MATLAB in Aero Engine Design

Vineet K Reddy, Sravan Shakker, Sushovan Roychowdhury, Asha Koshy
GKN Aerospace India | 10 April 2014
Index

➢ About GKN
➢ Aero-engine: simulation for design and certification
➢ Challenges and MATLAB-advantage
➢ Examples of MATLAB usage in the design process
➢ Future
GKN: Engineering that moves the world

Every day, we drive the wheels of hundreds of millions of the world’s cars, we help thousands of aircraft fly safely, we deliver the power to move earth and harvest crops, and we make the essential components for industries that touch lives across the globe.

Our four divisions are all market leaders, each outperforming their markets, with a leading market share in their segment. This breadth gives us unrivalled expertise and experience in cutting-edge technology and engineering.

GKN Driveline  GKN Powder Metallurgy  GKN Aerospace  GKN Land Systems
Aerospace Engine Systems has a strong presence in today’s aircraft fleets

- More than 90 percent of all new large commercial aircraft engines use GKN components

- For these aircraft GKN provides:
  - Engine components
  - Engine technology
  - Engine technical support
  - Engine MRO services
  - Parts Repair
Aero Engine Component Design

Every Engine need to be certified by
• FAA - Federal Aviation Administration (USA)
• EASA - European Aviation Safety Agency

Analysis Inputs
• Performance Data
• Flow information
• Thermal loads
• Mechanical & Internal pressure loads

Analysis Outputs
• Metal Temperatures
• Deformations
• Stresses & Strains

Derived Outputs
• Crack initiation & propagation life
• Safety Margins
Typical Flight Mission & Engine Loads

Loads

- Temperature
- Pressure
- Mechanical
Strength Verification of Design – Overview of Simulation Process

Temperature and pressure distribution

Generate Thermal load
Pressure load
Maneuver load

Stress analysis

Compute life

Aero-thermo equations/CFD

MATLAB

Thermal analysis
Excel files

MATLAB

FEA

MATLAB

Crack initiation and propagation
The MATLAB-advantage

- **Challenges:**
  - Large number of DoF (typically in millions)
  - Vector and Tensor variables
  - Multiple iterations

- **MATLAB features in use:**
  - Handling of large data (typically Excel files with mixed data type)
  - Vector/matrix operations and sorting
  - Easy automation
  - User friendly IDE

MATLAB Usage - Examples

- Thermal Analysis
- Life Analysis – Generic process
- Stress History Generation – Linear Response
- Stress History Generation – Nonlinear Response
- Iterative Solution Control
Thermal Analysis

MATLAB is used for processing test data

- MATLAB is used to process 100’s of ‘.csv’ files containing test data
- Files containing insufficient information are identified and removed
- The frequency of available data is calculated and reduced
- Data from all files is written into single text file for further usage

Generating thermal loads

- Generally multiple range selections are required to read inputs from an excel file, MATLAB and Excel interaction time reduced by using “xlsread1” macro
  - Xlsread1 saves time by avoiding open/close of excel server on each function call
- A MATLAB code is used to generate thermal loads for life estimation of a component. The MATLAB code is generic such that it can handle variable input depending on mission types
- The code is flexible to accommodate intermediate intervention, e.g. test data overriding generated load in special cases – a semi-automated process in such a case
Generating Thermal Loads

MS EXCEL
• Performance data
• Material data

Model calibration parameters
• Air temp inputs
  • Factors from test data and CFD analysis
• HTC inputs
  • Text book correlations and CFD analysis

MATLAB
• Case specific equations (time points)
• Database for transient response functions
• Switches to control mission types
• Interpolation of data

ANSYS readable LOADs
• Bulk Temperature
• Heat transfer coefficients
Weld Life Analysis

