFLR4 meshing example - Sphere

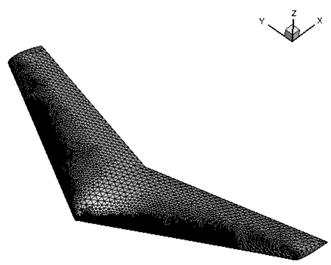

Figure 3 AFLR4 meshing example - Wing

0.1.2 Clearance Statement

This software has been cleared for public release on 05 Nov 2020, case number 88ABW-2020-3462.

0.2 AIM Attributes

The following list of attributes are available to guide the mesh generation with the AFLR4 AIM.

- **AFLR_GBC** [Optional FACE attribute: Default STD_UG3_GBC] This string FACE attribute informs AFLR4 what BC treatment should be employed for each geometric FACE. The BC defaults to the string STD_UG3←_GBC if none is specified.

Predefined AFLR Grid BC string values are:

AFLR_GBC String	Description
FARFIELD_UG3_GBC	farfield surface same as a standard surface except w/AFLR4
STD_UG3_GBC	standard surface
-STD_UG3_GBC	standard BL generating surface
BL_INT_UG3_GBC	symmetry or standard surface that intersects BL region
TRANSP_SRC_UG3_GBC	embedded/transparent surface converted to source nodes by AFLR
TRANSP_BL_INT_UG3_GBC	embedded/transparent surface that intersects BL region
TRANSP_UG3_GBC	embedded/transparent surface
-TRANSP_UG3_GBC	embedded/transparent BL generating surface
TRANSP_INTRNL_UG3_GBC	embedded/transparent surface converted to internal faces by AFLR
FIXED_BL_INT_UG3_GBC	fixed surface with BL region that intersects BL region

Within AFLR4 the grid BC determines how automatic spacing is applied. Their are four basic Grid BC types that are each treated differently.

1. Faces/surfaces that are part of the farfield should be given a FARFIELD_UG3_GBC Grid BC. Farfield faces/surfaces are given a uniform spacing independent of other faces/surfaces with different Grid BCs.

2. Faces/surfaces that represent standard solid surfaces should be given either a STD_UG3_GBC or -STD_←UG3_GBC (BL generating) Grid BC. Standard surfaces are given a curvature dependent spacing that may be modified by proximity checking.

3. Faces/surfaces that intersect a BL region should be given either a BL_INT_UG3_GBC (standard boundary surface) or TRANSP_BL_INT_UG3_GBC (embedded/ transparent surface with volume mesh on both sides) Grid BC. A common example for the BL_INT_UG3_GBC Grid BC is a symmetry plane. Faces/surfaces set as BL intersecting surfaces are excluded from auto spacing calculations within AFLR4 and use edge spacing derived from their neighbors.

4. Surfaces set as transparent surfaces will have a volume mesh on both sides. They can have free edges and can have non-manifold connections to standard solid surfaces and/or BL intersecting surfaces. Vertices in the final surface mesh are not duplicated at non-manifold connections. Transparent surfaces use curvature driven surface spacing as used on standard solid surfaces. However, at non-manifold connections with standard solid surfaces they inherit the surface spacing set on the solid surface they are attached to. They are also excluded from proximity checking. Typical examples of transparent surfaces include wake sheets or multi-material interface surfaces.

- **AFLR4_Cmp_ID** [Optional FACE attribute]
  EGADS attribute AFLR4_Cmp_ID represents the component identifier for a given face/surface. Component IDs are used for proximity checking. Proximity is only checked between different components. A component is one or more CAD surfaces that represent a component of the full configuration that should be treated individually. For example, a wing-body-strut-nacelle configuration could be considered as four components with wing surfaces set to component 1, body surfaces set to component 2, nacelle surfaces set to 3, and store surfaces set to 4. If each component is a topologically closed surface/body then there is no need to set components. If component IDS are not specified then component identifiers are set for each body defined in the EGADS model or topologically closed surfaces/bodies of the overall configuration. Proximity checking is disabled if there is only one component/body defined. Note that proximity checking only applies to standard surfaces. Component identifiers are set by one of three methods, chosen in the following order.

1. If defined by EGADS attribute AFLR4_Cmp_ID then attribute sets component identifier.

2. Else, if multiple bodies are defined in the EGADS model then bodies index is used to set component identifier.

3. Else, component identifiers are set an index based on topologically closed surfaces/bodies of the overall configuration.

- **AFLR4_Isolated_Edge_Refinement_Flag** [Optional FACE attribute: Integer Range 0 to 2]
  Isolated edge refinement flag. If Flag = 0 then do not refine isolated interior edges. If Flag = 1 then refine isolated interior edges if the surface has local curvature (as defined using cier). If Flag = 2 then refine all isolated interior edges. An isolated interior edges is connected only to boundary nodes. Isolated edges are refined by placing a new node in the middle of of the edge. Note that if not set then the isolated edge refinement flag is set to the global value AFLR4_mier.

