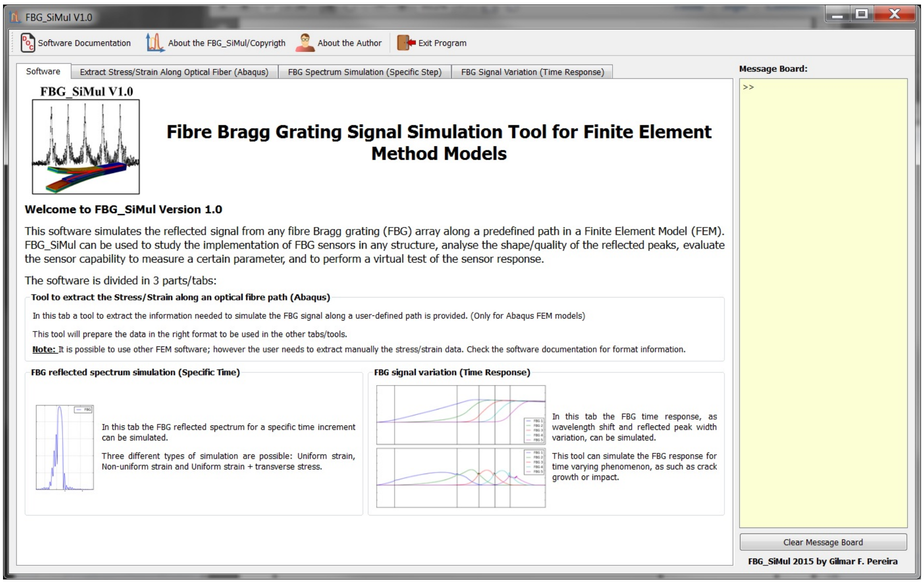
FBG_SiMul V1.0
USER-MANUAL

First version of the software FBG_SiMul user-manual.

FBG_SiMul V1.0, 2015–2016



Created by

Gilmar F.Pereira
Malcolm McGugan; Lars P.Mikkelsen

Technical University of Denmark
DTU- Wind Energy

January 2016

Contact: gfpe@dtu.dk; gilmar_fp@outlook.com

Contents

Section 1

Info.

1.1 Copyright

The FBG_SiMul author strong believes in:

- the freedom to use the software for any purpose,
- the freedom to change the software to suit your needs,
- the freedom to share the software with your friends and neighbors, and
- the freedom to share the changes you make.

1.2 About the software and the author

The FBG_SiMul is a software that simulates the reflected signal from a fibre Bragg grating (FBG) array along a predefined path in a finite element method model. The opportunity to develop a tool that could be used to virtually simulate the implementation of FBG sensors in different structures, without been asked to the user an extensive knowledge about the FBG sensors was the main motivation to create this software. Together with this user-manual you can find a folder containing files that are used in the tutorial section 7. If you found this software useful to your work, please give some recognition to the authors and cite this software together with the article DOI.

Message from the Software main author, Gilmar F. Pereira.

Hi!! First of all, thanks for your interest in this software, my name is Gilmar F. Pereira and I'm FBG_SiMul main developer. I created the FBG_Simul while I was a PhD student at Denmark Technical University, as part of my thesis topic. I work with damage tolerance design and structural health monitoring of composite structures, and if you want to know more about the theory behind the code please take a look in some of my articles. You can find my complete list of publications here: DTU Orbit.

I would like to acknowledge:

- The seventh Framework Programme (FP7) for funding the project MareWint (Project reference: 309395) as Marie-Curie Initial Training;
- My supervisors, Lars P. Mikklesen and M. McGugan, for all the support during the software development;
- Denmark Technical University, Department of Wind Energy as my host institution. (Please visit DTU Wind website: DTU Wind)

For any question/feedback about the software feel free to contact me: gfpe@dtu.dk.

All the best and have a good FBG simulation,
Check out my Linkedin profile: Gilmar LinkeDin.

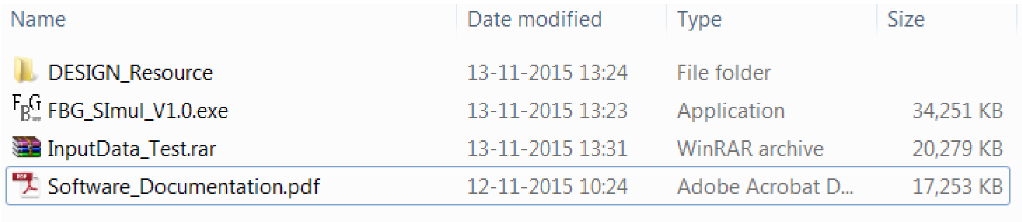
1.3 Software files

The software is provided in two formats:

- Standalone file, as **.exe** format; no dedicated software is requirement (no need to install python), but it is impossible to modify the source code;
- Python file; the user can modified the code, but python software/programming language compiler needs to be installed in the computer;

1.3.1 Standalone version

The Standalone version of the FBG_SiMul software is a **.exe** file that can be executed in any Windows machine, and it does not requires Python to be installed. This format can be useful for users that do not have any programming knowledge, or for an easy distribution of the software, as only 1 file is needed. However, the code cannot be accessed or modified.



Name	Date modified	Type	Size
DESIGN_Resource	13-11-2015 13:24	File folder	
FBG_SImul_V1.0.exe	13-11-2015 13:23	Application	34,251 KB
InputData_Test.rar	13-11-2015 13:31	WinRAR archive	20,279 KB
Software_Documentation.pdf	12-11-2015 10:24	Adobe Acrobat D...	17,253 KB

Figure 1.1: FBG_SiMul standalone version.

To run the software in Standalone, double-click the file **"FBG_Simul_V1.0.exe"** using a Windows machine. The folder **"Design_Resource"** contains pictures used by the software. The **"Input-Data_test.rar"** contains examples of input files that can be used by the software (files used in the tutorial presented in section 7).

1.3.2 Python version

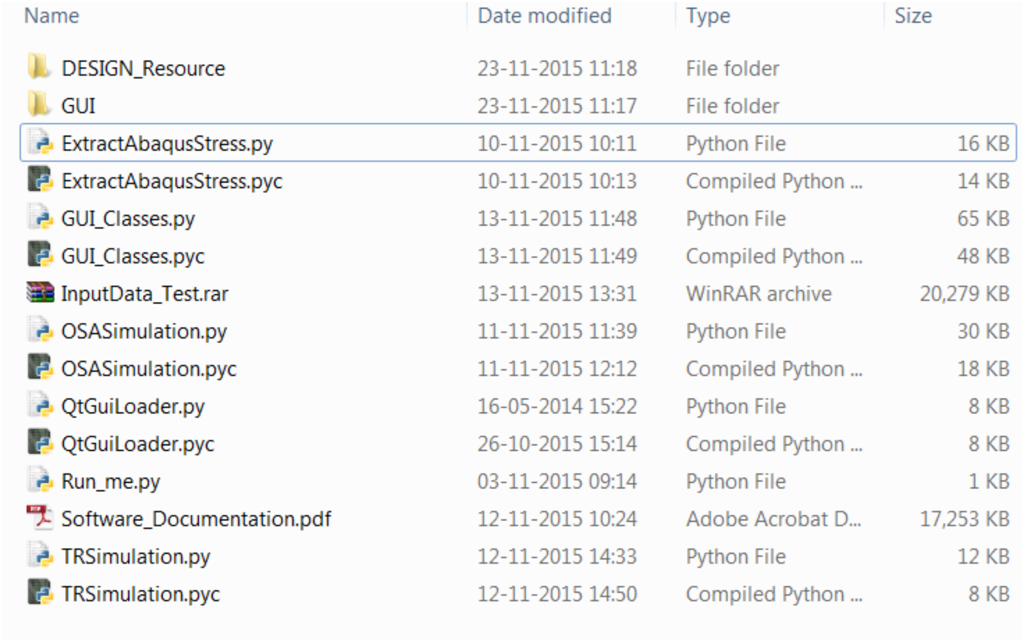
The other option to run the FBG_SiMul software is by using the Python version. This version requires the following software and modules to be installed:

- Python 2.7.5;
- Modules: numpy; matplotlib; PyQt4; math; and cmath;
- Suggested: QTdesigner;

Python web-page: Python.org;
Portable python version with all the packages and modules need (Suggested): winpython.sourceforge.net;

Note, all these software's are open source, meaning that you can installed without the need to purchase a license.

The FBG_SiMul was developed using python *2.7.5* and *QT designer*. It source code can be found in the folder **"Python_version"**, as shown in figure 1.2.



Name	Date modified	Type	Size
DESIGN_Resource	23-11-2015 11:18	File folder	
GUI	23-11-2015 11:17	File folder	
ExtractAbaqusStress.py	10-11-2015 10:11	Python File	16 KB
ExtractAbaqusStress.pyc	10-11-2015 10:13	Compiled Python ...	14 KB
GUI_Classes.py	13-11-2015 11:48	Python File	65 KB
GUI_Classes.pyc	13-11-2015 11:49	Compiled Python ...	48 KB
InputData_Test.rar	13-11-2015 13:31	WinRAR archive	20,279 KB
OSASimulation.py	11-11-2015 11:39	Python File	30 KB
OSASimulation.pyc	11-11-2015 12:12	Compiled Python ...	18 KB
QtGuiLoader.py	16-05-2014 15:22	Python File	8 KB
QtGuiLoader.pyc	26-10-2015 15:14	Compiled Python ...	8 KB
Run_me.py	03-11-2015 09:14	Python File	1 KB
Software_Documentation.pdf	12-11-2015 10:24	Adobe Acrobat D...	17,253 KB
TRSimulation.py	12-11-2015 14:33	Python File	12 KB
TRSimulation.pyc	12-11-2015 14:50	Compiled Python ...	8 KB

Figure 1.2: FBG_SiMul *Python* version.

The software was coded using an object-oriented approach and its structure is shown in figure 1.3. The functions used to connect the user-interface (QT Designer) with the code are in the folder **"GUI"**.

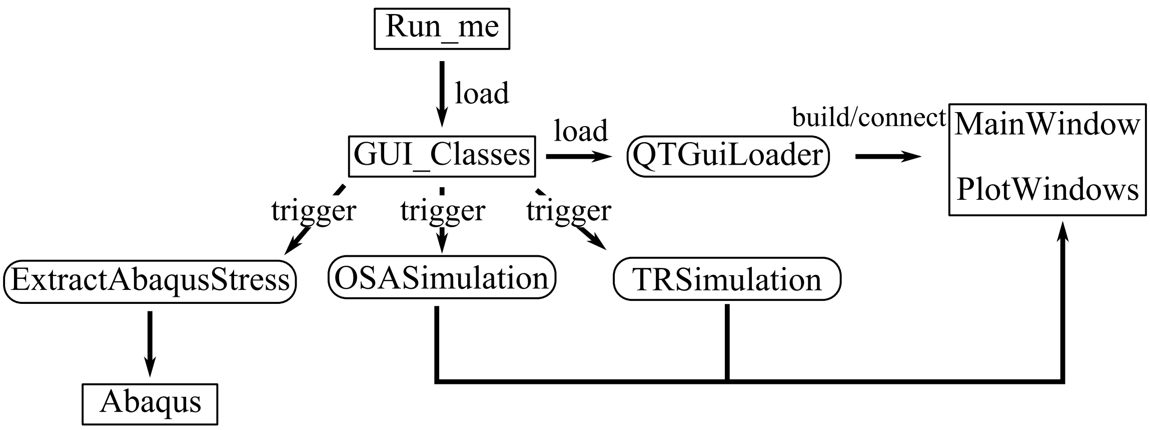


Figure 1.3: Python programming structure.

1.3.2.1 Files/Structure

⋆ "Run_me.py"

To start FBG_SiMul code run the file "Run_me.py".
This file will load the classes and packages needed to run the software.
start/load ⇒ class "GUI_Classes".

⋆ "GUI_Classes.py"

This class contains all the functions needed to load and start the user-interface; it checks if the input data have the right format; and it triggers the FBG analyses classes.

start/load ⇒ "ExtractAbaqusStress.py";
start/load ⇒ "OSASimulation.py";
start/load ⇒ "TRSimulation.py";

Main Functions:

- **__init__**: It converts the QTdesign file in a compiled Python Script using the class "QTGUILoader"; it connects all the actions in the user-interface with the Python Code (Ex: pressing buttons).
- **actionSubmitToAbaqus**: It checks if all the input data was inserted and have the correct format; it calls the class ExtractAbaqusStress, which builds the python script that is later submitted to Abaqus;
- **actionOsaGenerate**: It checks if all the input data was inserted and have the correct format; It calls the class OSASimulation, which simulates the FBG reflected spectrum for a specific time increment;
- **actionTimeResponseGenerate**: It checks if all the input data was inserted and have the correct format; It calls the class TRSimulation(), which simulates the FBG time response;

All the other functions and functionalities are explained in the file "GUI_Classes.py" header.

⋆ "QTGuiLoader.py"

"GUI_Classes.py" start ⇒ "QTGuiLoader.py" ⇒ "MainWindow.py" and "PlotWindow.py"

Class that loads a QWidget designed in QT Designer as MainWindow, Dialog or Widget. This class connects the actions (ex: pressing a button) with the actions (ex:actionSubmitToAbaqus) in the "GUI_Classes.py".

⋆ "ExtractAbaqusStress.py"
"GUI_Classes.py" triggers ⇒ "ExtractAbaqusStress.py"

Class file to extract the stress and strain along a predefined path in a FEM (Abaqus) model. This class writes a Python file that later is submitted to Abaqus.

Class input:

- **Odbpath**: String (Mandatory); path to Abaqus .Odb file;
- **Writefolder**: String (Mandatory); path where the output file will be written;
- **PathCoordinates**: Dictionary (Mandatory); dict. with the paths coordinates;
- **RotateAxis**: Integer (Mandatory); command to rotate the coordinate system; 0- Default; 1- User-Defined; 2- Vector with new axis direction;
- **DisplacementStep**: Integer (Mandatory); command to extract the stress/strain file for all time increments or for a specific time increment; 0- all time increments; 1- Specific time increment;
- **SpecificDisplacement**: Integer (Mandatory); number of the specific time increment, if DisplacementStep=0 the SpecificDisplacement="None";
- **UserDefinedCoordinate**: String (Mandatory); string with the name of the user defined coordinatesystem, if RotateAxis=0 or =2, the UserDefinedCoordinate="None";
- **VectorNewCoordinate**: Dictionary (Mandatory); dict. with the new coordinate system directions; VectorNewCoordinate[0] - origin point; VectorNewCoordinate[1] - x direction; VectorNewCoordinate[2] - y direction; if RotateAxis=0 or =1, the VectorNewCoordinate="None";

Main Functions:

- **__init__**: It loads all the input data;
- **createPyFile**: It creates the Python file that later is submitted to Abaqus;
- **submitAbaqus**: It submits the python file that was generated before to Abaqus; options: with or without Abaqus-GUI;

⋆ "OSASimulation.py"
"GUI_Classes.py" triggers ⇒ "OSASimulation.py"

Class file to simulate the FBG reflected spectrum for a specific time increment.

Class input:

- **Filename**: string (Mandatory); input file path;
- **NumberFBG**: Integer (Mandatory); number of FBG sensors per optical fibre;
- **FBGlength**: Float (Mandatory); length of the FBG;
- **Tolerance**: Float (Mandatory); value added (±) to the FBG length;
- **SkipRow**: Integer (Mandatory); number of rows skipped during data loading;
- **FBGPosition**: List (Mandatory); list with the grating position along the path;
- **InputUnits**: Integer (Mandatory); input file units: 0- meters; 1- mm;
- **FBGOriginalWavel**: List (Mandatory); list with the FBG original (non-deformed) wavelength;
- **PhotoElasticParam**: Float (Mandatory); photo-elastic parameter;
- **InitialRefractiveIndex**: Float (Mandatory); initial refractive index parameter;
- **MeanChangeRefractiveIndex**: Float (Mandatory); mean change of the refractive index parameter;
- **FringeVisibility**: Float (Mandatory); fringe visibility parameter;
- **MinBandWidth**: Float (Mandatory); simulation minimum bandwidth;
- **MaxBandWidth**: Float (Mandatory); simulation maximum bandwidth;
- **SimulationResolution**: Float (Mandatory); simulation resolution, as wavelength increment from MinBandWidth to MaxBandWidth;
- **FBGDirection**: Integer (mandatory); Longitudinal direction of the optical fibre in the FEM model: 0- xx direction, 1-yy direction, 2-zz direction;

Main Functions:

- **__init__**: It loads the input file containing the stress and strain along the fibre path. Next, it creates a local variable where the strain and stress is sorted per FBG sensor.

  Variable format: *Self.FBGArray["FBG-number"]["Stress/strain"]*;

- **undeformedFBG**: Functions that simulates the FBG reflected spectrum for an non-deformed state.

  Output variable format: *Self.OReflect["wavelength/reflec"]*;

- **UniformStrain**: Functions that simulates the FBG spectrum considering only uniform strain contribution.

  Output variable format: *Self.USReflect["wavelength/reflec"]*;

- **NonUniformStrain**: Functions that simulates the FBG spectrum considering non-uniform strain contribution.

  Output variable format: *Self.NUSReflect["wavelength/reflec"]*;

- **TransverseStrain**: Functions that simulates the FBG spectrum considering uniform strain and transverse stress contribution.

  Output variable format: *Self.TSZReflect["wavelength/reflec"]*;

- **FBGOutputSum**: It summarises the FBG reflected spectrum shape as wavelength shift ($\Delta\lambda_b$) and peak width variation ($\Delta\lambda_{WV}$).

  Output variable format: *self.FBGOutSum["FBG-number"]["WaveShift/WaveWidth"]*

⋆ "TRSimulation.py"
"GUI_Classes.py" triggers ⇒ "TRSimulation.py"

Class file to simulate the FBG time response, using multiple input files.

