

fe-safe 2018

fe-safe USER GUIDE



Trademarks

fe-safe, Abaqus, Isight, Tosca, the 3DS logo, and SIMULIA are commercial trademarks or registered trademarks of Dassault Systèmes or its subsidiaries in the United States and/or other countries. Use of any Dassault Systèmes or its subsidiaries trademarks is subject to their express written approval. Other company, product, and service names may be trademarks or service marks of their respective owners.

Legal Notices

fe-safe and this documentation may be used or reproduced only in accordance with the terms of the software license agreement signed by the customer, or, absent such an agreement, the then current software license agreement to which the documentation relates.

This documentation and the software described in this documentation are subject to change without prior notice.

Dassault Systèmes and its subsidiaries shall not be responsible for the consequences of any errors or omissions that may appear in this documentation.

© Dassault Systèmes Simulia Corp, 2017.

Third-Party Copyright Notices

Certain portions of *fe-safe* contain elements subject to copyright owned by the entities listed below.

© Battelle
© Endurica LLC
© Amec Foster Wheeler Nuclear UK Limited

fe-safe Licensed Programs may include open source software components. Source code for these components is available if required by the license.

The open source software components are grouped under the applicable licensing terms. Where required, links to common license terms are included below.

IP Asset Name	IP Asset Version	Copyright Notice
Under BSD 2-Clause		
UnZip (from Info-ZIP)	2.4	Copyright (c) 1990-2009 Info-ZIP. All rights reserved.
Under BSD 3-Clause		
Qt Solutions	2.6	Copyright (c) 2014 Digia Plc and/or its subsidiary(-ies) All rights reserved.

1 Introduction

1.1 Background

SIMULIA, the Dassault Systèmes brand for realistic simulations, offers *fe-safe*[®] – the most accurate and advanced fatigue analysis technology for real-world applications.

fe-safe empowers you to better tailor and predict the life of your products. It has been developed continuously since the early 1990's in collaboration with industry, ensuring that *fe-safe* provides the capabilities required for real industrial applications. It continues to set the benchmark for fatigue analysis software and is testimony to the fact that, not only is accurate fatigue analysis possible, but it is possible regardless of the complexity of the model and the fatigue expertise of its users.

fe-safe was the first commercially available fatigue analysis software to focus on modern multiaxial strain-based fatigue methods. It analyses metals, rubber, thermo-mechanical and creep-fatigue and welded joints, and is renowned for its accuracy, speed and ease of use.

Consistent and accurate correlation with test results ensures that *fe-safe* maintains its position as the technology leader for durability assessment and failure prevention.

fe-safe and the add-on modules *fe-safe/Rubber*, *fe-safe/TURBOLife* and *Verity*[®] in *fe-safe*, are available worldwide via SIMULIA and our network of partners.

For further information please visit the [fe-safe pages](#) of the Dassault Systèmes website

1.1.1 *fe-safe*

fe-safe is a powerful, comprehensive and easy-to-use suite of fatigue analysis software for finite element models. It is used alongside commercial FEA software, to calculate:

- where fatigue cracks will occur
- when fatigue cracks will initiate
- the factors of safety on working stresses (for rapid optimisation)
- the probability of survival at different service lives (the 'warranty claim' curve)
- whether cracks will propagate

Results are presented as contour plots which can be plotted using standard FE viewers. *fe-safe* has direct interfaces to the leading FEA suites.

For critical elements, *fe-safe* can provide comprehensive graphical output, including fatigue cycle and damage distributions, calculated stress histories and crack orientation. To simplify component testing and to aid re-design, *fe-safe* can evaluate which loads and loading directions contribute most to the fatigue damage at critical locations.

Sophisticated techniques for identifying and eliminating non-damaged nodes, make *fe-safe* extremely efficient for large and complex analyses, without compromising on accuracy.

Typical application areas include the analysis of machined, forged and cast components in steel, aluminium and cast iron, high temperature components, welded fabrications and press-formed parts. Complex assemblies containing different materials and surface finishes can be analysed in a single run.

For engineers who are not specialists in fatigue, *fe-safe* will automatically select the most appropriate analysis method, and will estimate materials' properties if test data is not available.

Specialist engineers can take advantage of user-configurable features. Powerful macro recording and batch-processing functions make repetitive tasks and routine analyses straightforward to configure and easy to run.

fe-safe includes the *fe-safe* Material Database (see below), to which users can add their own data, and comprehensive materials data handling functions.

fe-safe also incorporates powerful durability analysis and signal processing software, *safe4fatigue* (see below) at no additional cost, on all platforms.

Summary of capabilities

- Fatigue of Welded Joints

fe-safe includes the BS708 analysis as standard. Other S-N curves can be added. *fe-safe* also has an exclusive license to the *Verity* Structural Stress Method developed by Battelle. Developed under a Joint Industry Panel and validated against more than 3500 fatigue tests, *Verity* is bringing new levels of accuracy to the analysis of structural welds, seam welds and spot welds

- Vibration Fatigue

fe-safe includes powerful features for the analysis of flexible components and structures that have dynamic responses to applied loading. Steady state modal analysis, random transient analysis and PSDs are amongst the analysis methods included

- Test Program Validation

fe-safe allows the user to create accelerated test fatigue programs. These can be validated in *fe-safe* to ensure that the fatigue-critical areas are the same as those obtained from the full service loading. Fatigue lives and fatigue damage distributions can also be correlated

- Critical Distance – will cracks propagate?

Critical distance methods use subsurface stresses from the FEA to allow for the effects of stress gradient. The data is read from the FE model by *fe-safe*, and the methods can be applied to single nodes, fatigue hot-spots or any other chosen areas including the whole model

- Property Mapping

Results from casting or forging simulations can be used to vary the fatigue properties at each FE node. Each node will then be analyzed with different materials data. Temperature variations in service, multiaxial stress states and other effects such as residual stresses can also be included

- Vector Plots

Vector plots show the direction of the critical plane at each node in a hotspot, or for the whole model. The length and colour of each vector indicates the fatigue damage

- Warranty curve

fe-safe combines variations in material fatigue strengths and variability in loading to calculate the probability of survival over a range of service lives

- Damage per block

Complex loading histories can be created from multiple blocks of measured or simulated load-time histories, dynamic response analyses, block loading programs and design load spectra. Repeat counts for each block can be specified. *fe-safe* also exports the fatigue damage for each 'block' of loading (for example, from each road surface on a vehicle proving ground, or for each wind state on a wind turbine). This shows clearly which parts of the duty cycle are contributing the most fatigue damage. Re-design can focus on this duty cycle, and accelerated fatigue test programs can be generated and validated

- Material database

A material database is supplied with *fe-safe*. Users can add their own material data and create new

databases. Material data can be plotted and tabulated. Effects of temperature, strain rate etc can be seen graphically. Equivalent specifications allow searching on US, European, Japanese and Chinese standards

- Automatic hot-spot formation

fe-safe automatically identifies fatigue hot-spots based on user-defined or default criteria. Hot-spots can be used for rapid design change studies and design sensitivity analysis

- Manufacturing effects

Results from an elastic-plastic FEA of a forming or assembly process or from surface treatments such as cold rolling or shot peening can be read into *fe-safe* and the effects included in the fatigue analysis. Estimated residual stresses can also be defined for areas of a model for a rapid 'sensitivity' analysis

- Surface detection

fe-safe automatically detects the surfaces of components. The user can select to analyse only the surface, or the whole model. Subsurface crack initiation can be detected and the effects of surface treatments taken in to account

- Surface contact

Surface contact is automatically detected. Special algorithms analyse the effects of contact stresses. This capability has been used for bearing design and for the analysis of railway wheel/rail contact

- Virtual strain gauges (single gauges and rosettes) can be specified in *fe-safe* to correlate with measured data. *fe-safe* exports the calculated time history of strains for the applied loading. FE models can be validated by comparison with measured data

- Parallel processing

Parallel processing functionality is included as standard – no extra licences are required

- Signal processing

Signal processing, load history manipulation, fatigue from strain gauges, and generation of accelerated testing signals are among the many features included as standard

- Structural optimisation

fe-safe can be run inside an optimisation loop with optimisation codes to allow designs to be optimised for fatigue performance. *fe-safe* interfaces to Isight and Tosca from SIMULIA, and Workbench ANSYS®.

- *fe-safe/Rotate*

fe-safe/Rotate speeds up the fatigue analysis of rotating components by taking advantage of their axial symmetry. It is used to provide a definition of the loading of a rotating component, through one full revolution, from a single static FE analysis. From a single load step, *fe-safe/Rotate* produces a sequence of additional stress results as if the model had been rotated through a sequence of different orientations.

fe-safe/Rotate is particularly suitable where the complete model exhibits axial symmetry, for example: wheels, bearings, etc.. However, the capability can also be used where only a part of the model exhibits axial symmetry, for example to analyse the hub of a cam. The remainder of the model (the non-axially symmetric parts) can be analysed in the conventional way.

fe-safe/Rotate is included as a capability in the standard *fe-safe*. Since it is for use with finite element model data, it is not available as an extension to *safe4fatigue*.

fe-safe/Rotate is an integrated part of the interface to the FE model, and is currently available for ANSYS results (*.rst), Abaqus Fil and ASCII model files only.

Use of *fe-safe/Rotate* is discussed in section 21.

- *fe-safe* Custom Module Framework (CMF)

fe-safe Custom Module Framework allows users to create and modify fatigue analysis methods. Confidential algorithms are created in plug-in libraries using a C++ API. Using the Custom Module Framework, algorithms can be added to those supplied with *fe-safe* to operate seamlessly in the *fe-safe* environment.

fe-safe uses its own powerful fatigue loading capabilities to assemble the tensor time histories, which are passed to the custom fatigue algorithm. Stress, strain and temperature variation and node-by-node material property variations are supported, as well as custom FE variables. User-defined material properties may be retrieved from material databases. After analysis, standard and user-defined contours, logs and histories are returned to *fe-safe* to make use of its reporting capabilities.

Batch and distributed processing are also supported.

For further information and assistance with the usage of the API please contact your local SIMULIA support representative.

1.1.2 *safe4fatigue*

safe4fatigue is an integrated system for managing advanced fatigue and durability analyses from measured or simulated strain signals, peak/valley files and cycle histograms. Results may be in the form of cycle and damage histograms, cycle and damage density diagrams, stress-strain hysteresis loops or plots of fatigue damage.

safe4fatigue has been optimised for use on Windows and Linux platforms. Interfaces to many common data acquisition systems and data structures are included. Alternatively, data can be acquired using *fe-safe* data acquisition tools.

safe4fatigue incorporates powerful signal processing functionality, including modules for amplitude analysis, frequency analysis and digital filtering. The signal processing modules can also be purchased separately, for installations where fatigue analysis is not required.

safe4fatigue includes the *fe-safe* Material Database (see above), and comprehensive material data handling functions.

Typical applications of *safe4fatigue* include automotive and aerospace component validation, 'road load' data analysis, on-line fatigue damage analysis, accelerated prototype testing and civil engineering structure monitoring.

Powerful macro recording and batch processing functions make repetitive tasks and routine analyses straightforward to configure and easy to run.

safe4fatigue is included in *fe-safe* at no additional cost.

1.1.3 *fe-safe/TURBOLife*

fe-safe/TURBOLife has been developed in partnership with AMEC Foster Wheeler to assess creep damage, fatigue damage and creep fatigue interactions. *fe-safe/TURBOLife* creep fatigue algorithms have been successfully applied to nuclear power plant components, power station boilers, gas turbine blades, steam turbine components, automotive exhaust components and turbocharger impellers.

fe-safe/TURBOLife is licensed separately and is an additional module to the standard *fe-safe*. Since this module is for use with finite element model data, it is not available as an extension to *safe4fatigue*.

Use of *fe-safe/TURBOLife* is discussed in the separate *fe-safe/TURBOLife* User Manual.

1.1.5 *Verity®* in *fe-safe*

Verity is the original, patented mesh-insensitive Structural Stress Method developed by the Battelle Institute that allows engineers to predict failure locations and calculate fatigue lives for welded joints

and structures.

Verity in fe-safe is licensed and sold separately, and is an additional module to the standard *fe-safe*. Since this module is for use with finite element model data, it is not available as an extension to *safe4fatigue*.

Verity in fe-safe allows both welded and non-welded areas to be analysed in a single operation and displayed as a single fatigue life contour plot.

Use of *Verity® in fe-safe* is discussed in the separate *Verity® in fe-safe* User Manual.

1.2 *fe-safe* – data

1.2.1 *fe-safe* material database

fe-safe and *safe4fatigue* are supplied with a comprehensive database containing fatigue properties for commonly used materials.

Material data is managed within the main application environment. Functions are available for creating new material records, editing, sorting and plotting material properties and approximating fatigue parameters. All material data management functions are available in both *fe-safe* and *safe4fatigue*.

1.2.2 Material data archive

Dassault Systèmes SIMULIA maintains an extensive archive of materials fatigue data. Material data from the archive is made available to customers on request. Requested material data will normally be added to the *fe-safe* material database, for distribution with future releases.

1.2.3 Material testing and material search services

To supplement the materials data archive, Dassault Systèmes SIMULIA offers a material testing service and a material data search service.

1.3 New features in this release of *fe-safe*

A list of new features in this version is documented in the release notes.

1.4 *fe-safe* – Documentation

1.4.1 *fe-safe* User Guide

The *fe-safe* User Guide is split into separate documents, comprehensively covering the use of the software, fatigue theory and signal processing theory, as follows:

User Guide & Technical Notes

User Guide Appendices

Tutorials

Fatigue Theory Reference Manual

This document is based on the publication “*Modern Metal Fatigue Analysis*” by John Draper, Founder and former CEO of Safe Technology Limited.

Signal Processing Reference Manual

This is based on the course notes for the “*Signal Processing*” training course by John Draper.

1.5 A complete copy of the user guide is included in the *fe-safe* software, via the online help, and in the *fe-safe* installation directory in Adobe® PDF format.

1.6 *fe-safe* – services

1.6.1 Training

Dassault Systèmes SIMULIA provides training courses in:

- Theory and Application of Modern Durability Analysis
- Practical hands-on *fe-safe* training

Courses are available in-house and can be tailored to customers' requirements.

1.6.2 Software development

Dassault Systèmes SIMULIA operates a continuous programme of product development, and collaborates closely with leading engineering companies to ensure that *fe-safe* provides the capabilities required for real industrial applications. Many developments are the result of specific requests or suggestions from new or existing customers.

New features undergo rigorous testing with sample test data and with customer models. Consistent and accurate correlation with real-world test results ensures that *fe-safe* maintains its position as the technology leader for durability assessment and failure prevention.

1.6.3 Software customisation

fe-safe provides unique capabilities. The software is tailored to satisfy specific requirements. Customisation projects vary in size and complexity, from interfacing to bespoke data formats or in-house software, to adding major new analysis features.

2 Getting started

2.1 Installing the *fe-safe* software suite

The *fe-safe* software suite should be installed as described in section 3. Before running the software for the first time a licence key is required. Details of the various licensing options are discussed in section 4.

Systems administrators should familiarise themselves with sections 3 and 4.

2.2 Running the program

In Windows, the program is started by selecting the *fe-safe* menu option from the Windows “Start” menu, i.e.:

```
Start >> Programs >> fe-safe x.y >> fe-safe
```

Where x.y is the version number of *fe-safe*.

On Linux platforms, the program is started by running the script *fe-safe* located in `<install_dir>/linux_a64/code/bin`

The licence key determines whether the software runs as *fe-safe* or *safe4fatigue*.

2.3 The *fe-safe* user interface

The user interface, which is common to all platforms, is shown in *Figure 2.3-1*. It consists of:

- The FEA-fatigue dialogue box (not used in *safe4fatigue*);
- A window listing the loading files (data files);
- A window containing the material databases;
- A window to show details of the open FEA file (not used in *safe4fatigue*);
- A message window.

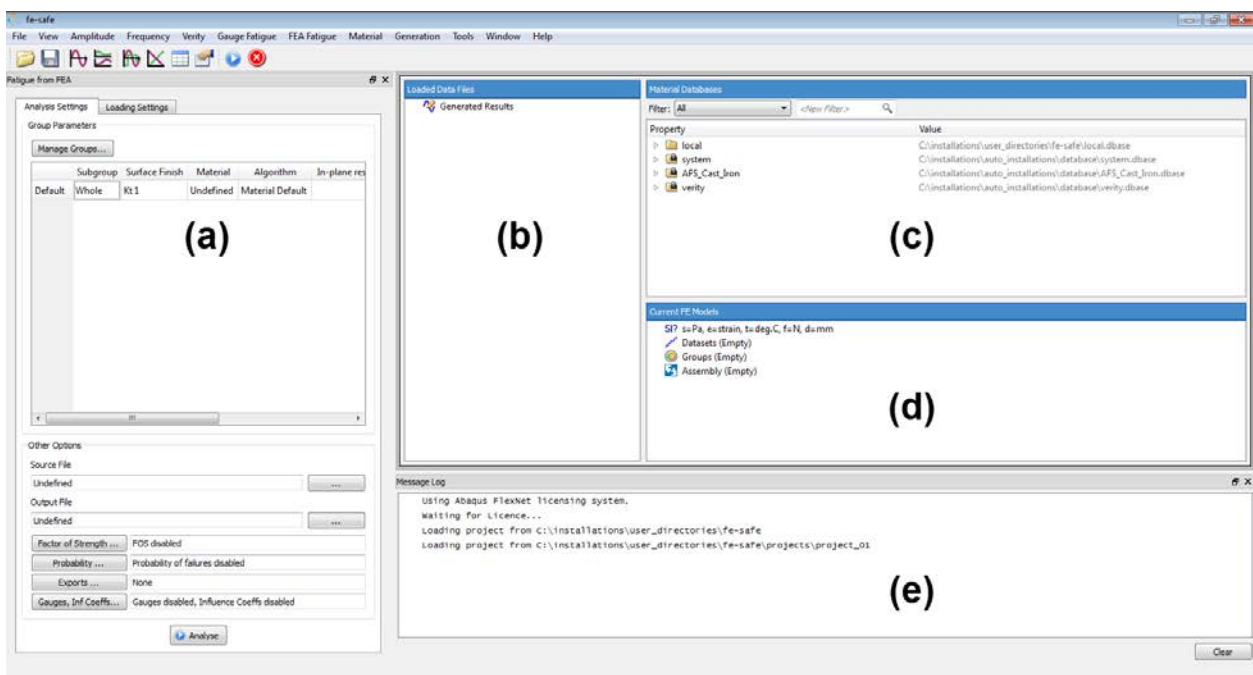


Figure 2.3-1 The *fe-safe* user interface

2.4 Getting started with *safe4fatigue*

The following sections from Volume 1 of this User Guide apply to *safe4fatigue*:

Section 7	Using <i>safe4fatigue</i>
Section 8	Material properties
Section 9	Signal generation methods
Section 10	Signal processing methods
Section 11	Fatigue analysis from measured signals [1]: using S–N curves
Section 12	Fatigue analysis from measured signals [2]: strain–life methods
Section 23	Macros and batch mode operation

Some of the material properties described in section 8 apply only to *fe-safe*.

safe4fatigue users should also familiarise themselves with the Fatigue Theory and Signal Processing Reference Manuals .

Section 2.4.1 below describes a simple signal processing operation using *safe4fatigue*.

Section 2.4.2 below describes a typical fatigue analysis from a measured signal using a strain–life method.

Note that all of the functionality of *safe4fatigue* is available in *fe-safe*.

2.4.1 A simple signal processing operation

This example demonstrates configuration and execution of a signal processing operation in *safe4fatigue*.


Step 1: Open the data file

Open a data file by selecting **File >> Data Files >> Open Data File(s)...**




Opened data files are shown in the **Loaded Data Files** window (b). Files may also be opened by dragging the file name into the **Loaded Data Files** window (drag and drop).

Step 2: Examine the data.

Expand the filename to display all data channels contained in the file. Highlight one or more channels in the **Loaded Data Files** window. Highlighted channels will be included in the analysis.

Detailed information about a channel can be displayed by highlighting the channel then clicking on the **Properties** icon: .

Data can be plotted by clicking on the **Plot** icon: .

If more than one channel of data is selected, stacked plots () , overlaid plots () or cross-plots () can be produced.

Data can be presented in a tabular numerical format by clicking on the **Numerical Display** icon: .

Step 3: Select the signal processing function

Select the required function from either the **Amplitude** or **Frequency** menu. Refer to section 10 for a full description of each signal processing function, including the inputs required, the output produced and analysis options.

For example: perform a peak–valley analysis of the signal by selecting **Amplitude >> Peak–Valley (and P–V Exceedence)...**

The output files produced are added to the list of files in the **Loaded Data Files** window.

2.4.2 A simple fatigue analysis from a measured signal using a local strain–life algorithm

This example demonstrates using *safe4fatigue* to perform a simple fatigue analysis from a measured signal using a local strain–life algorithm.

Step 1: Open the data file

Open a data file by selecting **File >> Data Files >> Open Data File(s)...**

Opened data files are shown in the **Loaded Data Files** window (*b*). Files may also be opened by dragging the file name into the **Loaded Data Files** window (drag and drop).

Step 2: Select the loading history

Expand the filename to display all data channels contained in the file. Highlight one or more channels in the **Loaded Data Files** window. The highlighted channel is the loading history that will be used for the analysis.

Step 3: Select the material database

In the **Material Databases** window, highlight the database that contains the material to be used for the fatigue analysis, for example `local.dbase`.

Step 4: Select the analysis method

Select the required analysis method from the **Gauge Fatigue** menu, for example: **Gauge Fatigue >> Uniaxial Strain Life from Time Histories...** (see section 12).

Step 5: Configure the analysis

Select a material from the drop–down list. The materials available in the list depend on the material database selected in Step 3.

Define a value for the stress concentration factor, K_t . By default, $K_t = 1$ (smooth finish).

Use the drop–down list to select whether or not to perform a mean stress correction.

A sensitivity analysis can be performed by selecting the **Perform Sensitivity Analysis** checkbox. If this option is checked the configuration options for the sensitivity analysis become available.

Using the default **Analysis Range** settings ensures that the full time history is included in the analysis.

Determine which output file types should be produced using options in the **Output Options** area of the **Local Strain Analysis from Time History** dialogue.

Step 6: Performing the analysis

Click **OK** to run the analysis.

When the analysis is complete:

- a summary results is displayed in a dialogue box;
- the output files selected in step 5 will be added to the list of files in the **Loaded Data Files** window.

2.5 Getting started with *fe-safe*

As all of the functionality of *safe4fatigue* is included in *fe-safe*, sections listed in 2.4 above are common to *fe-safe* and *safe4fatigue*. It is therefore advised for *fe-safe* users to familiarise themselves with those sections as well.

The following sections of this User Guide apply to *fe-safe*:

Section 5	Using <i>fe-safe</i>
Section 8	Material properties

Section 13	Defining fatigue loadings
Section 14	Fatigue analysis of elastic FEA results
Section 15	Fatigue analysis of elastic–plastic FEA results
Section 16	Fatigue analysis of welded steel joints
Section 17	Factor of strength and probability–based fatigue methods
Section 18	Conventional high temperature fatigue
Section 19	Fatigue analysis from frequency domain loading
Section 23	Macros and batch mode operation

Section 5 gives an introduction to the operation of *fe-safe*.

Section 8 describes how material properties are defined.

Section 13 describes how fatigue loadings are defined, from simple constant amplitude loading to complex multi–block loading definitions.

