Metris. Analysis Interface Module (AIM)

Marshall Galbriath
MIT

February 28, 2025

0.1 Introduction

0.1.1 metris AIM Overview

A module in the Computational Aircraft Prototype Syntheses (CAPS) has been developed to interact with the unstructured mesh adaptation software `metris`.

Currently Metris only supports 2D (Area_Mesh) adaptation, but this will be expeanded in the future.

An outline of the AIM's inputs and outputs are provided in AIM Inputs and AIM Outputs, respectively.

The metris AIM can automatically execute metris, with details provided in AIM Execution. The specific executable can be changed with the 'metris' input string.

0.2 AIM Inputs

The following list outlines the metris inputs along with their default value available through the AIM interface.

- **metris = "metris"**
  metris executable

- **Passes = 30**
  Number of metris internal adaptation iterations

- **Mesh = NULL**
  An Area_Mesh link for mesh adaptation

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

- **MetricFieldFile = NULL**
  Metric field file in libMeshb format.

0.3 AIM Execution

If auto execution is enabled when creating an metris AIM, the AIM will execute metris just-in-time with the command line:
```
metris $(cat metrisInput.txt) > metrisOutput.txt
```

where preAnalysis generated the file "metrisInput.txt" which contains commandline arguments for metris.

The metris analysis directory is assumed to contain a metric.meshb file. This file will be generated automatically with preAnalysis using ScalarFieldFile or HessianFieldFile inputs, or can be generated manually via system calls to metris and setting MetricFieldFile.

The analysis can be also be explicitly executed with caps_execute in the C-API or via Analysis.runAnalysis in the pyCAPS API.

Calling preAnalysis and postAnalysis is NOT allowed when auto execution is enabled.

Auto execution can also be disabled when creating an metris AIM object. In this mode, caps_execute and Analysis.runAnalysis can be used to run the analysis, or metris can be executed by calling preAnalysis, system call, and posAnalysis as demonstrated below with a pyCAPS example:
```
print ("\n\preAnalysis......")
metris.preAnalysis()
print ("\n\nRunning......")
metris.system("metris $(cat metrisInput.txt) > metrisOutput.txt"); # Run via system call
print ("\n\postAnalysis......")
metris.postAnalysis()
```

0.4   AIM Outputs

List of available outputs from the metris AIM

- **Mesh**
  The output Area_Mesh or Volume_Mesh for a link

- **xyz**
  Grid coordinates. Useful for constructing scalar, hessian, or metric fields