- **InputList**: List (Mandatory); list containing a path for every input files;
- **NumberFBG**: Integer (Mandatory); number of FBG sensors per optical fibre;
- **FBGlength**: Float (Mandatory); length of the FBG;
- **Tolerance**: Float (Mandatory); value added (±) to the FBG length;
- **SkipRow**: Integer (Mandatory); number of rows skipped during data loading;
- **FBGPosition**: List (Mandatory); list with the grating position along the path;
- **InputUnits**: Integer (Mandatory); Input file units: 0- meters; 1- mm;
- **FBGOriginalWavel**: List (Mandatory); list with the FBG original (un-deformed) wavelength;
- **PhotoElasticParam**: Float (Mandatory); photo-elastic parameter;

- **InitialRefractiveIndex**: Float (Mandatory); initial refractive index parameter;
- **DirectionalRefractiveP11**: Float (Mandatory); photo-elastic parameter in direction 11;
- **DirectionalRefractiveP12**: Float (Mandatory); photo-elastic parameter in direction 12;
- **FBGDirection**: Integer (mandatory), Longitudinal direction of the optical fibre in the FEM model: 0- xx direction, 1-yy direction, 2-zz direction;

Main Functions:

- **__init__**: It loads the input files containing the stress and strain along the path; next it creates a local variable for each FBG sensor containing the average, the maximum and the minimum stress and strain.

  Output variable format: *Self.FBGArrayTR["FBG-number"]["Stress/strain"]*;

- **calculate**: This function simulates the time response of the FBG array.

  Output variable format: *self.FBGTimeResponse["FBG-number"]["WaveShift/WaveWidth"]*

⋆ **GUI Files**: MyPlotMainWindowUI; PlotWindow_OSA; PlotWindow_TR
"QTGuiLoader.py" compiles ⇒ GUI Files

The GUI files were build using the tool QT designer, and the file QTGuiLoader compiles them to python code and links the actions (ex: press button) with the functions in "GUI_Classes.py".

Section 2

The FBG_SiMul software

Welcome to the FBG_SiMul V1.0 software.

This software was developed to support the implementation of FBG sensors into different type of structures or application. The software removes the need of an user expert in fibre optic technology, becoming more obvious the sensor response of a structural health monitoring solution using FBG sensors.

2.1 How is this done?

The shape and response of the FBG reflected spectrum (measured signal) depends on the way that the grating is deformed, i.e., if the grating is deformed uniformly or not, or if it gets ovalized due to transverse loads. Thus, the FBG_SiMul simulates the FBG signal based on the strain magnitude and the strain gradient along and transverse to the grating. If you want to know more about how the software simulates the signal based on the stress and strain field, please take a look on the paper publish simultaneous with the FBG_SiMul software[1].

The FBG_SiMul software conceptual structure is shown in figure 2.1. First, the software extracts a file/s from a FEM model, containing the stress and strain along a predefined path. This can be made for a specific/single time increment, or for multiple time increments (ex: dynamic models, time dependent behaviour). Next, the software will identify the elements that lay inside of each FBG by considering it length, and it will create a local variable per FBG sensor. This variable contains all information needed to simulate the FBG response, as the number of elements per grating, and the stress and strain field. Two simulation options are given to the user: reflected spectrum simulation for a specific time increment, where the shape of the reflected signal can be evaluated, useful to determine the quality of the measurement, or the presence of non-uniform strain and transverse stress; FBG time

[1] G. Pereira, M. McGugan, L.P. Mikkelsen, FBG_SiMul V1.0: Fibre Bragg Grating Signal Simulation Tool for Finite Element Method Models, SoftwareX, 2016

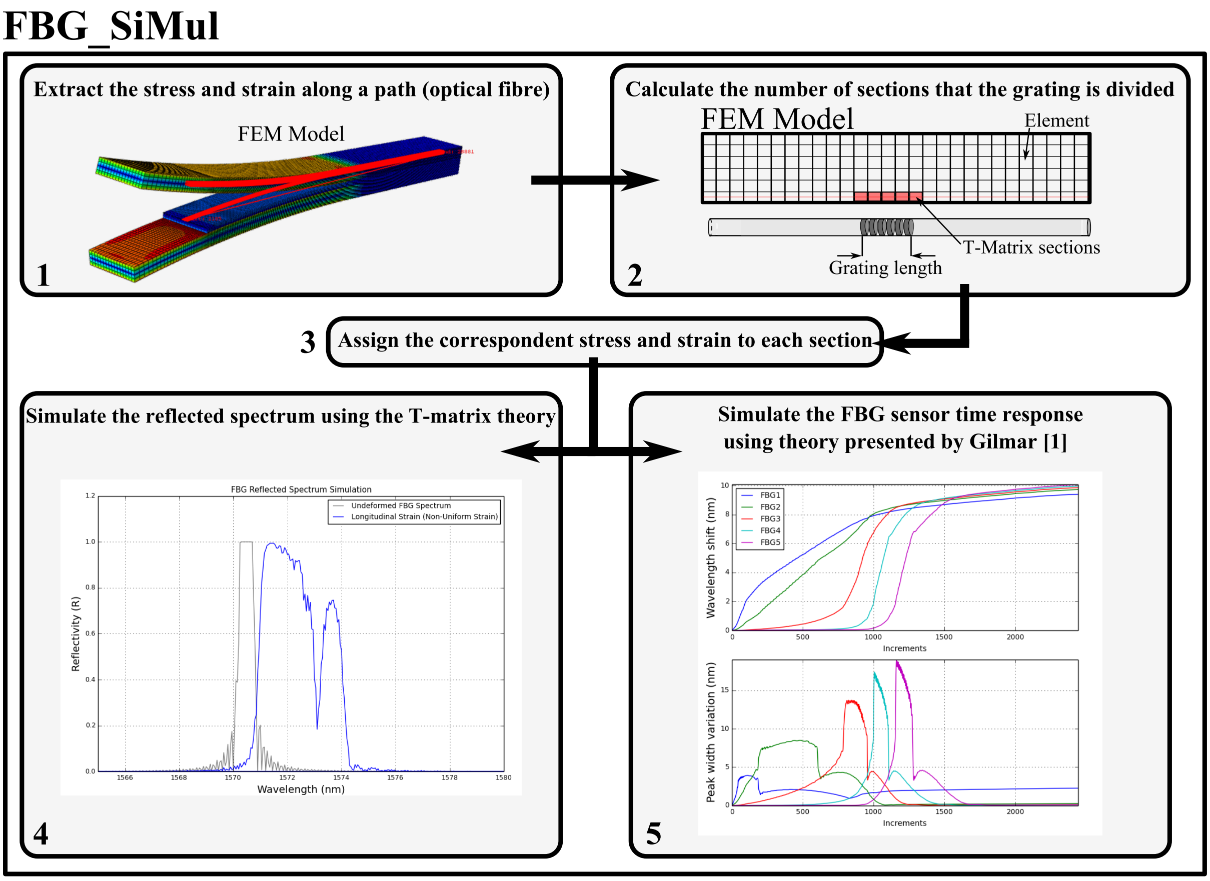


Figure 2.1: FBG_SiMul software conceptual structure.

response, where the sensor response is simulate along time, which is useful to simulate the behaviour of the sensor to time-dependent or dynamic loading.

To do this the FBG_SiMul software is divided in 4 tabs:

- **Tab 1- Software:** Software cover page that contains information about the different tabs and functionalities, and buttons to open the user-manual and information about the software and the author;
- **Tab 2- Extract Stress/Strain along Optical Fibre (Abaqus):** Tab to extract the stress and strain file from a FEM model (stage 1 in figure 2.1);
- **Tab 3- FBG Spectrum Simulation (Specific Step):** In this tab, the stress and strain is assigned to each FBG, and its reflected spectrum is simulated (stage 2 and 3 in figure 2.1);
- **Tab 4- FBG Signal variation (Time Response):** In this tab, the stress and strain from multiple files is assigned to each FBG, and its time response is simulated (stage 2 and 4 in figure 2.1);

Section 3

Tab 1: Software

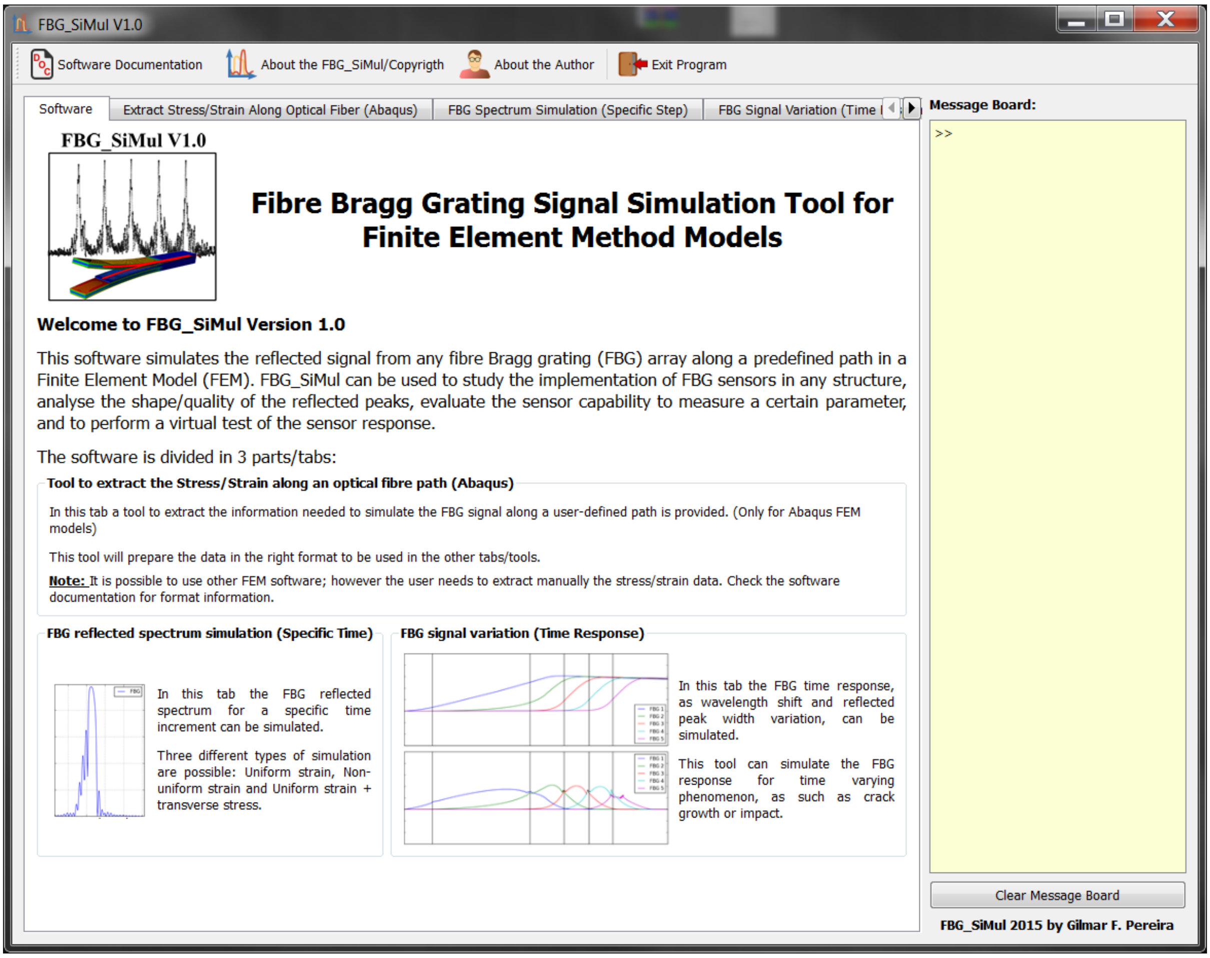


Figure 3.1: Tab 1: Software cover page.

This is the FBG_SiMul cover page, where some information about the software structure is presented.

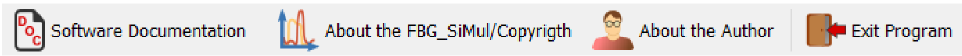


Figure 3.2: FBG_SiMul toolbox.

In the top of the software window a toolbox contain some useful buttons (see figure 3.2):

- **Software Documentation:** Push button that opens the software documentation pdf;
- **About FBG_SiMul/copyrigth:** Information about the software copyright;
- **About the Author:** Information about the author and a link to the author publications list;
- **Exit Program**: Push button that closes the software;

A message board located on the right side of the software window is always visible (see figure 3.1). In here, error messages are presented to help the user to locate possible input errors; and, messages are presented when the main functions are successfully completed.

Section 4

Tab 2: Extract Stress/Strain along Optical Fibre (Abaqus)

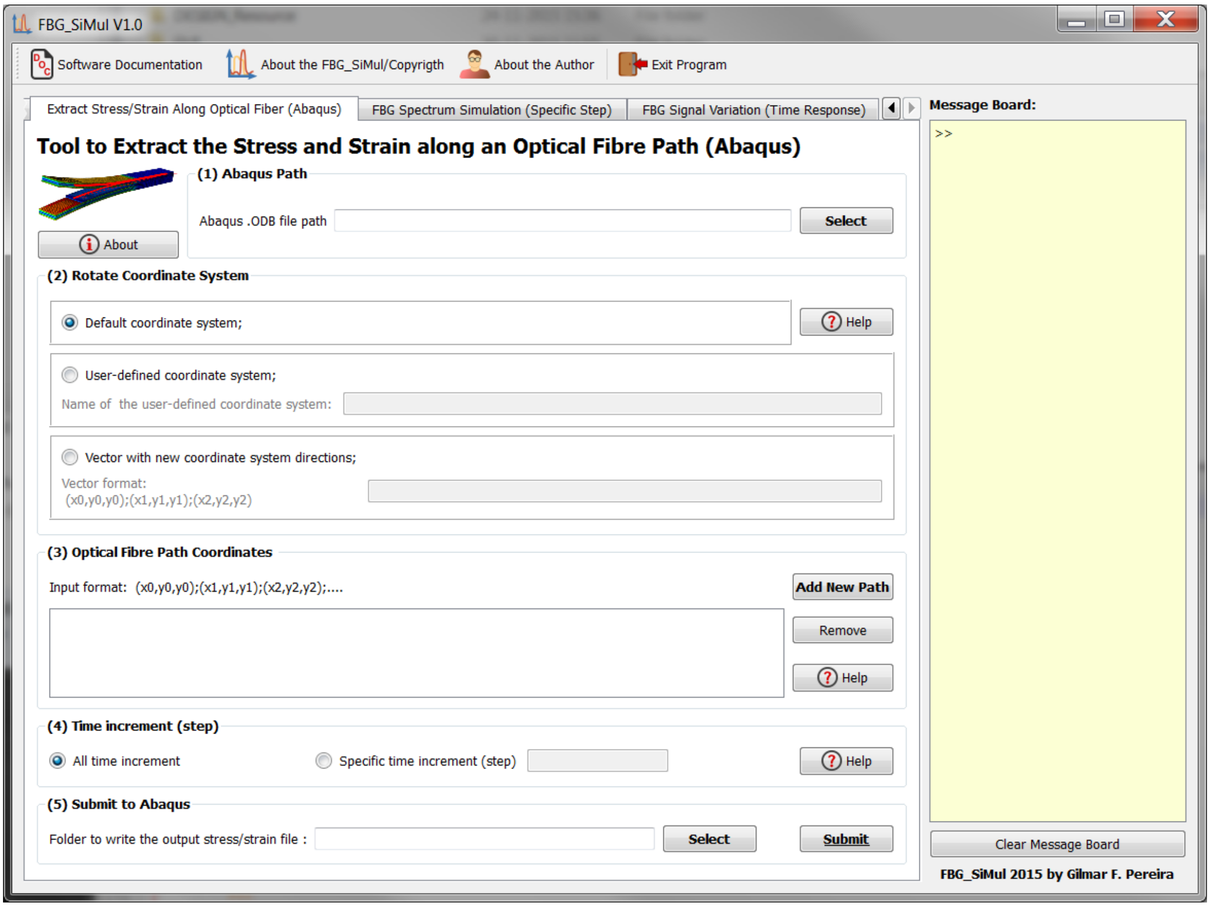


Figure 4.1: Tab: Extract Stress/Strain along Optical Fiber (Abaqus).

This tab provides a tool to extract the stress and strain along a pre-defined path in a finite element model (FEM). The output file/s generated at this tab will be used later to simulate the FBG spectrum.

Why is this tab important?

In order to simulate the FBG spectrum, the stress and strain along the grating length are needed. Thus, this tab will help the user to extract the stress and strain along a pre-defined path, and it will ensure that the output file have the right format and it is named correctly.