Sections 14 to 19 discuss the various fatigue analysis algorithms used in *fe-safe*, including analysis of elastic FE models, elastic–plastic FE models and welded joints, factor of strength and probability–based methods, conventional (isothermal) high temperature fatigue and frequency–based fatigue.

The examples below demonstrate the simple steps required to configure and run analyses in *fe-safe*.

2.5.1 A simple analysis from FEA

A simple analysis of a linear elastic FEA model could consist of importing the nodal stress results for an applied load, then calculating fatigue lives for a time history of the applied load. In the following section, letters in brackets, e.g. (e), refer to *Figure 2.3-1*.

Setting up the analysis:

Step 1: Open the FEA file

The FEA results file is opened using **File >> FEA Solutions >> Open Finite Element Model...** The file(s) can then be pre–scanned to speed up reading times. *fe-safe* reads in the model reporting its progress in the message window (e). A summary of the file is shown in the **Current FE Models** window (d). Information on named element groups is shown in the **Fatigue from FEA** dialogue box (a). Elements in un–named groups are shown as ‘Default’.

See Appendix G, for details regarding interfacing to all supported FE file formats.

Step 2: Open the loading file

The loading file is opened using **File >> Data Files >> Open Data File(s)...** It will be shown in the **Loaded Data Files** window (b). It may also be opened by dragging the file name into the **Loaded Data Files** window (drag and drop).

Step 3: Apply the loading to the FEA stresses

To associate the loading file with the FEA stresses, the loading file is highlighted, and the stress dataset in the **Current FEA Models** window (d) is highlighted. In the **Fatigue from FEA** dialogue box (a), after selecting the **Loading Settings** tab, the **Add... >> A LOAD * dataset** option is selected. If the loading represents a real–life duty cycle, e.g. 100 track miles, or 50 flight hours, this may be defined by editing the **Loading is equivalent to...** item in the loading tree. Loading definition can be saved and then imported using the **File >> Loadings >> Open FEA Loadings File...** menu item (see section 13).

Step 4: Select the material

The material is selected by highlighting it in the **Material Databases** window (if no materials are listed

expand the database name).

With the material highlighted, go to the **Group Parameters** section of the **Fatigue from FEA** dialogue box, point the cursor to the **Material** cell in the relevant group row and double-click then confirm your selection. To change the material for all groups, double-click the **Material** column header.

Note that once the material has been selected, the appropriate analysis algorithm is shown in the **Algorithm** column.

Step 5: Define the output file

A default output file is shown in the **Output Options** of the **Fatigue from FEA** dialogue box (a). This will be of the same file type as the FEA input file (for example, an Abaqus .odb file, an ANSYS .rst file). Other output formats can be selected (See Appendix G)

Step 6: Perform the analysis

To perform the analysis, press the **Analyse** button. A summary table will be displayed showing all the inputs defined for the analysis. Click **Confirm** to start the analysis. The analysis will proceed, with progress reported in the **Message Log** window (e). The **Analyse** button changes to **Abort** to allow the user to stop the analysis. On completion, the results will be exported to the output file, for display as contour plots using appropriate FEA viewer. A summary of the results can be shown by using **View >> View FEA Fatigue Results Log**. Each new analysis result is appended to this log file.

The *fe-safe* analysis method:

The *fe-safe* analysis has comprised the following steps:

- (a) The elastically-calculated FEA nodal stress tensor is read.
- (b) Each of the 6 components of the stress tensor is multiplied by the time history of the applied loading, to produce a time history of each of the 6 components of the stress tensor.
- (c) The time histories of the in-plane principal stresses are calculated. (The out-of-plane stress is checked for possible contact loading – the following steps assume no contact).
- (d) The time histories of the three principal strains are calculated from the stresses.
- (e) For a strain-life analysis (for example, a Brown-Miller analysis), a multi-axial cyclic plasticity model is used to convert the elastic stress-strain histories into elastic plastic stress-strain histories. For an S-N curve analysis this step is omitted.
- (f) For a shear strain or Brown-Miller analysis, the time histories of the shear and normal strain and the associated normal stress are calculated on three possible planes. For an S-N curve analysis a plane perpendicular to the surface is defined, and the time history of the stress normal to this plane is calculated.
- (g) On each plane the fatigue damage is calculated. For each plane the individual fatigue cycles are identified using a 'Rainflow' cycle algorithm, the fatigue damage for each cycle is calculated and the total damage is summed. The plane with the shortest life defines the plane of crack initiation, and this life is written to the output file.
- (h) During this calculation, *fe-safe* may modify the endurance limit amplitude. If all cycles (on a plane) are below the endurance limit amplitude, there is no calculated fatigue damage on this plane. If any cycle is damaging, the endurance limit amplitude is reduced to 25% of the constant amplitude value, and the damage curve extended to this new endurance limit.
- (i) Steps (a) to (h) are repeated for each node.

2.5.2 An analysis with two load histories applied to an FEA model

This analysis could be for a component with two or more loads applied to a component, each load having its own time history of loading. The FEA analysis will consist of a linear elastic FEA for each load applied separately, producing two stress datasets. Fatigue lives will be calculated for the component with both load histories applied together. This is called a 'scale and combine' analysis. *fe-safe* allows up to 4096 load histories to be applied simultaneously.

Setting up the analysis:

The analysis follows the same sequence as before, with the following exceptions.

- Two loading history files will be opened (or one file containing at least two channels of loading data).
- The FEA model will contain at least two stress datasets.
- Step 3 is performed twice, i.e.:
 - i. The first loading file is highlighted, as is the stress dataset to which it is applied. In the **Fatigue from FEA** dialogue box (a), the **Loading Settings** tab is selected, and the **Add... >> A Load * dataset** option is used.
 - ii. The second loading file is highlighted, as is the stress dataset to which it is applied. In the **Fatigue from FEA** dialogue box (d), the **Loading Settings** tab is selected, and the **Add... >> A Load * dataset** option is used.

The **Analyse** button initiates the analysis, as before.

fe-safe will prohibit the use of uniaxial fatigue methods when multiple load histories are applied. This is because the principal stresses may change their orientation during the loading history.

The *fe-safe* analysis method:

The analysis method has only one change. At steps (f) and (g) above *fe-safe* will use a critical plane procedure to search for the plane of crack initiation. (see section 7.5 of the Fatigue Theory Reference Manual).

fe-safe does not peak/valley the loading histories before using them in the analysis. This means that *fe-safe* is not assuming that a peak or valley in the principal stresses will always be caused by a peak or valley in the loading. This is the most rigorous assumption. However, the user may request that *fe-safe* performs a multi-channel peak/valley extraction on the signals as a default setting. (Alternatively, the user may produce peak/valley signals as a separate operation (see section 10). This will reduce the analysis time, but may lead to inaccuracies in the calculated lives. (see section 4 of the Fatigue Theory Reference Manual for further discussion of multi-channel peak valley operations). If the user has selected the peak/valley option, it is strongly recommended that the analysis is repeated for a selection of the most critical elements with the peak/valley option turned off, to compare the fatigue lives.

2.5.3 An analysis with a sequence of FEA stresses

In the previous examples, loading was applied in the form of load history files. For some analyses the FEA may be used to model a series of events, with the stress results being written for each event. Examples are the analysis of an engine crank shaft, with the stresses calculated at every 5° of rotation of the crank shaft, through two or three complete revolutions. The stress history at each node is then defined by the sequence of FEA solutions. *fe-safe* will analyse this sequence of stresses.

fe-safe allows the stresses to be scaled, and applied in any sequence, in which case the FEA must be a linear elastic analysis. However, if no scale factors are applied to the stresses, then the FEA need not be a linear analysis. Nor need it be an elastic analysis; the analysis of inelastic (elastic-plastic) FEA is

discussed in section 15. The following description assumes a linear elastic FEA.

Setting up the analysis:

Open the FEA file – as in 2.5.1.

Define the required sequence of stress datasets in the loading tree (see section 13), which can be accessed through the **Loading Settings** tab on the **Fatigue from FEA** dialogue box. Adding multiple datasets can be simplified by a manual editing: a continuous list of datasets can be specified with a hyphen, e.g. datasets 1 through 10 would be '1–10', a list of datasets incrementing or decrementing by a fixed amount can be specified by adding the increment within parenthesis after the end dataset number, e.g. datasets 1, 4, 7 and 10 would be '1–10(3)'.

Select the material and define the output file – as in 2.5.1.

Perform the analysis by pressing the **'Analyse'** button

2.5.4 Other features that can be included

Complex loading conditions

The analyses of single loads, multiple loads, block loading and sequences of FEA stress results can be combined to simulate complex loading conditions.

Named element groups

For each named group of elements or nodes, the user can specify different surface finish corrections, materials, analysis algorithms, mean stress corrections, and residual stresses. See section 5 for more details.

Factors of strength (FOS)

For a specified target life, *fe-safe* will calculate, at each node, the factor of strength (FOS), which, when applied to the elastic FEA stresses, will give the required life. This shows how much the component is over- or under-strength in terms of FEA stresses. See section 17 for more details.

Failure rates

For one or more specified target lives, *fe-safe* will combine statistical variability of material data, and variability in loading, to estimate the failure rate. Data from a series of target lives can be used to derive a 'warranty claim' curve. See section 17 for more details.

Fatigue reserve factor (FRF)

For a specified target life, *fe-safe* will calculate fatigue reserve factors using Haigh-type diagrams. Factors can be calculated for the stress amplitude at a constant mean stress; the mean stress at a constant stress amplitude; and the factor to be applied to both the stress amplitude and the mean stress. See section 17 for more details.

Haigh diagram

A Haigh diagram, showing the most damaging cycle at each node, can be created and plotted. The results for all nodes on the model, or on selected element groups, are superimposed on a single diagram. This provides a visual indication of the stress-based FRF's for the complete model. See section 14 for more details.

Plot materials data

Materials data can be exported as plot files and plotted in *fe-safe*. Data can be plotted at different temperatures and strain rates. Results from several materials can be overlaid. See section 8 for more details.

Approximate materials data

In the absence of specific test data, the materials strain–life and S–N data can be approximated and saved in the database. See section 8 for more details.

Load sensitivity analysis

For components with more than one applied load direction, *fe-safe* will remove each load direction in turn and recalculated the fatigue life for selected critical locations. This can be used to simplify component testing, by showing which load directions are important and the potential failure locations. It should also be used as a check on design sensitivity, as it may show that the life is shorter if some loads are omitted. See section 22 for more details.

Export detailed results

For selected elements, additional detailed results can be exported, and plotted in *fe-safe*. These can include:

- Time histories of stress tensors, principal stresses and strains, and the damage parameters (normal stress/strain, shear strain, etc) on the critical plane. These results can be plotted and further analysed (e.g. Rainflow cycle counted) in *fe-safe*. See section 7 for more details.
- A list of the most damaged *n* nodes. See section 22 for more details.
- A ranked list of nodes eliminated as non–damaged. See section 22 for more details.
- A traffic light contour plot showing the fatigue results as ‘pass’, ‘fail’ or ‘marginal’. See section 22 for more details.

Saving standard analyses

The set–up for the analysis can be saved for re–use. This allows standard analysis procedures to be created and saved. They can be applied to the same or a different model. Individual settings can be changed at run time. See section 5 for more details.

Batch analysis

The standard analyses can be re–run interactively or in batch mode. See section 23 for more details.

Elastic–plastic FEA

Elastic–plastic FEA results can be analysed for certain loading sequences. See section 15 for more details.

Additional effects

Additional scale factors can be included to allow for additional effects (for example size effects, environmental effect, etc.). See section 5 for more details.

Saving the loaded FEA model

The file created by *fe-safe* to hold the FEA model information can be saved for rapid re–use. See section 5 for more details.

Export diagnostics

Detailed diagnostics can be written to a log file. See section 22 for more details.

2.6 Getting started with *fe-safe/Rotate*

fe-safe/Rotate is an additional module included in *fe-safe* as standard, which speeds up both the FEA and the fatigue analysis of rotating components by taking advantage of their axial symmetry. The *fe-safe/Rotate* module is described in detail in section 21.

2.7 Getting started with *fe-safe/TURBOLife*

fe-safe/TURBOLife is an optional module for *fe-safe*, which performs creep-fatigue crack initiation calculations for engineering components under thermo-mechanical loadings. The *fe-safe/TURBOLife* module is described in detail in the separate *fe-safe/TURBOLife* User Manual.

3 This chapter has been released in a separate document

4 This chapter has been released in a separate document

5 Using *fe-safe*

5.1 Introduction

fe-safe is a suite of software for fatigue analysis from finite element models. It calculates:

- fatigue lives at each node on the model – and thereby identifies fatigue crack sites;
- stress-based factors of strength for a specified target life – these show how much the stresses must be changed at each node to achieve the design life;
- probability of failure at the design life, at each node;
- probability of failure at a specified series of lives, to produce a ‘warranty curve’.

The results of these calculations can be plotted as 3-D contour plots, using the FEA graphics or third party plotting suites. The fatigue results can be calculated from nodal stresses or elemental stresses.

In addition, *fe-safe* can output:

- the effect of each load on the fatigue life at critical locations – to show if fatigue testing can be simplified, and for load sensitivity analysis;
- detailed results for critical elements, in the form of time histories of stresses and strains, orientation of critical planes, etc.

fe-safe also includes a powerful suite of signal processing software, *safe4fatigue* (see section 7). This allows the analysis of measured load histories and *fe-safe* results output. The facilities include:

- plotting and digital listing;
- manipulation, for example editing, scaling, filtering, integrating/differentiating;
- amplitude analysis, for example Rainflow cycle counting, level crossing analysis;
- frequency domain analysis, for example PSD, transfer function;
- fatigue analysis for strain gauge data and other time history and Rainflow matrix data.

5.2 Starting *fe-safe*

5.2.1 Running the *fe-safe* program

In Windows, *fe-safe* is started by selecting the *fe-safe* menu option from the Windows “Start” menu, i.e.:

```
Start >> Programs >> fe-safe x.y >> fe-safe
```

On Linux platforms, the program is started by running the script `fe-safe` located in `<install_dir>/linux_a64/code/bin`

These methods assume that *fe-safe* has been installed and configured as described in section 3, and that an appropriate licence key has been installed as described in section 4.

5.2.2 Project directory

On starting *fe-safe* the *fe-safe* landing page is shown, which can be used to create a new project or load an existing one as shown in *Figure 5-1* below.

The project directory is used to store configuration files for an FEA Fatigue analysis, together with the loaded FEA Models (FESAFE.FED directories), and analysis results to maintain a record of the entire analysis and to reference the files later.

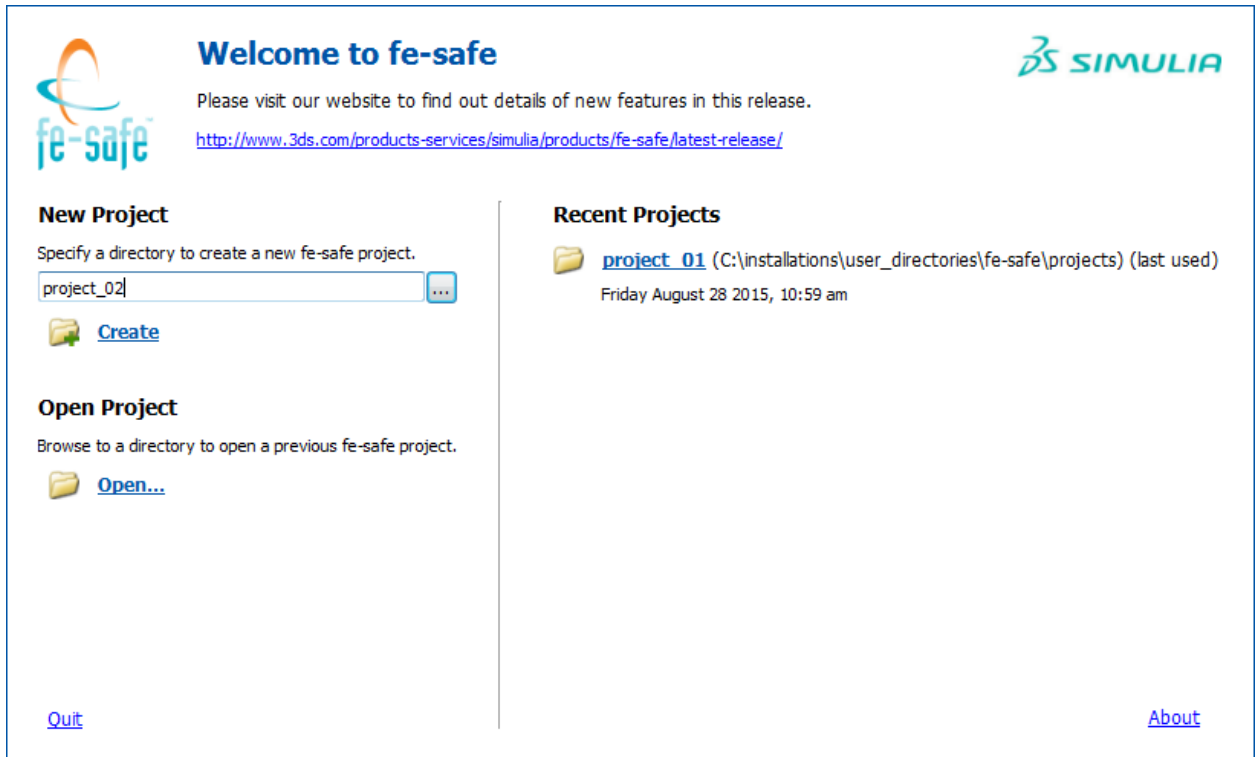


Figure 5-1

5.2.3 Creating a new project with default settings

To create a new project with the default settings, from within *fe-safe*, use the main menu option **File >> Project >> New Project...**

The default configuration can be customised to suit user requirements, detailed information can be found in Sections 5.10 and 5.11.

5.2.4 The *fe-safe* user interface

The user interface, which is common to all platforms, is shown in *Figure 5-2*. It consists of:

- a. The FEA-fatigue dialogue box;
- b. A window listing the loading files (data files);
- c. A window containing the material databases;
- d. A window to show details of the open FEA file;
- e. A message window.

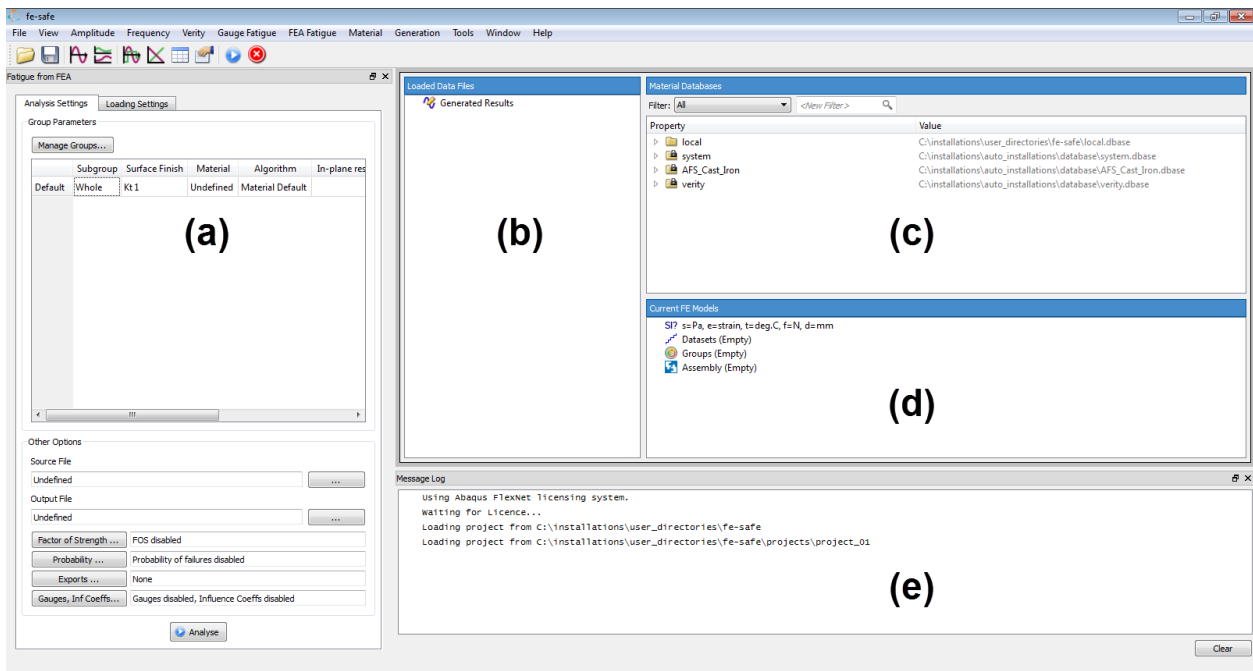


Figure 5-2: The *fe-safe* user interface

The layout of the user interface can be adjusted to suit user preference and the screen size.

On Windows platforms, the **Current FE Models** and **Loaded Data Files** windows support “drag-and-drop” methods. This means that selecting files in another Windows application (for example Windows Explorer), and then dragging them into the appropriate *fe-safe* window can automatically load the files.

When a file is “dragged-and-dropped” to the **Loaded Data Files** window, the file is added to the list of available data files.

When a file is “dragged-and-dropped” to the **Current FE Models** window, *fe-safe* starts the process of importing the model.

Tip: If the *fe-safe* application is not visible, or is partly obscured by another application, then drag the files to the *fe-safe* icon on the Windows taskbar, and hover over it for a couple of seconds (without releasing the mouse button) until *fe-safe* becomes visible.

5.3 Inputs required for a fatigue analysis

To perform a fatigue analysis, *fe-safe* requires three inputs:

- The stresses at each point in the model: *fe-safe* can use elastic stresses from an elastic finite element (FE) analysis, or elastic-plastic stresses and strains from an elastic-plastic FE analysis. If necessary, *fe-safe* will perform a plasticity correction in order to use elastic FE stresses with strain-based fatigue algorithms.
- A description of the loading: load histories can be imported from industry-standard file formats or entered at the keyboard. Complex loading conditions can also be defined, including combinations of superimposed load histories, sequences of FEA stresses and block loading. Loading histories and other time-series data are contained in files referred to as *data files*.
- Materials data: fatigue properties of the component material(s) are required; a comprehensive material database is provided with *fe-safe*.

5.4 Importing datasets from FE models

5.4.1 Supported file types

fe-safe can read Finite Element analysis data (i.e. stresses, strains and temperatures) and geometry data from the following third-party file types:

FE package	File type	File extension	Versions supported
Abaqus	FIL	.fil	
	ODB output database	.odb	Windows: 6.13 to 2018 ³ Linux 6.13 to 2018
ANSYS	RST (results file)	.rst	Versions 5.5 to 17.0
NASTRAN	f06 (PRINT) file	.f06 ¹	CSA/Nastran v98 and MSC.Nastran for Windows v4.5 NX Nastran
	OP2 output file	.op2	MSC.Nastran NX Nastran
I-DEAS	UNV	.unv	SDRC I-DEAS / Master Series versions 1 to 12 NX I-deas 5 to 6.3 (<i>fe-safe</i> detects the file version)

1 Importing geometry data is not supported.

2 Requires the corresponding .neu file with mesh data, importing geometry also requires the corresponding .pnu file with geometry data.