Typical representation of butt weld in structure

Row of nodes on top side

Node pair

Plate 1

Row of nodes on root side

Plate 2

Schematic illustration of incremental loads

Incremental loads

Flight mission points

MATLAB

Orient stresses

Identify worst incremental load

Build mission stress

Rain Flow Cycle counting

Crack propagation analysis

Limit load sustenance check

Vector operation

Matrix sorting

Build time series data

File I/O operation

Technical computing
Damage Tolerance Analysis – Linear Response

MATLAB

File I/O

Matrix operations: Sorting, Slicing

Matrix multiplication

Load Cases

FE stress solution for uncoupled loads

Stress solution for all load cases

Mechanical + Thermal stress history for life calculation

Load Case | $F_{X1}$ | $F_{Y1}$ | $F_{X2}$ | $F_{Y2}$ |
---|---|---|---|---|
1 | 50 | 64 | 30 | 98 |
2 | -49 | 87 | -44 | -61 |
3 | 97 | 46 | 52 | -29 |
. | 79 | -74 | -62 | 2 |
i | 52 | -70 | 96 | 54 |
. | -61 | -81 | 88 | 13 |
n-1 | 23 | 93 | 100 | 13 |
n | -75 | -70 | -5 | 69 |

Node | 1 | 2 | 3 | . | j | . | m-1 | m |
---|---|---|---|---|---|---|---|---|
$F_{X1}$ | 0.95 | -0.99 | -0.51 | 0.91 | -0.03 | -0.55 | 0.85 | 0.69 |
$F_{Y1}$ | 0.07 | 0.05 | -0.72 | -0.82 | -0.92 | 0.64 | 0.13 | -0.9 |
$F_{X2}$ | -0.45 | -0.73 | -0.1 | -0.75 | 0.99 | -0.83 | -0.38 | -0.35 |
$F_{Y2}$ | -0.84 | -0.21 | -0.32 | -0.55 | -0.26 | -0.16 | -0.7 | -0.6 |

Stress at 1st node due to $F_{X1}=1$

Magnitude of $F_{X1}$ in 1st load case
Damage Tolerance Analysis – Nonlinear Response

Two uncoupled loads on a structure

Response Surface

<table>
<thead>
<tr>
<th>Python</th>
<th>ANSYS</th>
<th>MATLAB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compute convex hull to choose load cases</td>
<td>Non Linear FE analysis</td>
<td>Generate Response Surface</td>
</tr>
<tr>
<td>Gift wrapper algorithm</td>
<td>PDE</td>
<td>2D curve fitting</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Matrix Operations</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Crack Initiation / Propagation analysis</td>
</tr>
<tr>
<td></td>
<td></td>
<td>File I/O operation</td>
</tr>
</tbody>
</table>
Example Matlab code - life analysis

```matlab
% Main script to prepare NASCRO input files for weld.
% This script is called from another script, that takes in the user data.
%
% The script executes the following steps for all nodes in a given weld:
% 1) Identify the worst incremental load for each
% 2) Print load cases in file "loadable.txt" This
% 3) Run VAGMELD to extract stresses for all load c
% 4) Run ADDNASCRG.
% 5) Stress values are combined in the file "loada"s
% Note each "loadable.txt" is unique for every
% 6) Run cycle counting procedure RFC. Program per
% and groups cycles into blocks of loads which
% 7) Create NASCRO input file.

%%
% Go to working directory
% cd (p)
%
% Error check. If exceedance file doesn't exist, stop.
% if exists('exceedance_FAL.txt')
% error('Exceedance data file FALs does not exist. Exit.')
% end
% Write exceedance data file NaNs. Use.
% p = [p1, 'exceedance_FAL.txt'];
% fprintf(fid, 'NaN
Matlab

Function with variable-length input argument list

Optimized performance from code
Iterative Solution Control

- Use mean value theorem based algorithm to guess next $S_{\text{max}}$
- Check for convergence

- Generate input deck
- Invoke NASGRO
- Extract results from NASGRO output

- $S_{\text{min}} = R \times S_{\text{max}}$
- For a given $R$, estimate value of $S_{\text{max}}$, which will result in target life
- No closed end relation exists between $S_{\text{max}}$ and life
MATLAB usage in near-future

➢ Probabilistic assessment of structures

➢ Predictive response surface from historical stress analyses

➢ Optimization
  – meet target strength and life with optimum structure
Thank You