Integrated Tool for Strain Extraction in Virtual Testing

Ajit Chirme, Dattatraya Parle, Niranjan Awati, Ravi Udali and Sachin Kulshrestha

Product Life Cycle and Engineering Solutions

Infosys Technologies Ltd., Pune, India

E-Mail: venkata_udali@infosys.com

Abstract: Aircraft manufacturers conduct extensive testing to meet certification and regulatory requirements prior to delivery of an aircraft to end customer. The physical testing especially full scale testing requires huge test facilities like test rigs etc., which are expensive and time consuming. Hence, the number of physical tests is optimized by conducting large number of virtual tests through finite element analysis prior to experiments. The results, specifically the strains, obtained from virtual tests are correlated with experimental data. The manual extraction of strains at the strain gage locations from the huge finite element data is laborious, time consuming and error prone. Hence it is essential to automate the process of extraction of strains from the reference CAD and FE models.

This paper presents an automated approach to extract strains of aircraft structural models from widely used CAD and FE environments. The developed approach has been implemented as an integrated tool in widely used CATIA V5 and Abaqus environments. The integrated tool eliminates tedious procedures involved in manual extraction of strains using FEA models at appropriate strain gage locations. This integrated tool is a quick, inexpensive and effective technique for predicting structural strains. This work uses Abaqus-Python, CATIA-VB and Visual Basic environment to develop integrated tool for identifying location, direction and strains in the strain gages present in an aircraft structure. Substantial effort savings are achieved through the developed tool while extracting strains which is of the order of few days to few hours without any errors.

Keywords: Aircraft structure, Strain Gage, Abaqus-Python, CATIA V5-VB, Abaqus Automation

1. Introduction

Strains at various locations of an aircraft are computed from FE simulations and compared with the physical test data to meet certification and regulatory requirements prior to the delivery of an aircraft to its end customer. The full scale testing of an aircraft requires huge test facilities like test rigs etc., which are costly and time consuming (Linde, 2004). Properly installed and calibrated strain gages are used to determine the strains in landing gear, wings and empennage assemblies (Kottkamp, 2002). The number of physical tests to be conducted is optimized by conducting large number of inexpensive virtual tests through finite element analysis. The results such as strains, obtained from such virtual tests are validated with experimental data.

Abaqus, due to its advanced finite element analysis capability and user friendly functionalities, is widely used in aerospace industry as virtual testing tool. Similarly, CATIA V5 is CAD software widely used for 3D modeling of aircraft structures. Although the GUI of Abaqus and CATIA V5 is user-friendly, manually performing some intricate and repeating tasks takes a lot of time and leads to errors. Hence, customization of these tools is required to meet the specific requirements of an engineer. One such activity which requires customization is strain extraction which requires interaction between CAD tool (CATIA V5) and FE tool (Abaqus). The manual extraction of strains at the strain gage locations from the huge finite element data is laborious, time consuming and error prone. Hence, it is essential to automate the process of extraction of strains from the referenced CAD and FE models.

2. Literature Review

Many physical tests are performed on aircraft structures when new design principles, materials, manufacturing processes or load conditions are to be developed and deployed. The physical tests are performed either on single components of small size specimen or on a full-scale test of aircraft fuselage/wing. The various in-services loading conditions are applied to the specimen of component of full aircraft model. Small scale experimental tests help in reducing the costs along with realizing the experimental objectives faster and easier (Walker, 1952). The tests may include quasi-static loads up to failure, repeated load cycles, low or high energy impact. It includes also environmental effects, such as moisture or temperature. All these tests need to meet strong certification criterion for the aircraft structures, and the means of compliance to this regulation are done in accordance with the airworthiness authorities. These physical tests are very expensive and time consuming. Hence, recent times virtual testing has become popular in aerospace industries. Virtual Testing is usually described as the capability to provide by simulation blind prediction of the real-world structure physical behavior. The prediction is expected to provide the structure strength value in order to ensure a proper sizing against in-service conditions. In past, the use of structural analysis in commercial aircraft design and certification has been focused on linear finite element analysis for the calculation of internal load distributions and on the use of analytical stressing methods, both for initial sizing and then more detailed calculations for final certification. This stress based approach, when combined with structural testing both to demonstrate the aircraft structure integrity and adequacy of methods, has proven itself to be highly reliable in the development of safe aircraft structures. In recent years, advanced nonlinear analysis methods have been used increasingly to obtain more accurate assessments of the actual strength of aircraft test structures, both for risk mitigation prior to test and subsequent to a failure event (Imbert, 2007). Nonlinear finite element analysis has been employed with great effect to increase confidence in the large-scale and expensive structural tests that are required before certification, as well as to understand in more detail the likelihood, causes and consequences of structural failure (Prior, 2009).