3.Abaqus =< 2016 ODB files are not supported on Windows XP

Detailed information on interfacing to the various FE data formats is given in Appendix G.

fe-safe endeavours to maintain interface support to the latest versions of supported third-party FE packages.

5.4.2 Extracting datasets from the source FE model

A dataset in *fe-safe* is a set of stresses, strains, forces or temperatures from an FE model. In the FE software these sets of data may be referred to as steps, increments or datasets, depending on the FEA suite being used.

To extract data from an FE model select **Open Finite Element Model...** from the **FEA Solutions** section of the **File**

menu. The type of model being imported is determined by the extension of the model file name.

By default *fe-safe* will ask the user if they wish to pre-scan the model(s). Selecting **Yes** will allow for user control of which datasets to read using pre-scan mode. Selecting **No** will extract datasets based on the settings on the **Import** tab, of the **Analysis Options** dialogue, and at positions specified in the appropriate interface options dialogue, based on FE file type, as discussed in Appendix G.

To configure the extraction without pre-scan mode use the **Import** tab on the **FEA Fatigue >> Analysis Options** dialogue, it is found in **Full-Read Options** section. The default settings are:

- All stress datasets will be selected;
- All non-ANSYS temperature datasets will be selected;
- If **Read strains from FE Models** is selected then all strain datasets will be selected.
- If **Read forces from FE Models** is selected then all force datasets will be selected.

When pre-scanning files, all datasets will be located and the basic information extracted. A maximum of 256000 datasets can be pre-scanned, an error message is shown if an attempt is made to load more than 256000 datasets. The **Select Datasets to Read** dialogue will then be displayed showing all datasets for the selected position, see *Figure 5-3*. For each load step a separate line appears in the pre-scan list which acts as a header for all increments and datasets identified in this step. For general details on pre-scan file, see Appendix E.

The **Positions** combo box lists all nodal and elemental locations that contain datasets. Changing the **Positions** combo box will change the datasets displayed in the **Datasets** list, see *Figure 5-3*.

Using the checkboxes in the **Quick select** section along with **Apply to Dataset List** button can be used to select ranges of datasets. Otherwise, datasets can be selected manually.

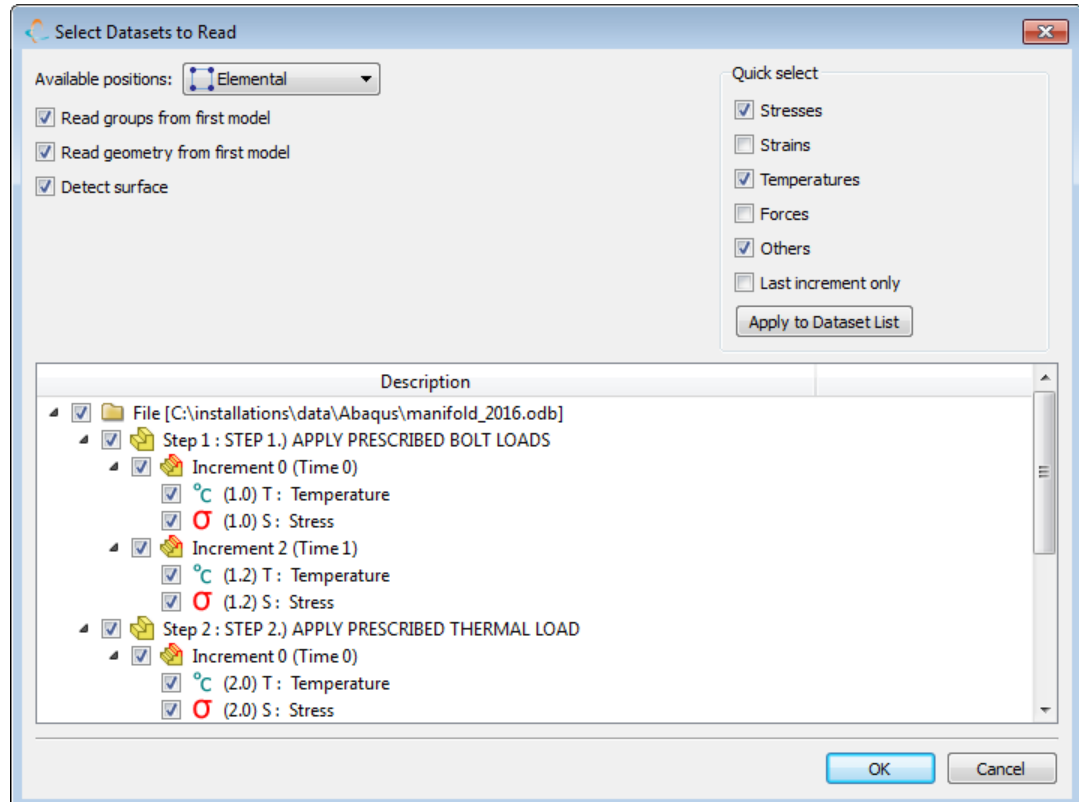


Figure 5-3

Each time a model is opened, the user is prompted to define the units.

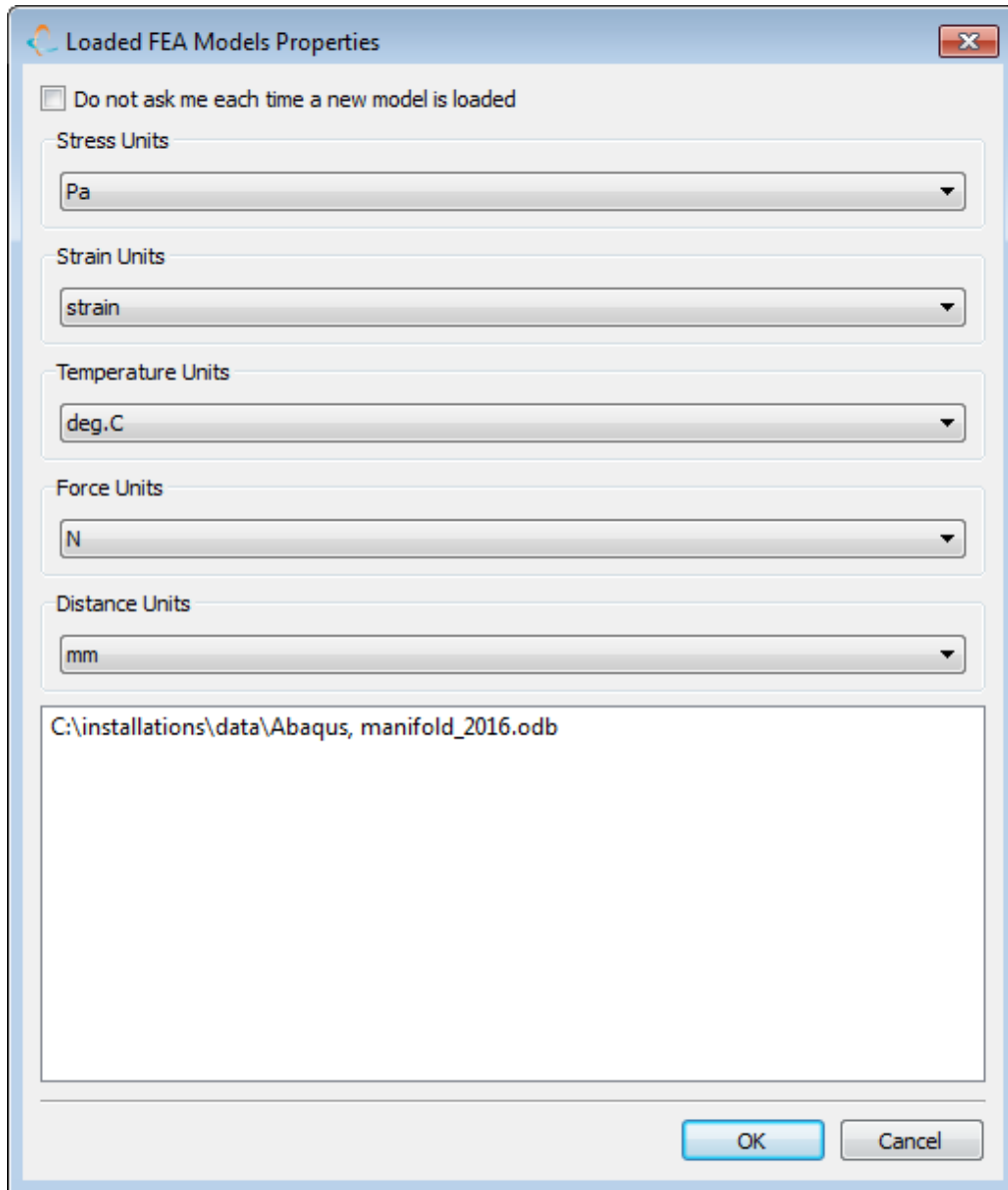


Figure 5-4

For stresses the units can be MPa, KPa, Pa, psi, ksi. For strain the units can be strain(m/m) or microstrain (μE). For temperatures the units can be °C, °F or Kelvin. For forces the units can be N, KN, MN, lbf or klb. For distance the units can be mm, m or in. For all the above unit types a user-defined unit can be set, which requires configuring a conversion scale factor to SI units (MPa, strain, °C, N and mm). The units are then displayed in the **Current FE Models** window.

When the model is imported, pertinent data extracted from the model is written to the “Loaded FE Model” FED directory (see Appendix E) in the project folder. The FED directory stores stress, strain, force and temperature data extracted from the imported FE model.

As data is being extracted from the FE model, the message log reports:

- the names of element or node groups (for nodal datasets node groups are imported, for elemental datasets element groups are imported);
- maximum and minimum direct and shear stresses in each dataset;
- a summary of the temperature datasets found.

Saving the FED directory

The FED directory may be saved for re-use in a later analysis, selecting **Archive FED Directory...** from the **File** menu. A saved FED directory can be retrieved later, in the same way as loading any other FE model file, using the **Open Finite Element Model...** option.

Note that when a FED directory is opened using the **Open Finite Element Model...** option, the contents of the file are used directly, without creating a new FED directory. If a model is to be analysed repeatedly in *fe-safe*, it should be saved to a named FED directory after the first analysis, in order to save read-in time on subsequent analyses.

The following apply to reading FE Model data without pre-scanning.

Reading strain datasets

Strain datasets can be extracted as the model is being imported by checking the **Read strains from FE models** option in the **Analysis Options** dialogue, **Import** tab. When pre-scanning this option does not apply, the strain datasets must be selected in the **Select Datasets to Read** dialogue. This is only required when reading results from an elastic-plastic FE analysis.

Reading force datasets

Force datasets can be extracted as the model is being imported by checking the **Read forces from FE models** option in the **Analysis Options** dialogue, **Import** tab. When pre-scanning this option does not apply, the force datasets must be selected in the **Select Datasets to Read** dialogue. This is only required when reading results for use with Verity.

Reading temperature datasets

If the source FE model contains temperature datasets, these will be summarised in the message log window when the model is being imported, and will also be listed in the **Current FE Models** window.

Appending datasets to the loaded FE model

Additional datasets, (for example additional stress datasets or temperature datasets), can be appended to the loaded FE model using the **Append Finite Element Model...** option. Appended datasets can also be imported from a file having a different file format, providing that the data relates to the same model, and the element and node numbers correspond. Element or node group information is loaded from the first file only (i.e. the file opened using **Open Finite Element Model...**).

Referencing datasets

In all cases, the index used to reference stress and strain datasets is the one displayed in the **Current FE Models** window, which may not be the same as the step number in the source FE model file. Note also that the numbering of stress datasets in the open FE model may change, for example if the model is re-imported after the status of the **Read strains from FE models** option (in the **General FE Options** dialogue) is changed.

5.5 Managing groups

5.5.1 Group information

A named group in *fe-safe* is a set of elements or nodes that can come from an FE model, or can be user defined in which case it can contain both elements and nodes. In the FE software these lists of element or node numbers may be referred to as sets, groups, contour colours or material designations, depending on the FEA suite being used.

fe-safe extracts group information for both element and node groups in the source FE model.

A summary of the element or node groups is displayed in the **Current FE Models** window by expanding the Groups list.

Tip: When pre-scanning is enabled, read just the group information from the first file by deselecting all the datasets in the file.

5.5.2 Group names

fe-safe reads the original group name from the model, then derives a group name for use in *fe-safe* by removing any spaces or illegal characters and replacing with an underscore, then cropping the name if necessary so that it has no more than 120 characters. If this results in more than one instance of the same derived group name then *fe-safe* tags a number to the end of the group name to make it unique.

5.5.3 Group management

The loaded groups can be further managed to control which sections of the model are to be analysed to simplify the fatigue analysis configuration. This can be done in the **Select Groups to Analyse** dialogue, which can be accessed by either:

- using the **FEA Fatigue >> Manage Groups...** menu, or
- right-clicking in the **Current FE Models** window and selecting **Manage Groups...**, or
- clicking the **Manage Groups...** button in the **Group Parameters** area of the **Fatigue from FEA** dialogue

The dialogue is displayed in *Figure 5-5* below:

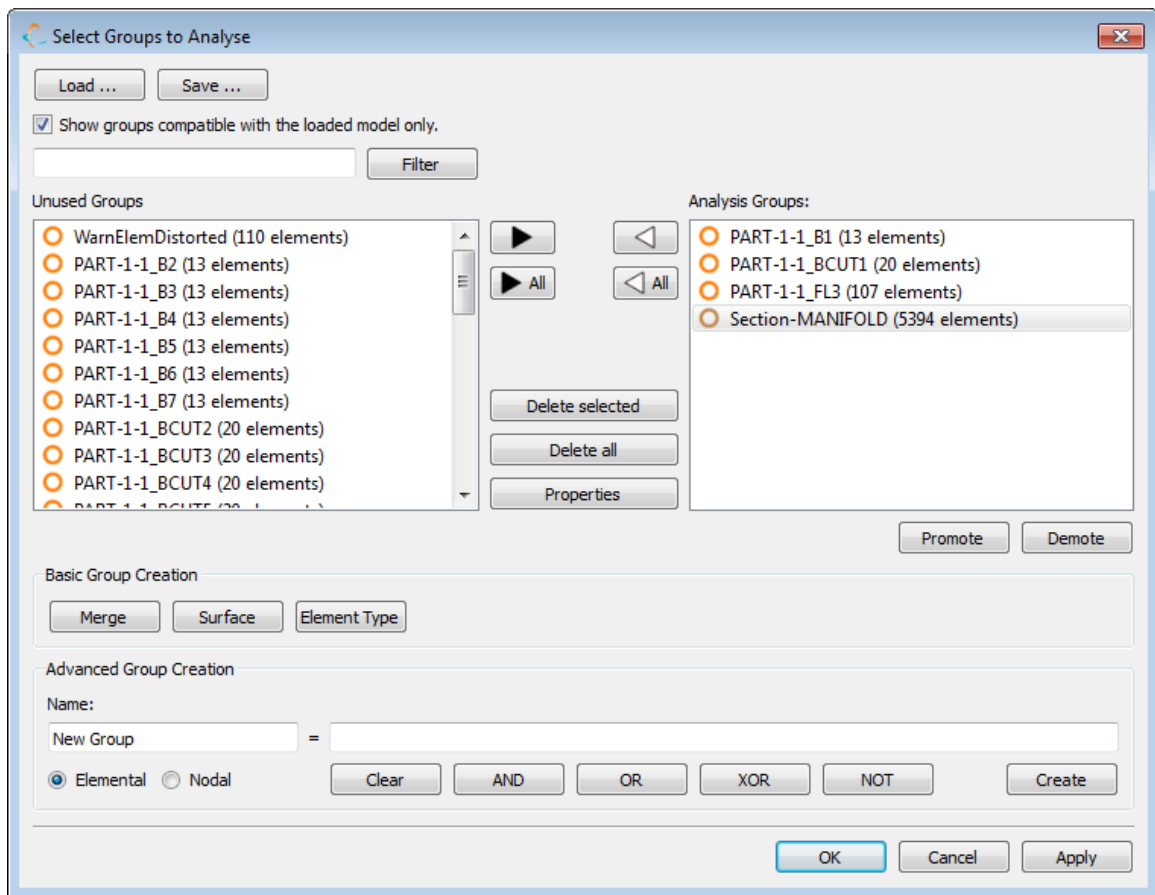


Figure 5-5

A checkbox in the top left hand corner of the dialogue toggles to view all the Groups or only those compatible with the loaded model. While incompatible group may be used in an analysis, they require greater overhead to process and can only be used if the FE mesh is available. When a model contains a large number of groups it may become difficult to locate those of interest. To simplify the navigation a filter can be applied to the list of groups. This filter is case insensitive and does not support the use of wildcards.

User-defined ASCII element/node groups can be imported and exported, using the **Load** and **Save** buttons respectively, or they can be created directly through the **Basic Group Creation** and the **Advanced Group Creation** options at the bottom of the dialogue. These are described in the next section.

Individual or multiple groups can be moved between the list of **Unused Groups** on the left and the list of **Analysis Groups** on the right by first selecting the groups to move and then clicking on either the ► or the ◀ button.

Groups in both lists can be renamed as required, within the naming conventions described in section 5.5.2 above, by selecting a group in either list box and clicking the **Properties** button. This opens the Group Properties dialogue shown in *Figure 5-6* below, where the new name can be set in the **User Name** field.

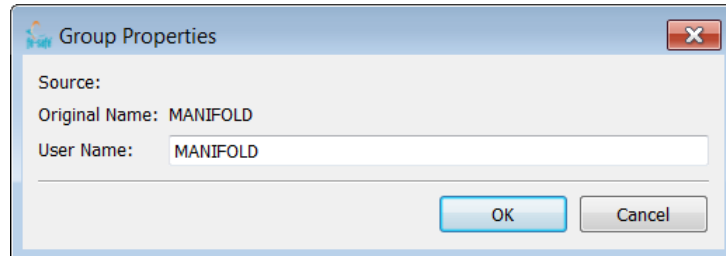


Figure 5-6

The **Group Properties** dialog also contains read-only fields with the original group name and the source file of the model.

Groups to be analysed can be re-ordered (promoted / demoted) and the importance of the groups order is discussed further in section 5.6.9 below.

Any changes made can be applied by clicking either the **Apply** or **OK** buttons, which will result in the groups from the **Analysis Groups** list being added to the **Group Parameters** table within **Fatigue from FEA** dialogue.

5.5.4 User defined groups

User defined groups can be added to your loaded FE models in two ways: either by importing an ASCII file of group items or by defining the group as a combination of existing groups and / or a manually entered lists of items.

To import an ASCII file of group information use the **Load...** button in the **Select Groups to Analyse** dialogue, or the **File** menu item **Open User Defined ASCII FE Group File...** The ASCII group files contain a list of element or node IDs; for elemental stress data element IDs are required and for nodal stress data node IDs are required. A type selection dialogue will be shown to confirm whether the loaded group is elemental or nodal.

Within the ASCII file **GROUP** and **END** tokens can be used to allow multiple named groups to be defined, otherwise the group name will be derived from the stem of the file name. For more information on the format of the file see Appendix E.

Upon adding the group(s) to the **Current FE Models** window the group names are validated to ensure they are unique. If they are not then the group is renamed to a unique name and a message will be shown in a pop-up window - see *Figure 5-7*.

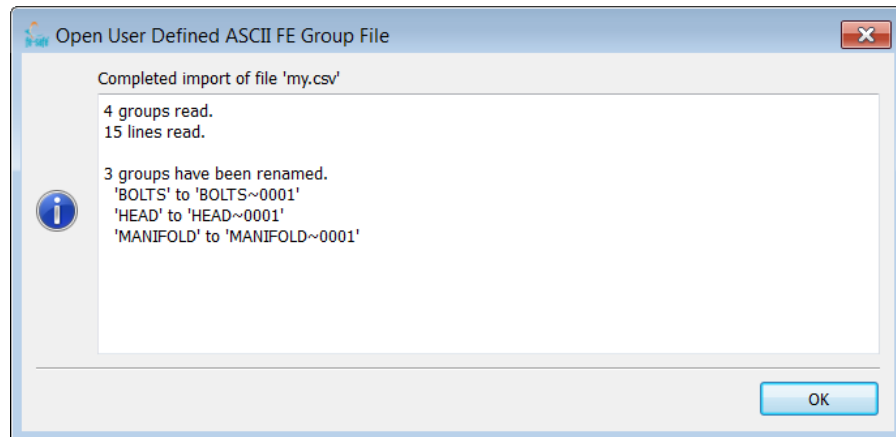


Figure 5-7

Alternatively user defined groups can be created directly through the **Basic Group Creation** and the **Advanced Group Creation** options in the **Select Groups to Analyse** dialogue, see *Figure 5-5*.

New groups can be added to the **Unused Groups** list box on the left by using the **Basic Group Creation** options **Merge** and **Surface** or by using the more complex but flexible **Advanced Group Creation** equation editor.

The two basic options allow one to create a union of two or more groups selected from the list of groups or an intersection of the selected groups with the SURFACE group of the loaded model. This second option will only succeed if the **Detect surface** option was selected when loading the model.

The equation editor allows boolean operators to be used in creating new groups from the existing ones. Double clicking the group name in either of the **Unused Groups** or **Analysis Groups** lists will insert it in the equation editor.

The following boolean operators can be typed in or inserted using the relevant buttons:

- AND – intersection of two or more groups
- OR - union of two or more groups
- XOR – the exclusive or of two or more groups
- NOT – excludes ID's from the selected group

Additionally parentheses can be used to further refine the equation.

Individual item ID's can be manually entered in the equation, delimited by a comma or the OR operator. Adding a continuous list of ID's can be simplified by using with a hyphen, e.g. 1-5 will create a group comprising of ID's 1,2,3,4,5, a list of ID's incrementing or decrementing by a fixed amount can be specified by adding the increment within parentheses, e.g. 1-5(2) will create a group comprising of ID's 1,3,5.

The wildcard operator * can be used to create unions between multiple groups. For example inserting a * character alone in the equation field will create a new group comprising the union of all items within groups in the current model.

Radio buttons at the bottom of the dialogue are used to indicate if the new group is to be nodal or elemental. This choice will determine the item type when no type is specified. To mix element and node IDs in the same group, prefix 'e' to element IDs and 'n' to node IDs, e.g. e1-10, n100 will create a group with elements 1 to 10 and node 100. Note that element and node IDs will only be checked against the mesh at analysis thus the group operators (AND, XOR etc.) will not check consider if a node is on an element.

The source field shown in **Properties**, see *Figure 5-6* above, for a user defined group will contain the equation string rather than the path to the parent model.

5.6 Configuration options

5.6.1 Defining analysis groups

By default, *fe-safe* will analyse the surface elements of the model, but it is also possible to add the sub-surface elements to the analysis (i.e. analyse all elements in the model), as described in section 0 below. Furthermore, individual element groups or node groups can be excluded from the analysis by clicking on **Algorithm** in the **Group Parameters** region of the **Fatigue from FEA** dialogue, then selecting **Do not analyse** in the **Group Algorithm Selection** dialogue.

The following information can be configured individually for each element or node group in the **Group Parameters** region of the **Fatigue from FEA** dialogue, see *Figure 5-8*:

- Analysis subgroup
- Surface finish factor
- Material
- Analysis algorithm
- In-plane residual stress
- Additional SN data scale factor
- SN data knock-down curve

The screenshot shows a dialog box titled "Group Parameters" with a "Manage Groups..." button. Below the button is a table with the following data:

	Subgroup	Surface Finish	Material	Algorithm	In-plane residual stress	SN Scale	Knock-Down
BOLTS	Surface	1	Undefined	Material Default		1	Disabled
HDTOP	Surface	1	Undefined	Material Default		1	Disabled
HEAD	Surface	1	Undefined	Material Default		1	Disabled
MANIFOLD	Surface	1	Undefined	Material Default		1	Disabled
Default	Surface	1	Undefined	Material Default		1	Disabled

Figure 5-8

Group properties for nodes and elements in multiple groups are handled as described in section 5.6.9 below.