(If you want to know more about this please take a look on the article that was published together with the software)

Note: This tool was developed for ABAQUS FEM models. Nevertheless, if you want to simulate the FBG response by using a different FEM software, please extract the stress/strain along the optical fibre path manually, and ensure that the output file have the following format: (see figure 4.2)

X	LE11_B1-Ins1	LE22_B1-Ins1	LE33_B1-Ins1	s11_B1-Ins1	s22_B1-Ins1	s33_B1-Ins1
0.	1.12888E-03	-1.54449E-03	226.703E-06	1.80016E+03	-4.00785E+06	684.788E+05
5.0081E-03	748.226E-06	-2.22759E-03	1.02841E-03	72.426E+03	-6.39418E+06	684.788E+05
10.0163E-03	891.128E-06	-2.70833E-03	1.2019E-03	-183.114E+03	-8.0043E+06	496.052E+05
15.0244E-03	839.193E-06	-2.49985E-03	1.02647E-03	-360.335E+03	-7.61629E+06	49.2387E+03
20.0325E-03	831.48E-06	-2.48874E-03	1.0334E-03	-341.1E+03	-7.55664E+06	99.5774E+03
25.0396E-03	837.537E-06	-2.44396E-03	989.868E-06	-302.477E+03	-7.43431E+06	29.8163E+03
30.0472E-03	839.293E-06	-2.38654E-03	943.746E-06	-253.433E+03	-7.26464E+06	-25.6479E+03
35.0557E-03	848.987E-06	-2.3517E-03	918.342E-06	-166.325E+03	-7.12319E+06	-15.1241E+03
40.0638E-03	862.026E-06	-2.30326E-03	879.288E-06	-60.6781E+03	-6.94081E+06	-22.8389E+03
45.0719E-03	873.314E-06	-2.26161E-03	857.699E-06	71.8022E+03	-6.7424E+06	38.1901E+03
50.0798E-03	862.469E-06	-2.17531E-03	809.561E-06	142.085E+03	-6.4608E+06	27.6612E+03
55.0884E-03	839.991E-06	-2.11914E-03	788.754E-06	137.555E+03	-6.29434E+06	26.7881E+03
60.0962E-03	834.115E-06	-2.09026E-03	769.834E-06	138.841E+03	-6.21747E+06	-239.886
65.1032E-03	825.111E-06	-2.0791E-03	769.471E-06	125.431E+03	-6.18708E+06	5.10757E+03
70.1109E-03	817.827E-06	-2.05892E-03	758.758E-06	117.134E+03	-6.1357E+06	-10.6767E+03
75.1195E-03	811.708E-06	-2.04344E-03	754.634E-06	121.895E+03	-6.08399E+06	-1.61467E+03
80.1273E-03	806.867E-06	-2.02559E-03	747.937E-06	133.125E+03	-6.02347E+06	5.53519E+03
85.1356E-03	803.177E-06	-2.00968E-03	744.454E-06	155.186E+03	-5.95884E+06	27.9912E+03

Figure 4.2: Input file format.

- **8 columns separated by tab;**
- **Variables per Column:** -1st-: FBG path length/distance; -2nd-: Strain in direction 11; -3rd-: Strain in direction 22; -4th-: Strain in direction 33; -5th-: Stress in direction 11; -6th-: Stress in direction 22; -7th-: Stress in direction 33;

4.1 (1) Abaqus Path

In station **(1) Abaqus path**, the user selects the .ODB Abaqus file.

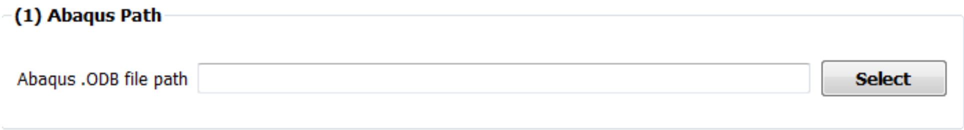


Figure 4.3: Abaqus Path selection.

4.2 (2) Coordinate System Rotation

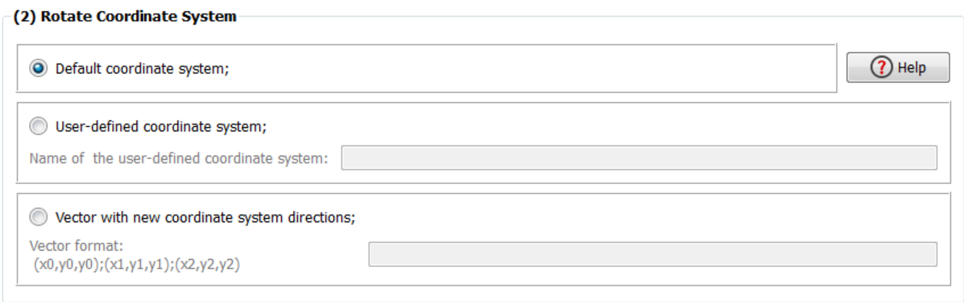


Figure 4.4: Coordinate System Rotation.

In station **(2) Rotate Coordinate System**, the user have the possibility to rotate the coordinate axis. This allows the user to match the model axis direction with the optical fibre direction, as shown in figure 4.4 and 4.5.

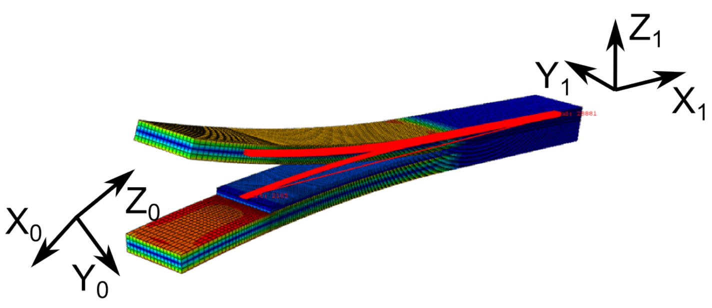


Figure 4.5: FEM model coordinate system rotation.

Rotate Coordinate System options:

- **Default**: The default model coordinate system is used;
- **User-defined coordinate system**: A user-defined coordinate system, created during the model development, is used (see figure 4.6). **Input format:** name of the user-defined coordinate system. (Example: Assemby_Blade-1_ORI-1);

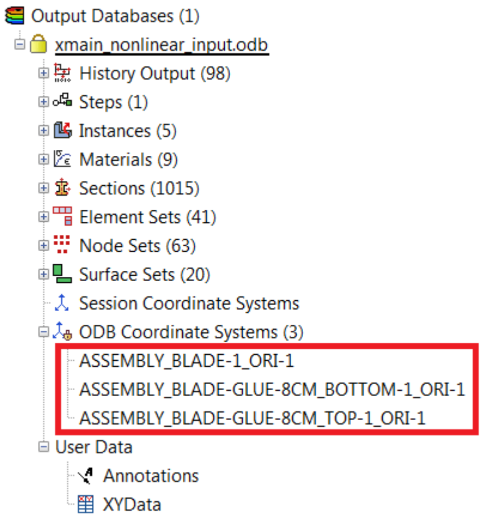


Figure 4.6: User defined coordinate system created in the Abaqus FEM model.

- **Vector with the new coordinate system directions**: A vector with new coordinate system directions is inserted by the user. **Input format**: $(x_0, y_0, z_0);(x_1, y_1, z_1);(x_2, y_2, z_2)$. The index 0 represents the origin coordinate, the index 1 represents the coordinated for the direction 1/xx, and the index 2 represents the coordinated for the direction 2/yy.

  Example (figure 4.7) : *(0,0,0);(0,1,0);(1,0,0)*

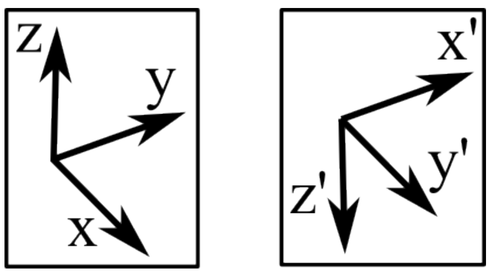


Figure 4.7: Example of vector with new coordinate system directions.

4.3 (3) Optical Fibre Path Coordinates

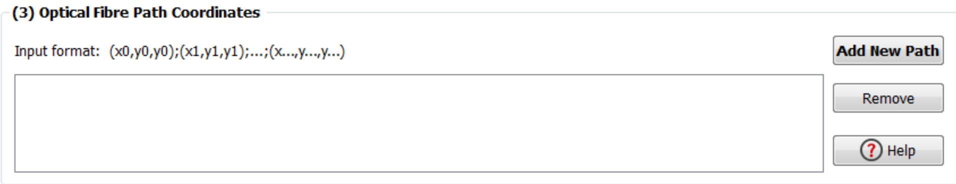


Figure 4.8: Optical fibre path coordinates.

In station **(3) Optical Fibre Path Coordinates**, the user is asked to insert the optical fibre path coordinates. This path represents the virtual location of the optical fibre line in the FEM model.

Each path should have a minimum of two points; but multiple paths can be inserted.

Push Buttons:

- **Add new path**: Open dialogue window to insert the optical fibre path.

  Input format: $(x_0, y_0, z_0); (x_1, y_1, z_1); ...(x_{...}, y_{...}, z_{...})$ (see figure 4.9);

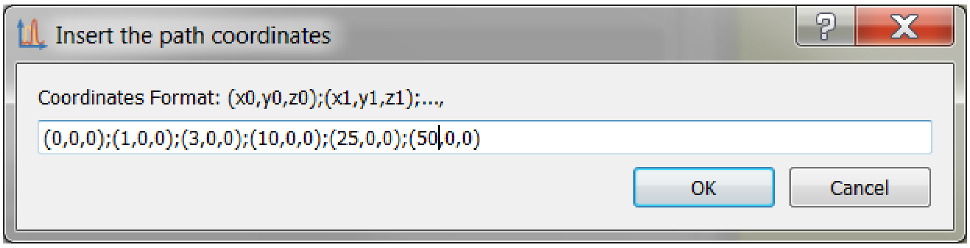


Figure 4.9: Add new path coordinates dialogue window.

- **Remove button**: Deletes the last inserted path.

A path number is assigned to each new path inserted, as shown in figure 4.10. This path number is used to name the output file (keep it noted).

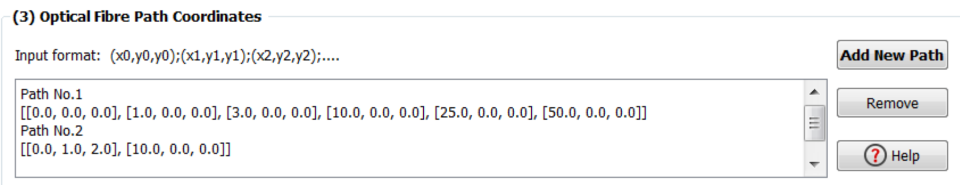


Figure 4.10: Optical fibre path coordinates numbering.

4.4 (4) Time Increment (Step)

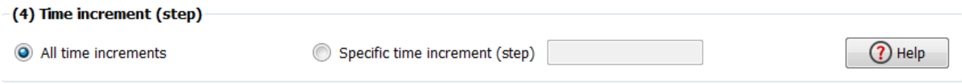


Figure 4.11: Time increment selection tool.

In station **(4) Time increment**, the user can select between two time increment options to extract the stress/strain: All time increments or specific time increment.

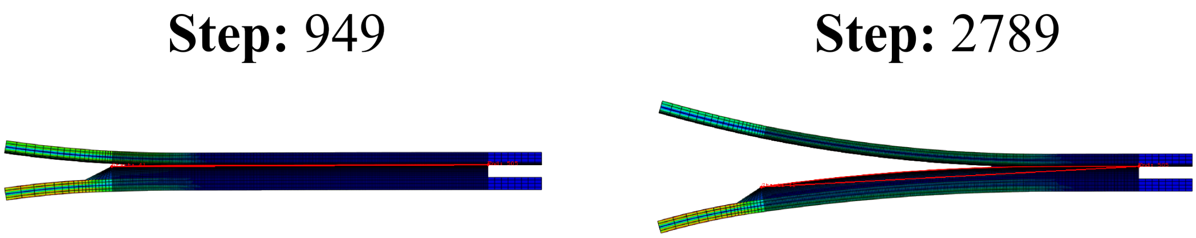


Figure 4.12: Different time increments in a FEM model (Delamination of a double cantilever beam).

A FEM model can have only one time increment (Ex: a linear loading model), or several time increments (Ex: a crack growth model), as shown in figure 4.12. Based on this, the user can choose to extract a specific time increment, by inserting the increment number, or extract all the time increments.

Note: The software will name the output files accordingly with the increment number.

4.5 (5) Submit to Abaqus

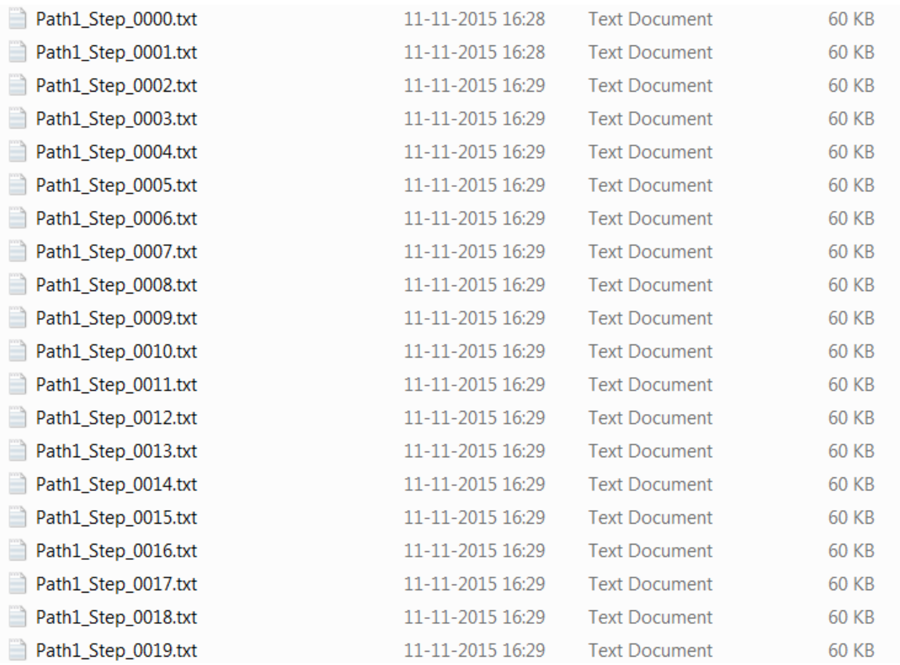
Figure 4.13: Submit to Abaqus.

In station **(5) submit to Abaqus**, the user is asked to select a folder where the output file/s will be saved. The output files are named in the following format: *Pathxx_Step_yyyy.txt*, where xx is the path number defined in the section 4.3- Optical Fibre Path Coordinates, and yyyy is the time increment number.

Push Buttons:

- **Submit**: Before submitting to Abaqus, the software will preform an input check; if all the data is correct, an Abaqus python script is created in a temporary folder and submitted to Abaqus; Abaqus GUI interface will open and the data is generated; After all data has been generated simply close Abaqus (the software will delete the temporary folder and all auxiliary files).

An example of the files obtained by this tab is shown in figure 4.14. Each file have the same format as the file presented in figure 4.2.



Path1_Step_0000.txt	11-11-2015 16:28	Text Document	60 KB
Path1_Step_0001.txt	11-11-2015 16:28	Text Document	60 KB
Path1_Step_0002.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0003.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0004.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0005.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0006.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0007.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0008.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0009.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0010.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0011.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0012.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0013.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0014.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0015.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0016.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0017.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0018.txt	11-11-2015 16:29	Text Document	60 KB
Path1_Step_0019.txt	11-11-2015 16:29	Text Document	60 KB

Figure 4.14: File output.

Section 5

Tab 3: FBG Spectrum Simulation (Specific Step)

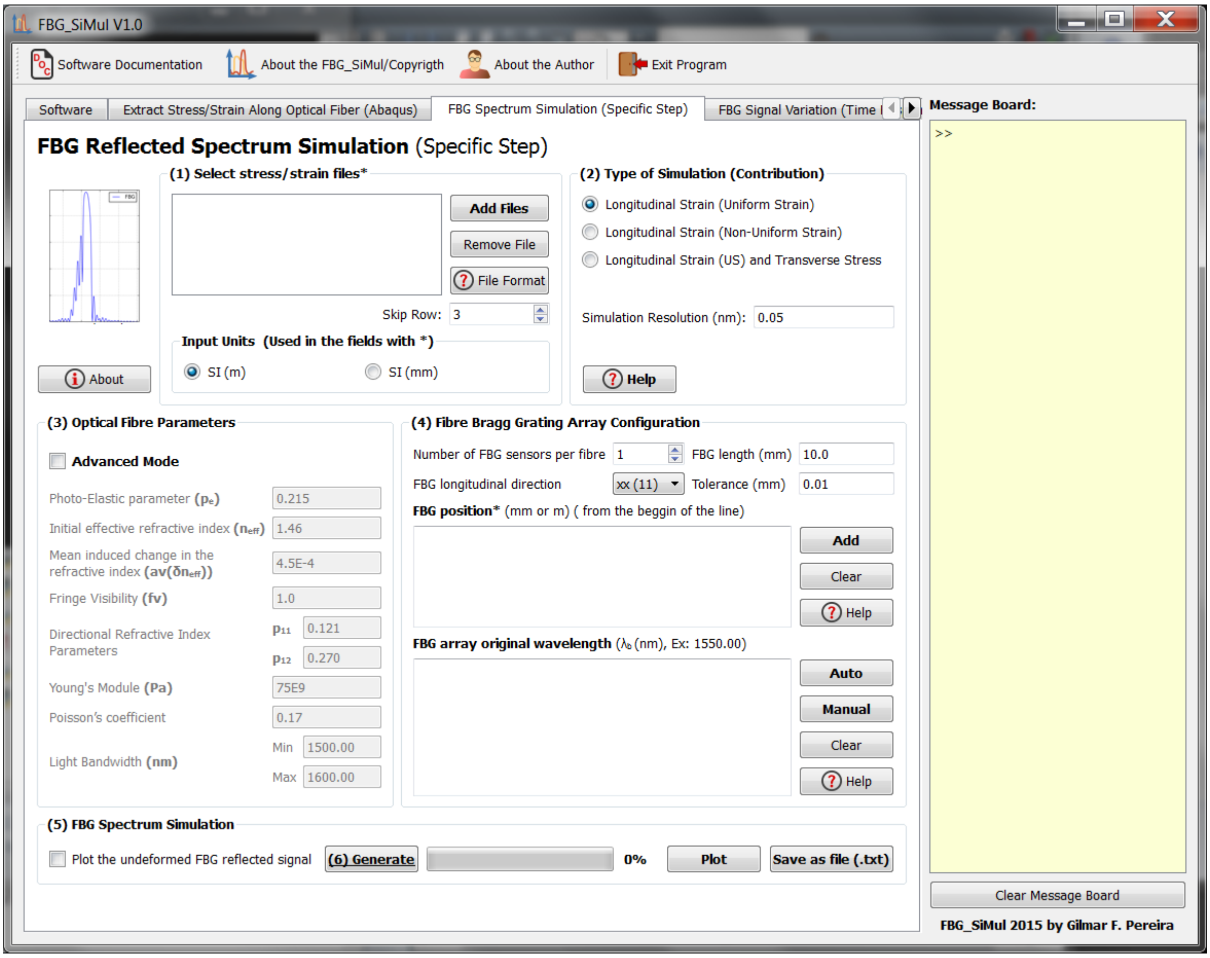


Figure 5.1: Tab: FBG Spectrum Simulation (Specific Step).