In late 90's, Dassault Systems developed the concept of digital mock-up (DMU) for aerospace industry. It was one of the ground-breaking technologies that helped aerospace manufacturers tremendously and freed them from building expensive, time-consuming physical prototypes. Using DMU, aircraft industry could create virtual (digital) models of new aircraft quickly (Walker, 1952). Few researchers from AIRBUS (Gaudin, 2008) presented the AIRBUS vision on the key numerical technologies that are developed in order to provide a significant Virtual Testing capability for aircraft structures at industrial level. They also highlighted the expected benefits, limitations and use of a virtual testing capability in the context of the aircraft certification process. Ostergaard (Ostergaard, 2011) focused on the virtual testing to predict aircraft structural strength. Virtual testing is a concept with several attributes and is to be understood as the simulation of aircraft structure using advanced linear/nonlinear finite element analysis. According to him virtual testing predicts the aircraft structural strength using combination of analysis software, methods, people skills and experience with a high level of confidence.

Based on the review of available literature it is observed that virtual testing has become popular using CAD and CAE tools. Further, time spent on CAD and FEA tools can be reduced by automating manual processes through customization of these tools using its application programming interfaces (API's). Although customization of CAD and FEA tools is commonly done for various repetitive activities in aerospace domain but there isn't much published literature available on it.The objective of the current work is to develop an integrated strain extraction tool which can save substantial effort and time in aircraft testing.

3. Integrated Strain Extraction Tool

In the existing process, strain is extracted manually from FEA models by extracting corresponding strain gage locations from CAD models. The user has to open CAD part in CATIA V5 and measure x, y, z co-ordinates for particular strain gage. Knowing positional co-ordinates of strain gage, user locates corresponding (nearest) node in FEA model, user measures strain of this node in Abaqus results file using Abaqus/CAE. For 100-150 strain gage locations in typical empennage model of aircraft, the manual process requires one complete man-day. This process further complicates and requires more skill and time for composite structures. The time required for extraction of strain increases further with the complexity and size of the aircraft. For a typical aircraft, this process takes few days to a week time. Further, the manual process requires expertise in both CATIA V5 and Abaqus tools for extracting the strain. Hence, there is need and scope for automation of such complex, tedious and non value added process. This not only simplifies strain extraction process but also reduces strain extraction time to few minutes for typical empennage model eliminating errors. Also, the tool can be operated by non-expert or infrequent user of CATIA V5 and Abaqus. Thus, the objective of the current work is to develop an integrated tool to extract strain from aircraft structure for virtual testing.

The manual process of strain extraction has been re-engineered to develop integrated automate process using the automation utilities of CATIA V5 and Abaqus. This integrated process has been implemented in a integrated tool. The overall process flow of integrated tool for strain extraction has been shown in Fig.1. The tool takes CATIA V5 model which contains strain gage location information and Abaqus result file as inputs.

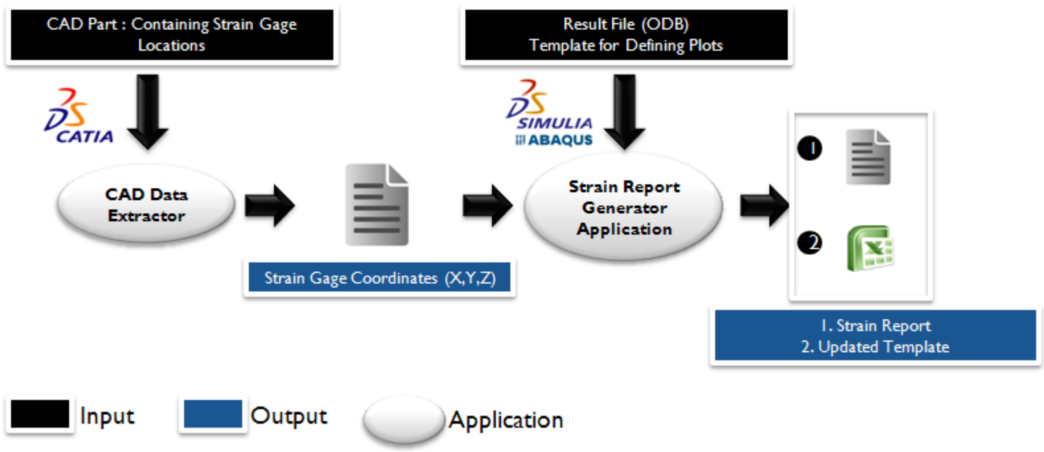


Figure 1. Integrated tool process flow for stain extraction in aircraft structure.