- **AFLR4_Edge_Refinement_Weight** [Optional FACE attribute: Default 0.0, Range 0 to 1]
  EGADS attribute AFLR4_Edge_Refinement_Weight represents the edge mesh spacing scale factor weight for a given face/surface. Edge mesh spacing can be scaled on a given face/surface based on the discontinuity level between adjacent faces/surfaces on both sides of the edge. The edge mesh spacing scale factor weight set with AFLR4_Edge_Refinement_Weight is used as an interpolation weight between the unmodified spacing and the modified spacing. A value of one applies the maximum modification and a value of zero applies no change in edge spacing. Note that no modification is done to edges that belong to farfield or BL intersecting face/surface.

- **AFLR4_Scale_Factor** [Optional FACE or EDGE attribute: Default 1.0]
  EGADS attribute AFLR4_Scale_Factor represents the AFLR4 mesh spacing scale factor for a given face/edge. Curvature dependent spacing can be scaled on the face/edge by the value of the scale factor set with AFLR4_Scale_Factor.

0.3 AIM Inputs

The following list outlines the AFLR4 meshing options along with their default value available through the AIM interface.

Please consult the AFLR4 documentation for default values not present here.

- **Proj_Name = NULL**
  This corresponds to the output name of the mesh. If left NULL, the mesh is not written to a file.
- **Mesh_Quiet_Flag = False**
  Complete suppression of mesh generator (not including errors)
- **Mesh_Format = "AFLR3"**
  Mesh output format. Available format names include: "AFLR3", "VTK", "TECPLOT", "STL" (quadrilaterals will be split into triangles), "FAST", "ETO".
- **Mesh_ASCII_Flag = True**
  Output mesh in ASCII format, otherwise write a binary file if applicable.
- **Mesh_Gen_Input_String = NULL**
  Meshing program command line string (as if called in bash mode). Use this to specify more complicated options/use features of the mesher not currently exposed through other AIM input variables. Note that this is the exact string that will be provided to the volume mesher; no modifications will be made. If left NULL an input string will be created based on default values of the relevant AIM input variables.
- **ff_cdfr**
  Farfield growth rate for field point spacing.
  The farfield spacing is set to a uniform value dependent upon the maximum size of the domain, maximum size of inner bodies, max and min body spacing, and farfield growth rate. ;
  ff_spacing = (ff_cdfr-1)$*$L+(min_spacing+max_spacing)/2 ;
  where L is the approximate distance between inner bodies and farfield.
- **min_ncell**
  Minimum number of cells between two components/bodies.
  Proximity of components/bodies to each other is estimated and surface spacing is locally reduced if needed. Local surface spacing is selectively reduced when components/bodies are close and their existing local surface spacing would generate less than the minimum number of cells specified by min_ncell. Proximity checking is automatically disabled if min_ncell=1 or if there is only one component/body defined.
- **mer_all**
  Global edge mesh spacing scale factor flag.
  Edge mesh spacing can be scaled on all surfaces based on discontinuity level between adjacent surfaces on both sides of the edge. For each surface the level of discontinuity (as defined by angerw1 and angerw2) determines the edge spacing scale factor for potentially reducing the edge spacing. See erw_ids and erw_list. This option is equivalent to setting erw_ids equal to the list of all surface IDS and the edge mesh spacing scale factor weight in erw_list equal to one. Note that no modification is done to edges that belong to surfaces with a grid BC of farfield (ff_ids) or BL intersecting (int_ids).

- **no_prox**
  Disable proximity check flag.
  If no_prox=False then proximity of components/bodies to each other is estimated and surface spacing is locally reduced if needed.
  If no_prox=True or if there is only one component/body defined then proximity checking is disabled.

- **abs_min_scale**
  Relative scale of absolute minimum spacing to reference length. The relative scale of absolute minimum spacing to reference length (ref_len) controls the absolute minimum spacing that can be set on any component/body surface by proximity checking (see min_ncell).
  Note that the value of abs_min_scale is limited to be less than or equal to min_scale.

- **BL_thickness**
  Boundary layer thickness for proximity checking.
  Proximity of components/bodies to each other is estimated and surface spacing is locally reduced if needed. Note that if the Reynolds Number, Re_l, is set then the BL_thickness value is set to an estimate for turbulent flow. If the set or calculated value of BL_thickness>0 then the boundary layer thickness is included in the calculation for the required surface spacing during proximity checking.

- **Re_l**
  Reynolds Number for estimating BL thickness.
  The Reynolds Number based on reference length, Re_l, (if set) along with reference length, ref_len, are used to estimate the BL thickness, BL_thickness, for turbulent flow. If Re_l>0 then this estimated value is used to set BL_thickness.