5.6.2 Defining subgroups

By default, *fe-safe* will try to detect elements on the surface of the component and limit the fatigue analysis to those elements only. The surface detection is activated by selecting the **Detect surface** option in the **Select Datasets to Read** dialogue, see *Figure 5-3* above. If geometry reading was enabled at the **Select Datasets to Read** dialogue, then surface detection can also be executed after the model data was loaded by right-clicking in the **Current FE Models** window and selecting the **Detect Surface** option, see *Figure 5-9* below.

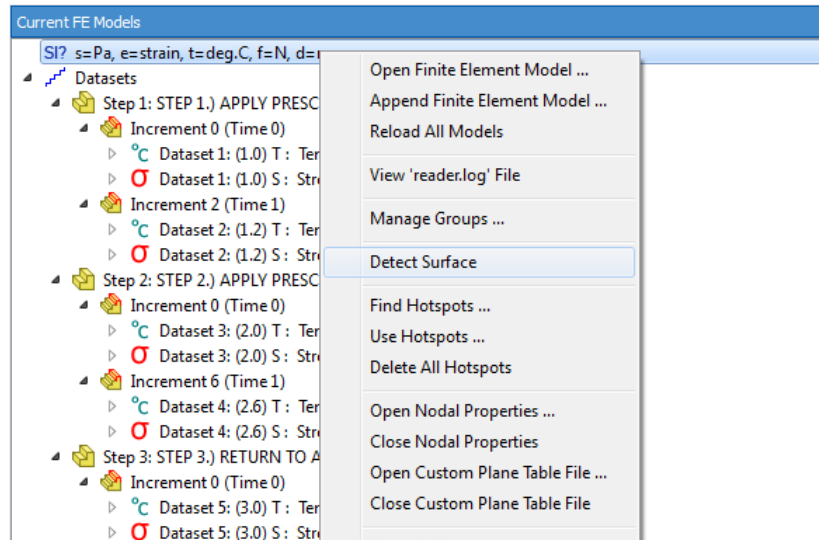


Figure 5-9

When surface detection is successfully completed new element and nodal groups will be created, named ELEM SURFACE and NODAL SURFACE respectively, and a new entry named Surface will be added to the Assembly section in the **Current FE Models** window, see *Figure 5-10* below.

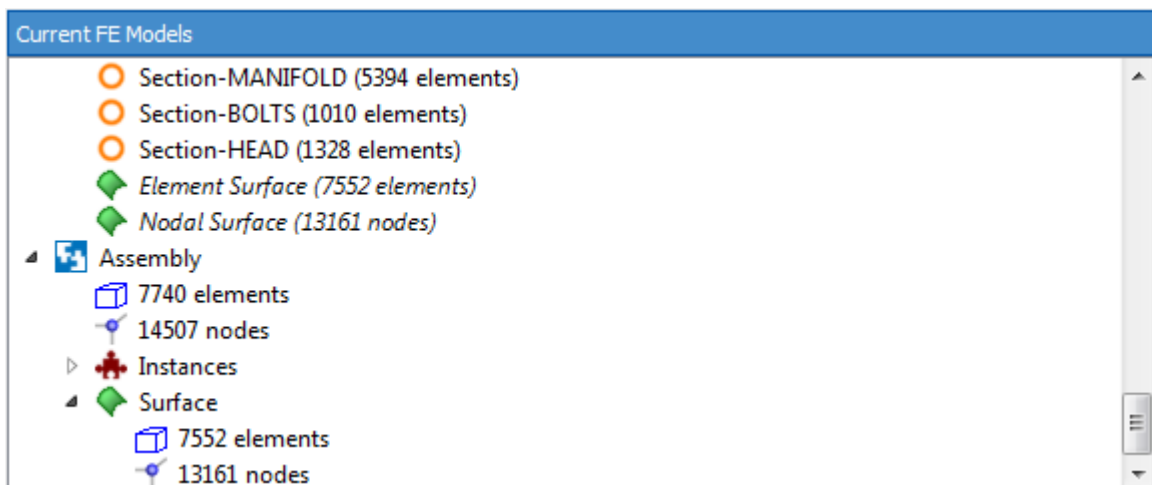



Figure 5-10

The subgroup option (i.e. analysis of the surface elements or the whole group) for an element group is defined by double-clicking on **Subgroup** in the **Group Parameters** region of the **Fatigue from FEA** dialogue. A dialogue box will appear where one of the two options must be selected.

5.6.3 Defining the material

The material for an element or node group is defined by highlighting the required material in the **Material Databases** window, then double-clicking on **Material** in the **Group Parameters** region of the **Fatigue from FEA** dialogue. A dialogue box will appear asking for confirmation that the material should be changed to the highlighted material. More information on materials and material properties required for the fatigue analysis can be found in section 8.

5.6.4 Defining the analysis algorithm

fe-safe will select a recommended fatigue algorithm for each element or node group, based on the properties of the material defined for that group. This method can be modified by double-clicking on **Algorithm** in the **Group Parameters** region of the **Fatigue from FEA** dialogue, then selecting **Select an algorithm to be used** in the **Group Algorithm Selection** dialogue. Clicking the  button displays a drop-down menu of available fatigue algorithms.

The algorithms available in *fe-safe* are discussed in more detail in the following sections:

- Section 14: *Fatigue analysis of elastic FEA results*
- Section 15: *Fatigue analysis of elastic-plastic FEA results*
- Section 16: *Fatigue analysis of welded steel joints*
- Section 17: *Design life and probability-based fatigue methods*

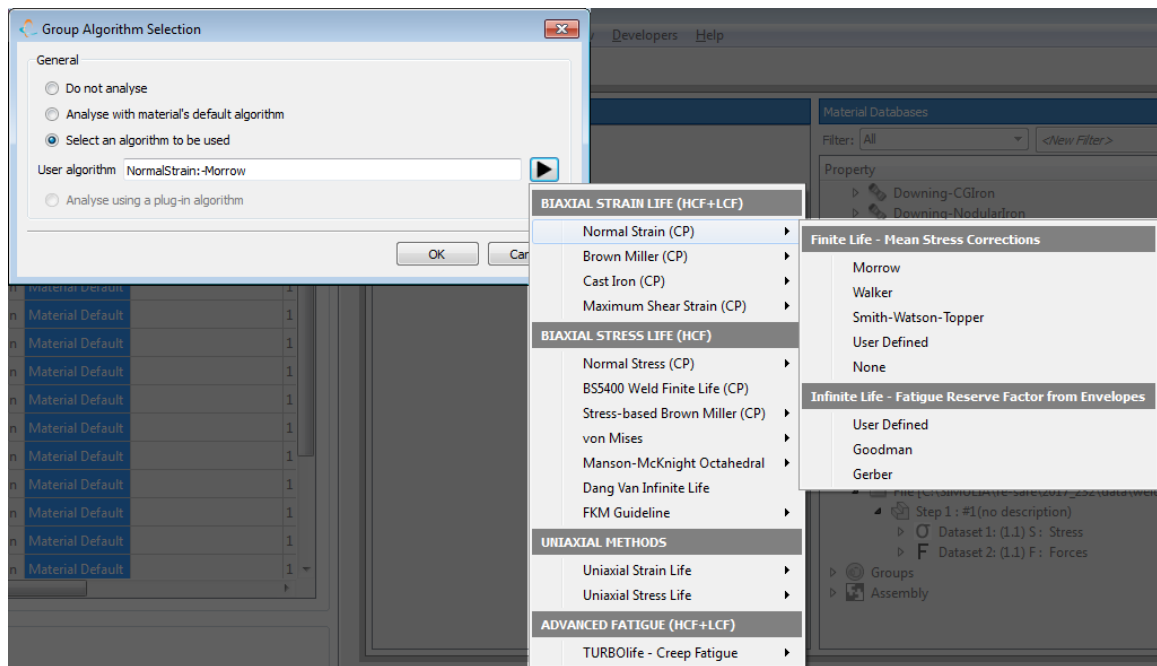


Figure 5-11: Defining the analysis algorithm

5.6.5 Defining the surface finish factor

When a model is loaded, the surface finish factor K_t for each element or node group defaults to 1.0. This can be modified by clicking on **Surface** in the **Group Parameters** region of the **Fatigue from FEA** dialogue. This will open the **Surface Finish Definition** dialogue, where either the surface finish can be selected from a drop-down list of pre-defined Ra ranges, the surface finish can be defined as a Rz range or a value for K_t can be entered, as shown in *Figure 5-12*.

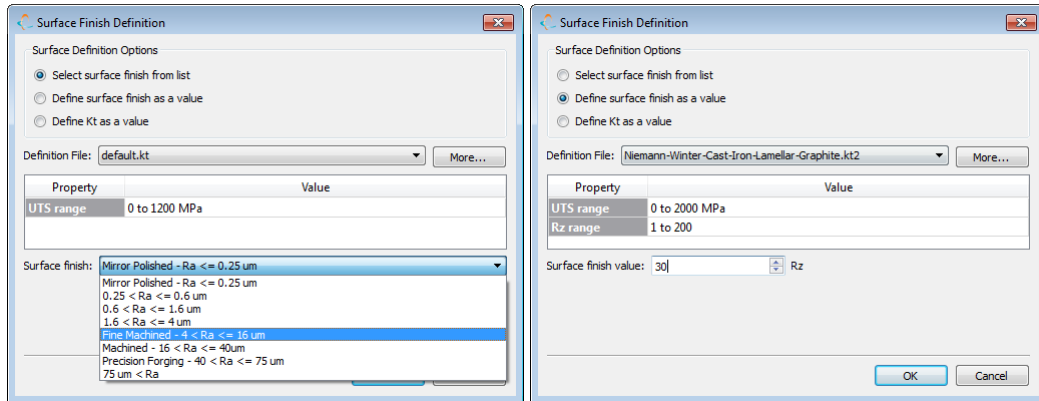


Figure 5-12

A default surface finish definition file (`default.kt1`) is included in the installation. Several additional surface finish definition files are also available:

- `juvinall-1967.kt2`
- `rcjohnson-1973.kt3`
- `Niemann-Winter-Cast-Iron-Lamellar-Graphite.kt4`
- `Niemann-Winter-Cast-Iron-Nodular-Graphite.kt4`
- `Niemann-Winter-Cast-Steel.kt4`
- `Niemann-Winter-Malleable-Cast-Iron.kt4`
- `Niemann-Winter-Rolled-Steel.kt4`
- `FKM-Guideline.kt5`

These files are stored in the `\kt` subdirectory of the *fe-safe* installation directory, and their format is described in Appendix E.

¹ UNI 7670, Meccanismi per apparecchi di sollevamento, Ente Nazionale Italiano Di Unificazione, Milano, Italy.

² Data extracted from "Fundamentals of Metal Fatigue Analysis", Bannantine, Comer and Handrock – page 13.

³ Data extracted from "Fundamentals of Metal Fatigue Analysis", Bannantine, Comer and Handrock – page 14.

⁴ Data extracted from "Maschinenelemente Band 1", Niemann, Winter & Höhn – chapter 3.

⁵ Data based on calculations in FKM Guideline 6th Edition, 2012 – Section 4.3.1.4

The surface finishes defined in the default definition file are shown in *Figure 5-13* below. When a surface finish type is selected from a list (see *Figure 5-12*, left) the material's UTS is used to derive the value of K_t from the selected curve.

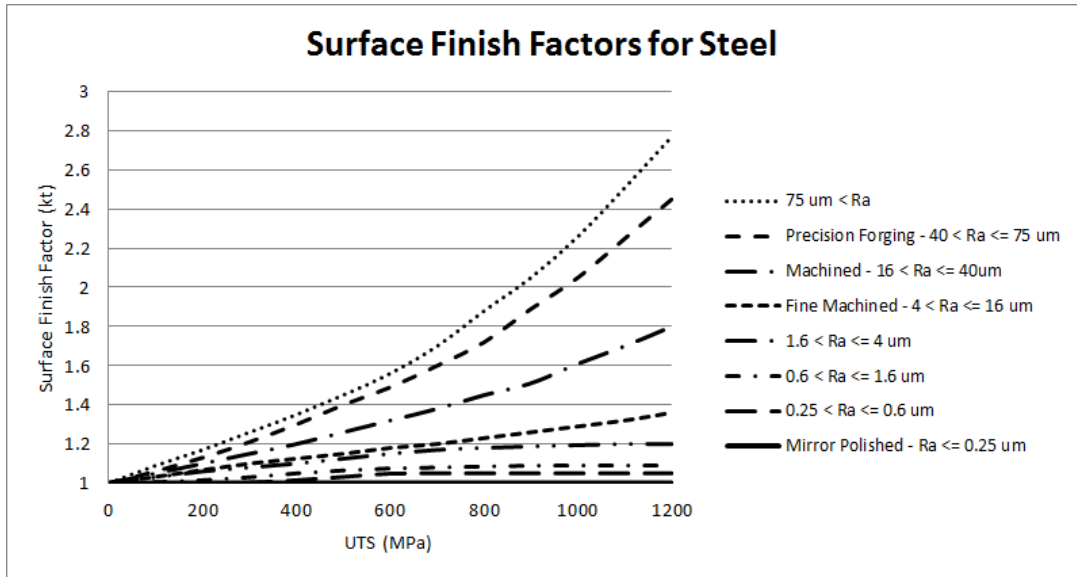


Figure 5-13: Stress concentration factor based on Ra, K_t versus UTS

Sample surface finishes defined in the Rz range definition file are shown in *Figure 5-14* below. A surface finish definition file is firstly selected from a list, and then the specific surface finish value is entered in the Rz range field (see *Figure 5-12*, right). A new surface definition curve is generated by interpolating the existing data for the defined Rz value and the material's UTS is used to derive the value of K_t from the generated curve.

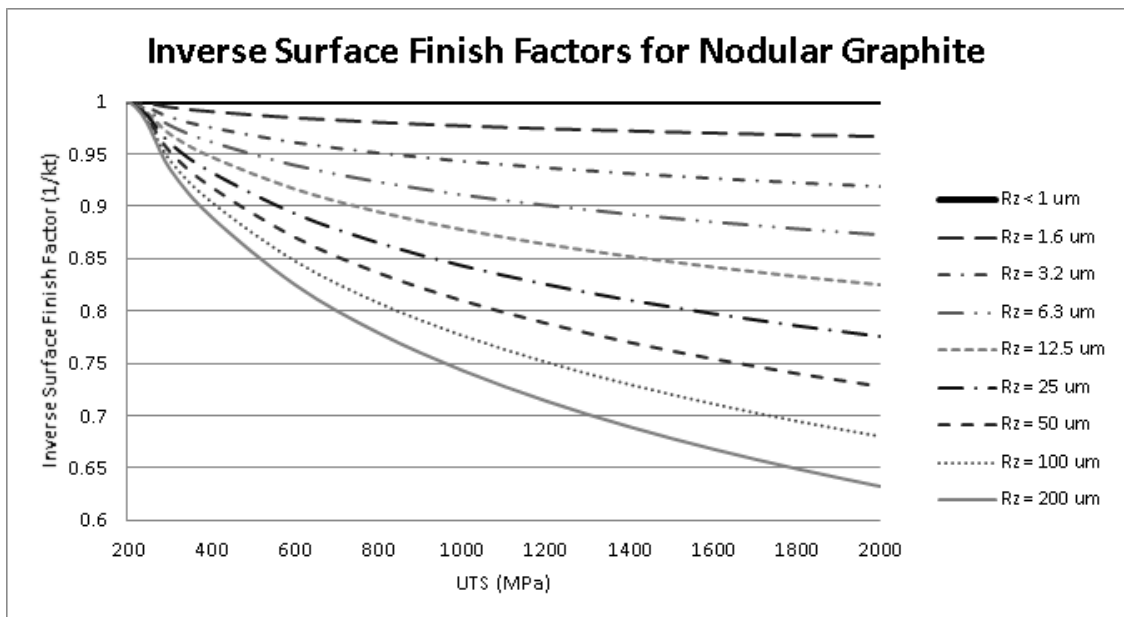


Figure 5-14: Inverse stress concentration factor ($K_{F, \dots} = 1/K_t$) based on Rz

Surface finish factors are applied using a multiaxial Neuber's rule: the elastic stress is multiplied by the surface finish K_t and this stress is used with the biaxial Neuber's rule to calculate elastic-plastic stress-strain. This means that surface finish effects are more significant at high endurance where the stresses are essentially elastic.

Since the surface finish is a stress-dependent property, the surface finish factor can be used to incorporate other stress-dependent phenomena, e.g. a size factor. To incorporate multiple stress-dependent properties, simply multiply the scale factors for each property, and enter it as a user-defined surface finish factor.

5.6.6 Including in-plane residual stresses

In-plane residual stresses can be defined for an element or node group by clicking on **In plane residual stress** in the **Group Parameters** region of the **Fatigue from FEA** dialogue. This opens the **In Plane Residual Stress** dialogue.

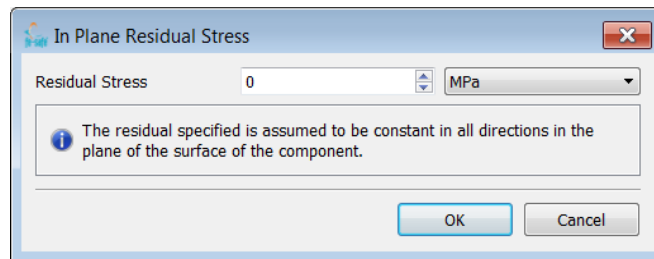


Figure 5-15

The residual stress can be defined in units of MPa or ksi, and is assumed to be constant in all directions in the plane of the surface of the component.

No elastic plastic correction is applied to this stress value. The value is applied by adding it to the mean stress of each cycle when calculating the life. For Factor of Strength (FOS) analyses (see section 17) the residual stress is not scaled by the FOS scale factor.

Residual stresses can also be included as an initial stress condition in a fatigue loading.

5.6.7 Additional SN data scale

Additional SN data scale factor can be defined for an element or node group by double-clicking on **SN Scale** in the **Group Parameters** region of the **Fatigue from FEA** dialogue.

The scale factor will be uniformly applied to scale all stress data points in the defined material SN curve, as well as the s_f parameter of the strain-life curve if stress-based analyses are performed using the elastic-plastic stress life curve derived from the local strain parameters. This option applies to stress-based analyses only and therefore will only be enabled if a stress-based algorithm is selected. Stress type analyses include the modules Uniaxial Stress Life, Normal Stress, Von Mises and Verity structural stress method for welded joints.

5.6.8 SN data knock-down curve

A series of additional SN data scale factors in form of a knock-down curve can be applied for an element or node group by double-clicking on **Knock-Down** in the **Group Parameters** region of the **Fatigue from FEA** dialogue.

If an additional effects curve is defined for the selected material (see the Material Properties section) and the Knock-Down parameter is enabled, scale factors will be extracted from the curve and applied to scale all stress data points in the defined material SN curve, interpolating and extrapolating the available data points as necessary. Format of the knock-down curve is described in Appendix E, for more details on the application of the additional effects curve see Section 14.

This option applies to stress-based analyses only and therefore will only be enabled if a stress-based algorithm is selected. Stress type analyses include the modules Uniaxial Stress Life, Normal Stress, Von Mises and Verity structural stress method for welded joints.

5.6.9 Group properties for nodes and elements in multiple groups

Groups and their properties are displayed in the **Group Parameters** area of the **Fatigue from FEA** dialogue as shown above in *Figure 5-8*.

Where an item (element or node) is present in more than one group, its properties are taken from the first group in the list of which it is a member. For example if node 888 appears in Bolt and Manifold in *Figure 5-8* above then its properties will be those of Bolt. The groups order can be edited in the **Select Groups to Analyse** dialogue, using **Promote** and **Demote** buttons to order the list of **Analysis Groups**, see section 0.

Note that any elements in groups with **Algorithm** set to **Do not analyse** will be subtracted from the analysis (marked not to analyse). If any of those elements are used in a group further down the list they still will not be analysed.

If an item is not present in any of the loaded groups then its properties are set to the defaults, which are listed at the bottom of the table as a **Default** group. A ** mark next to the name of a group indicates a group with parameters different to the defaults.

For elemental data types *fe-safe* does not import node number information but considers all nodes on an element to belong to that element. Therefore, all nodes on an element inherit the same properties as the element. If the same node appears on another element *fe-safe* will analyse it separately using the properties that apply to that element.

Note that when elemental results are exported and displayed in an FE viewer, some nodes could have multiple values. The way these values are displayed is handled by the viewer - usually either the average value or the lowest value is plotted. For details on how the data is displayed in the viewer, refer to the documentation supplied with the viewer. Additional information on post-processing can be found in Appendix H.

The order and usage of groups can significantly affect the outcome of an analysis, as the following examples will illustrate.

Assume that:

- five groups (grp_1, grp_2, grp_3, grp_4, and grp_5) are imported from a model in that order;
- groups grp_1, grp_2 and grp_5 have the same properties as the Default group;
- grp_3 has different properties to the Default group;
- grp_4 has different properties to both the Default group and may or may not have the same properties as grp_3.

The table in the group parameters area of the 'Fatigue from FEA' dialogue will appear as:

grp_1	
grp_2	
grp_3 **	
grp_4 **	
grp_5	
Default	

Figure 5-16

where the ** indicates that the group has parameters that are different to the Default group.

Example 1:

If node 888 belongs to groups grp_3 and grp_4, then node 888 will take the properties of grp_3, containing that node

Example 2:

If node 999 belongs to groups grp_1, grp_2, and grp_4 then node 999 will take the properties of grp_1, (which are the same as the Default group).

Example 3:

If all groups, with the exception of grp_1, are set to 'Do not analyse', then node 888 will not be analysed as grp_3 was set not to be analysed and node 999 will take the properties of grp_1

Example 4:

If all groups, including the Default group, with the exception of grp_4 are set to 'Do not analyse', then neither node will be analysed as grp_1 and grp_3 were both set not to be analysed and they are higher on the list than grp_4.

Promoting grp_4 to the top of the list will ensure all the nodes in that group take the properties of grp_4. This can be accomplished using the Manage Groups dialogue described in section 0. Once grp_4 is promoted, the table will appear as:

grp_4 **	
grp_1	
grp_2	
grp_3 **	
grp_5	
Default	

Figure 5-17

Example 5:

If all groups including the Default group, with the exception of grp_4 are set to 'Do not analyse', then both nodes will take the properties of grp_4 as it is higher on the list than grp_1, grp_2, or grp_3.

Determining which group the properties for a node or element came from can be done using a request to export nodal information described in section 22.

5.6.10 Defining the source filename

The source file is the file from which the FE model information will be extracted and used as a base for the fatigue results export. *fe-safe* automatically assumes the source filename to be the name of the first imported FE results file. This can be edited as required, e.g. allowing exporting results to a different file format.

5.6.11 Defining the output filename

The output file is the file to which the fatigue results will be exported. *fe-safe* automatically generates an output filename based on the name of the imported FE model file. The auto-generated name can be edited as required.