The tool consists of two applications integrated together, see Fig. 2. The user interface is developed using Visual Basic which is further integrated with CATIA V5 and Abaqus/CAE for data processing. The user needs to launch CAD Data Extractor first followed by the Strain Report Generator.

Figure 2. Integrated tool user interface

Fig. 3 shows CATIA V5 based application known as CAD data extractor. This tool user interface interacts with CATIA V5 API's to locate strain gages and its references in CATIA V5 Part file (11). After identifying all such locations it creates ASCII file as shown in Fig 4. The output file contains information of strain gage name, location and direction of strain to be extracted.

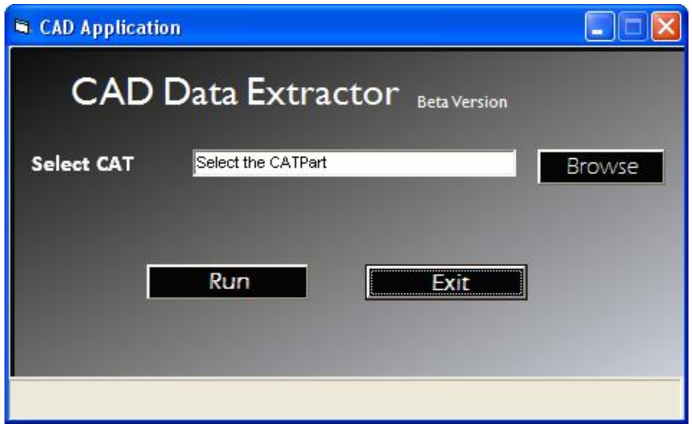
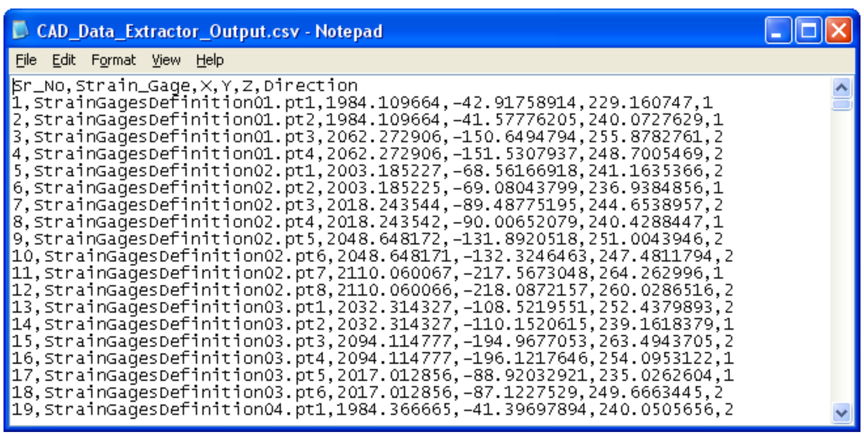


Figure 3. CAD Data Extractor user interface.




```
Sr_No,Strain_Gage,X,Y,Z,Direction
1,StrainGagesDefinition01.pt1,1984.109664,-42.91758914,229.160747,1
2,StrainGagesDefinition01.pt2,1984.109664,-41.57776205,240.0727629,1
3,StrainGagesDefinition01.pt3,2062.272906,-150.6494794,255.8782761,2
4,StrainGagesDefinition01.pt4,2062.272906,-151.5307937,248.7005469,2
5,StrainGagesDefinition02.pt1,2003.185227,-68.56166918,241.1635366,2
6,StrainGagesDefinition02.pt2,2003.185225,-69.08043799,236.9384856,1
7,StrainGagesDefinition02.pt3,2018.243544,-89.48775195,244.6538957,2
8,StrainGagesDefinition02.pt4,2018.243542,-90.00652079,240.4288447,1
9,StrainGagesDefinition02.pt5,2048.648172,-131.8920518,251.0043946,2
10,StrainGagesDefinition02.pt6,2048.648171,-132.3246463,247.4811794,2
11,StrainGagesDefinition02.pt7,2110.060067,-217.5673048,264.262996,1
12,StrainGagesDefinition02.pt8,2110.060066,-218.0872157,260.0286516,2
13,StrainGagesDefinition03.pt1,2032.314327,-108.5219551,252.4379893,2
14,StrainGagesDefinition03.pt2,2032.314327,-110.1520615,239.1618379,1
15,StrainGagesDefinition03.pt3,2094.114777,-194.9677053,263.4943705,2
16,StrainGagesDefinition03.pt4,2094.114777,-196.1217646,254.0953122,1
17,StrainGagesDefinition03.pt5,2017.012856,-88.92032921,235.0262604,1
18,StrainGagesDefinition03.pt6,2017.012856,-87.1227529,249.6663445,2
19,StrainGagesDefinition04.pt1,1984.366665,-41.39697894,240.0505656,2
```

Figure 4. Output of CAD data extractor.