In this tab, the user can simulate the FBG reflected spectrum by using the stress/strain file generated in the previous tab. The theory used is presented in the article published together with this software[1]. The output from this tab is a plot or a .txt file of the FBG reflected signal(s) at a specific time increment.

In order to generate the plot some basic information about the length scales (Station (5)) and the fibre Bragg gratings properties (station (4)) must be given. The user can also include non-uniform strain effects and transverse strain effects in the plot of FBG reflected signal(s) (station (2)).

Tasks that can be done on this tab include:

- Finding the best sensor locations for your structure/loading;
- Planning the number and position of FBGs along a proposed optic fibre line;
- Checking if the wavelength shift is the expected and optimising available bandwidth;
- Comparing two or more proposed measurement lines;
- Evaluating possible distortion effects (measurement error);
- Investigating complex phenomena effects on the FBG response;
- Non-uniform strain effects;
- Transverse strain effects;
- Crack propagation;
- Singularities (material interfaces, glue lines, stress concentration features, etc.);

[1] G. Pereira, M. McGugan, L.P. Mikkelsen, FBG_SiMul V1.0: Fibre Bragg Grating Signal Simulation Tool for Finite Element Method Models, SoftwareX, 2016

5.1 (1) Select stress/strain files

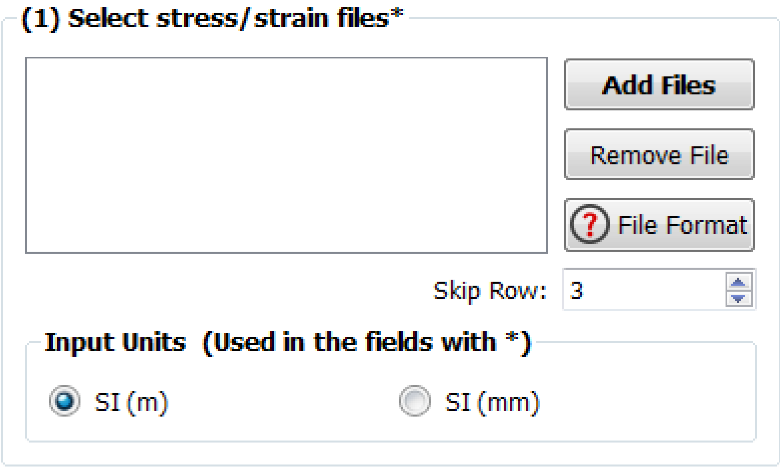


Figure 5.2: Select stress/strain files.

In station **(1) Select stress/strain files**, the user selects the stress/strain files (or series of files) they wish to work with. The files used here need to follow the format presented in section 4.

Several files can be added to the list, but only one file (highlighted by the user) will be used to generate the spectrum simulation.

Push Buttons:

- **Add Files**: opens a dialogue window to insert/load the input files.
- **Remove File**: removes the selected file.

The code can ignore the headings of the input file, using the value in **Skip Row**. The default value (Skip Row=3) is used for the files obtain through Tab 1, however the user can change the number of rows to skip in order to accommodate their own .txt file formatting.

Additionally, the user can select the units that the model was build, as metres (m) or millimetres (mm). This option will impact the values loaded from the input file, station (1), and the FBG position, station (4).

5.2 (2) Type of simulation (contribution)

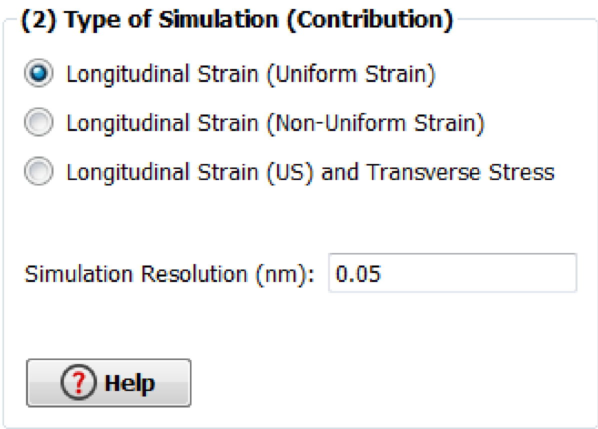


Figure 5.3: Type of simulation (contribution).

In station **(2) Type of simulation (contribution)**, the user chooses the type of simulation to be performed. If you want to know more about this please take a look on the article that was published together with this software.

Type of simulations:

- **Longitudinal Strain (Uniform Strain)**: In this option, only uniform strain acting in the longitudinal direction of the FBG sensor is considered; The shape of the reflected peak remains the same, and the wavelength shift $\Delta\lambda_b$ is calculated using an average strain along the grating length.
- **Longitudinal Strain (Uniform + Non-Uniform Strain):** In this option, the effect of Non-uniform strain acting in the longitudinal direction of the FBG sensor is considered. The strain variation along the sensor is used to calculate the change on the grating periodicity, which it is then used to simulate the shape of the reflected peak. The output from this method is the wavelength shift, $\Delta\lambda_b$, and the peak width variation, $\Delta\lambda_{wv}$.
- **Longitudinal Strain (Uniform Strain) and Transverse Stress:** In this option, the effect of uniform strain along the longitudinal direction of the FBG sensor and the transverse stress acting perpendicularly to the sensor are considered. The birefringent behaviour caused by the transverse stresses is added to the signal, and used to calculated the shape of the reflected peak. The output from this method is the wavelength shift, $\Delta\lambda_b$, and the peak width variation, $\Delta\lambda_{wv}$.

Additionally, the user can specify the simulation resolution. The simulation resolution correspond to the light bandwidth discretization, i.e., the amount of points that the reflected spectrum will be calculated. For a resolution of 0.05nm and a bandwidth of 1000nm (1500nm-to-1600nm light bandwidth), the spectrum simulated contains 20.000 points. Note that this parameter have a big impact on the simulation time.

5.3 (3) Optical Fibre Parameters

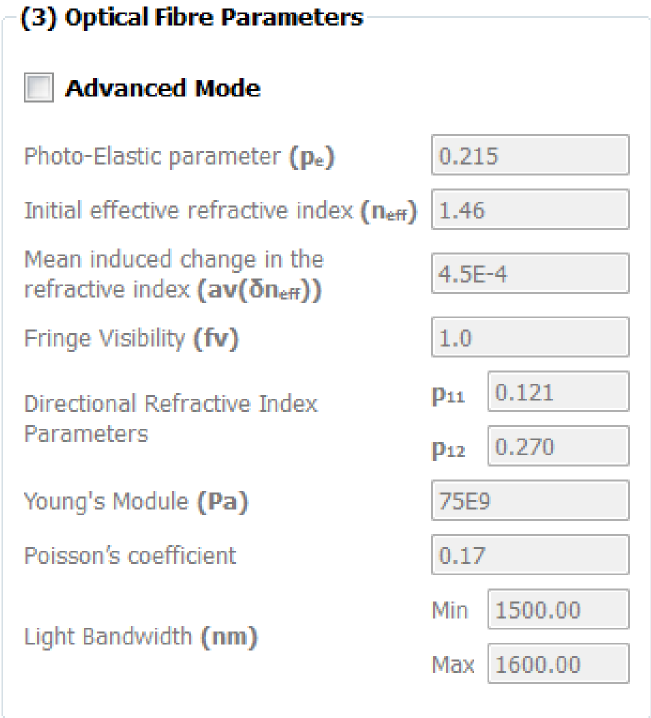


Figure 5.4: Optical Fibre Parameters.

In station **(3) Optical Fibre Parameters**, the user can choose to toggle/activate the advanced mode and specify their own values for the optical fibre parameters.

Parameters used by each type of simulation:

- **Longitudinal Strain (Uniform Strain)**: Photo-Elastic (p_e); Initial effective refractive index (n_{eff}); Mean induced change in the refractive index ($\delta\overline{n_{eff}}$); Fringe Visibility (fv);
- **Longitudinal Strain (Non-Uniform Strain):** Photo-Elastic (p_e); Initial effective refractive index (n_{eff}); Mean induced change in the refractive index ($\delta\overline{n_{eff}}$); Fringe Visibility (fv);
- **Longitudinal Strain (US) and Transverse Stress:** Photo-Elastic (p_e); Initial effective refractive index (n_{eff}); Directional Refractive Index (p_{11} and p_{12}); Young's Module; Poisson's coefficient;

The light bandwidth parameters define the light interval where FBG spectrum will be simulated. Note that all FBG original reflected peaks (λ_b) should be within this interval.

5.4 (4) Fibre Bragg Grating Array Configuration

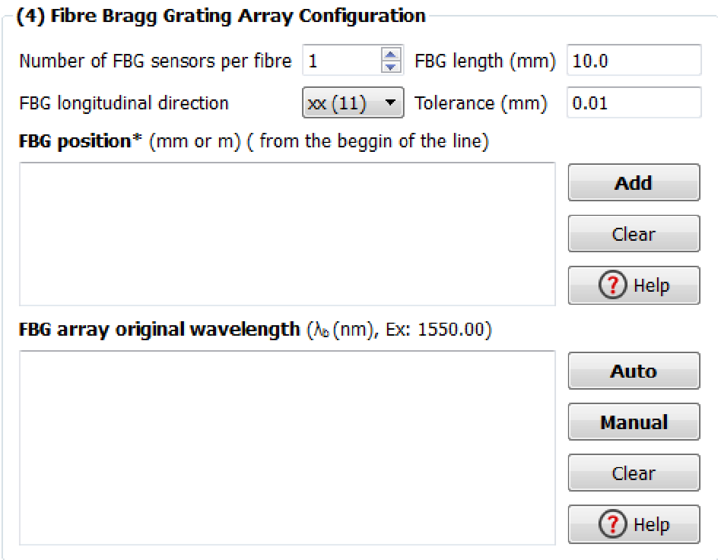


Figure 5.5: Fibre Bragg Grating Array Configuration.

In station **(4) Fibre Bragg Grating Array Configuration**, the user is asked to define the FBG array geometric configuration, as the number of sensors, its position, and length; and, to define the FBG original wavelength, which is given by the grating periodicity defined by the grating manufacturing process.

Geometric Parameters:

- **Number of FBG sensors per fibre:** Here, the user defines the number of "virtual" FBG sensors that will be simulated in the path previously extracted (Tab 2); A minimum of 1 FBG sensor is required;
- **FBG length (mm):** The user defines the FBG sensor length in millimeters; The default FBG length is between 5 and 10 mm;
- **FBG longitudinal direction:** Here, the user needs to specify the longitudinal direction of the FBGs accordingly with the FEM model coordinate axis. In most cases the fibre optic is aligned in the xx (11) direction, but depending on the model this measurement axis can be changed; Direction options: xx(11); yy(22); and zz(33);
- **Tolerance (mm):** The tolerance parameter defines the software length tolerance when is assigning the stress and strain to the correspondent FBG. With a tolerance of zero the simulation will only consider elements that are fully contained within the FBG gauge length. As example, if

no tolerance is defined for a FBG sensor with 5mm of length starting at 0 mm in the fibre optic line, the values correspondent to x=5.001 is not included (see red box figure 5.6); by specifying a tolerance value the influence of adjacent overlapping elements can also be included in the simulated response. A value of 10^{-3} FBG length could be considered appropriate.

X	LE11_Path1_Step:231	LE22_Path1_Step:231	LE33_Path1_Step:231	s11_Path1_Step:231	s22_Path1_Step:231	s33_Path1_Step:231
0.	1.15911E-03	-149.045E-06	0.	5.75273	1.33325	0.
500.009E-03	1.71432E-03	-176.076E-06	0.	8.5886	2.20278	0.
1.00001	2.23306E-03	-374.117E-06	0.	10.9242	2.11705	0.
1.50001	2.47143E-03	-434.36E-06	0.	12.0533	2.23756	0.
2.00001	2.65778E-03	-497.236E-06	0.	12.9073	2.24973	0.
2.50001	2.78618E-03	-536.396E-06	0.	13.5034	2.27971	0.
3.00001	2.88203E-03	-564.323E-06	0.	13.9507	2.30889	0.
3.50001	2.95204E-03	-584.904E-06	0.	14.2771	2.32929	0.
4.00001	3.00183E-03	-597.31E-06	0.	14.5133	2.35537	0.
4.50001	3.03612E-03	-605.516E-06	0.	14.6765	2.37509	0.
5.00001	3.05748E-03	-608.783E-06	0.	14.7816	2.39698	0.
5.50001	3.06821E-03	-608.744E-06	0.	14.8374	2.4167	0.
6.00001	3.07072E-03	-606.133E-06	0.	14.8552	2.43483	0.
6.50001	3.06525E-03	-600.422E-06	0.	14.8371	2.45458	0.
7.00001	3.0549E-03	-594.002E-06	0.	14.795	2.46913	0.

Figure 5.6: Example of the tolerance parameter.

- **FBG Position (mm or m):**

  In here, the user defines the FBGs position along the path. By pressing the **Add** button, the user will be prompted to input the length from the beginning of the line to the beginning of that particular FBG grating. This distance is defined by the first column of the input file (see figure 5.6), and should be inserted with the input units (mm or m) defined in station (1).

- **FBG array original wavelength ($\lambda_b(nm)$):**

  Each FBG has a original wavelength (λ_b) that is defined during the manufacturing process. The λ_b is the wavelength of the FBG reflected peak in it unstrained state. The array of original wavelength values must lie within the light bandwidth specified in station (3) (by default:between 1500 to 1600nm). The user can either press the **Auto** button to distribute the FBG's λ_b along the available bandwidth with equal spacing, or may choose to input each FBG original wavelength manually by pressing **Manual** button, perhaps in order to maximise the range of particular sensors based on the known loading regime.

5.5 (5) FBG Spectrum Simulation

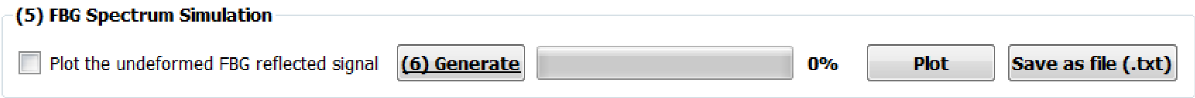


Figure 5.7: FBG spectrum simulation.

In station **(5) FBG Spectrum Simulation**, the user generates the reflected spectrum from the input file and the FBG parameters defined. By toggling the option **"plot the undeformed FBG reflected signal"** the software will generate two signals: the reflected spectrum of the FBG array in an unstrained state, and the reflected spectrum of the FBG array in the FEM model; this allows the user to compare the signal evolution. Once the simulation is completed a message appears in the message board, and then it is possible to display the output as an image, or save it as a .txt file.

Warning messages will be presented in the message board if any error or parameter is missing.

Push button:

- **Plot:** Plots the simulated reflected peak.

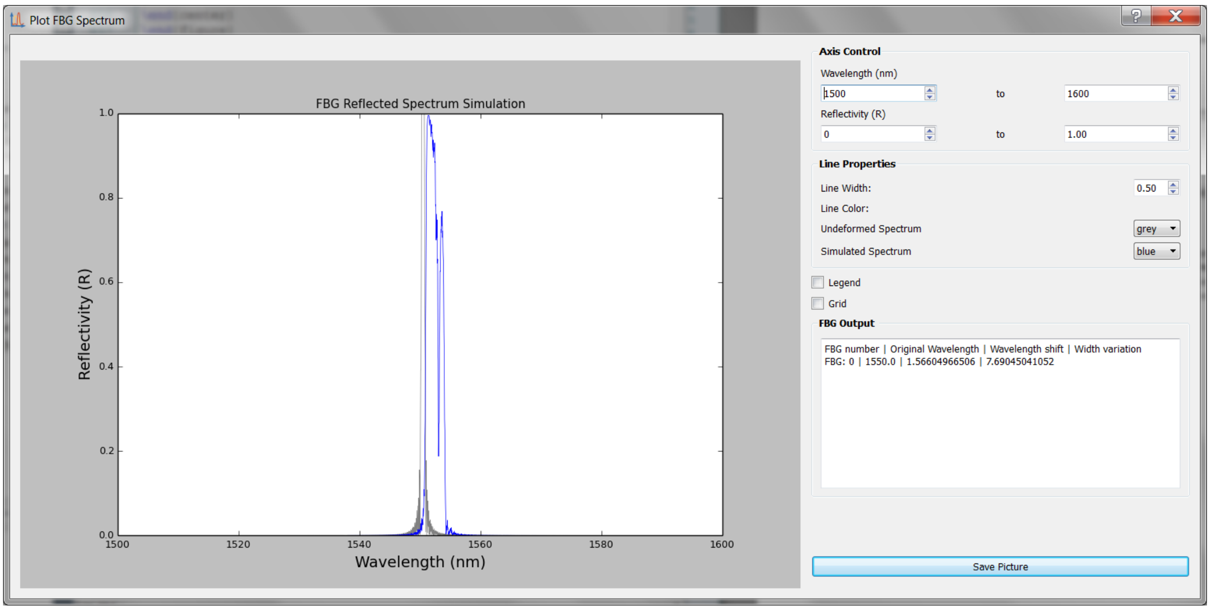


Figure 5.8: Plot window.