The type of file being exported is determined by the filename extension. The output file type is normally the same as the input file type. However, it is also possible to export fatigue results to a different format to enable the results to be viewed in a particular viewer. Not all combinations of input type and output type are compatible.

For all input types, results can be exported to an ASCII CSV output file.

For more information see Appendix G and Appendix H.

5.6.12 Saving analysis configuration settings

Analysis configuration settings can be saved for re-use by selecting the **Save FEA Fatigue Definition File...** option from the **Project** section of the **File** menu. The current settings are written to a user-specified *fe-safe Project Definition* file (extension `.stlx`). The default location for these files is in the project directory, see Appendix C.

The *fe-safe Project Definition* file saves references to locations of the files used in the analysis (e.g. source FE model file, the fatigue loading definition file, etc.) as follows:

- if the files used were placed outside the Project Directory, absolute paths are used, e.g.:
D:\Data\Files_Repository\FEA_Files\Project99\my_file.op2
- if the files used were placed inside the Project Directory, relative paths are used, e.g.:
jobs\job_01\fe-results\fesafe.fer

A loading definition file (extension `.ldf`) file will also be created at the same time as the project definition file, if a `current.ldf` file (for the current job) is used. This file will have the same root name as defined for the project definition file above, but with extension `.ldf`.

Configuration settings can be retrieved using the **Open FEA Fatigue Definition File...** option. A dialogue appears giving the user the option to reload the finite element model (or models) if required, for example:

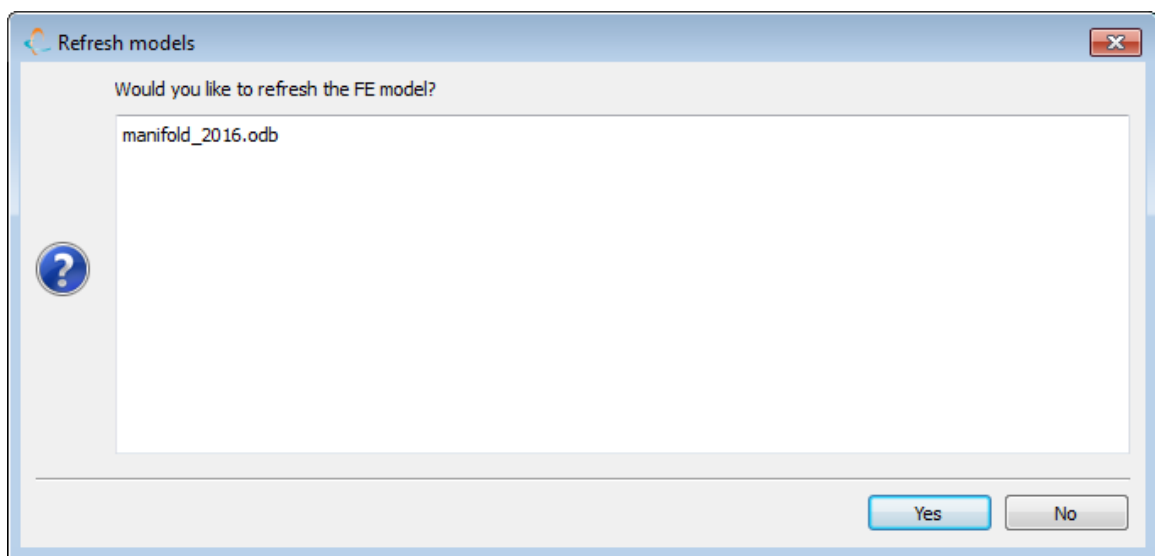


Figure 5-18

When the file is opened, the loaded settings will overwrite the current project and job settings. As the file is opened, any paths defined in the file are interpreted assuming the following path hierarchy:

- Absolute path (as defined in the `.stlx` file)
- Location of the `.stlx` file
- Current project path

Any paths defined in the referenced `.ldf` file will also be interpreted in a similar way and the loading definition will then be saved as the new `current.ldf` (for the current job).

Legacy Keyword format and Stripped Keyword (`*.kwd` and `*.xkwd`) files can also be used to open analysis configuration settings from analyses completed in an earlier version of *fe-safe*.

Configuration file can be used in command line or batch processes (see section 23).

The use of configuration files is discussed in detail in Appendix C and Appendix E.

5.7 Analysis options

General analysis options can be configured in the **FEA Fatigue >> Analysis Options** dialogue.

5.7.1 Import tab

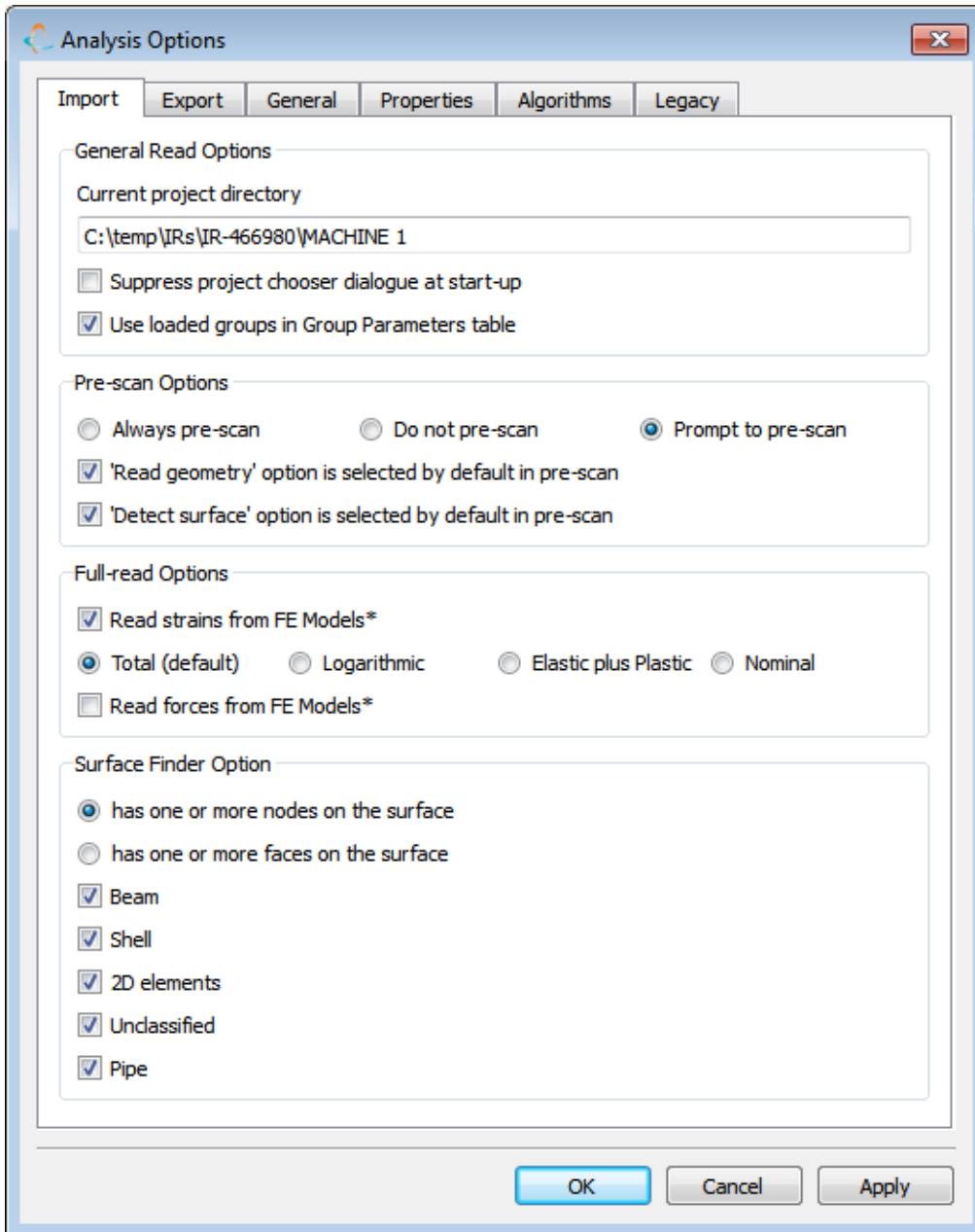


Figure 5-19 Configuring import options

Current project directory

Location of the **Project directory** (see Appendix E) is displayed here. This location can only be set when starting the application or by creating a new Project using menu option File >> Project >> New Project... (see section 5.2).

Suppress project chooser dialogue at start-up

A prompt to choose a project directory is shown at every start-up. This option can be used to suppress it from being shown again or to restore it if already suppressed.

Use loaded groups in Group Parameters table

When checked and a model is opened, the groups loaded from that model will automatically be added to the list of groups in the **Group Parameters** table in the **Fatigue from FEA** window. Settings that differ from the default analysis, material etc can be set for each of the groups added. Groups can be added, removed and reordered via a dialogue accessed from the 'Manage Groups..' button in the **Fatigue from FEA** window.

Pre-scan options

- **Always pre-scan:** files will be pre-scanned automatically without prompting the user.
- **Do not pre-scan:** files will not be pre-scanned and the whole file will be loaded each time.
- **Prompt to pre-scan:** user will be asked each time if the file(s) should be pre-scanned (only if the pre-scan file is invalid or not present).

Default pre-scan window options

These options control the default settings for the **Read geometry** and **Detect surface** checkboxes in the **Select Datasets to Read** dialogue, see *Figure 5-3*. By default both these options are selected.

Read strains from FE Models

Checking this option reads strains as well as stresses from the FE model. These are used only when performing strain-based analysis. Note that this option takes effect the next time a model is loaded and is not applicable to pre-scanned models.

Read forces from FE Models

Checking this option reads forces as well as stresses from the FE model. These are used only when performing a Verity analysis to get structural stresses. Note that this option takes effect the next time a model is loaded and is not applicable to pre-scanned models.

Surface finder options

These options are used to configure the surface-finder algorithm: surface elements can be defined as having either at least one surface node (**has one or more nodes on the surface** option) or at least one surface face (**has one or more faces on the surface** option).

Additionally, the following elements with non-solid geometry can be treated as surface elements: planar elements (**2D elements**), elements with reduced geometry (**Beam, Pipe, Shell**) and any other elements not classified by *fe-safe* (**Unclassified**). Note that when elements with non-solid geometry are used in conjunction with the **has one or more nodes on the surface** option, all solid elements sharing their nodes will be set as surface elements as well.

5.7.2 Export tab

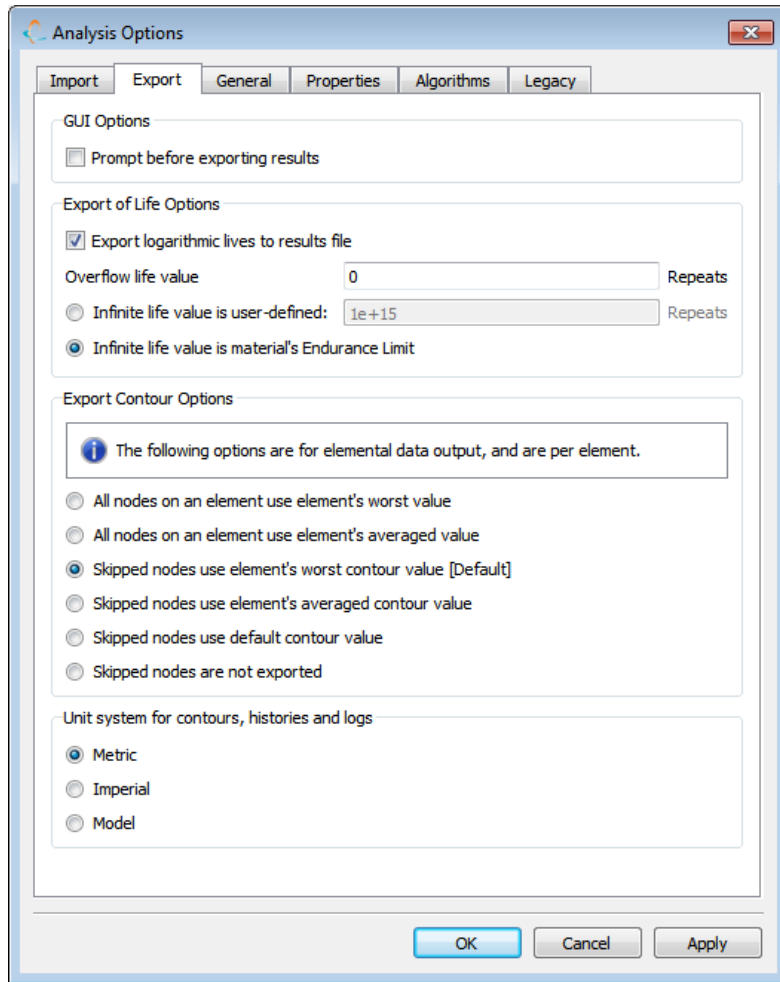


Figure 5-20 Configuring export options

Prompt before exporting results

When checked a prompt will be displayed after every analysis but before the results are exported, giving a choice not to export.

Export logarithmic lives to results file

This option will normally be set to export the base-10 log of lives, for viewers that do not have a logarithmic plotting capability for contour plots. With this option set, a life of (e.g.) 10^6 miles will be written as 6.0.

Note: viewing fatigue hotspots in logarithmic scale on contour plots enables pinpointing locations of lowest lives most easily as opposed to in a linear scale. Disabling this option should be made carefully, as contours will be less clear in linear scale, and viewers should be used to contour in log scale.

Overflow life value

At each node, *fe-safe* checks the integrity of the stresses before performing the fatigue life calculation. If the stresses are rejected (perhaps because the stresses are so high that the endurance would be less than one fatigue cycle) *fe-safe* displays a warning message. To avoid gaps in the output file, a fatigue life will still be written. This value is termed the 'overflow' value, and can be user-defined. If a negative value is specified then this will be used for both the fatigue life and the log of the life.

Infinite life value is user-defined

Where no damage occurs at a node, the supplied infinite-life value is written to the Life result for contouring. In versions prior to 5.00 this value was 1e15.

Infinite life value is material's Endurance Limit

Where no damage occurs at a node, the value written to the Life result is the constant-amplitude endurance limit of its material (as defined by database ID Const_Amp_Endurance_Limit), where defined; and 1e15 otherwise. Note that the material property is defined in terms of reversals (2Nf), whereas life is reported in repeats (Nf), so the value reported will be half that in the material definition.

Export contour options

Skipped nodes referenced in the radio button descriptions in this region refer to nodes on an element that are not analysed, for instance while using the Verity in *fe-safe* module. These export settings could affect the way that viewers display fatigue results for elements with skipped nodes.

- **All nodes on an element use element's worst value:** all nodes on an element will be assigned the worst contour value of that element.
- **All nodes on an element use element's averaged value:** all nodes on an element will be assigned the average contour value of that element.
- **Skipped nodes use element's worst contour value:** nodes not analysed (e.g. when analysing weld lines) will be assigned the worst contour value of the remaining nodes on element.
- **Skipped nodes use element's averaged contour value:** nodes not analysed (e.g. when analysing weld lines) will be assigned the average contour value of the remaining nodes on element.
- **Skipped nodes use default contour value:** nodes not analysed (e.g. when analysing weld lines) will be assigned the default contour value:

Contour	Default value
Life / Log of Life	Infinite life
FOS	Max FOS
FRF	10
Dang Van - radial	10
Dang Van - vertical	10
Dang Van - survived	1
Prismatic Hull FRF (PH-FRF)	10
Probabilities	-100
SRP PP	0
SRP PC	1
SRP CP	2
SRP CC	3
All other contours	0

- **Skipped nodes are not exported:** nodes not analysed (e.g. when analysing weld lines) will not be assigned any value.

5.7.3 General tab

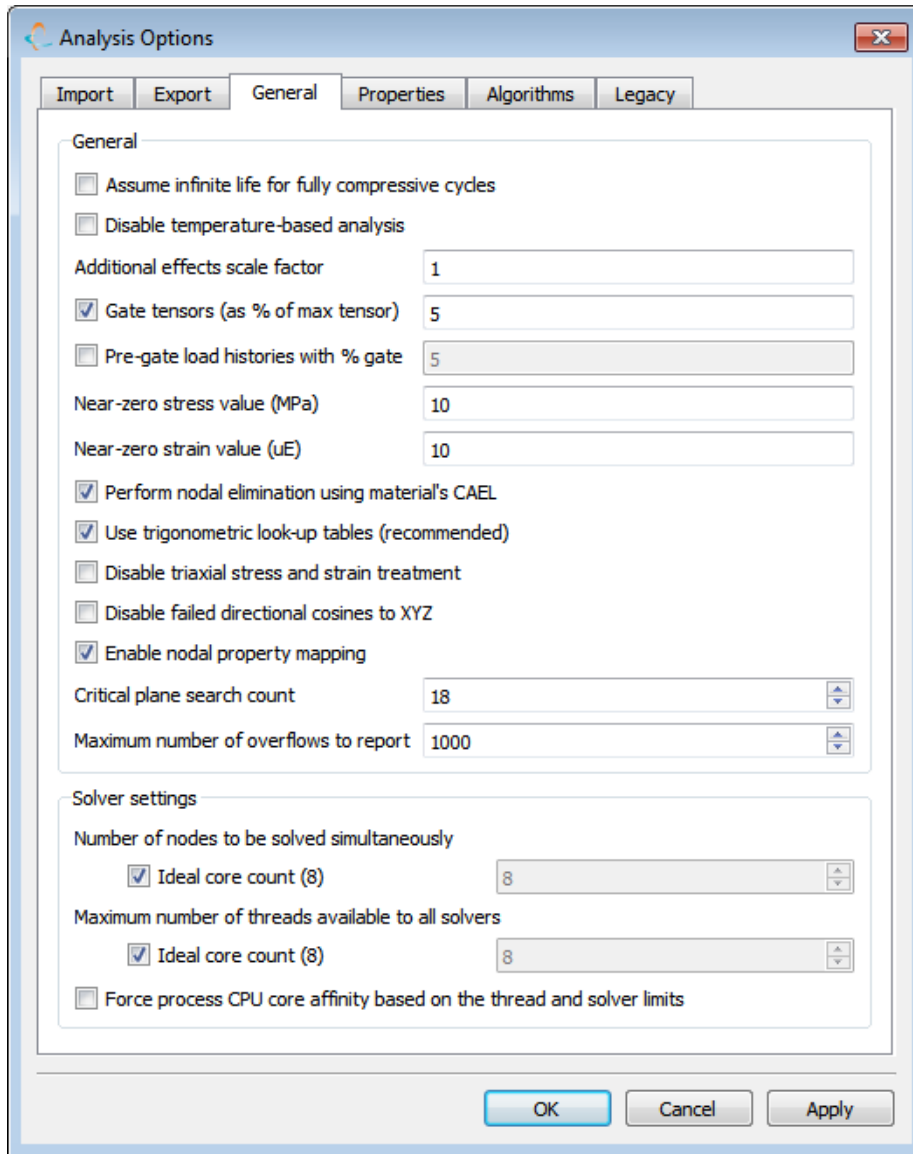


Figure 5-21 Configuring general options

Assume infinite life for fully compressive cycles

When checked, cycles with entirely compressive stress ranges will cause no damage, regardless of their magnitude. For infinite life FRF calculations the envelope is modified to close when the amplitude magnitude is less than the mean stress for negative mean stresses.

Note: The criterion is always based on the stress range of a cycle, even for strain-based algorithms.

Disable temperature-based analysis

Checking this item will disable temperature-dependent fatigue analysis in *fe-safe*.

For conventional fatigue analysis, including high temperature fatigue (see section 18), checking this option causes the temperatures from the loaded temperature dataset to be ignored. Instead, material data corresponding to a temperature of 0°C is used (subject to the interpolation/extrapolation conditions described in section 8.6.3), regardless of the temperature for the node in the temperature dataset.

Additional effects scale factor

This scale factor is applied to the imported stresses to allow additional phenomena to be incorporated into the analysis, for example:

- corrosion effects
- confidence levels

Gating and analysis speed control

The **gating** parameters allow for a reduction in the amount of time required to run an analysis.

Gate tensors (as % of max tensor)

Omits samples from the calculated stress or strain histories so as to eliminate cycles whose amplitude is less than the given percentage of the largest cycle. This is performed separately on each analysis plane of each loading block. Gating the tensors is generally a safe speed-up to perform because it ensures that the peaks and valleys on each analysis plane are retained.

Pre-gate load histories with % gate

This is not the case if you enable pre-gating of the load histories. This will generally find the larger cycles but may produce non-conservative fatigue lives by missing damaging cycles. This can be used as a quick analysis and followed by a more detailed analysis of just the critical sections of the component.

Perform nodal elimination using material's CAEL

This is also a safe speed-up to perform. For scale-and-combine loading blocks, the worst-case values in the load histories and the FE datasets are used to estimate the worst possible stress and strain ranges in the final loading. Any nodes whose worst possible strain or stress range is beneath the constant amplitude endurance limit of the material are ignored, as no damage can occur. An 80% safety factor is used.

Use trigonometric look-up tables

When the look-up tables are used the trigonometric function results related to the plane being analysed are acquired from a table of pre-generated values at 0.05 degree intervals.

Disable triaxial stress and strain treatment

When triaxial stresses are detected at a node the critical-plane analysis is performed in each of the triaxial planes. This can be disabled using this check box. In this case the 'surface' orientation is identified from the stress tensors (even for strain-based analyses). See Technical Note 3 (TN-003) for an in-depth discussion of triaxial stress treatment.

Disable failed directional cosines to XYZ

The default method used in *fe-safe* to evaluate the directional cosines for a node follows the sequence below:

1. calculate the directional cosines for the largest stress sample in the loading
- if 1 fails, then:
2. work through the remaining points in the stress history until a cycle is found for which the directional cosines are solvable
- if 2 fails, then:
3. use directional cosines for the global axis, XYZ.

In most cases this default behaviour will evaluate accurately the directional cosines in either step 1 or step 2. The user has the option to disable step 3 by selecting **Disable failed directional cosines to XYZ**.

If this option is selected and the directional cosines cannot be evaluated from the stress history (steps 1 and 2), then the fatigue evaluation for this node is aborted and a "non-fatigue failure" error is recorded in the log file.

Enable nodal property mapping

Checking this item will enable a feature to map material properties by node depending on inputs from manufacturing analyses such as casting or forging.

Critical plane search count

When performing a critical plane search the number of planes to search can be changed with this option. The search angle interval will be 180 / plane count. This defaults to 18 for all algorithms which use critical plane searching.

Note: In *safe4fatigue* the plane count is set to 16 and cannot be modified.

Maximum number of overflows to report

Overflow indicates that the stress/strain range of a load cycle is too high even for a single repeat of that cycle, i.e. that a static, rather than fatigue, failure is expected at the given location. Overflow messages, along with their cycle information, are stored and then reported at the end of the analysis. This option limits the number of overflow messages kept to prevent any adverse effects on the available computational resources.

Note: It is recommended to exclude from the analysis areas where overflows are expected, e.g. discontinuities, singularities or constraints.

Solver settings

By default, *fe-safe* will employ all the available processors or processor cores on the system that is running the analysis to produce the fastest result. If this is causing undesirable slow-downs in other applications, the solver controls can be used to reduce the number of cores used.

Two options are available: to control either the number of nodes analysed simultaneously (memory intensive) and the number of simultaneous analysis threads (since in critical-plane analysis each plane can be analysed in parallel as a separate thread).

Note: Each node solver will use at least one thread.