Flow chart for CAD data extractor is shown in Fig. 5. Once application is launched it opens CATPart in CATIA V5, application process the tree of CATPart to extract the strain gage information.

CAD data extractor is a standalone application developed in VB that connects to CATIA via the APIs to extract the strain-gauge location properties. Strain gauge locations are provided as Point explicit (in CATIAV5 models). Point explicit is the breed of points that lets user / automation queries the co-ordinates of itself. Points are housed or stored in geometrical sets in a "hive-recursive" fashion, such that the user / automation would have to traverse recursively to identify the points of the specific above mentioned type bearing the X, Y, Z co-ordinates in the model (virtual) space. Models are of the type "Part" (.CATPart in CATIA V5) the needs the automation to access the MECMOD and PARTITF interfaces for automatic extraction of the points.

The user can select the hive node in the tree and start the automation, which then in turn would recursively access all the points to "query" the co-ordinates of the points. Typically the geometrical sets can contain a range of wireframe geometry, so that the automation has an added task of filtering the geometry to bare minimum points required. The automation is programmed to traverse any depth and length of the hive to extract the points, such that no format imposition is laid on the model being made.

As the automation progresses the hive it would leave behind the imprint of point (Name vs co-ordinate) in a comma separated file format (.CSV). Additionally the automation can export the model of the points un-hived (flat in on geometrical set against the earlier recursive form) in an IGES (Initial graphics exchange specification) format. IGES since it is a universal CAD format it can then be opened in any of the downstream CAD or analysis packages.

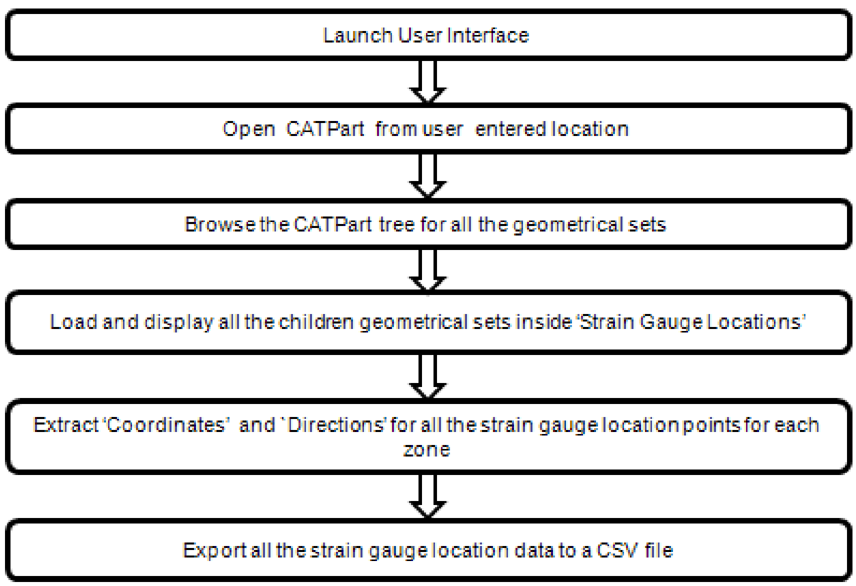


Figure 5. CAD Data Extractor flow chart.

The Second application which is Strain Report Generator is shown in Fig. 6. This application requires following inputs:

- Abaqus results file i.e. ODB: This is FE analysis result file for input CAD model. This file should contain strain information.
- Output generated by CAD Data Extractor: This is comma separated CSV file containing strain gage location information, see Fig. 4.
- Plot generation template file: This is MS-Excel template file in which strain generator application updates information such as part instance, node, element, etc for further processing. Fig. 7 shows the sample template file.
- Strain Result component to be extracted.
- Abaqus Version. The tool finds out the installed Abaqus Versions on the workstation and asks the user to select the version on which the tool would operate.

