AFLR2 Analysis Interface Module (AIM)

Marshall Galbraith
MIT ACDL

July 26, 2024

0.1 Introduction

0.1.1 AFLR2 AIM Overview

A module in the Computational Aircraft Prototype Syntheses (CAPS) has been developed to interact with the unstructured, surface grid generator AFLR2 [2] [1].

The AFLR2 AIM provides the CAPS users with the ability to generate "unstructured, 2D grids" using an "Advancing-Front/Local-Reconnection (AFLR) procedure." Both triangular and quadrilateral elements may be generated.

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`.

0.2 AIM attributes

The following list of attributes are required for the AFLR2 AIM inside the geometry input.

- **capsGroup** This is a name assigned to any geometric entity to denote a "boundary" for further analysis. Recall that a string in ESP starts with a $. For example, attribute `capsGroup $Wing`.
- **capsMesh** This is a name assigned to any geometric entity in order to control meshing related parameters. Recall that a string in ESP starts with a $. For example, attribute `capsMesh $Wing`.

0.3 AIM Inputs

The following list outlines the AFLR2 meshing options along with their default value available through the AIM interface.

- **Proj_Name = "aflr2_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.

- **Tess_Params = [0.025, 0.001, 15.0]**
  Body tessellation parameters. Tess_Params[0] and Tess_Params[1] get scaled by the capsMeshLength attribute. (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.

- **Mesh_Quiet_Flag = False**
  Complete 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.
- **Edge_Point_Min = NULL**
  Minimum number of points on an edge including end points to use when creating a surface mesh (min 2).
- **Edge_Point_Max = NULL**
  Maximum number of points on an edge including end points to use when creating a surface mesh (min 2).
- **Mesh_Sizing = NULL**
  See Mesh Sizing for additional details.

0.4 AIM Outputs

The following list outlines the AFLR2 AIM outputs available through the AIM interface.

- **NumberOfElement**
  Number of elements in the surface mesh
- **NumberOfNode**
  Number of vertices in the surface mesh
- **Area_Mesh**
  The resulting mesh that can be linked to an anlaysis input.

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:

- **edgeDistribution = "Even"**
  Edge Distribution types. Options: Even (even distribution), Tanh (hyperbolic tangent distribution).
- **numEdgePoints = 2**
  Number of points along an edge including end points. Must be at least 2.

- **initialNodeSpacing = [0.0, 0.0]**
  Initial (scaled) node spacing along an edge. [first node, last node] consistent with the orientation of the edge.

- **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.

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