The CPU core affinity can also be set when the number of threads to be used is less than the optimal core count. Only a specific set of CPU cores will be used (determined by the operating system) to run all analysis and non-analysis operations. For some operating systems, reducing a process's pool of cores is permanent and cannot be increased, so it is recommended that *fe-safe* is restarted after increasing core usage or removing the limit entirely.

5.7.4 Properties tab

Temperature

Two options are available for interpolation of the temperature-dependent material properties - linearly (default) or between temperatures on a logarithmic Kelvin scale, (zero Kelvin = -273.15°C).

An optional warning can be generated in the case that material data are extrapolated past the temperature limits defined in the material properties. See Section 8.6 for additional details on extrapolation of material data.

SN data

Checking this item will generate a warning in the case that material data are extrapolated past the SN data limits defined in the material properties. See Section 8.6 for additional details on extrapolation of material data.

Stress ratio (R-ratio)

Checking this item will generate a warning in the case that the calculated stress ratio of a loading cycle is beyond the defined SN stress-ratio limits defined in the material properties.

5.7.5 Algorithms tab

Cast Iron section

This tab defines the cast iron damage rate equation parameters for the Cast Iron algorithm. For more information see section 14.

Modal Analysis section

This tab defines parameters for analysis from steady-state modal FEA, for more information see section 25.

PSD section

This tab defines parameters for analysis using PSD data, for more information see section 27.

Plugin section

This tab defines options used with custom-built plugin algorithms, for more information see the custom framework documentation.

Default to using EN data for strain analysis

Checking this item will ensure the strain-based analyses are performed using E-N curve data rather than the elastic-plastic strain life curve derived from the local strain parameters

Default to using SE data for cyclic curve

Checking this item will ensure the plasticity correction is performed using cyclic stress-strain curve tabular data rather than the cyclic curve derived from the cyclic stress strain curve parameters K' and n' .

Enable custom nodal plane mapping

Checking this item will enable a custom feature for critical plane searching at each node.

Safety Factors section

Enhanced Safety-Factor Options

For Safety Factor analyses such as Factor of Strength (FOS) and Fatigue Reserve Factor from Envelopes (FRF), enhanced scaling options are available in order to apply a correction factor based on critical distances either **using critical-distance point method** or **using critical-distance line method** if appropriate. A limit to apply corrections **only where uncorrected factors are between** specified values is possible. An option to **export raw correction factors as a contour** in addition to the safety factor contour is available.

Band Definitions for FOS Calculations

This region defines the bands between which the Factor of Strength calculations are performed. For more information see section 17.2.

Stress Analysis section

The following configuration options apply to Stress type analyses.

Use stress-life curve defined using SN datapoints

Selecting this radio button will ensure the stress-based analyses are performed using S-N curve data rather than the elastic-plastic stress life curve derived from the local strain parameters. Stress type analyses include the modules Uniaxial Stress-Life, and Biaxial Stress-Life such as Normal Stress and Von Mises.

Use stress-life curve defined using sf' and b

Selecting this radio button will ensure the stress-based analyses are performed using the elastic-plastic stress life curve derived from the local strain parameters rather than S-N curve data. Stress type analyses include the modules Uniaxial Stress Life, and Biaxial Stress-Life such as Normal Stress and Von Mises.

Use sf' and b if no SN datapoints

Checking this item will ensure that if there is no S-N curve data for the current material then the elastic-plastic stress life curve derived from the local strain parameters will be used.

Note: Exercise care when deselecting this option in conjunction with the **Use stress-life curve defined using SN datapoints** option above. If a component contains more than one material type, some parts of the model may use a material which has S-N data available, whilst other parts may use a material with no S-N data available, but analysis will still be aborted with the Error: *The defined SN curve is not valid. This must be defined correctly for the selected analysis type.*

No plasticity correction (for HCF only)

This radio button will ensure there is not a plasticity correction applied to the elastic stress tensors read from FE models for stress-type analysis.

Note: Exercise care when configuring this option (**No plasticity correction**) as it should only be used for High Cycle Fatigue problems.

Apply Neuber plasticity correction (for HCF and LCF, requires K' and n')

This radio button will ensure there is a Neuber-type plasticity correction applied to the elastic stress- tensors read from FE models for stress-type analysis.

Note: This option should not be selected if stress results from elastic-plastic FEA are used.

Von Mises section

In evaluating the von Mises stress for the Von Mises fatigue life algorithm and for exporting the Von Mises stress the sign can be assigned in one of two ways: either using the sign of the hydrostatic stress or the sign of the largest principal stress. This is configured from this tab.

5.7.6 Legacy tab

Enable non-standard fatigue modules

This option will enable additional analysis modules.

5.8 Populating Data Tables Using Interpolating and Extrapolating

When inputting data in a numerical table such as the material parameters or the user defined load history, partially filled tables can have the remaining data interpolated or extrapolated using either the **Populate** button or the column or row context menu (right mouse button to access).

5.8.1 Minimal data requirements

To use interpolation and extrapolation the row or column a minimum of 2 data points is required within the same data set e.g. with a material's SN data for multiple temperatures, interpolation/extrapolation will occur separately for S and for N so on a row with N and one S value specified the row will not be populated.

When selecting a logarithmic interpolation option, columns or rows with zero or less will be skipped.

5.8.2 Interpolation and Extrapolation Behaviour

Interpolation is done first on missing values between know points, then the any missing start of end values are extrapolated to, the derived values will be in green and italicised e.g.

Row	Col 1	Col 2
1		
2		
3	1	
4		
5		
6		
7		
8	2	
9		
10		

Row	Col 1	Col 2
1		<i>0.6</i>
2		<i>0.8</i>
3	1	
4		<i>1.2</i>
5		<i>1.4</i>
6		<i>1.6</i>
7		<i>1.8</i>
8	2	
9		
10		

Figure 5-22

With linear population of missing values the delta between values is uniform, whereas with logarithmic population of values the values are first converted to logarithmic scale, populated linearly then converted back to the original scale.

Note that only missing values are populated, if the values of a derived value are changed and the table is populated a second time the derived value will not update – if this behaviour is required any derived values that need regenerating must be deleted first.

5.8.3 2D Table Behaviour

For 2D tables (which are usually used for material parameter data) the **Populate** menu allows for rows or columns to be populated first (major axis) before populating the other (minor) axis. Where the major axis could not be populated the minor axis will be tried, if this succeeds the major axis will be re-checked e.g. with linear populating rows first:

	Nf	S (200 degC)	S (300 degC)	S (400 degC)
1	10000	377	201	138
2	1e+07	196	107	72
3				
4		170		50
5				
6				

	Nf	S (200 degC)	S (300 degC)	S (400 degC)
1	10000	377	201	138
2	1e+07	196	107	72
3	1.999e+07	183	108.5	34
4	2.998e+07	170	110	50
5				
6				

Figure 5-23

Nf is a separate data set, as it is a single column the last two values are extrapolated from the first two. In this case it would be more appropriate to use the logarithmic population of just the Nf column before the linear population of the S values.

For material parameters with constraints on the data, invalid values will be highlighted red.

5.9 Data File Options

The Data Files Option can be configured in the **Tools >> Data File Options** dialogue, as shown in *Figure 5-24*:

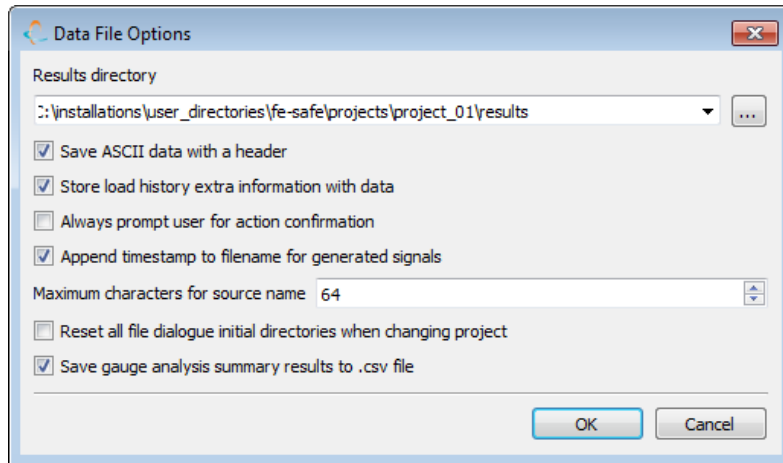


Figure 5-24

Results directory

This is the default directory for storing the results of signal processing and fatigue analysis from measured signals, for more details see sections 9, 10, 11 and 12. By default this directory is in the project directory <ProjectDir>/results, see section 3.

5.10 Controlling default setting values

Default values of settings can now be recorded permanently, so that they remain set even after clearing all settings, see section 5.2.3. The settings in *fe-safe* are divided into two groups:

- **Project settings** – stores settings related to a particular project, see section 5.6.12. They are recorded in a series of files under the project directory and are applied wherever that project is opened, so that the project can be transferred to a different workstation with no extra setup needed.
- **User settings** – stores user preferences not specific to a single project. They are recorded in the “user.stli” file in the user directory, and are not typically transferred between workstations.

When clearing settings, the two categories of settings are reset to factory defaults. However the factory defaults can be overridden and will apply whenever a new project is started or an existing project is cleared.

To choose which default settings to record, open the **Tools >> Project Default Settings...** dialogue (or **Tools >> User Default Settings...** dialogue) as shown in *Figure 5-25*. The tree on the left displays all settings that are currently different to the factory default. Settings names shown in the list match the options descriptions used in the GUI relating to that setting; clicking on a particular setting displays details of the setting in a panel on the right.

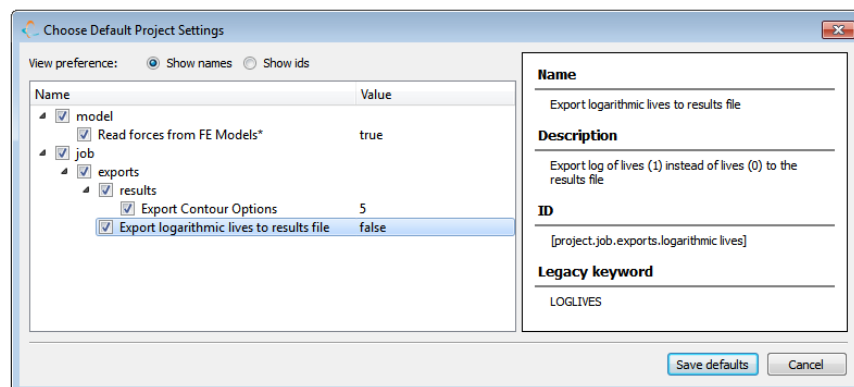


Figure 5-25

The check-boxes next to each setting determine which settings are saved. Only those that are selected will be recorded in the defaults.

Note: It is currently not possible to control the defaults of group-related settings (e.g. algorithm or material).

When the **Save defaults** button is clicked, the selected defaults will be recorded to one of the two following files (depending on the type):

- <UserDir>/project.stld
- <UserDir>/user.stld

5.11 Establishing defaults across an organisation

The default settings files can be applied to other installations of *fe-safe* within an organisation to ensure consistent preferences and settings. It requires at least one user (or administrator) to install the software with factory default settings and record suitable defaults to the two files. These files can then be made available to other users *before* they install *fe-safe*; for example they could be stored on a network drive.

When the next user installs the software, the saved default configuration files can be selected to be applied, see the Installation and Licensing Guide for more details.

These files will then be copied into the installation directory. When *fe-safe* starts for the first time, the defaults file will be copied into the user directory and the defaults within the file will be applied to the software. Any subsequent changes the user makes to their defaults will only be applied locally and will not affect the original default files.

5.12 Additional capabilities

- Load histories can be scaled and manipulated in many ways - see sections 10.3 and 13.
- Several load histories, and material data from several materials, can be plot-overlaid or cross plotted- see section 7.
- The load history can have the peaks and valleys extracted before use, to speed up the analysis – see section 10.3.
- The material's data can be temperature-dependent - see section 8.5.10.
- Material data can be approximated if fatigue test data is not available - see section 8.4.7.
- Elastic-plastic stress-strain pairs can be read from the FE model to allow analysis of elastic-plastic FEA results – see section 15.
- Factors of strength (FOS) at each node can be calculated for a specified design life, to be displayed as contour plots - see section 17.
- Probability of failure at each node can be calculated for the design life or a series of lives, to be displayed as contour plots - see section 17.3
- Fatigue reserve factors (FRF) can be calculated, to be displayed as contour plots - see section 17.
- A load sensitivity analysis can be performed to show which load directions are most damaging and the potential failure locations – see section 22.
- A Haigh diagram showing the most damaging cycle at each node can be created and plotted – see section 14.13.
- Additional detailed results for selected elements can be exported and plotted in *fe-safe* – see section 22.
- The analysis set-up can be saved for use with different FEA models or for use in batch operations - see section 23.

5.13 Project Management

From the file menu projects can be:

- **New Project ...** : This will let the user specify a new project which will have the default settings (see Sections 5.10 and 5.11 regarding defaults)
- **Open Project ...** : This will open an existing project
- **Close Project** : The current project will be closed. This can be useful to release a licence for an add-on module while retaining the client licence.
- **Export Project ...** : This allows all used project files to be exported to another directory or an archive, see below
- **Import Project ...** : This allows an archived project to be imported into a new or existing project, see below
- **Open FEA Fatigue Definition File ...** : This can be used to open project settings (see Section 5.6.12)
- **Save FEA Fatigue Definition File ...** : This can be used to save project settings in a single file (see Section 5.6.12)

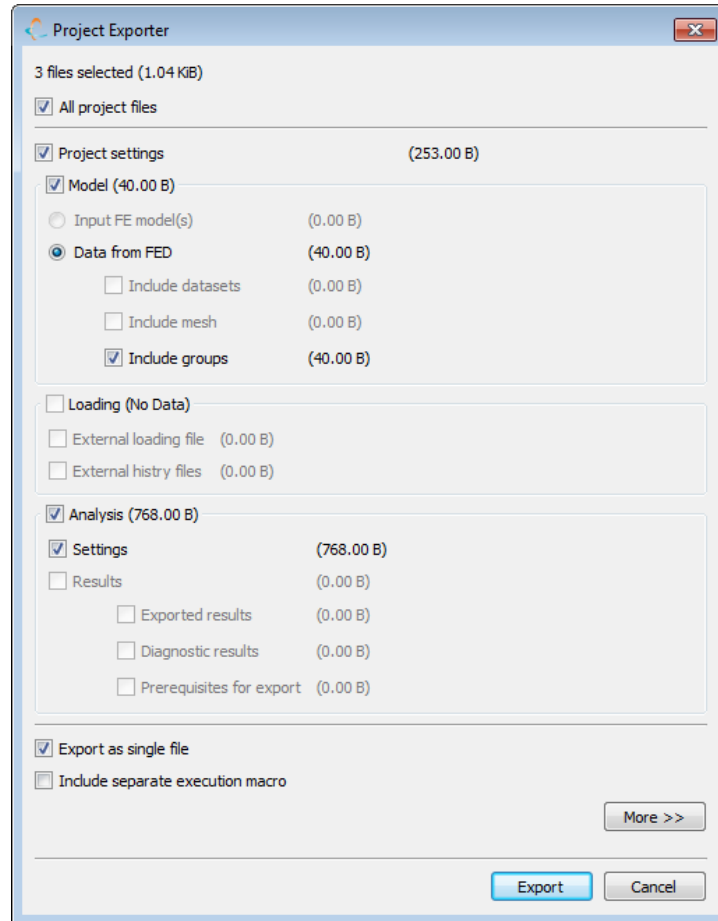
See Section 23.2 for project command line options, and Section 23.6 for project macro commands

5.13.1 Exporting project files

When exporting project files the user can specify all or subsets of project files to export. The target location to export to can either be a directory or a single archive file, determined by the option **Export as single file**.

The **Include separate execution macro** will create a macro that opens the project and tries to run a fatigue analysis, regenerating as much data as possible.

Files external to the project that are selected for export will be copied to a location relative to the exported project, e.g. exporting to *c:\Archive\project_01* will cause external files to be copied to *c:\Archive\project_01\external_files* (or one of its subdirectories). The exported project settings will reflect the new relative locations which the external files are now in.



5.13.2 Importing a project archive

Importing a project can be done with or without an open project. When there is an existing project *fe-safe* will ask the user if the operation would cause any files to be overwritten.

6 This chapter has been released in a separate document

7 Using *safe4fatigue*

7.1 Introduction

safe4fatigue is a suite of software for signal processing and graphics display and fatigue analysis from strain gauge data. The files produced by the *fe-safe* **Exports and Outputs** function (see section 22) can also be displayed using the graphics described in this section.

The functions available include:

- File Handling;
- Plotting, Printing and Exporting;
- File editing;
- Amplitude Analysis including File Modification and Digital Filters;
- Frequency Domain Analysis;
- Fatigue Analysis;
- Signal Generation.

A file may contain single or multiple channels of time history data (e.g. measured signals obtained from a data acquisition system), results produced in *safe4fatigue* (e.g. a Rainflow cycle histogram, a time-at-level distribution) and results files produced by the *fe-safe* Export and Diagnostic options described in section 22.

Section 7.2 gives an overview of the *safe4fatigue* user interface. Section 7.3-7.6 describes the file handling; file plotting, printing and exporting; and file editing. The Amplitude, Frequency and Fatigue analysis functions, and Signal Generation, are described in sections 9 to 12.

7.2 Overview of *safe4fatigue*

The user interface, which is common to all platforms, is shown in *Figure 7.2-1*. It consists of:

- a. The FEA-fatigue dialogue box;
- b. A window listing the loading files (data files);
- c. A window containing the materials databases;
- d. A window to show details of the open FEA file;
- e. A message window.

The layout of the user interface can be adjusted to suit user preference and the screen size.

On Windows platforms, the **Current FE Models** and **Loaded Data Files** windows support “drag-and-drop” methods. This means that selecting files in another Windows application (for example Windows Explorer), and then dragging them into the appropriate *fe-safe* window can automatically load the files.

When a file is “dragged-and-dropped” to the **Loaded Data Files** window, the file is added to the list of available data files.

Tip: If the *fe-safe* application is not visible, or is partly obscured by another application, then drag the files to the *fe-safe* icon on the Windows taskbar, and hover over it for a couple of seconds (without releasing the mouse button) until *fe-safe* becomes visible.

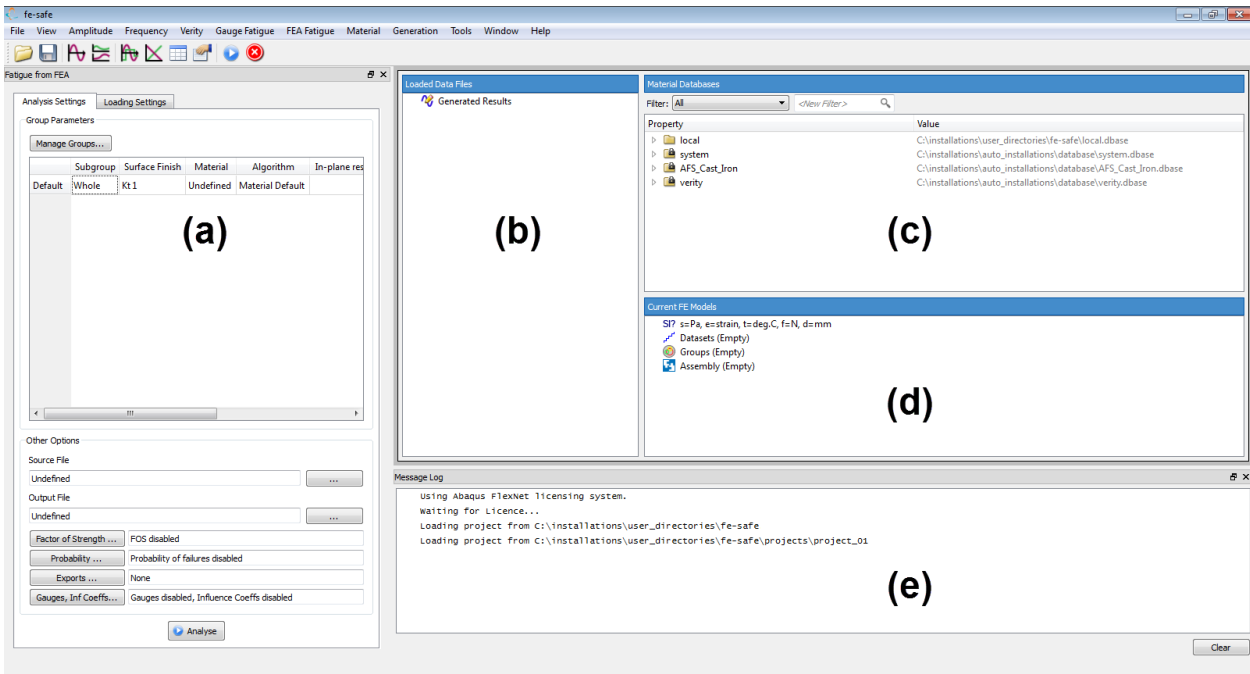



Figure 7.2-1: The *fe-safe* user interface

The **Current FE Models** window and the **Fatigue from FEA** dialogue box, normally displayed in *fe-safe*, are not required for *safe4fatigue* analysis, the **Material Databases** window is required for the fatigue analysis functions.

Note that almost all the operations performed in *safe4fatigue* are written to a macro recording file, and can be used in batch commands. See section 23 for a description of the macro recording and batch command system.

7.2.1 Analysing a load history file

The file is opened using **File >> Data Files >> Open Data File(s)**. It will appear in the **Loaded Data Files** window. If it is a multi-channel file the channels will be listed.


Files may be plotted by highlighting the file (or the channel in the file) and selecting the  icon on the Toolbar, or selecting **View >> Plot** (see section 7.5.6)

Multiple files, or multiple channels in files, may be plotted by highlighting the required channels using either the CTRL key, for highlighting individual channels, or the SHIFT key, for highlighting ranges of channels. This capability to process multiple files and channels applies to most of the signal manipulation and analysis functions in *safe4fatigue*. For example, several channels can be analysed in a single process using the analysis functions described in sections 9 to 12.

In the following examples a single channel file will be used.

To filter the signal (see section 10) the required channel is highlighted and the required filtering function is selected. An output file is generated automatically, and its name is displayed in the **Generated Results** section of the **Loaded Data Files** window. The filename shows that the file has been filtered. This information is also entered into the file header, and can be displayed by accessing the file properties (see section 7.5.21).

To calculate a Rainflow cycle histogram (see section 10) highlight the required signal and select **Amplitude >> Rainflow (and Cycle Exceedence) from Time Histories**

The results files are generated automatically. The 3-D cycle histogram can be displayed by highlighting the filename and selecting the Toolbar  icon or selecting **View >> Plot**. This plot can be rotated, scaled and manipulated (see section 7.5.22).

Results files can be re-scaled, integrated and manipulated using the **Amplitude** functions (see section 10).

7.3 Supported file types

fe-safe and *safe4fatigue* support the following proprietary data file types:

- industry standard binary DAC file (*.dac)
- analogue multi-channel AMC file (*.amc)
- ASCII single and multi-channel data files (*.txt, *.asc, etc.)
- ASCII histogram files (*.txt, *.asc, etc.)

fe-safe and *safe4fatigue* support the following third-party data file types:

- Servotest SBF and SBR files (*.sbf, *.sbr)
- Snap-Master file (*.sm?)
- MTS RPCIII binary data file (*.rsp)
- Adams multi-column ASCII tabular data (*.tab)
- ANSYS Modal Coordinates File (*.mcf)
- ASAM MDF4 binary data file (*.mf4)

The interfaces to these file formats operate without conversion. In other words, no translation is required - *fe-safe* reads the data directly from the file. Data from different file formats can be included in the same plot, or analysed at the same time. These file formats are discussed in Appendix E and Appendix F.