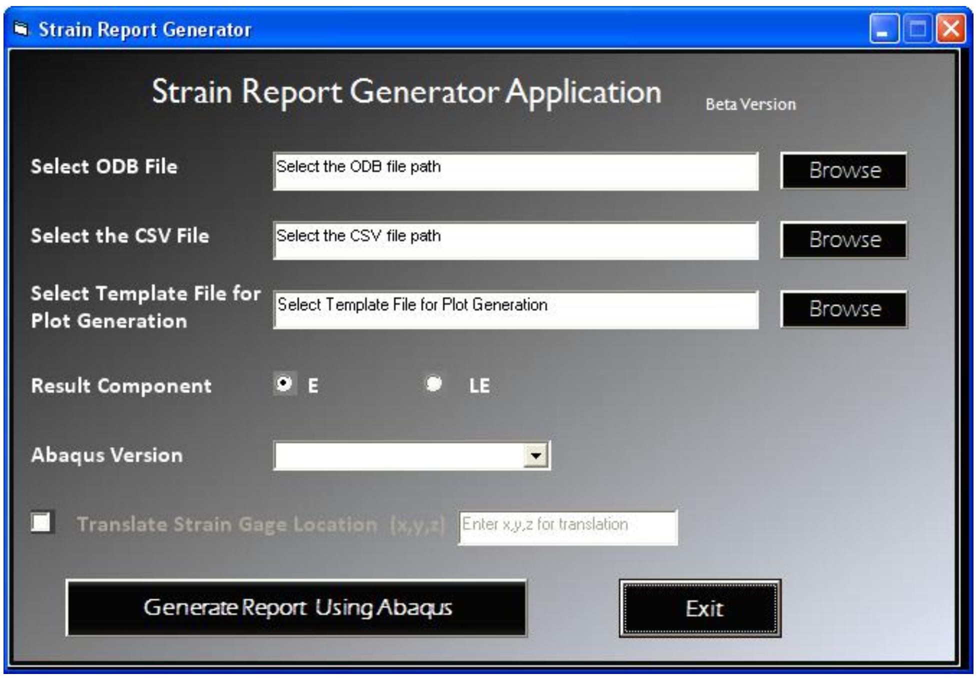


Figure 6. Strain report generator application.

Strain Gage FEA Information					
Sr. No	Gage	Part Instance	Element	Node	Field Variable

Figure 7. Plot generation template file.

Finally, strain report as shown in Fig. 8 is generated by strain report generator application. Report is generated for nodes (see Fig. 8a) as well as attached elements (see Fig. 8b). Each report is two columns: Left column is % loading in case of nonlinear analysis and right column is corresponding strain values.

X	LE:LE11 PI: SIDE_CONSTRAINT_1-2 E: 2185 Centroid
0.00000000E+000	0.00000000E+000
1.00000001E-001	-1.30662511E-006
2.00000003E-001	-2.87773446E-006
3.00000012E-001	-4.77799676E-006
4.00000006E-001	-7.11348184E-006
4.49999988E-001	-8.47223055E-006
5.00000000E-001	-1.00095231E-005
5.50000012E-001	-1.17553755E-005
6.00000024E-001	-1.37146117E-005
6.49999976E-001	-1.59460615E-005
6.99999988E-001	-1.85074241E-005
7.50000000E-001	-2.15508680E-005
8.00000012E-001	-2.53968919E-005
8.50000024E-001	-3.09426032E-005
8.99999976E-001	-3.75619129E-005
1.00000000E+000	-5.62589194E-005

(a)

X	LE:LE11 (Avg: 75p) PI: SIDE_CONSTRAINT_1-2 N: 5827
0.00000000E+000	0.00000000E+000
1.00000001E-001	-3.48513169E-007
2.00000003E-001	-8.10724771E-007
3.00000012E-001	-1.42632530E-006
4.00000006E-001	-2.23675602E-006
4.49999988E-001	-2.72391890E-006
5.00000000E-001	-3.28689293E-006
5.50000012E-001	-3.94494782E-006
6.00000024E-001	-4.70270061E-006
6.49999976E-001	-5.59701721E-006
6.99999988E-001	-6.65102880E-006
7.50000000E-001	-7.93145318E-006
8.00000012E-001	-9.56586518E-006
8.50000024E-001	-1.18718544E-005
8.99999976E-001	-1.46042112E-005
1.00000000E+000	-2.22686704E-005

(b)

Figure 8. Extracted sample strain report for strain gages.

