Torrence Gue

Bilal Sharqi

A2SRL Lab

19 August 2020

Final Report: Modal Assurance Criterion Test Script

**INTRODUCTION**

From the month of June 2020 to August 2020 I have put together a test analysis script with the help of Bilal Sharqi that computes the modal assurance criterion (MAC) for the experimental porotype aircraft X-HALE as well as a non-uniform beam. The script, created and ran in Python, imports simulated NASTRAN data and collected experimental data from a ‘.mat’ library to calculate MAC for each possible combination of simulated mode to experimental mode.

**BACKGROUND**

Modal Assurance Criterion

Modal Assurance Criterion (MAC) quantifies the similarity of an experimental set and numerical set of eigenvectors, each representing a modal shape of a structure. The MAC Equation is as follows[1]:

where the variables represent the following:

MAC Matrix of dimensions { n modes x n modes}

Simulated analytical matrix of eigenvectors

Experimental test matrix of eigenvectors

Transpose of

Transpose of

In the case of operating in a 3D space in which wing displacement is key, the z-component of the eigenvector was isolated for modal analysis. With regards to the X-HALE all 36 data points and associated z-components of eigenvectors were used in the MAC calculation

Cross Orthogonality Testing

Though MAC can provide a useful analysis comparing simulated and numerical mode sets, cross orthogonality testing also examines the similarity between two eigenvector mode sets by analysing the orthogonal characteristics of each modal eigenvector data set. The cross orthogonality matrix equation is calculated below using the System Equivalent Reduction and Expansion Process (SEREP)[2] below:

where:

Cross orthogonality matrix

Expanded GVT mode shape

Transposed

Reduced Mass Matrix of the simulated data set

The reduced mass matrix and expanded GVT mode shape can be calculated as follows[2]:

Reduced Mass Matrix:

where intermeditary matrix is calculated as:

where:

Simulated NASTRAN modal analysis matrix of dimension [modes x d.o.f]

Transpose of

Expanded GVT mode shape:

Because of the definition

Where the following matricies represent:

Master eigenvector Matrix, dimensions [master dof x modes]

This matrix is the reduced number NASTRAN eigenvectors that correlate with experimental data

Slave eigenvector Matrix, dimensions [(all dof - master dof) x modes]

This matrix is the remaining number NASTRAN eigenvectors that do not correlate to the experimental data

Measured eigenvector matrix obtained from experimental test

Due to the similarity in processed between MAC and cross orthogonality testing (often abbreviated XOR), a script that can calculate MAC can also be manipulated to caclculate cross orthogonality.

**SUMMARY**

Functions and Key Components of Script

The MAC script works in several major steps:

* Importing Data: The first portion of the script. Import data here to be manipulated and arranged later in the script. Import Data specifications have been laid out in the Appendix of the MAC\_user\_friendly script
* Declaring Test Operations: Declare which tests you wish to conduct. You have the option to calculate the MAC of the XHALE manually, or using the automated grid sorting tool. You also have the option to complete an individual MAC of two specific numerical and experimental mode shapes. Additionally you can conduct a MAC of a non-uniform beam. Note: You will need to determine the mode frequencies you wish to compare before running the MAC. The ‘help\_Sort’ tool may assist you with this. Instructions listed in the script
* Matching Coordinate Data: After tests have been declared, the organizeGrid function helps match the most similarly placed simulated NASTRAN coordinates with the simulated experimental coordinates. This excludes the manual X-HALE MAC calculation, which exists as a preventative measure in case the grid tolerances are outside of the current organizeGrids restrictions.
* Numerical Data Normalization: Z-comonent numerical data was normalzied by the highest local value of the mode shape data set. This was different than the experimental mode set which was normalized by the highest absolute value of the experimental mode shape data set
* Reducing Coordinate Data: After data was normalzied, the data was then stripped of any grid points that did not align with the experimental data set as well as mode shape data that the user did not want to be used in the MAC comparison. This is done using the removeGrid\_Freq function
* Creating Phi Matrix: The ‘phi’ matrix is created by isolating each z-component of the eigenvector in each modal shape and compiling it into a two dimension matrix of dimension {frequencies analysed x degrees of freedom in z (how many accelerometers there are)}. This is done to each the experimental and numerical data sets for future mac comparisons
* Calculating MAC Matrix: Mac matrix is calculated using the calculateMAC function which takes in the phi matrix of the simulated and experimental data as an input.
* Displaying MAC Matrix and Associated Mode Graphs: MAC matrix is displayed as a 3D bar graph as well as mode shapes are displayed to help visualize the change in shape of the X-HALE and/or non-uniform beam