In the plot window, the user have the possibility to change the axis scale and the display layout. Also, a table with the summarize FBG output is displayed, where the FBG number, the original wavelength $\lambda_b(nm)$, the wavelength shift $\Delta\lambda_b(nm)$, and the width variation $\Delta\lambda_{wv}(nm)$ are shown. The user can save the plot by pressing the **Save Picture** push button.

- **Save as file (.txt)**: Save the simulated reflected signal as .txt file.

```
File name: D:/Work/MAREWINT/ResearchWork/FBG_Simulation_Software/20151112-FBG_SImul_V1.0/DCB-Path (Crack at 15 mm).txt
Type of Simulation: Longitudinal Strain (Non-Uniform Strain)

 ------FBG array configuration------
Number of FBG sensors: 1
FBG length (mm): 10.0
Tolerance (mm): 0.01
FBG position (mm): [15.0]
FBG original wavelength (nm): [1550.0]

FBG number | Original Wavelength | Wavelength shift | Width variation
FBG: 0 | 1550.0 | 1.56604966506 | 7.69045041052

Wavelength (nm)         Reflectivity:Undeformed Shape          Reflectivity: Longitudinal Strain (Non-Uniform
Strain)
1500.000000       0.000001        0.000000
1500.050000       0.000015        0.000014
1500.100000       0.000018        0.000016
1500.150000       0.000000        0.000000
1500.200000       0.000018        0.000017
1500.250000       0.000014        0.000013
1500.300000       0.000001        0.000001
1500.350000       0.000021        0.000020
1500.400000       0.000011        0.000009
1500.450000       0.000003        0.000003
1500.500000       0.000023        0.000021
1500.550000       0.000007        0.000006
1500.600000       0.000006        0.000006
1500.650000       0.000023        0.000022
1500.700000       0.000004        0.000003
1500.750000       0.000009        0.000009
1500.800000       0.000022        0.000020
1500.850000       0.000001        0.000001
1500.900000       0.000013        0.000013
1500.950000       0.000020        0.000018
1501.000000       0.000000        0.000000
1501.050000       0.000017        0.000017
1501.100000       0.000017        0.000015
1501.150000       0.000000        0.000000
1501.200000       0.000021        0.000020
```

Figure 5.9: Simulated reflected spectrum output as .txt file.

In the header of the .txt output file is presented information that describes and summarises the simulation, and in the main body the reflectivity in function of the wavelength.

Section 6

Tab 4: FBG Signal variation (Time Response)

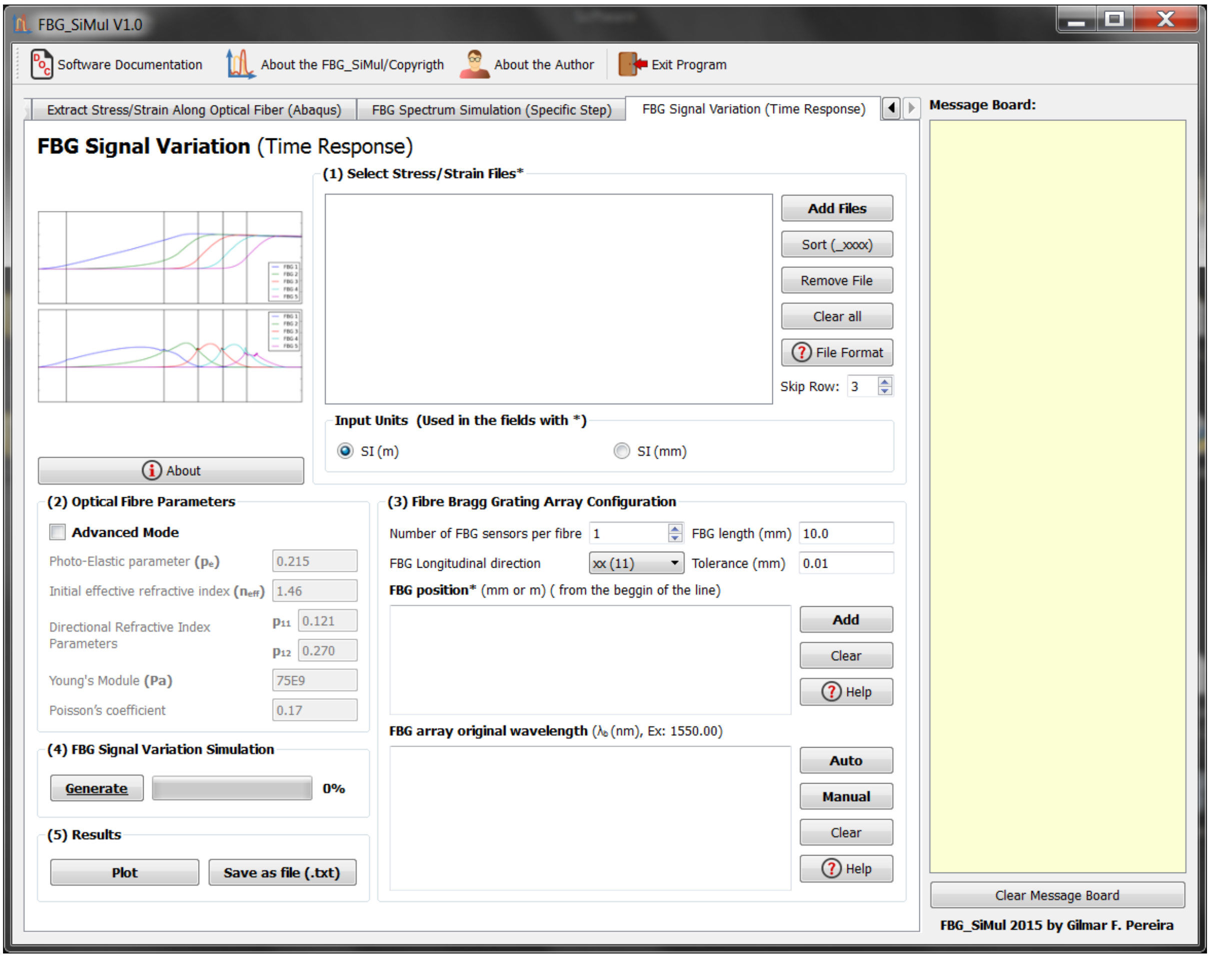


Figure 6.1: Tab: FBG Signal variation (Time Response).

In this tab, the user can simulate the FBG response using multiple input files (time response). Similar to the Tab 3- FBG Spectrum Simulation, this tool calculates the reflected signal of the FBG based on the stress/strain; however multiple input files are selected, and the output is presented as wavelength shift variation $\Delta\lambda_b$ and peak width variation $\Delta\lambda_{wv}$. This tool is useful to simulate the behaviour of the sensor to time-dependent or dynamic loading.

Advanced mode allows the user to specify various parameters of the optic fibre itself. However, standard values are already given, which are suitable for most fibre optic systems. (station (2))

Note: The type of simulation (contribution) can not be select in this stage. This because all the contributions, uniform strain, non-uniform strain and transverse stress, are included in the simulation of the sensor response.

Tasks that can be done on this tab include:

- Finding the best sensor locations for your structure/loading;
- Planning the number and position of FBGs along a proposed optic fibre line;
- Check the signal variation during the loading case;
- Comparing two or more proposed measurement lines;
- Evaluating possible distortion effects (measurement error);
- Investigating complex phenomena effects on the FBG response;
- Crack propagation;
- Damage detection;

6.1 (1) Select stress/strain files

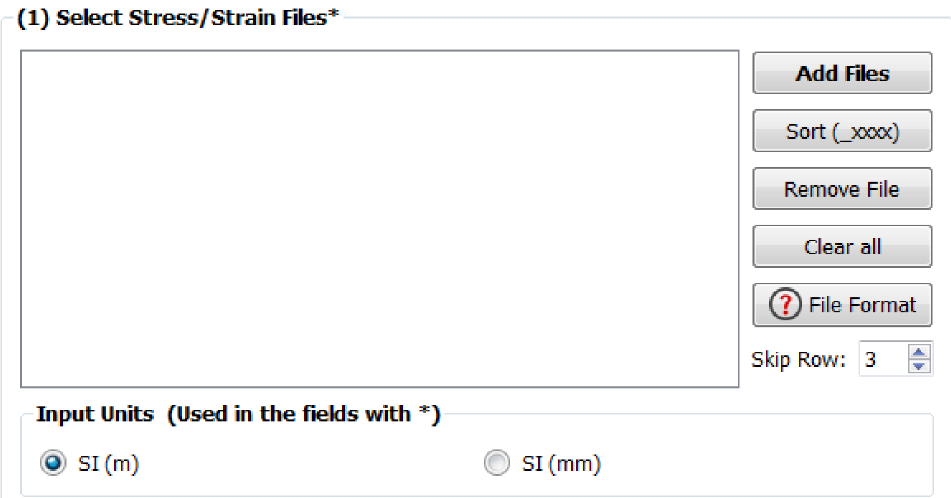


Figure 6.2: Select stress/strain files.

In station **(1) Select stress/strain files**, the user selects the stress/strain files (series of files) they wish to work with. The files need to follow the format presented in section 4, and the order that the files are displayed in the text box will be used as the order to simulate the time response.

Push Buttons:

- **Add Files**: opens a dialogue window to insert/load the input files;
- **Sort (_xxxx)**: sorts the input files based on the step number;
- **Remove File**: removes the selected file;
- **Clear all**: removes all the files;

The code can ignore the headings of the input file, using the value in **Skip Row**. The default value (Skip Row=3) is used for the files obtain through Tab 1, however the user can change the number of rows to skip in order to accommodate their own .txt file formatting.

Additionally, the user can selected the units that the model was build, as metres (m) or millimetres (mm). This option will have an impact on the values loaded from the input file, station (1), and the FBG position, station (3).

6.2 (2) Optical Fibre Parameters

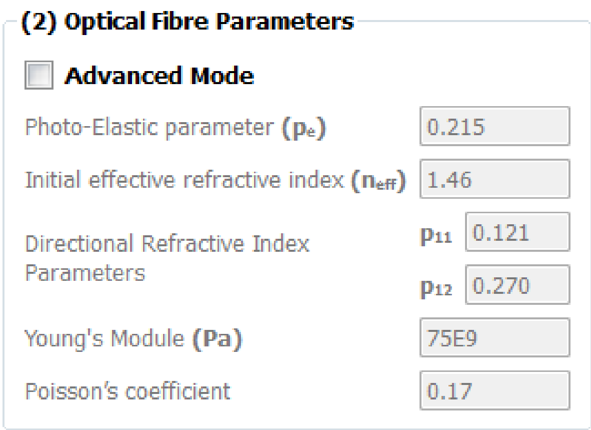


Figure 6.3: Optical Fibre Parameters.

In station **(2) Optical Fibre Parameters**, the user can choose to toggle/activate the advanced mode and specify their own values for the optical fibre parameters.

6.3 (3) Fibre Bragg Grating Array Configuration

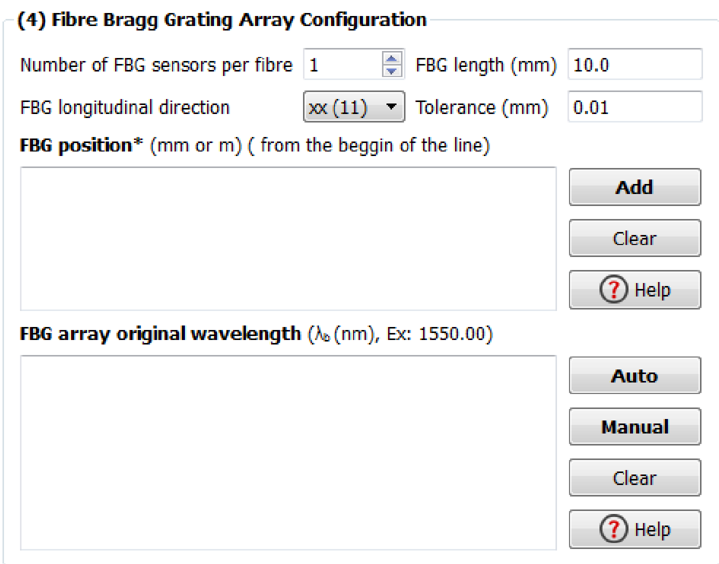


Figure 6.4: Fibre Bragg Grating Array Configuration.

In station **(3) Fibre Bragg Grating Array Configuration**, the user is asked to define the FBG array geometric configuration, as the number of sensors, its position and length; and to define the FBG original wavelength, which is given by the grating periodicity defined by the grating manufacturing process.

Geometric Parameters:

- **Number of FBG sensors per fibre:** Here, the user defines the number of "virtual" FBG sensors that will be simulated in the path previously extracted (Tab 2); A minimum of 1 FBG sensor is required;

- **FBG length (mm):** The user defines the FBG sensor length in millimeters; The default FBG length is between 5 and 10 mm;

- **FBG longitudinal direction:** In here, the user needs to specify the longitudinal direction of the FBGs accordingly with the FEM model coordinate axis. In most cases the fibre optic is aligned in the xx (11) direction, but depending on the model or the instrumentation being simulated this measurement axis can be changed; Direction options: xx(11); yy(22); and zz(33);

- **Tolerance (mm):** The tolerance parameter defines the software length tolerance when is assigning the stress and strain to the correspondent FBG. With a tolerance of zero the simulation will only consider elements that are fully contained within the FBG gauge length. As example, if no tolerance is defined for a FBG sensor with 5mm of length starting at 0 mm in the fibre optic line, the values correspondent to x=5.001 is not included (see red box figure 6.5); by specifying a tolerance value the influence of adjacent overlapping elements can also be included in the simulated response. A value of 10^{-3} FBG length could be considered appropriate.

X	LE11_Path1_Step:2 31	LE22_Path1_Step:2 31	LE33_Path1_Step:2 31	s11_Path1_Step:23 1	s22_Path1_Step:23 1	s33_Path1_Step:23 1
0.	1.15911E-03	-149.045E-06	0.	5.75273	1.33325	0.
500.009E-03	1.71432E-03	-176.076E-06	0.	8.5886	2.20278	0.
1.00001	2.23306E-03	-374.117E-06	0.	10.9242	2.11705	0.
1.50001	2.47143E-03	-434.36E-06	0.	12.0533	2.23756	0.
2.00001	2.65778E-03	-497.236E-06	0.	12.9073	2.24973	0.
2.50001	2.78618E-03	-536.396E-06	0.	13.5034	2.27971	0.
3.00001	2.88203E-03	-564.323E-06	0.	13.9507	2.30889	0.
3.50001	2.95204E-03	-584.904E-06	0.	14.2771	2.32929	0.
4.00001	3.00183E-03	-597.31E-06	0.	14.5133	2.35537	0.
4.50001	3.03612E-03	-605.516E-06	0.	14.6765	2.37509	0.
5.00001	3.05748E-03	-608.783E-06	0.	14.7816	2.39698	0.
5.50001	3.06821E-03	-608.744E-06	0.	14.8374	2.4167	0.
6.00001	3.07072E-03	-606.133E-06	0.	14.8552	2.43483	0.
6.50001	3.06525E-03	-600.422E-06	0.	14.8371	2.45458	0.
7.00001	3.0549E-03	-594.002E-06	0.	14.795	2.46913	0.

Figure 6.5: Example of the tolerance parameter.

- **FBG Position (mm or m):**

  Here the user defines the FBGs position along the path. By pressing the **Add** button the user will be prompted to input the length from the beginning of the line to the beginning of that particular FBG grating. This distance is defined by the first column of the input file (see figure 5.6), and should be inserted with the input units (mm or m) defined in station (1).

- **FBG array original wavelength ($\lambda_b(nm)$):**

  Each FBG has a original wavelength (λ_b) that is defined during the manufacturing process. The λ_b is the wavelength of the FBG reflected peak in it unstrained state. The array of original wavelength values must lie within the light bandwidth specified in station (3) (by default:between 1500 to 1600nm). The user can either press the **Auto** button to distribute the FBG's λ_b along the available bandwidth with equal spacing, or may choose to input each FBG original wavelength manually by pressing **Manual** button, perhaps in order to maximise the range of particular sensors based on the known loading regime.

6.4 (4) FBG signal variation simulation and (5) Results

In station **(4) and (5)**, the user generates the FBG time response signal from the input files and FBG parameters defined. Warning messages are presented in the message board if any error or parameter is missing.

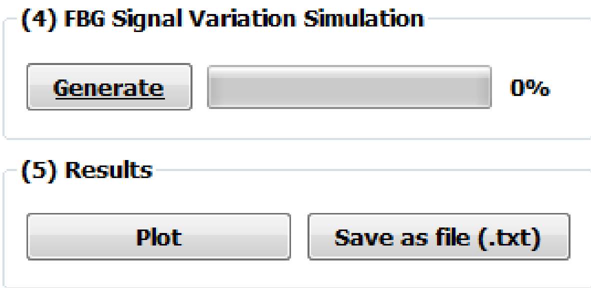


Figure 6.6: FBG time response simulation.

Push buttons:

- **Plot:** Plot of the FBG signal variation simulation.