Dassault Systèmes UK Ltd endeavours to maintain interface support to the latest versions of supported third-party data files.

7.4 Running *safe4fatigue*

7.4.1 Analysis program operation

Each analysis module displays a dialogue box for defining the input data and the output results file, together with the specific input required by the module.

7.4.2 Input files

For *safe4fatigue* all input data files and signals are selected in the **Loaded Data Files** window. To open a new data file select **File >> Data Files >> Open Data File(s)**.

7.4.3 Start/end time

The user may select a portion of the data for analysis. The default values are for the complete signal.

7.4.4 Output file name

This will be generated automatically from the name of the input data signal, the date and the type of analysis being performed. At the end of the analysis the results will be appended to the **Generated Results** section of the **Loaded Data Files** window, to allow plotting and further analysis.

7.4.5 Closing files

Files may be closed and removed from the **Loaded Data Files** window by highlighting the file name(s) and selecting **File >> Data Files >> Close Data File(s)** or hitting the **Delete** key.

7.5 File handling, plotting, printing and exporting

This part of the *safe4fatigue* environment provides Plotting, Printing, Saving and Clipboard interfacing.

7.5.1 Loaded data files window

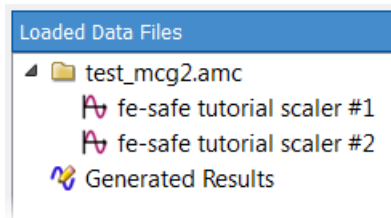


Figure 7.5-1 Loaded Data Files window

The **Loaded Data Files** window lists all the open data files. Each data file is the top-level item in the tree and has a number of signals associated with it as sub-items. Signals can be analysed or plotted by selecting them and then selecting the required operation. Most operations allow multiple signals to be selected at once, using the standard Windows functions of **<SHIFT>** or **<CTRL>** with mouse clicks.

This window also displays the contents of the **Generated Results**. Analysis results are placed in the **Results Archive** on completion of the analysis. Items in the **Results Archive** can be plotted and analysed in the same way as open data files.

A right mouse click over the **Loaded Data Files** window displays a menu. This duplicates some **File** menu options, as well as the following tasks specific to the **Loaded Data Files** window:

Refresh Refreshes the display of file names.

Expand All Expands all tree items in the window to see the contents of all files.

Collapse All Collapses all tree items to display only file names.

7.5.2 Open data file

Select **File >> Data Files >> Open Data File(s)**, select the Toolbar icon, or select **Open Data File(s)** from the context sensitive menu displayed by right mouse clicking over the **Loaded Data Files** Window.

The function is used to open existing data files, extract the signals within the file, and add them to the **Loaded Data Files** Window.

This operation can also be performed by dragging data files into the **Loaded Data Files** Window.

7.5.3 Close data file

Select **File >> Data Files >> Close Data File(s)**, select the Toolbar icon, hit the **Delete** key or select **Close Data File(s)** from the context sensitive menu displayed by right mouse clicking over the **Loaded Data Files** window.

The function is used to close the selected data files.

7.5.4 Exit

Select **File >> Exit** or click the cross in the top right hand corner of the screen to exit *fe-safe*.

7.5.5 Save data file(s) as

Select **File >> Data Files >> Save Data File(s) As**, or select the Toolbar icon.

This function allows the selected data files, or channels within a file, to be exported to a new format; to be exported to the same format with a different file name or a different sample rate; and to export a selected section of the data.

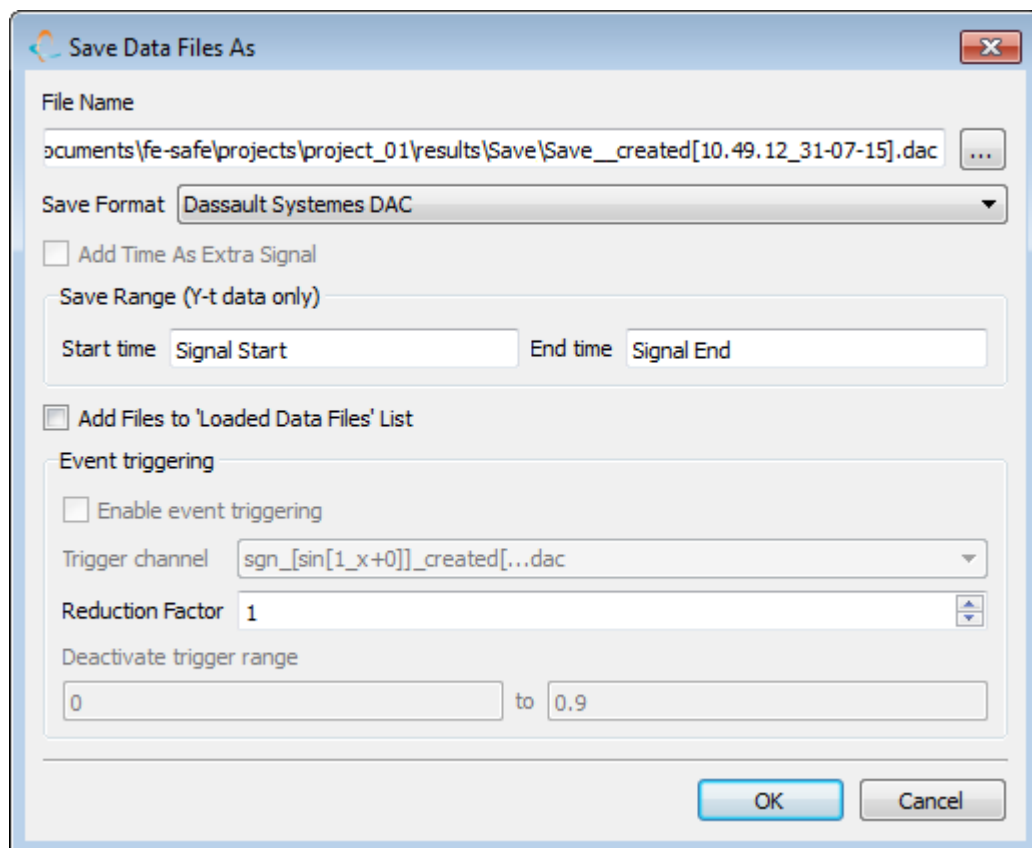


Figure 7.5-2 Save Data Files As window

The following options are available:

File Name: The name of the output file. Output file names are auto-generated, but can be modified by the user. If several signals are selected in the **Loaded Data Files** window and saved in single channel format (such as Dassault Systemes DAC) then the names will be auto generated by adding an underscore and a number. These names can be also modified by the user

Save Format: Allowable export file formats are:

- Dassault Systemes DAC
- Dassault Systemes DAC (UNIX)
- Dassault Systemes AMC
- Single column ASCII
- Multi-column ASCII
- Servotest binary file.

Note that in *fe-safe* DAC files are interchangeable between Windows and UNIX. The separate export functions are included to allow these files to be exported to other third-party software.

Add Time As Extra Signal: For ASCII output files an extra column can be added containing the sample time for each sample.


Start Time/End Time: The portion of data to save to the output file. The default values save the complete signal.

Reduction Factor: The output file can be down-sampled by exporting every *n*th value. For example, a reduction factor of 2 will cause alternate values to be saved.

Add Files To Open Data Files List: After saving the file, the file name can be added to the **Loaded Data Files** Window.

Event Triggering: These options, which apply to AMC files only, add an event trigger channel based on the configured criteria.

7.5.6 Plot

Select **View >> Plot** or the main toolbar icon .

This will create a plot window for each of the data signals selected in the **Loaded Data Files** window.

The plot window toolbar icons provide the following functions:



- 1 Copy the current plot window to clipboard
- 2 Print the current plot window
- 3-4 For the current plot window, scroll up or down, or tilt a histogram
- 5-6 For the current plot window, scroll left or right, or tilt a histogram
- 7-8 For the current plot window, move to the start or end of a file
- 9-10 Zoom the current plot window in and out
- 11 For the current plot window, toggle the display of max / min values
- 12 For the current plot window, toggle the display of cursor
- 13-14 Zoom mode options

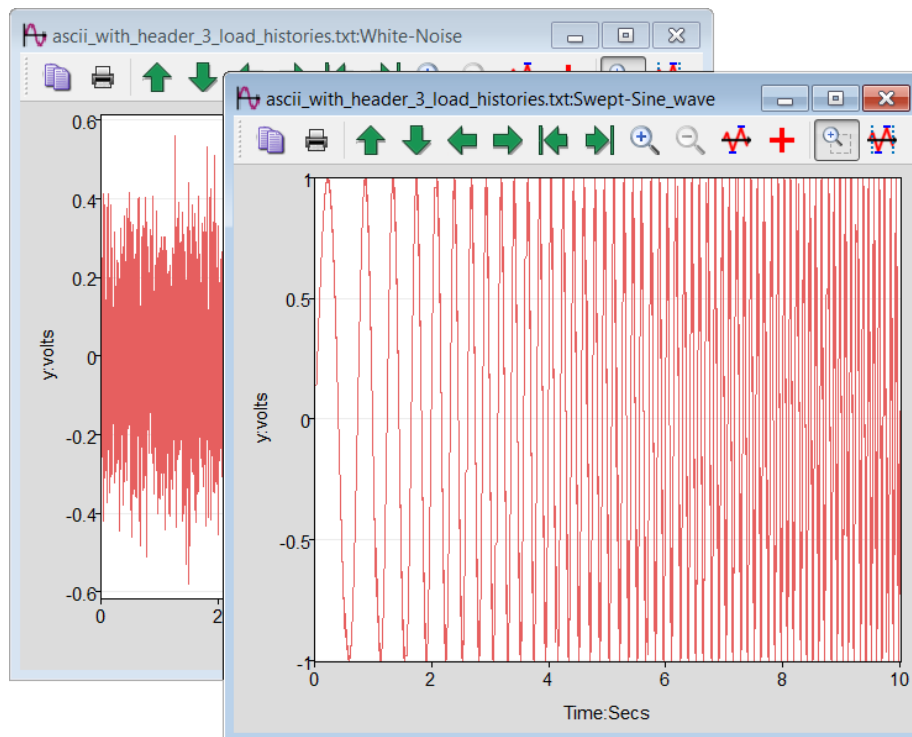



Figure 7.5-3 Single signal plots

7.5.7 Overlaid multiple plots

Select **View >> Overlay Plot(s)** or the main toolbar icon .

This will create a single window and superimpose each of the signals selected in the **Loaded Data Files** window.

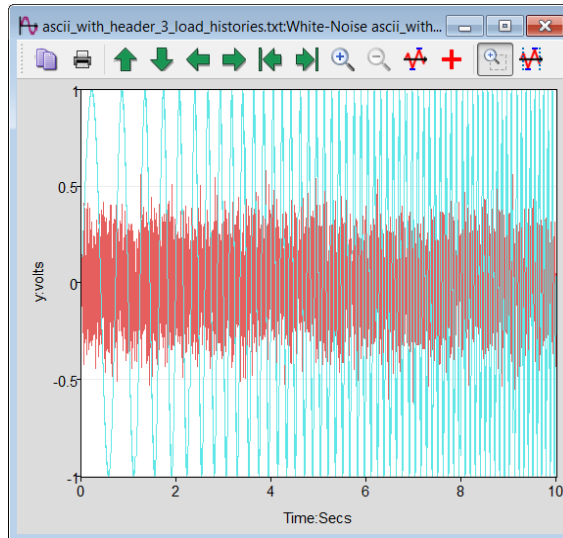



Figure 7.5-4 Overlaid plots

7.5.8 Stacked multiple plots

Select **View >> Stack Plot(s)** or the main toolbar icon .

This will create a single window and plot all of the signals selected in the **Loaded Data Files** window in a separate plot space.

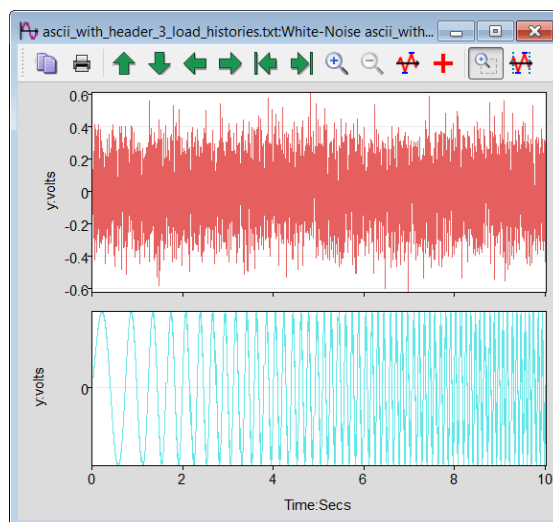


Figure 7.5-5 Stacked plots

7.5.9 Cross plot

Select **View >> Cross Plot** or the toolbar icon .

This will create a single window and plot the first two signals selected in the **Loaded Data Files** window as a cross plot.

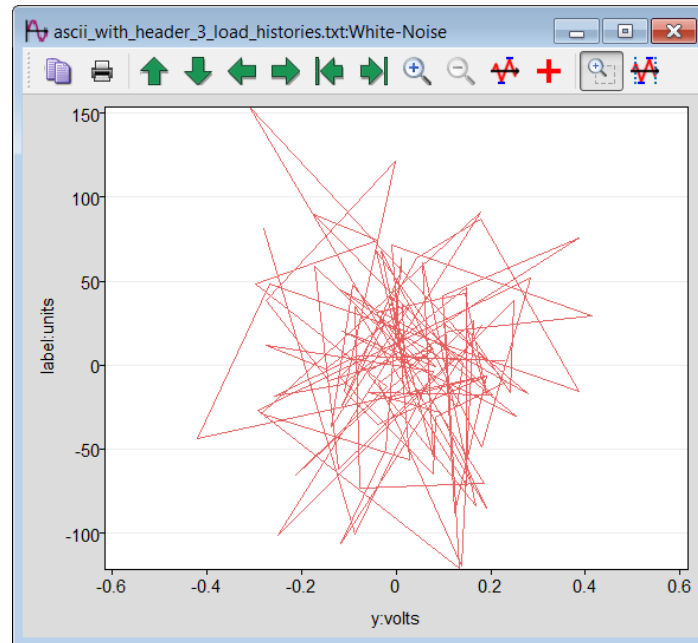



Figure 7.5-6 Cross plot


7.5.10 Numerical listing

Select **View >> Numerical Listing** or the main toolbar icon . This function can be used to view numerically the contents of a signal or results file. Multiple files and channels (and formats) can be listed together.

7.5.11 Print

In the plot window select the toolbar icon  to print the active plot window.

7.5.12 Copy

In the plot window select the toolbar icon  or select **Copy to Clipboard** from the context sensitive menu displayed by right mouse clicking over the active plot window.

The contents of the current plot window are copied to the clipboard for inserting into word processing and spreadsheet software.

7.5.13 Add plots


Select **Overlay selected plots** or **Stack selected plots** from the context sensitive menu displayed by right mouse clicking over the active plot window.

This will superimpose plots of the selected signals or add them in a separate space to the current plot window.

7.5.14 Add XY plot

Select **Add XY Plot** from the context sensitive menu displayed by right mouse clicking over the active plot window. This will superimpose a cross plot of the first two selected signals onto the plot in the first sector of the current window.

7.5.15 Cursor toggle

Select **Cursor Toggle** from the context sensitive menu displayed by right mouse clicking over the active plot window, or the plot window toolbar icon .

This will toggle on/off the cursor for picking values from a sequential plot.

For multiple plot spaces the cursor will be displayed for the first plot in each plot space.

Use the left and right keyboard arrows, or the arrow icons to move forwards and backwards one sample at a time, or use a mouse click to jump to a new location.

The cursor values will be displayed in all prints and copies. Cursor values can be converted into permanent text.

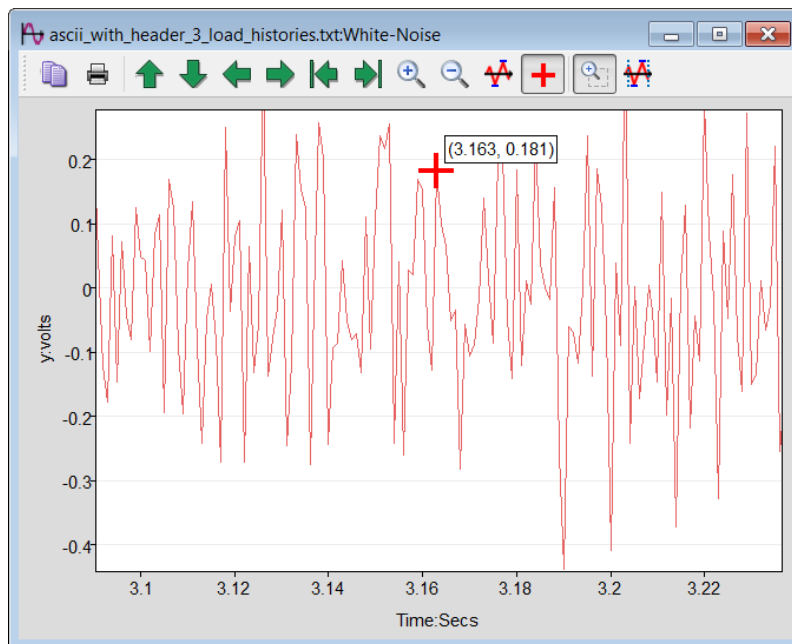



Figure 7.5-7 Value under cursor

7.5.16 Zooming

The mouse is used to define the required area of the plot.

The **Zoom In** and **Zoom Out** from the context sensitive menu displayed by right mouse clicking over the active plot window, or the plot window toolbar icons  can be used to zoom in and out of the selected area.

7.5.17 Add line...

This function can be used to add extra lines to a graph (e.g. limit lines).

Select **Add Line...** from the context sensitive menu displayed by right mouse clicking over the active plot window.

This displays the following dialogue box.

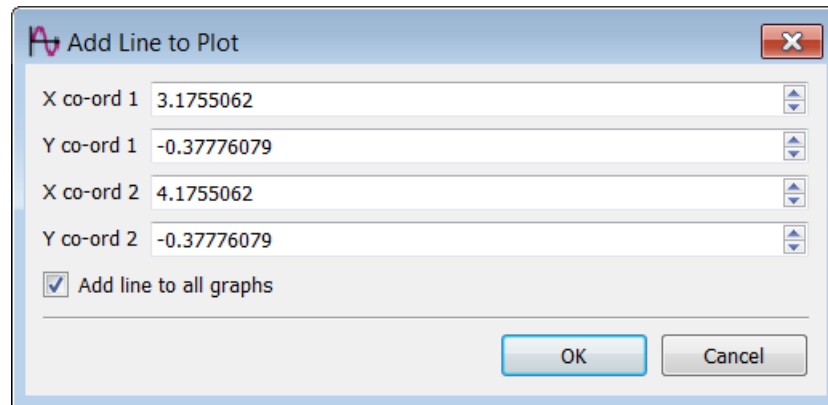


Figure 7.5-8 Adding lines to a plot

The co-ordinates for the start and end point of the line can be defined.


7.5.18 Remove line

Select the line, then select **Remove Line** from the context sensitive menu displayed by right mouse clicking over the active plot window. The line will be removed.

7.5.19 Remove all lines

Select **Remove All Lines** from the context sensitive menu displayed by right mouse clicking over the active plot window. All added lines on the graph will be removed.

7.5.20 Properties (<Ctrl + Enter>)

Select **View >> Properties** or the main toolbar icon , or select **Properties** from the context sensitive menu of a signal.

The **Properties** function displays information about the selected signal.

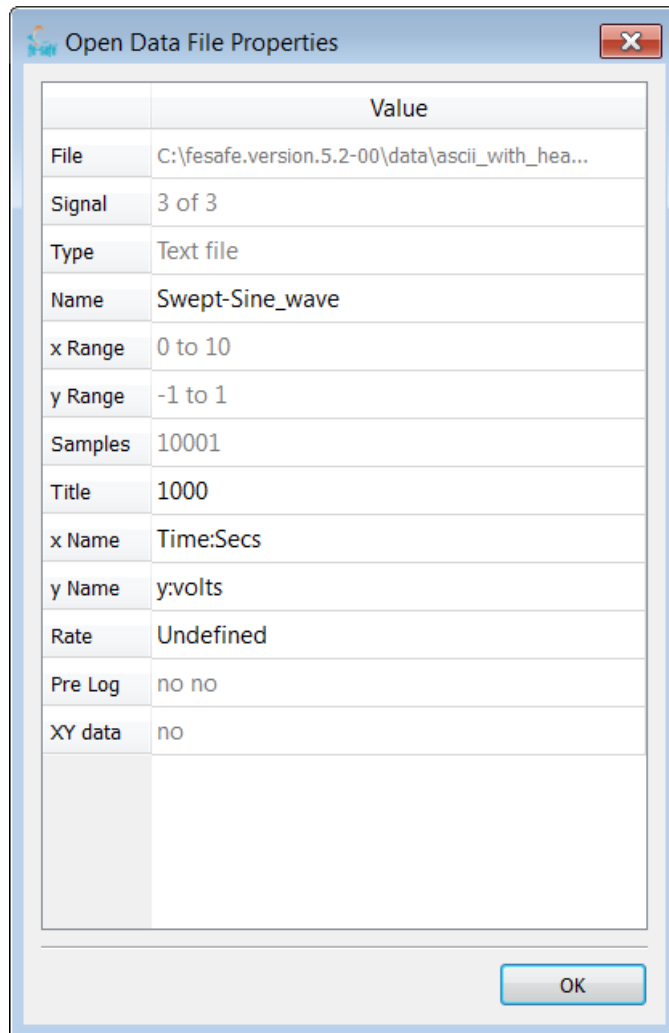


Figure 7.5-9 Signal properties

7.5.21 Plot configuration dialogue

Select **Properties** from the context sensitive menu for a plot window. The **Properties** dialogue box is displayed.