\*It should be noted that two data sets are used for each test, one which is for the calculation of the MAC, consisting of mode data, the other (often with some form of title called ‘vis’) is for the visualisation of the mode set, and is the sum of the mode data + coordinate data. (Static deformation is not included to prevent the over emphasis of deformation in the mode shape.) Each versionis processed the same way using the steps above.

\*Data is subject to additional processing not listed above, particularly see calcDisp section in Limitatations Section

Limitations of Script

There are a series of limitations limiting the validity of the modal assurance criterion listed below:

* Mapped grid points from the experimental test setup to the NASTRAN simulation have a possible error ranging from 5% to 11% (based on encoded tolerances) depending on how similar the experimental setup is to the NASTRAN simulation. Though grid points in close proximimity have similar modal behavior, this still can provide a source of error.
* The NASTRAN simulation demonstrates a conservation of length, in which the XHALE or non-uniform beam has conserved total length. This is not the case of the experimental test data. Further normalization or changes in test conditions may be required to achieve the same consistency.
* The given X-HALE test configuration was pinned on one side, limiting the degree of motion of the aircraft. This would alter the MAC results because it would create disimilar mode shapes between the free-free boundary condition of the simulated NASTRAN mode data and the fixed-free boundary condition of the XHALE experimental mode shape data.
* In previous iterations of the script, the MAC of the non-uniform beam only used the averaged values of the z-component of the experimental mode set. This reduced the number of experimental data points from 26 to 13 in order to be compared with the 13 most similiarly placed numerical data points. Since then, it was decided that instead of reducing the experimental mode data, 13 of the most similarly placed numerical data beam points would be created into 26 points using the x-rotation component of each grid point and the distance from the experimental accelerometer to the beam centerline. The calcDisp function would help create a predicted point in which a similar accelerometer ought to be in the simulated NASTRAN model. However due to unforseen circumstances, it has been difficult to access the accuracy of this new model. The x-rotation seems to be set to near zero for nearly all numeical mode data points. This may be due to accidentally manipulating the data prior to the MAC comparison.

Future Expansions of MAC Tool

Hopefully, the MAC\_user\_friendly script can be altered to include tests for cross orthogonality testing. The infrastructure to analyse and isolate the z-components of the eigenvectors already exists. If another user was to implement the cross orthogonality test method they would have to do five additional steps:

1. Use the current output of the orderPhi function as the Master or matrix, and transpose it to be dimensions [dof\_master x modes]
2. Use the remainder of the mode data and run it into the orderPhi function to use as the Slave or matrix. Make sure to transpose this data as well.
3. Calculate the expanded matrix using the equations listed above.
4. Import and process the mass matrix of the structure you wish to analyse it and reduce as instructed above.
5. Calculate Cross orthogonality matrix

**CONCLUSION**

The MAC application process seemed reasonably successful with the exception of the implementation of the calcDisp method to factor in torsion into the MAC script. Further documentation on the procedure to calculate MAC can be found within the MAC\_user\_friendly script itself as well as the ‘Reports and Associated Files’ tab under my branch of the repository. See weekly Reports folder for weekly analysis and detailed formulation of each method and resulting tables. Additional graphs are available upon request. Contact information is listed below.

Contact information:

Torrence Gue

Email: [tkgue@umich.edu](mailto:tkgue@umich.edu)

Cell: 773-308-8371

**CITATIONS**

[1]Pastor, M., Binda, M., & Harčarik, T. (2012). Modal Assurance Criterion. *Procedia Engineering,* *48*, 543-548. doi:10.1016/j.proeng.2012.09.551

[2] Pak, C.-G., and Truong, S., “Creating a Test Validated Structural Dynamic Finite Element Model of the X 56A Aircraft,” *15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2014.