Strain report generator is developed using Abaqus-Python scripting (12). Based on the inputs provided, the tool updates python script which runs on ODB file in Abaqus/CAE environment. It is observed that many instances CAD model and FEA model may not be at same location and there might exist certain offset. Hence, the translational transformation is required in CSV co-ordinates based on user input. Python script (*VBCommand.py*) below is updated with user inputs from UI shown in Fig. 6 using Visual Basic.

```
import abaqusStrainReport # User defined Module for Report Generation
odbFile="D:/Result_File.odb"
CSVFile="D:/CAD_Data_Extractor.csv"
PDTFile="D:/Plot_Template_Excel.xls"
ResComp=20 # Strain Component Indicator
TranslateReq ="No"
xT=0 # Translation Required in X Direction
```

```
yT=0 # Translation Required in Y Direction
zT=0 # Translation Required in Z Direction
abaqusStrainReport.fileProcessing(odbFile,CSVFile,PDTFile,ResComp,TranslateReq,xT,
yT,zT)
```

Above shown script is run at startup using the below command:

```
abq6102 cae startup="D:\Code\VBCommand.py"|| pause
```

VBCommand.py contains a user defined module named *abaqusStrainReport* which contains the application algorithm. This script uses various Abaqus modules as listed below:

```
from abaqus import *
from abaqusConstants import *
import __main__
import section,regionToolset,part,step,visualization,xyPlot,csv,sys,array
import displayGroupMdbToolset as dgm
import displayGroupOdbToolset as dgo
import connectorBehavior
import nearestNodeModule
```

The tool searches for nearest node for the given strain gage co-ordinates in ODB model within the specified tolerance. To search nearest node in OBD file based on x, y, z co-ordinates tool uses *nearestNodeModule* functionality provided in Abaqus/CAE 6.10. This required x,y,z location and ODB file path as input and returns the closest node label, Part Instance, Distance and the coordinates of the node.

```
NodeInfo = nearestNodeModule.findNearestNode(x,y,z,odbFile)
```

Once node is located, tool identifies attached elements by using connectivity information present in FE model and updates plot generation template file. The below function returns the attached elements to a given node :

```
def findElements(elements, nodeLabel):
    return [e.label for e in elements if nodeLabel in e.connectivity]
```

For each strain gage location, it creates node and element sets to extract strain information. Tool can extract two types of strains i.e. elastic strain and logarithmic strains based on user request. This process is repeated for all strain gages present in CAD model. Finally, tool generates strain report using XYData Objects and highlights nodes and elements representing strain gages in Abaqus/CAE. Fig. 9 shows the strain report generator tool architecture. The below code generated the XYData objects which are then used to create the report.

```
session.xyDataListFromField(odb=odb, outputPosition=NODAL, variable=(('LE',
INTEGRATION_POINT, ((COMPONENT, 'LE11'), )), ), nodeSets=(NodeSetNameXY, ))
```