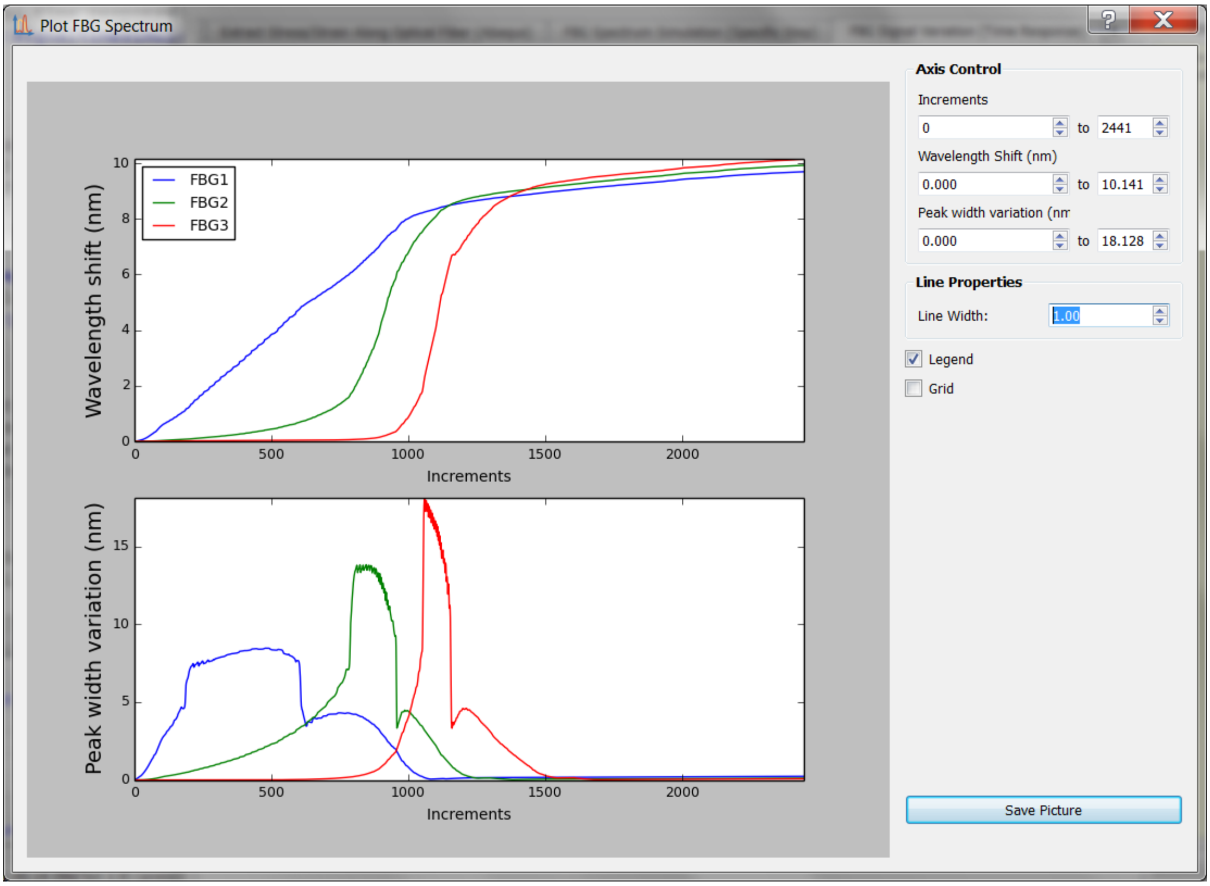


Figure 6.7: Plot window.

In the plot window the user have the possibility to change the axis scale and the display layout. The user can save the plot by pressing the **Save Picture** push button.

- **Save as file (.txt)**: Save the FBG signal variation simulation as .txt file.

```
Number of input files (increment number):  2442

 ------FBG array configuration------
Number of FBG sensors: 3
FBG length (mm): 10.0
Tolerance (mm): 0.01
FBG position (mm): [15.0, 30.0, 50.0]
FBG original wavelength (nm): [1525.0, 1550.0, 1575.0]

Increment number        FBG 1-Wavelength shift (nm)     FBG 2-Wavelength shift (nm)     FBG 3-Wavelength shift (nm)
FBG 1-Peak width variation (nm) FBG 2-Peak width variation (nm) FBG 3-Peak width variation (nm)
0       0.000000        0.000000        0.000000        0.000000        0.000000        0.000000
1       0.001674        0.000048        0.000126        0.010718        0.000135        0.000038
2       0.003347        0.000096        0.000251        0.021435        0.000269        0.000076
3       0.005021        0.000144        0.000377        0.032152        0.000404        0.000114
4       0.006696        0.000192        0.000503        0.042876        0.000538        0.000151
5       0.008379        0.000241        0.000629        0.053639        0.000669        0.000188
6       0.010110        0.000292        0.000756        0.064636        0.000786        0.000221
7       0.012024        0.000351        0.000887        0.076520        0.000847        0.000237
8       0.014093        0.000417        0.001023        0.089163        0.001005        0.000239
9       0.016056        0.000479        0.001155        0.101288        0.001181        0.000250
10      0.017949        0.000537        0.001286        0.113077        0.001341        0.000266
11      0.019866        0.000596        0.001417        0.125003        0.001509        0.000270
12      0.021949        0.000659        0.001550        0.137749        0.001802        0.000286
13      0.024292        0.000734        0.001688        0.151700        0.002131        0.000336
14      0.026610        0.000808        0.001825        0.165478        0.002365        0.000354
15      0.028847        0.000877        0.001961        0.178844        0.002502        0.000381
16      0.031292        0.000952        0.002098        0.193149        0.002732        0.000465
17      0.034005        0.001041        0.002240        0.208756        0.003101        0.000509
18      0.036619        0.001127        0.002382        0.223949        0.003458        0.000510
19      0.039314        0.001215        0.002523        0.239550        0.003951        0.000516
20      0.040150        0.001243        0.002560        0.244189        0.004118        0.000526
21      0.041408        0.001287        0.002616        0.251129        0.004314        0.000543
22      0.043138        0.001345        0.002698        0.260800        0.004527        0.000571
23      0.045567        0.001424        0.002819        0.274519        0.004798        0.000622
24      0.048682        0.001528        0.002963        0.291821        0.005272        0.000718
25      0.052006        0.001644        0.003111        0.310130        0.005829        0.000751
26      0.055209        0.001754        0.003258        0.327962        0.006401        0.000745
27      0.058652        0.001875        0.003406        0.346852        0.007184        0.000779
28      0.062350        0.002008        0.003557        0.366734        0.007835        0.000833
```

Figure 6.8: Simulated FBG time response as .txt file.

In the header of the .txt output file is presented information that describes and summarises the simulation, and in the main body is presented the wavelength shift $\Delta\lambda_b(nm)$ and the width variation $\Delta\lambda_{wv}(nm)$ for each FBG sensor.

Section 7

Example/Tutorial: Delamination of a Double Cantilever Beam

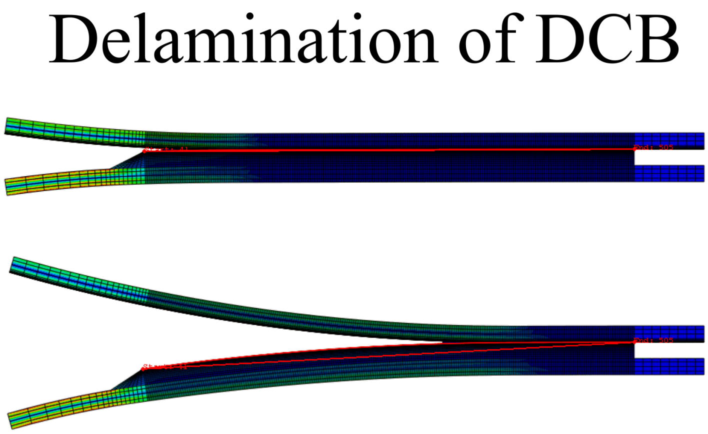


Figure 7.1: Tab: Delamination of a Double Cantilever Beam.

In this section, the software FBG_SiMul will be applied to a FEM model of delamination of a Double Cantilever Beam (DCB). Two "virtual" FBG sensor lines with an array of 5 gratings will be simulated using two distinct paths, with the objective to detect the crack growth.

Objectives/tasks:

- Extract the stress/strain along two different paths;
- Simulate the FBG Spectrum for different array configurations; optimize it to crack detection;
- Time response of the sensors during the crack growth process;

Note: The FEM model, and the input files are provided and can be found in the "InputData_Test.rar" file.

7.1 FEM Model

The DCB model used is the same as described by Pereira et al.[1]

The 2D Model was developed using the FEM software Abaqus, and the .CAE file is provided with this software. It was assumed plane stress conditions, and the delamination was modelled using 4-node cohesive elements along the delamination plane. The cohesive elements are modelled to express the cohesive law, meaning a progressive loss of the cohesion between the two crack faces with the local crack opening. The crack was modelled to occur between the interface of the adhesive and the glass fibre arm beam, and a cohesive element size of 0.5 mm was selected. The damage initiation was calculated using a quadratic stress criterion.

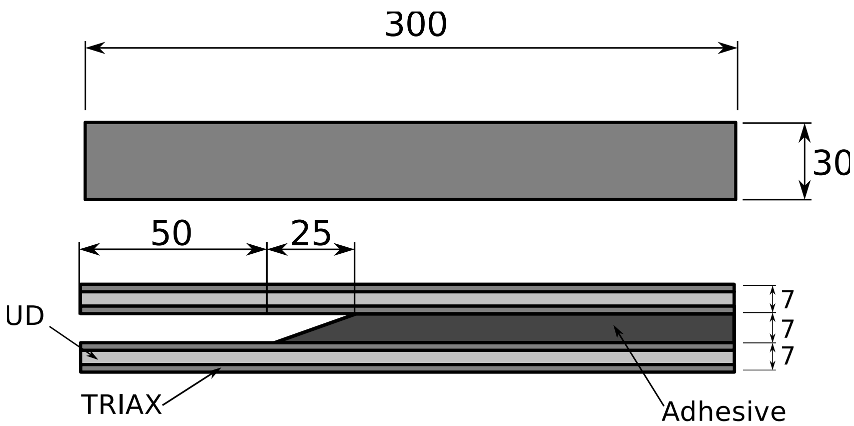


Figure 7.2: Double cantilever beam geometry dimensions. (doi:10.1371/journal.pone.0141495.g010)

The dimensions of the DCB specimen are shown in figure 7.2, and the material properties implemented in the FEM model are presented in table 7.1.

Table 7.1: Double cantilever beam material properties.

Composite Material		**Adhesive**
Triaxial Fabric (Composite)	Uniaxial Fabric (Composite)	Elastic
$E_1 = 44.3$ GPa $E_2 = E_3 = 12.9$ GPa $\nu_{12} = \nu_{13} = \nu_{23} = 0.23$ $G_{12} = G_{13} = G_{23} = 4393$ GPa	$E_1 = 23.8$ GPa $E_2 = E_3 = 15.05$ GPa $\nu_{12} = \nu_{13} = \nu_{23} = 0.513$ $G_{12} = G_{13} = G_{23} = 4.393$ GPa	$E = 4.56$ GPa $\nu = 0.35$

Interface (Cohesive Law)		
Penalty Stiffness	Damage (Quadratic stress)	Damage Evolution
$K = 4.2\,E12$ Pa;	$\sigma_n = 2.64$ MPa (Mode I) $\sigma_t = 22.15$ MPa (Mode II)	$\delta_{c1} = 1.4$ (Mode I) $\delta_{c2} = 0.37$ (Mode II)

[1] G.F. Pereira, L.P. Mikkelsen, M. McGugan, Crack Detection in Fibre Reinforced Plastic Structures Using Embedded Fibre Bragg Grating Sensors: Theory, Model Development and Experimental Validation., PLoS One. 10 (2015) e0141495. doi:10.1371/journal.pone.0141495.

The beams were modelled using a combination of two different laminates: unidirectional glass fibre (UD) and triaxial glass fibre (Triax). Moments were applied to the extremities of the beams to create a pure mode I- opening/delamination. **The model was build with SI (mm) units.**

7.2 FBG Paths

Two different fibre paths will be simulated from the FEM model: one path is 0.03 mm parallel from the crack delamination plane; and the second path is 4.4 mm parallel from the crack, as shown in figure 7.3.

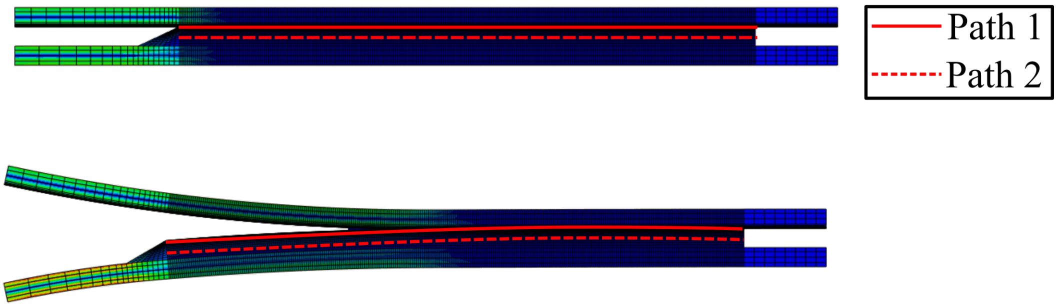


Figure 7.3: Virtual FBG paths.

The coordinates that define the crack plane and the two fibre paths are presented in table 7.2.

Table 7.2: Path points coordinates.

	Initial point coordinates (mm)	Final point coordinates (mm)
Crack Plane	-25 , 7.572 , 0	185 , 7.572 , 0
Path-1	-25 , 7.539 , 0	185 , 7.539 , 0
Path-2	-25 , 3.161 , 0	185 , 3.161 , 0

7.3 Stress/strain along the paths (Tab 2)

Now, we will extract the stress and strain files for the two fibre paths. To do this, we will use the Tab 2- Extract Stress/Strain along Optical Fiber (Abaqus).

Stress/strain along the paths tutorial (Tab 2):

1. Station (1): Select the .odb file- MI_05.odb.

2. Station (2): Beacuse the DCB model and the optical fibre longitudinal direction is aligned with the x axis, there is no need to rotate the coordinate system (see figure 7.4). Default option is selected.

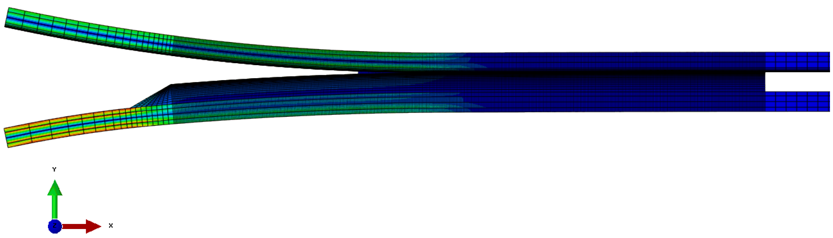


Figure 7.4: FEM axis direction.

3. Station (3): Insert the optical fibre coordinates for the two paths using the values presented in table 7.2.

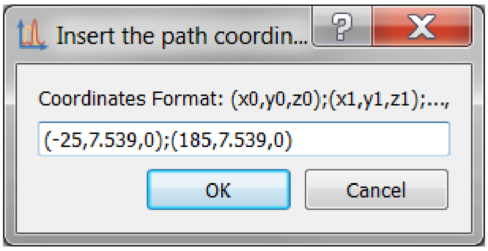


Figure 7.5: Dialogue window to insert the path coordinates: Path-1.

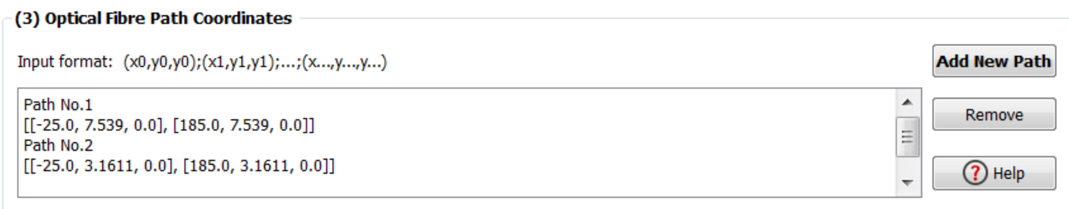


Figure 7.6: FBG virtual fibre paths.

4. Station (4): Because we want to analyse the crack growth, it is needed to extract all time increments. Select all time increments option;

5. Station (5): Select a folder to save the stress/strain files;

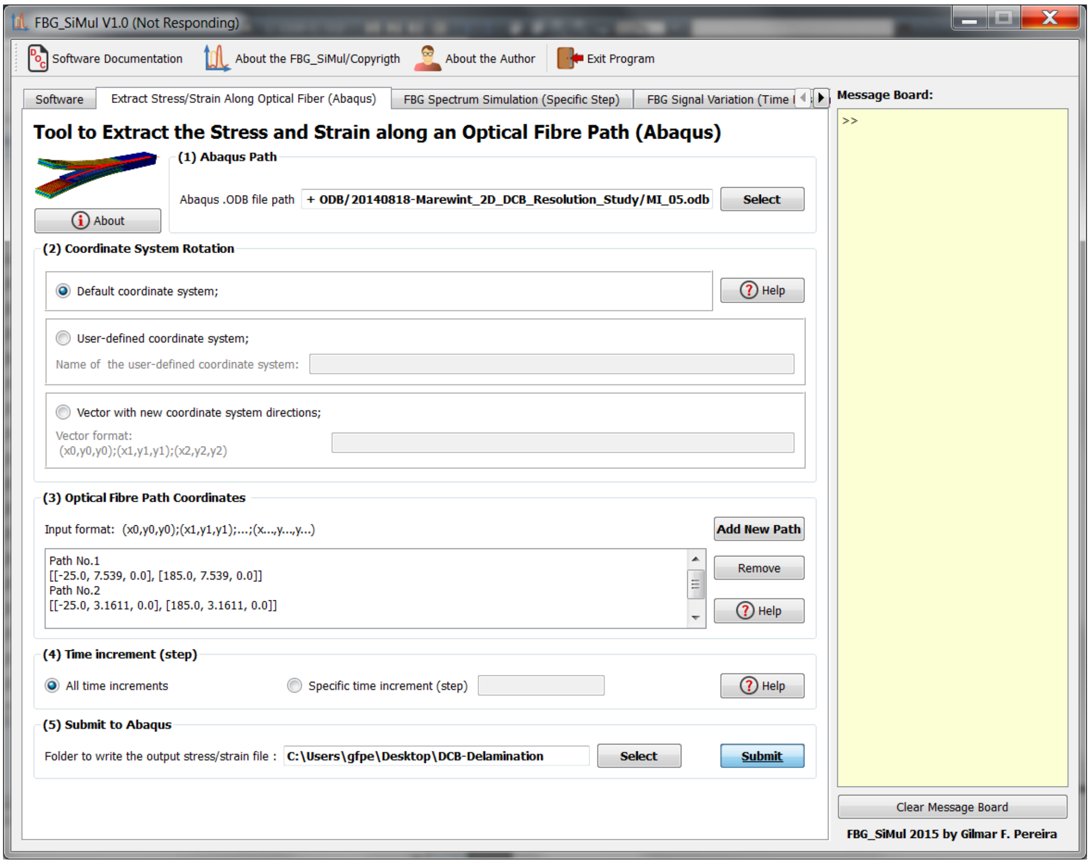


Figure 7.7: Tab 2: Stress and Strain along two optical fibre paths.

