AFLR3 Analysis Interface Module (AIM) Manual

Ryan Durscher
AFRL/RQVC

April 27, 2024

0.1 Introduction

0.1.1 AFLR3 AIM Overview

A module in the Computational Aircraft Prototype Syntheses (CAPS) has been developed to interact with the unstructured, volumetric grid generator AFLR3 [2] [1].

The AFLR3 AIM provides the CAPS users with the ability to generate "unstructured tetrahedral element grids" using an "Advancing-Front/Local-Reconnection (AFLR) procedure." Additionally, an "Advancing-Normal Boundary-Layer (ANBL) procedure" may be used "to generate a tetrahedral/pentahedral/hexahedral BL grid adjacent to" specified surfaces.

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

Example volumes meshes:

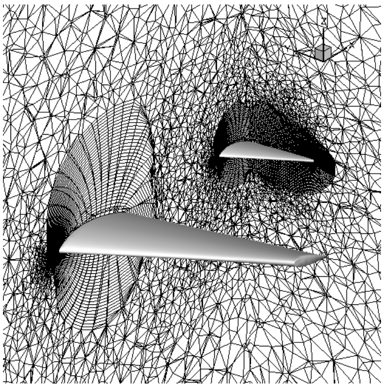

Figure 1 AFLR3 meshing example - Multiple Airfoils with Boundary Layer

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 Inputs

The following list outlines the AFLR3 meshing options along with their default value available through the AIM interface.

- **Proj_Name = "aflr3_CAPS"**
  Output name prefix for meshes to be written in formats specified by Mesh_Format. These meshes are not linked to any analysis, but may be useful exploring meshing parameters.

- **Mesh_Format = NULL**
  Optional list of string mesh formats to generate meshes not linked to analysis.
  Available format names include: "exodus", "fast", "libMeshb", "stl", "bstl", "su2", "tecplot", "ugrid", "vtk", and "bvtk".
  where the "b" prefix indicates binary version.

- **Mesh_Quiet_Flag = False**
  Suppression of mesh generator (not including errors)

- **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.

- **Multiple_Mesh = "SingleDomain"**
  If "SingleDomain": Generate a single volume mesh file is assuming multiple bodies define a single computational domain (i.e. CFD)

  If "MultiFile": Generate a volume mesh file for each body.

  If "MultiDomain": Generate a single mesh file containing multiple volume meshes for each body.

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

- **BL_Initial_Spacing = 0.0**
  Initial mesh spacing when growing a boundary layer that is applied to all bodies (scaled by capsMeshLength).

  Note: Both "BL_Initial_Spacing" and "BL_Thickness" must be non-zero for values to be applied. If "↩
  Multiple_Mesh" is False (default value) these values will not be applied to the largest body (if more than 1 body exist in the AIM), as that body is assumed to be a bounding box (e.g. a farfield boundary in a CFD simulation). Boundary spacing and thickness specified through the use of the "Mesh_Sizing" input (see Mesh Sizing for additional details) will take precedence over the values specified for "BL_Initial_Spacing" and "BL_Thickness".

- **BL_Thickness = 0.0**
  Total boundary layer thickness that is applied to all bodies (scaled by capsMeshLength).
  This is a lower bound on the desired thickness. The height can be limited with "nbl".

  Note: see "BL_Initial_Spacing" and "BL_Max_Layers" for additional details

- **BL_Max_Layers = 10000**
  Maximum BL grid layers to generate.

- **BL_Max_Layer_Diff = 0**
  Maximum difference in BL levels.
  If BL_Max_Layer_Diff $>$ 0 then the maximum difference between the number of BL levels for
  the BL nodes on a given BL boundary surface face is limited to BL_Max_Layer_Diff.
  Any active BL node that would allow the number of levels to be greater is
  terminated.
  If BL_Max_Layer_Diff = 0 then the difference in BL levels is ignored.

- **Surface_Mesh = NULL**
  A Surface_Mesh link.

0.3 AIM Outputs

The following list outlines the AFLR3 AIM outputs available through the AIM interface.

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

- **NumberOfNode**
  Number of vertices in the volume mesh
- **Volume_Mesh**
  The volume mesh for a link

0.4 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)

0.4.1 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:

- **numEdgePoints = 2**
  Number of points along an edge including end points. Must be at least 2.
- **boundaryLayerThickness = 0.0**
  Desired lower bound boundary layer thickness on a face. The minimum thickness in the mesh is is given by meshBLThickness = capsMeshLength ∗ boundaryLayerThickness
- **boundaryLayerSpacing = 0.0**
  Initial spacing factor for boundary layer mesh growth on as face.
  The spacing in the mesh is is given by
  meshBLSpacing = capsMeshLength ∗ boundaryLayerSpacing
- **tessParams = (no default)**
  Face tessellation parameters, example [0.1, 0.01, 20.0]. (From the EGADS manual) A set of 3 parameters that drive the EDGE discretization and the FACE triangulation. The first is the maximum length of an EDGE segment or triangle side (in physical space). A zero is flag that allows for any length. The second is a curvature-based value that looks locally at the deviation between the centroid of the discrete object and the underlying geometry. Any deviation larger than the input value will cause the tessellation to be enhanced in those regions. The third is the maximum interior dihedral angle (in degrees) between triangle facets (or Edge segment tangents for a WIREBODY tessellation), note that a zero ignores this phase.
- **bcType = (no default)**
  Sets the AFLR_GBC attribute on faces. Options:
  - Farfield or Freestream or FARFIELD_UG3_GBC
    Farfield surface same as a standard surface except w/AFLR4
  - Viscous or -STD_UG3_GBC
    Standard BL generating surface
  - Inviscid or STD_UG3_GBC
    Standard surface

  - Symmetry or BL_INT_UG3_GBC or BoundaryLayerIntersect
    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

- **bcType = (no default)**
  Sets the AFLR_GBC attribute on faces. Options:

  - Farfield or Freestream or FARFIELD_UG3_GBC
    Farfield surface same as a standard surface except w/AFLR4
  - Viscous or -STD_UG3_GBC
    Standard BL generating surface
  - Inviscid or STD_UG3_GBC
    Standard surface
  - Symmetry or BoundaryLayerIntersect or BL_INT_UG3_GBC
    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

0.4.2 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