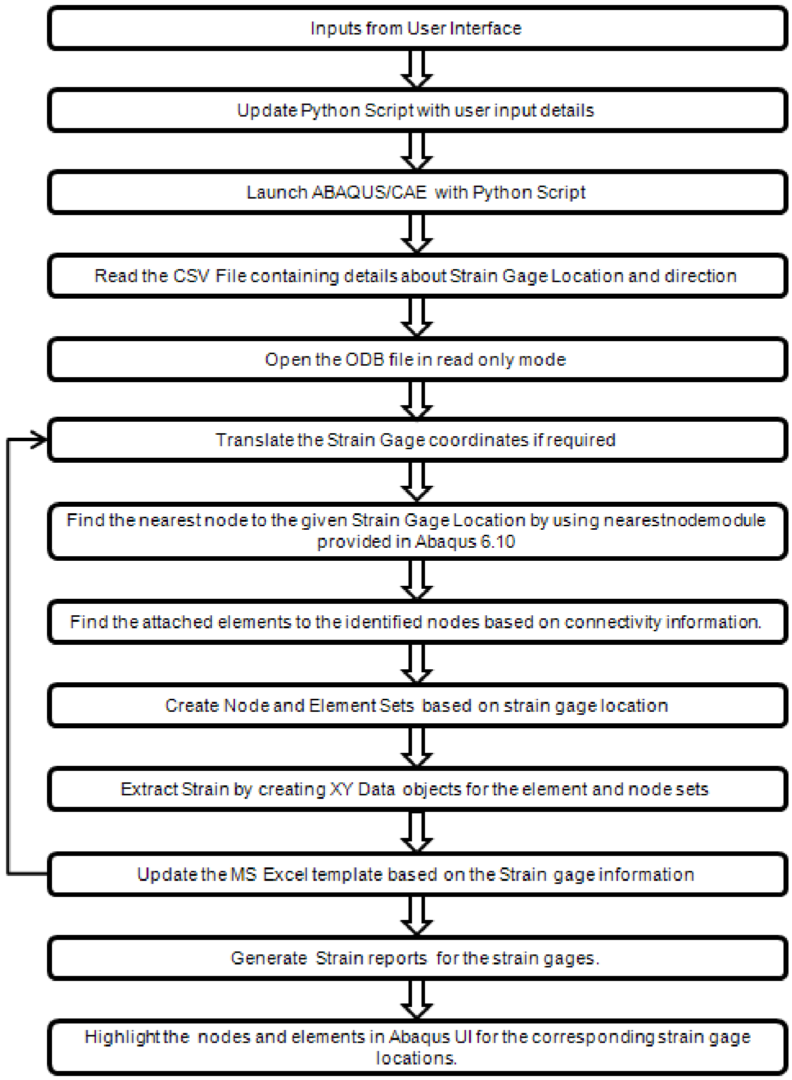


Figure 9. Strain report generator tool architecture.

4. Results and Discussion

In the current study, the tool is used to extract the strain data of a typical empennage box construction. Fig. 10 shows a typical empennage skin panel in CATIA V5. In this empennage panel there are 150 strain gage locations for which strain extraction is required. The extracted strain is used for calibration and virtual testing of aircraft. The manual stain extraction has taken one person day. The developed tool takes about 30 to 45 minutes for the strain extraction. This tool has helped reducing strain extraction time by about 90%. However, this time depends on many factors such as processor speed, number of strain gages and complexity of model.

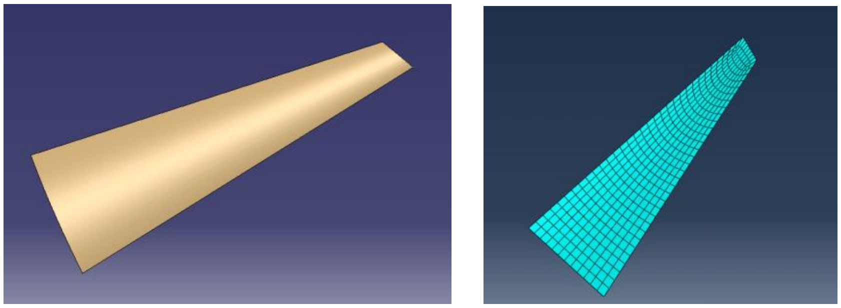


Figure 10. Representative empennage panel in CATIA V5 and Abaqus.

The tool architecture is such that it can be used for any general purpose structural component. Fig. 11 shows a typical sandwiched panel structure which has strain gages mounted. The developed tool is able to extract strains without any modifications.

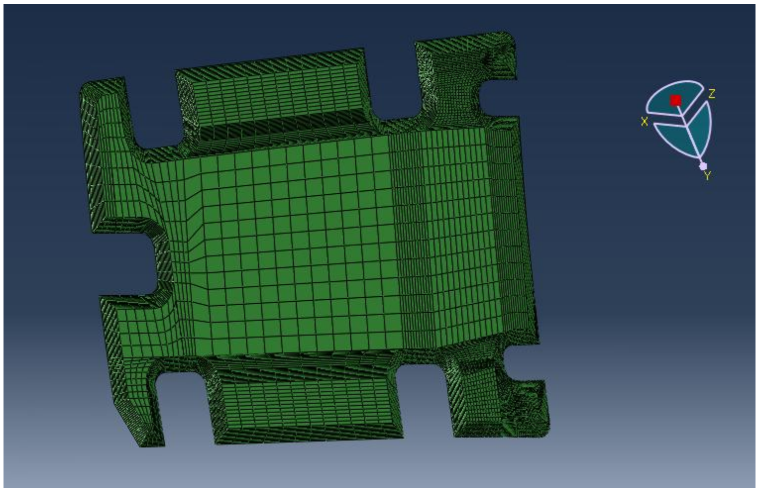


Figure 11. Sandwiched panel structure.

Strain report data is further used for calibration and validation. The validation of the integrated tool involves comparing manually extracted results with tool extracted results. In present study, it is observed that integrated tool and manually extracted strains are exactly matching. This indicates the accuracy of integrated tool. Further, extracted strain is validated against the test data.

After calibration and validation of strain data, virtual testing is carried out. Fig. 12 shows validation of FE strain data with test results for four strain gages located at Bay 1 of the structure. Validation process shows that FE results are very close to experimental results.