6. Press the push button **Submit**;

A temporary folder is created inside the selected output folder containing an Abaqus post-processing script; Abaqus software should open and start to generate the stress/strain files. After Abaqus finish to extract the stress/strain files, close the software, and FBG_SiMul should delete the temporary folder and files, and show the following message:

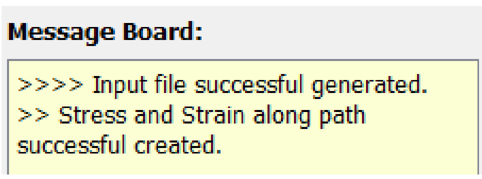


Figure 7.8: Extraction of Stress and Strain along a path completion message.

In the output folder, we can find the stress/strain files for the 2 paths and 3298 increments. The files are named as *Path**yy**_Step_**xxxx**.txt*, where **yy** is the path number and **xxxx** is the increment number.

Path1_Step_0000.txt	31-12-2015 15:08	Text Document	60 KB
Path1_Step_0001.txt	31-12-2015 15:08	Text Document	60 KB
Path1_Step_0002.txt	31-12-2015 15:08	Text Document	60 KB
Path1_Step_0003.txt	31-12-2015 15:08	Text Document	60 KB
Path1_Step_0004.txt	31-12-2015 15:08	Text Document	60 KB
Path1_Step_0005.txt	31-12-2015 15:08	Text Document	60 KB
Path1_Step_0006.txt	31-12-2015 15:08	Text Document	60 KB
Path1_Step_0007.txt	31-12-2015 15:08	Text Document	60 KB
Path1_Step_0008.txt	31-12-2015 15:08	Text Document	60 KB
Path1_Step_0009.txt	31-12-2015 15:08	Text Document	60 KB
Path1_Step_0010.txt	31-12-2015 15:08	Text Document	60 KB
Path1_Step_0011.txt	31-12-2015 15:08	Text Document	60 KB
Path1_Step_0012.txt	31-12-2015 15:08	Text Document	60 KB
Path1_Step_0013.txt	31-12-2015 15:08	Text Document	60 KB
Path1_Step_0014.txt	31-12-2015 15:08	Text Document	60 KB
Path1_Step_0015.txt	31-12-2015 15:09	Text Document	60 KB
Path1_Step_0016.txt	31-12-2015 15:09	Text Document	60 KB
Path1_Step_0017.txt	31-12-2015 15:09	Text Document	60 KB
Path1_Step_0018.txt	31-12-2015 15:09	Text Document	60 KB
Path1_Step_0019.txt	31-12-2015 15:09	Text Document	60 KB
Path1_Step_0020.txt	31-12-2015 15:09	Text Document	60 KB

Figure 7.9: Stress/Strain output files saved in the output folder.

7.4 Simulation of the Reflected Spectrum for a Specific Time Increment

In this section, we will simulate the reflected spectrum of the two FBG arrays for a specific time increment.

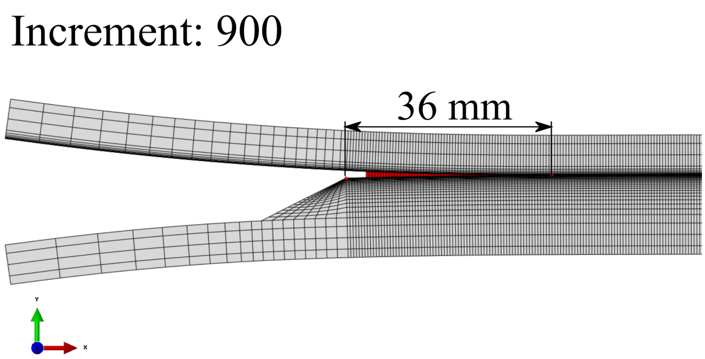


Figure 7.10: Crack tip position in the DCB specimen for the time increment 900.

At time increment **900** (see figure 7.10), the crack tip is located at 36 mm from the beginning of the optical fibre line. A 5 FBG array was defined as virtual measurement points, each with a 10 mm length and spaced 10 mm from each other. The first grating was defined 5 mm from the beginning of the optical fibre, as shown in figure 7.11.

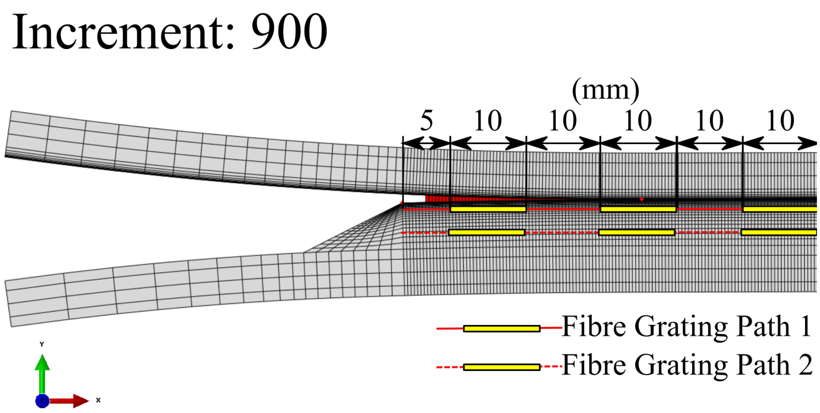


Figure 7.11: FBG array configuration in the DCB specimen.

Note that the FBG 2 is situated at the crack tip, where a large non-uniform strain filed is expected that create a distortion on the reflected peak signal (non-uniform strain).

Reflected peak simulation tutorial (Tab 3):

1. Station (1): Insert the two files- *Path1_Step_0900.txt* and *Path2_Step_0900.txt*;
2. Select the file Path1_Step_0900.txt; it should become highlighted (see figure 7.12); set the skip row to **3**; toggle the input units to **mm**, as the units of the FEM model.

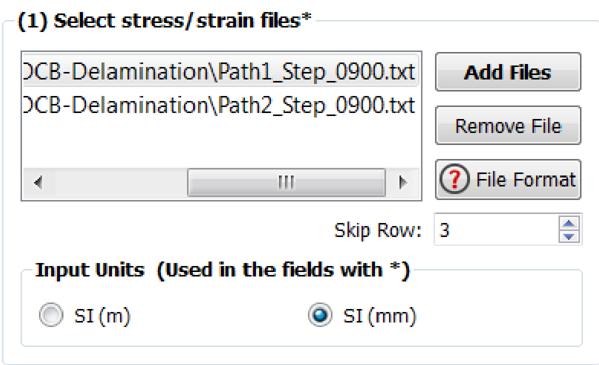


Figure 7.12: Select Stress/strain files for the time increment 900.

3. Station (2): We will analyse the three cases, but to start we will select **uniform strain**; leave the simulation resolution in the default value 0.05 nm;
4. Station (3): Leave the Optical Fibre Parameters as the default values;

5. Station (4): Set the number of FBG per array to **5**; the FBG length to 10 mm; the FBG longitudinal direction as **xx** (see axis of figure 7.11); and, leave the tolerance as the default value;

6. FBG position: Set the position of the FBG array as shown in figure 7.11; the distance is from the beginning of the line to the beginning of each grating; FBG position values (mm): 5; 25; 45; 65; 85;

7. Press the button **Auto** to distribute automatically the array of original wavelength (λ_b) along the available light bandwidth;

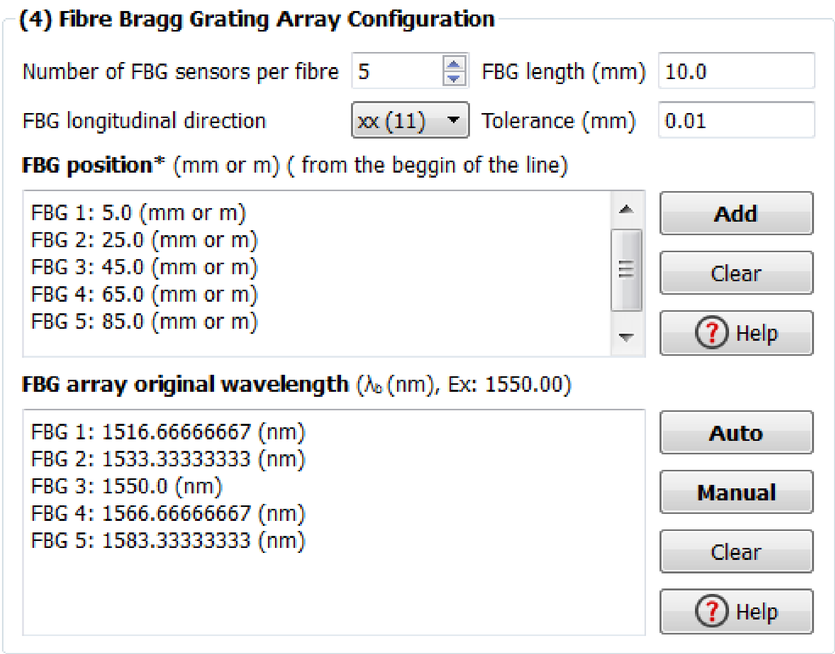


Figure 7.13: Fibre Bragg Grating Array Configuration.

8. Station (5): Toggle **"plot the undeformed FBG reflected signal"** and press the button **Simulate**;

9. If any errors occurs it will be presented in the message board; then, when the loading bar reaches 100% the reflected spectrum is successfully simulated; press **Plot** to visualize it or **Save as file** to save it as a .txt file;

10. Repeat this process to all type of simulations (uniform, non-uniform strain and transverse stress) and for both files *Path1_Step_0900.txt* and *Path2_Step_0900.txt*;

Results: All different type of simulations for the path-1.

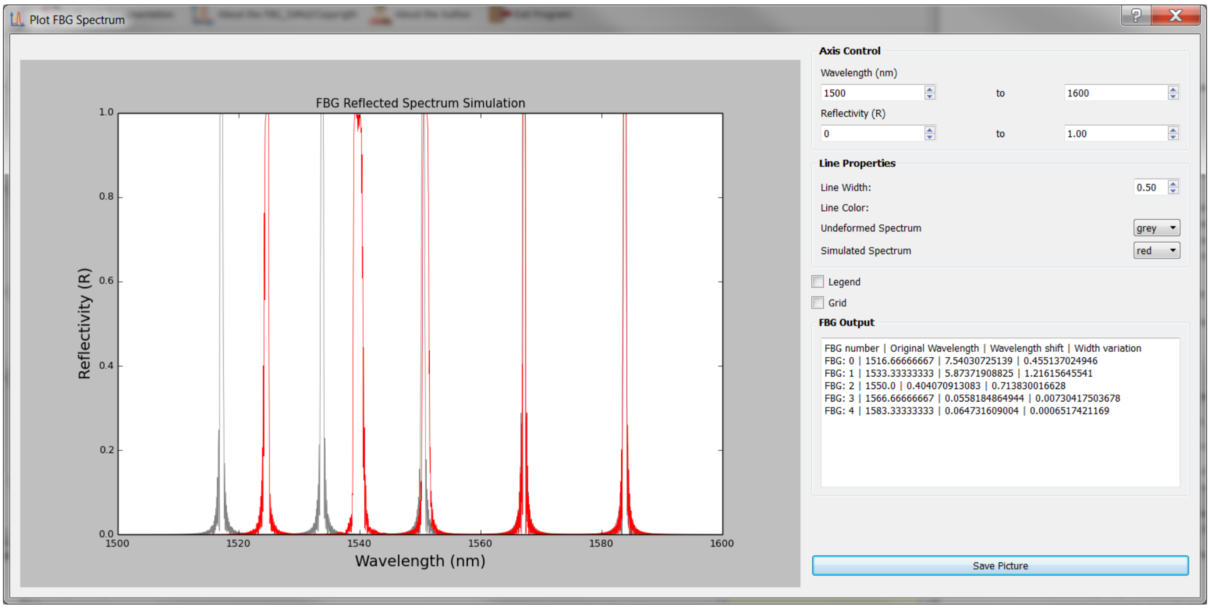


Figure 7.14: FBG_SiMul plot of the FBG simulation for non-uniform contribution.

A screen-shoot of the plot window containing the FBG simulation for the path-1, considering uniform and non-uniform strain contribution is shown in figure 7.14. To compare the three type of simulations the FBG reflected spectrum data was saved as .txt file, and plotted using an external software (see figure 7.15); and, the summarize FBG response data that is presented in tabled 7.3.

Table 7.3: FBG array response for the time increment 900.

FBG Sensor Number	**1**	**2**	**3**	**4**	**5**
Original Wavelength (λ_b)	1516.66	1533.33	1550.0	1566.66	1583.33
Wavelength shift ($\Delta\lambda_b$)	7.54	5.87	0.41	0.06	0.06
Width variation ($\Delta\lambda_{wv}$)	0.45	1.21	0.71	0.01	0.01

By analysing the results of the three simulation methods presented in figure 7.15, it is possible to determine the individual contribution of each simulation method to the sensor response. Thus, the strain field around the crack tip can be predicted and linked with the FBG response.

The uniform strain simulation describes the average longitudinal strain in each sensor. It is observed a large wavelength shift ($\Delta\lambda_b$)in the FBG 1 and 2, meaning that at this location the material is damaged and losing its compliance, which caused an increase on the strain. In the non-uniform strain simulation is observed an increase of the FBG 1, 2 and 3 peak width. Particularly in FBG 2, the reflected peak is highly distorted, which is caused by the proximity with the crack tip that causes a strain gradient (non-uniform strain). The transverse stress simulation does not show any peak split-

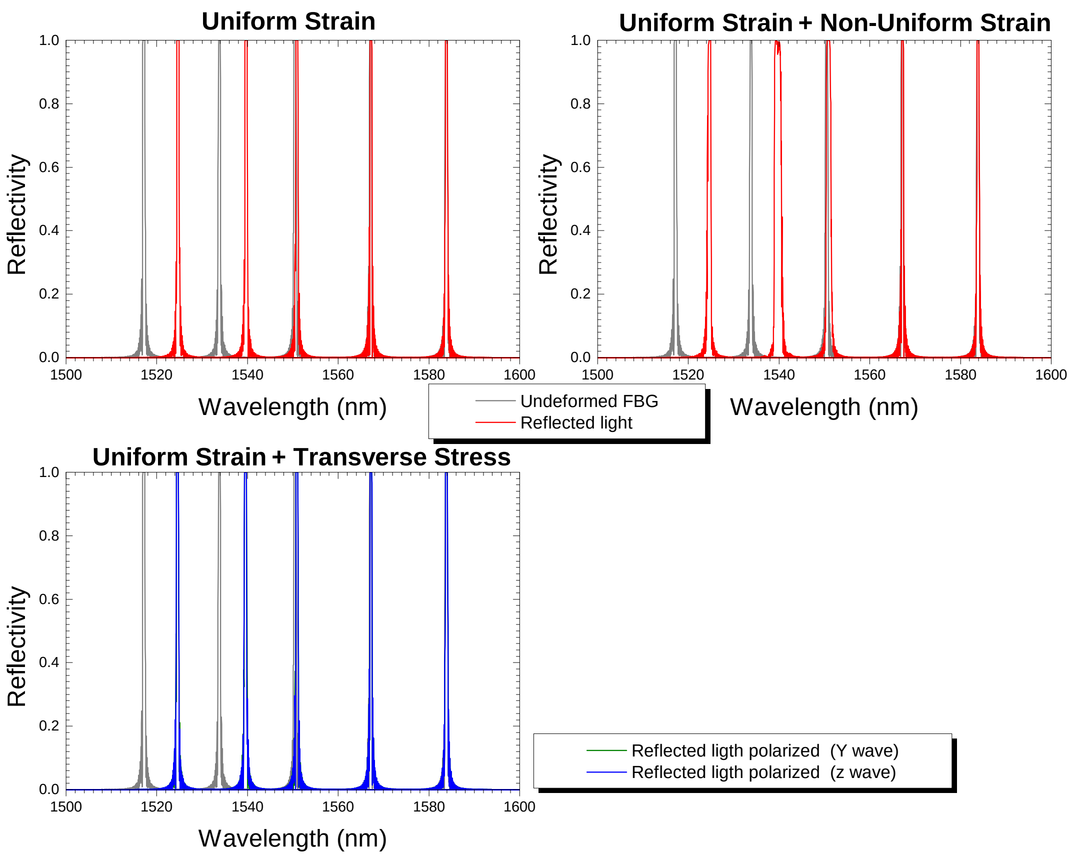


Figure 7.15: Simulate FBG reflected spectrum for the three simulation.

ting behaviour, because the magnitude of transverse stress is small. (In other FEM models/loading condition this can be different)

Thus, based on this simulations we can confidently say that it is the non-uniform strain field that will control the sensor response near the crack tip;

Results: Path-1 and Path-2 simulation comparison.

Now lets focus only on the FBG 2 (25 mm) and do an analyse of the two different fibre paths.