- **curv_factor**
  Curvature factor
  For surface curvature the spacing is derived from the curvature factor divided by the curvature.
  Curvature = 1 / Curvature_Radius
  Spacing = curv_factor / Curvature
  The resulting spacing between is limited by the minimum and maximum spacing set by min_scale and max_scale. Note that if curv_factor=0 then surface curvature adjustment is not used.

- **erw_all**
  Global edge mesh spacing refinement weight.
  Edge mesh spacing can be scaled on all surfaces (if mer_all=1) based on discontinuity level between adjacent surfaces on both sides of the edge.
  For each surface the level of discontinuity (as defined by angerw1 and angerw2) determines the edge spacing scale factor for potentially reducing the edge spacing. The edge mesh spacing scale factor weight is then used as an interpolation weight between the unmodified spacing and the modified spacing.
  A value of one applies the maximum modification and a value of zero applies no change in edge spacing. If the global edge mesh spacing scale factor flag, mer_all, is set to 1 then that is equivalent to setting AFLR_Edge_Scale_Factor_Weight on all FACEs to the value erw_all. Note that no modification is done to edges that belong to surfaces with a grid BC of farfield (FARFIELD_UG3_GBC) or BL intersecting. Also, note that the global weight, erw_all, is not applicable if mer_all=0.

- **max_scale**
  Relative scale of maximum spacing to
  reference length. The relative scale of
  maximum spacing to reference length (ref_len)
  controls the maximum spacing that can be set
  on any component/body surface.

- **min_scale**
  Relative scale of minimum spacing to
  reference length. The relative scale of
  minimum spacing to reference length (ref_len)
  controls the minimum spacing that can be set
  on any component/body surface.

- **Mesh_Length_Factor = 1**
  Scaling factor to compute AFLR4 'ref_len' parameter via:

  ref_len = capsMeshLength $*$ Mesh_Length_Factor

  where capsMeshLength is a numeric attribute that
  must be on at least one body and consistent if on multiple bodies.

  ref_len:
  Reference length for components/bodies in grid units. Reference length should
  be set to a physically relevant characteristic length for the configuration
  such as wing chord length or pipe diameter. If ref_len = 0 then it will be set
  to the bounding box for the largest component/body of interest.
  The parameters ref_len, max_scale, min_scale and abs_min_scale are all used to
  set spacing values on all component/body surfaces (those that are not on the
  farfield or symmetry plane-if any).

  max_spacing = max_scale $*$ ref_len
  min_spacing = min_scale $*$ ref_len
  abs_min_spacing = abs_min_scale $*$ ref_len

- **Mesh_Sizing = NULL**
  See Mesh Sizing for additional details.

- **Multiple_Mesh = True**
  If set to True (default) a surface mesh will be generated and output (given "Proj_Name" is set) for each body. When set to False only a single surface mesh will be created. Note, this only affects the mesh when writing to a file.

- **EGADS_Quad = False**
  Apply EGADS quadding to the AFLR4 triangulation.

0.4 AIM Outputs

The following list outlines the AFLR4 AIM outputs available through the AIM interface.

- **Done**
  True if a surface mesh was created on all surfaces, False if not.

- **NumberOfElement**
  Number of elements in the surface mesh

- **NumberOfNode**
  Number of vertices in the surface mesh

- **Surface_Mesh**
  The surface mesh for a link.

0.5 Mesh Sizing

NOTE: Available mesh sizing parameters differ between mesh generators.

Structure for the mesh sizing tuple = ("CAPS Mesh Name", "Value"). "CAPS Mesh Name" defines the caps↩ Mesh on which the sizing information should be applied. The "Value" can either be a JSON String dictionary (see Section JSON String Dictionary) or a single string keyword string (see Section Single Value String)

0.5.1 JSON String Dictionary

If "Value" is a JSON string dictionary (e.g. "Value" = {"edgeDistribution": "Even", "numEdgePoints": 100}) the following keywords (= default values) may be used:

- **bcType = (no default)**
  bcType sets the AFLR_GBC attribute on faces.

  See AFLR_GBC in AIM Attributes for additional details.

- **scaleFactor = (no default)**
  scaleFactor sets the AFLR4_Scale_Factor attribute on faces/edges.

  See AFLR4_Scale_Factor in AIM Attributes for additional details.

- **edgeWeight = (no default) [Range 0 to 1]**
  edgeWeight sets the AFLR4_Edge_Refinement_Weight attribute on faces.

  See AFLR4_Edge_Refinement_Weight in AIM Attributes for additional details.

0.5.2 Single Value String

If "Value" is a single string, the following options maybe used:

- (NONE Currently)

Bibliography

[1] David L Marcum. Unstructured grid generation using automatic point insertion and local reconnection. *The Handbook of Grid Generation*, pages 18–1, 1998. 1

[2] David L. Marcum and Nigel P. Weatherill. Unstructured grid generation using iterative point insertion and local reconnection. *AIAA Journal*, 33(9):1619–1625, Sep. 1995. 1