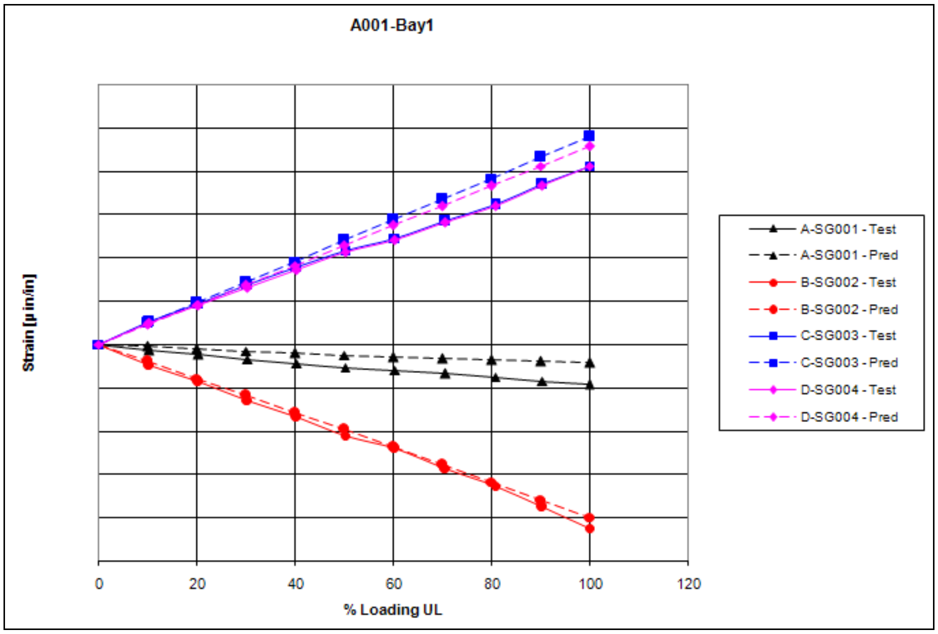


Figure 12. Validation Plot

5. Conclusions

In the current work a new process has been developed to extract the strains automatically. The developed process has been implemented in an integrated CAD and CAE environment using CATIA V5 and Abaqus tools. The results of the tool have been successfully calibrated and validated against test results. The integrated tool for strain extraction has helped in reducing the time by about 90%. Further, the use of tool eliminates all the manual errors. The integrated tool can be used for extraction strains of any generic structure with minimum customization, if required. The tool can be used by any novice user for extraction of strains.

6. References

1. Brill, D., R., "Advanced Airport Pavement Design Procedures", Airport Technology Research and Development, 2010
2. Gaudin, J., and Mahe, M., "Key Enabling Technologies For Virtual Testing Of Aircraft Structures - Airbus Vision", 8th. World Congress on Computational Mechanics (WCCM8) and 5th. European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS 2008) Venice (Italy) - July 2008
3. Imbert, J., F., “Airbus, challenges in aircraft structure analysis”, ESA/NAFEMS Seminar on Engineering Quality, Verification and Validation. Noordwijk, 6 December 2007
4. Kottkamp, E., Wilhelm, H., and Kohl, D., "Strain Gauge Measurements On Aircraft", Advisory Group For Aerospace Research And Development, AGARDograph No. 160 Vo1.7, 2002
5. Linde, P., Pleitner, J./ Rust, W.: Virtual Testing of Aircraft Fuselage Stiffened Panels, Proceedings of ICAS 2004, 24th International Congress of the Aeronautical Sciences
6. Prior, A., “Dassault Systems: Nonlinear Simulation of Large Scale Aircraft Structures — Implications for Certification Methodology and High Performance Computing Infrastructure”, NAFEMS World Congress, June 2009
7. Ostergaard, M., G., Ibbotson, A., R., Roux, O., L., Prior, A., M., "Virtual testing of aircraft structures", CEAS Aeronaut J (2011) 1:83–103
8. Walker, P. B., The Experimental Approach to Aircraft Structural Research, Journal of the Aeronautical Sciences, Vol. 19, No. 3, p. 145, March, 1952
9. www.cdcza.co.za/DMU%20in%20Aerospace_FINAL_Eng.doc
10. http://cordis.europa.eu/fetch?CALLER=FP6_PROJ&ACTION=D&DOC=15&CAT=PROJ&QUERY=0125d43f7184:23ab:79c5c95f&RCN=75782
11. CAA V5 Visual Basic Reference (V5R19)
12. Abaqus Scripting User’s Manual (6.10)

7. Acknowledgement

Authors acknowledge contribution of Dr Ravi Kumar GVV for his valuable guidance. Authors also acknowledge support of Mr. Devaraja Vaderahobli Holla, Mr. Avinash Patil, Mr Navin Kumar and Mr. Ramnath Suryaprakash during this work.