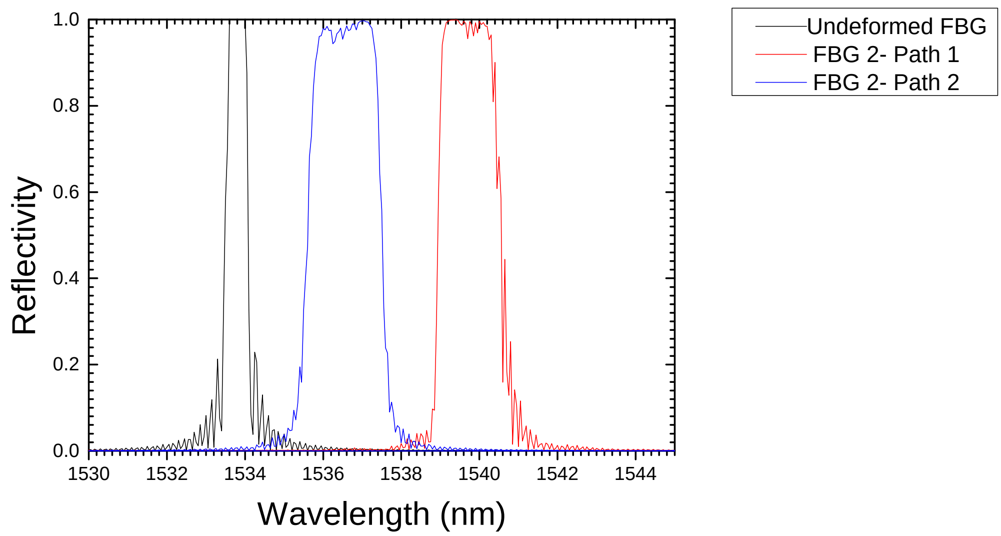


Figure 7.16: FBG 2 reflected peak simulation for the non-uniform strain contribution: path-1 and path-2 comparison.

Table 7.4: Summary of the FBG 2 reflected peak simulation for the non-uniform strain contribution: path-1 and path-2 comparison.

FBG Sensor Number 2	**Path-1**	**Path-2**
Original Wavelength (λ_b)	1533.33	1533.33
Wavelength shift ($\Delta\lambda_b$)	5.87	2.80
Width variation ($\Delta\lambda_{wv}$)	1.21	1.56

The two FBG paths reflected peak simulation considering non-uniform strain contribution is shown in figure 7.16 and summarized in table 7.4. The shape of the reflected spectrum is similar for both paths; yet, the path-2 wavelength shift is smaller, because the path is more distant from the crack plane, resulting in a higher local compliance and less strain along the sensor. On the other hand, the strain gradient is larger in the path-2, resulting in a higher width variation ($\Delta\lambda_{wv}$).

Initially the path-1 seemed more obvious to detect crack, but its proximity with the crack can promote the damage of the sensor. Thus, this simulation shows that a FBG array along the path-2 can also detect cracks successful, minimizing the risk of damaging the sensor. Also, it can be concluded from the simulation results (similar conclusions as presented by Pereira et al.[2]), that the FBG output

[2] G.F. Pereira, L.P. Mikkelsen, M. McGugan, Crack Detection in Fibre Reinforced Plastic Structures Using Embedded

parameters, wavelength shift ($\Delta\lambda_b$) and width variation ($\Delta\lambda_{wv}$), can be used to determine the presence of the crack and to track its growth. The wavelength shift is a parameter related to the strain level in the structure, where a rapid increase of its magnitude is caused by a damage event that reduces the stiffness of the structure. And in contrast, the width variation of the reflected peak is a parameter that only depends on the presence of a crack, independent of geometry and loading type. The width of the reflected peak increases when the crack is near the grating area, being low in magnitude before and after the crack passes.

7.5 Simulation of the FBG Time Response

In this section, we will simulate the FBG time response of the two FBG paths. The FBG array configuration is the same as presented in figure 6.4.

FBG Simulation tutorial (Tab 4):

1. Station (1): Insert all step files of the path-1 (later repeat the tutorial for the path-2);
2. If the files are not sorted press the button **Sort**;
3. Station (2): Leave the optical fibre parameters as the default values;
4. Station (3): Set the number of FBG per array to **5**; the FBG length to 10 mm; the FBG longitudinal direction as **xx**; and, leave the tolerance with the default value;
5. FBG position: Set the position of the FBG array as shown in figure 7.11; the distance is from the beginning of the line to the beginning of the grating; FBG position values (mm): 5; 25; 45; 65; 85;
6. Press the button **Auto** to distribute automatically the array original wavelength (λ_b) along the available light bandwidth;

Fibre Bragg Grating Sensors: Theory, Model Development and Experimental Validation., PLoS One. 10 (2015) e0141495. doi:10.1371/journal.pone.0141495.

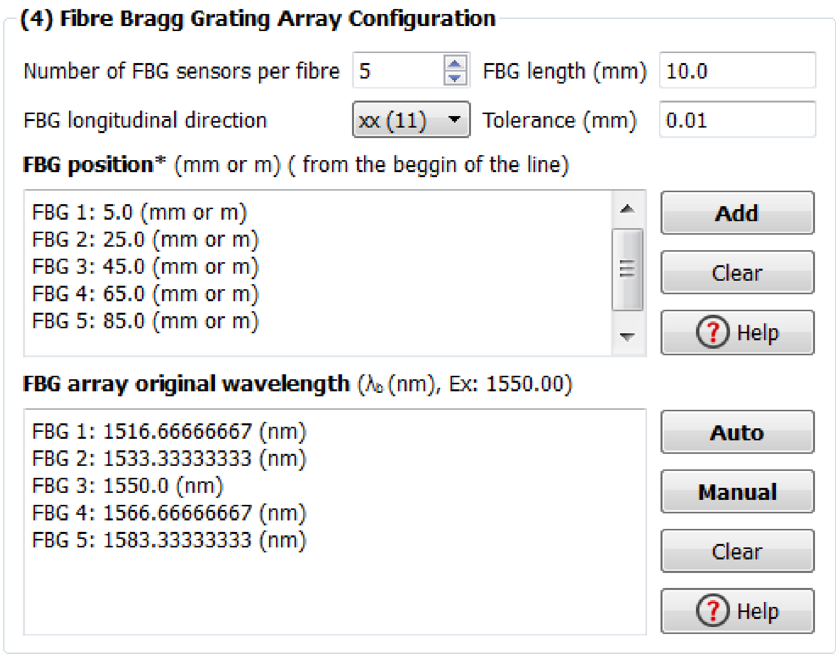


Figure 7.17: Fibre Bragg Grating Array Configuration.

7. Station (4): Press the button **Generate**; This can take some time because the software have to load multiple files;

8. If any errors occurs it will be presented in the message board; when the loading bar reaches 100% the FBG time response was successfully simulated; press **Plot** to visualize it or **Save as file** to save it as a .txt file;

9. Repeat this process for the path-2;

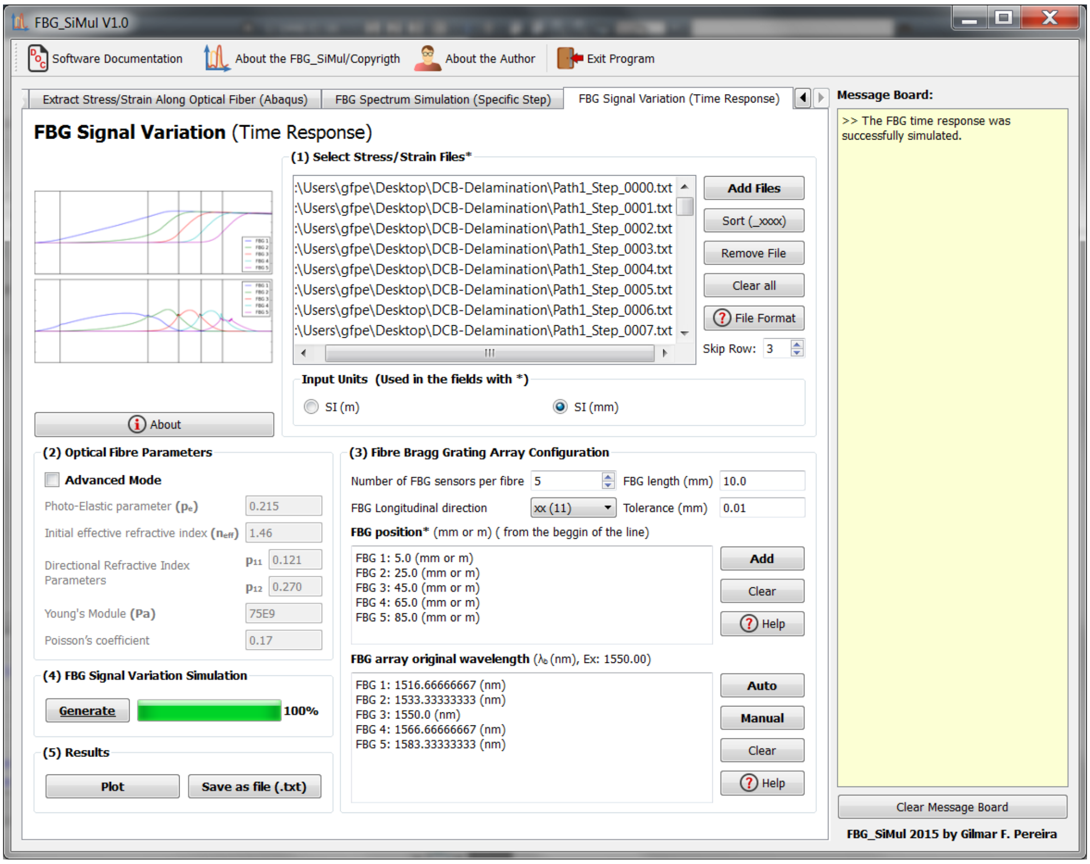


Figure 7.18: Tab 4: Time Repose Simulation Input.

Results: Paths comparison

The FBG_SiMul plot window showing the FBG time response for the path-1 is shown figure 7.19. The top plot shows the wavelength shift $\Delta\lambda_b$ caused by the longitudinal strain along the optical fibre, and the bottom plot shows the width variation of the reflected peak $\Delta\lambda_{wv}$ caused by the non-uniform strain and transverse stress.

A jump in the wavelength shift $\Delta\lambda_b$ can be observed when the crack passed the position of the grating. The damage/crack changes the material local compliance and the load distribution, making the area that surrounds the sensor less stiff and more deformed; therefore, an increase in the strain was measured. However, it is possible to observe some differences in the evolution (shape) of the wavelength shift $\Delta\lambda_b$ from each FBG, because the sensor and the crack position related to the applied moments is different. It is also observed that a variation in the width of the reflected peak $\Delta\lambda_{wv}$ occurs when the crack is near the grating, and that the original peak width is restored after the crack passes the grating.

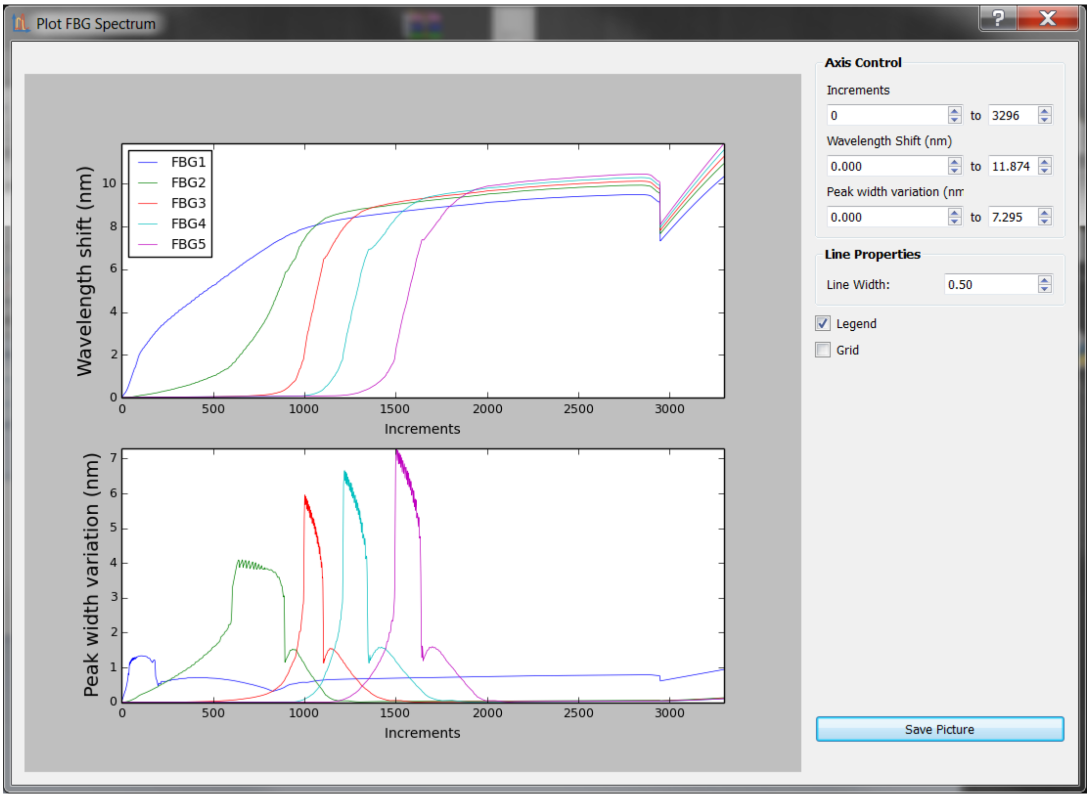


Figure 7.19: Path-1 FBG time response plot.

The FBG time response for the two fibre paths is shown in figure 7.20. As expected both paths show the same type of response, however the magnitudes are lower for the path-2. Due to its proximity with the crack, the gratings in the path-1 measure larger variation of the reflected peak width. However, the path-2 can also detect cracks successful, minimizing the risk of damaging the sensor.

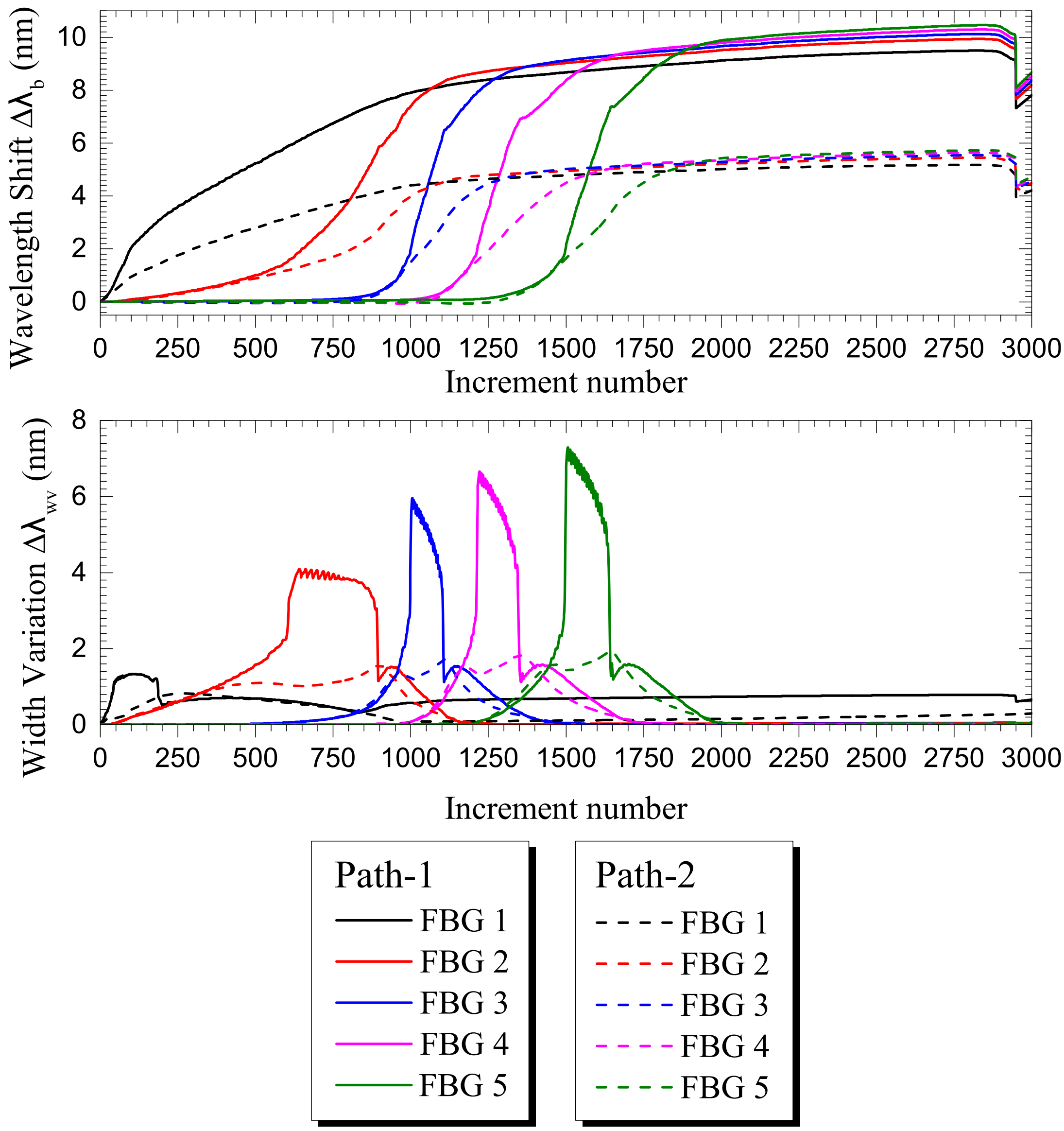


Figure 7.20: FBG time response plot: path-1 and path-2 comparison.