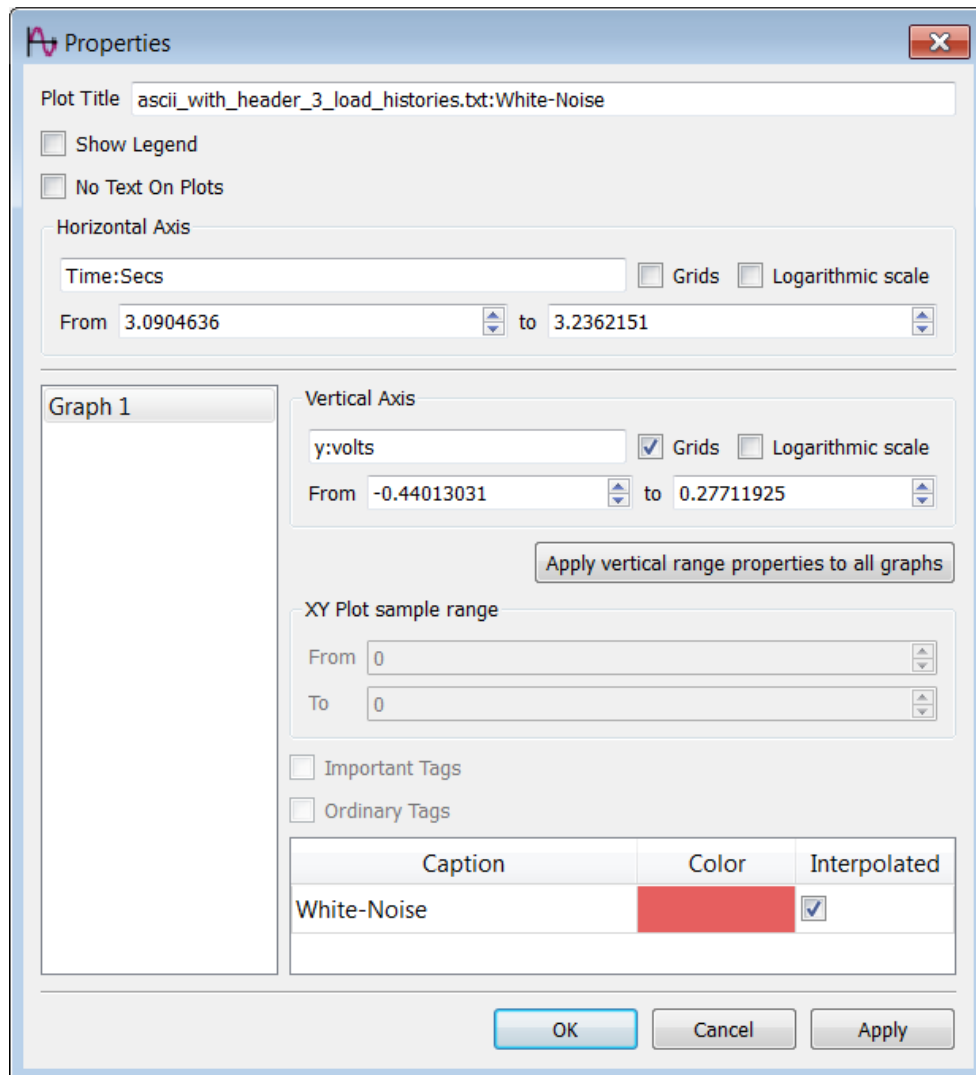


Figure 7.5-10 Plot configuration window

The plot title can be changed by over-typing.

The lower section allows individual plots within a plot space to be configured.

For line plots this allows the axis limits, log scaling, labels, grids and interpolation modes to be set. For histograms similar options are available, plus tilt/rotation controls and a check box to toggle between surface and tower plots.

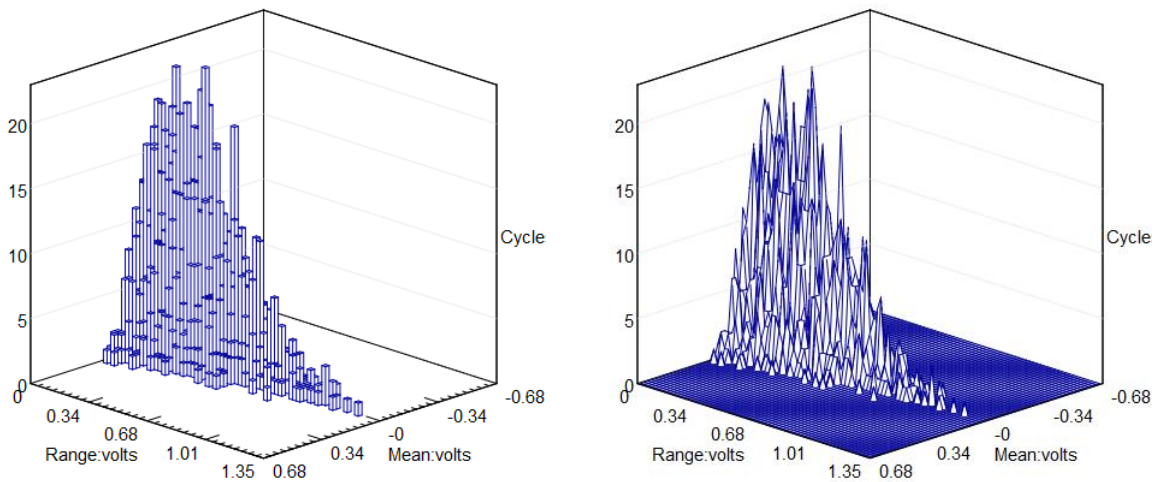
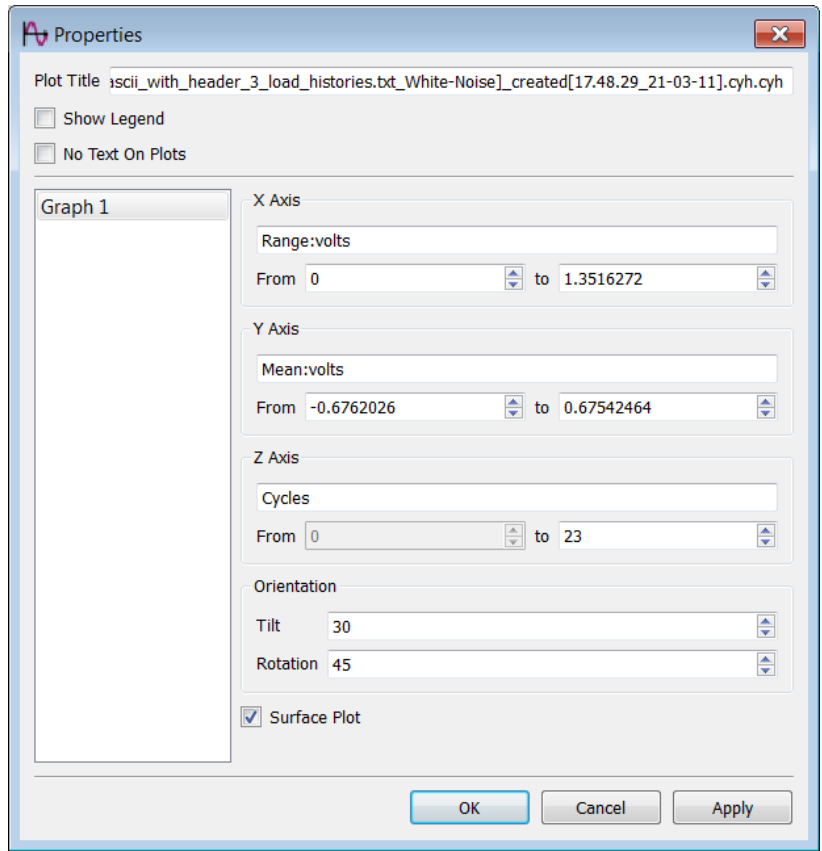


Figure 7.5-11 Histogram configuration window

7.5.22 Scrolling/tilting/rotating

These functions are accessed from the plot window toolbar or from the **Properties** dialogue for a plot window.

For sequential data plots the **left** and **right** arrows move forward and backwards one time base.

For histogram plots the **left** and **right** arrows control the rotation of a plot and the **up** and **down** arrows control the tilt.

If a histogram is plotted as towers and then the tilt is set to 90, this provides a colour contour plot of the data:

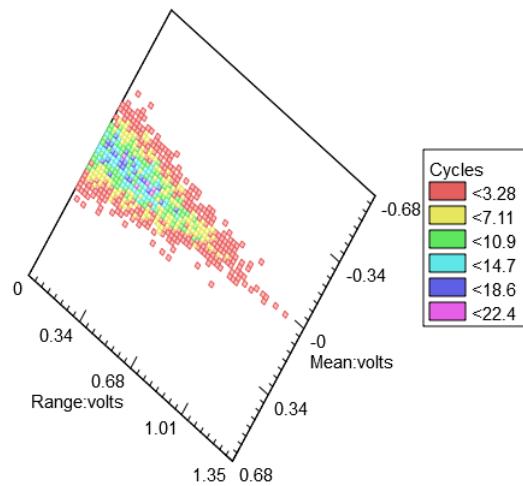


Figure 7.5-12 Contour plot of a histogram

7.5.23 Add text / edit text / remove text / move text

These options are available from the context sensitive menu for a plot window.

To add new text, select the **Add Text** item to display the following dialogue box:

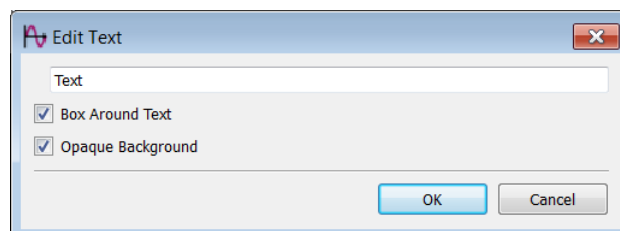


Figure 7.5-13

Enter the text and press **OK** to add the text to the plot.

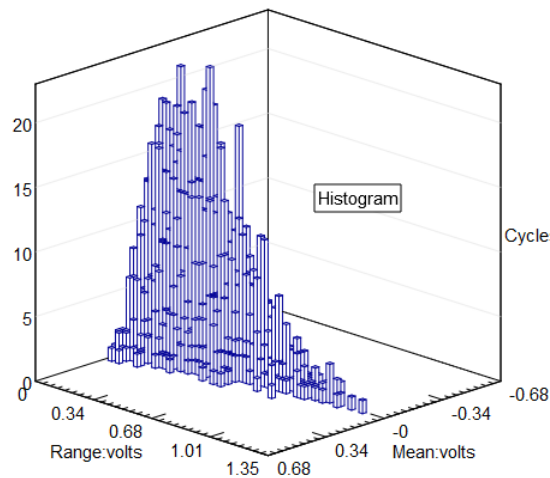


Figure 7.5-14 Annotating plots

To move text, drag it using the left mouse button.

To edit text, double click it or select **Edit Text** from the pop-up menu.

To remove a block of text, right click over the text and select **Remove Text**.

7.5.24 Abort current operation

This is accessed using the main toolbar.

During an analysis this will abort the analysis.

7.5.25 Close plots

This is accessed from the **Window** menu.

This will close all of the plot windows.

7.6 File editing

7.6.1 Introduction

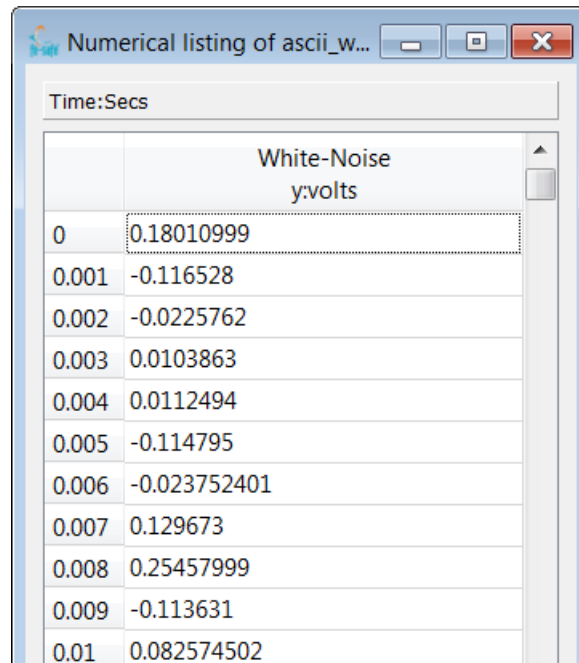
The file editor is a digital editor that can be used for editing time history and analysis results files. All file formats can be edited, including matrices from the Rainflow, Markov and other analysis functions. X-Y data files are excluded.

The editor stores the edits without modifying the input file, until the user selects to exit. An edited file can then be saved in any supported format. For example, a load history file in ASCII or binary format may be edited then saved as a binary DAC file. There is no limit to the file length.

7.6.2 Opening and displaying the input file

Input files are selected by highlighting the channel names in the **Loaded Data Files** window. Several channels can be selected from a multi-channel file. Several single-channel files and channels from different multi-channel files can also be selected concurrently. Channels with the same sample rate and number of samples are displayed in a single window, otherwise the channels are displayed in different windows.

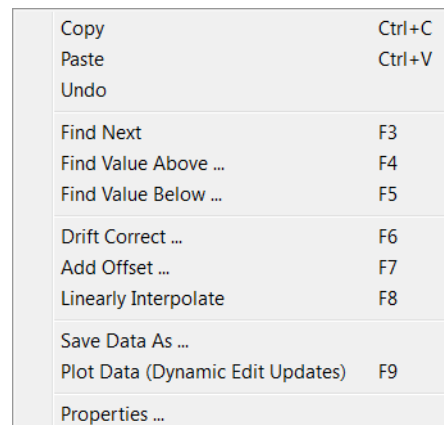
After selecting the required file(s) or channel(s), select **View >> Numerical Listing**. The first section of the file will be displayed. In this example a single channel file will be used.



Time:Secs	White-Noise y:volts
0	0.18010999
0.001	-0.116528
0.002	-0.0225762
0.003	0.0103863
0.004	0.0112494
0.005	-0.114795
0.006	-0.023752401
0.007	0.129673
0.008	0.25457999
0.009	-0.113631
0.01	0.082574502

Figure 7.6-1 Numerical listing and editing window

A context-sensitive **Edit** menu is displayed by clicking the right mouse button over the **Numerical Listing** window:



Copy	Ctrl+C
Paste	Ctrl+V
Undo	
Find Next	F3
Find Value Above ...	F4
Find Value Below ...	F5
Drift Correct ...	F6
Add Offset ...	F7
Linearly Interpolate	F8
Save Data As ...	
Plot Data (Dynamic Edit Updates)	F9
Properties ...	

Figure 7.6-2 The Pop-up **Edit** menu

After the first piece of data has been edited, the following prompt will be shown:

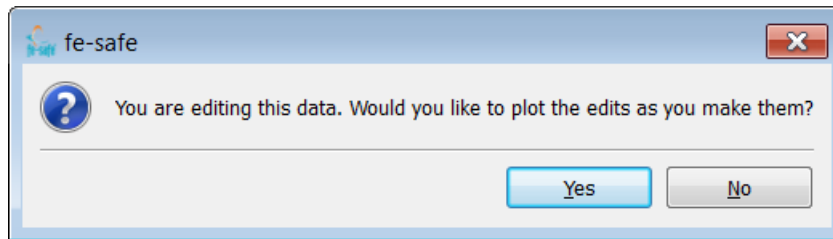


Figure 7.6-3

Clicking yes displays the numerical listing next to the signal. The signal is then updated after every edit.

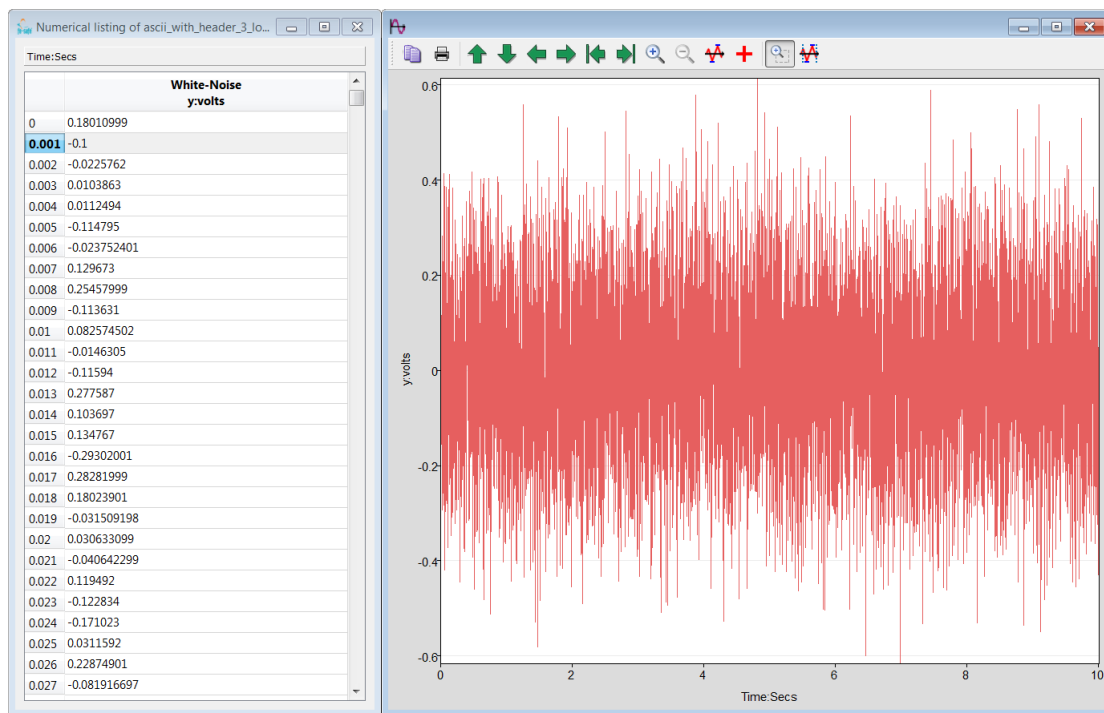


Figure 7.6-4 Edit window and graphical display window

With the cursor over the graphics window, click the right mouse button and select **Properties**. The properties of the plotted data can now be edited, for example to plot just the range displayed in the **Numerical Listing** window.

7.6.3 Editing the file

Editing a single value

Any data value can be edited by selecting the value, and typing a new value. The effect of the edit will be shown in the graphics window.

Several values can be selected (by dragging with **<Shift>** and **left mouse** button). Over-typing one value will change all the selected values.

A right mouse click displays the edit dialogue box.

The following functions are available.

Copy (<Ctrl> + C)

A section of the file can be highlighted and placed in the **Copy** buffer.

Paste (<Ctrl> + V)

The contents of the **Copy** buffer can be pasted into another section of the file, over-writing existing data points. The **Paste** operation starts at the selected data point.

Find next (F3)

Finds the next occurrence (from the current data point) of the condition set by **Find Value Above** or **Find Value Below**.

Find Value Above... (F4)

Sets **Find Next** to find the next value higher than the specified value.

Find Value Below... (F5)

Sets **Find Next** to find the next value lower than the specified value.

Add offset... (F7)

Adds a constant value to all the selected data points.

Drift correct...(F6)

Adds a non-constant value to all the selected data points, to remove 'drift' on a signal. For example, if a value of 100 is entered in the **Drift Correction** dialogue box :

- a value of zero will be added to the first selected point;
- a value of **-100** will be added to the last selected data point (note the minus sign).
- values obtained by interpolating between 0 and **-100** will be added to the intermediate data points.

Linearly Interpolate (F8)

For the selected data points:

- the first and last points are unchanged;
- the intermediate data points are replaced with values obtained by linearly interpolating between the first and last points.

7.6.4 Saving the edited file

From the **Edit** dialogue select **Save Data As ...**. The standard **Save Data As** dialogue is shown. The user may select the filename, file format, section of the file to be saved, and may choose to down-sample the data.

Alternatively, the user may close the **Numerical Listing** window, and select to save the edits. This action will display the **Save Data As...** dialogue.

7.6.5 Exiting without saving the edited file.

Close the **Numerical Listing** window, and select not to save the edits. This action will close the input file without making any edits.

8 Material properties

8.1 The *fe-safe* Material Database

fe-safe and *safe4fatigue* are supplied with a comprehensive database containing fatigue properties for commonly used materials.

Material data is managed within the main application environment. Functions are available for creating new material records, editing, sorting and plotting material properties and approximating fatigue parameters. All functions are available in *fe-safe* and *safe4fatigue*.

8.2 The Dassault Systèmes Material Data Archive

Dassault Systèmes maintains an extensive archive of material fatigue data. Material data from the archive is made available to customers on request. Requested material data will normally be added to the *fe-safe* **Material database**, for distribution with future releases.

8.3 Material Data Services

Dassault Systèmes offers a Material Data Search Service and a Material Testing Service.

8.4 Using the *fe-safe* Material Database

8.4.1 Viewing material data

The *fe-safe* **Material Database** window is always displayed:

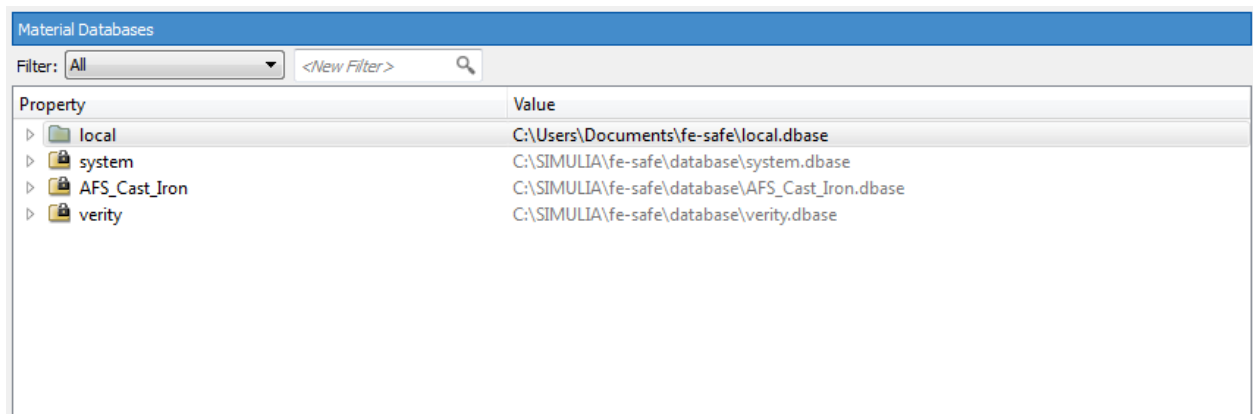


Figure 8-1

Material properties

Most of the functions described in this section can also be performed using a context-sensitive pop-up menu, which is available by clicking over the **Material Databases** window with the right mouse button:

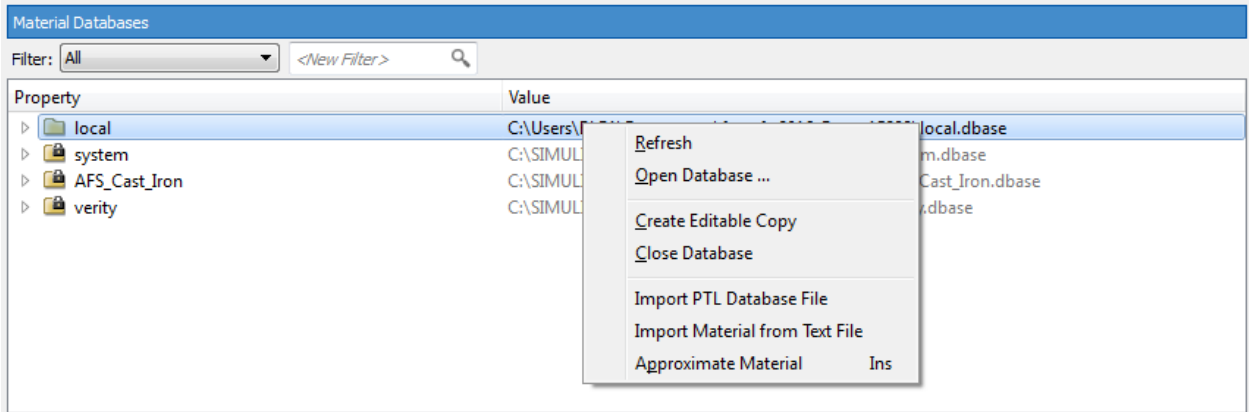


Figure 8-2

The **Material Databases** window presents the material data in an expandable tree view. Expanding the database view displays the material records in that database.

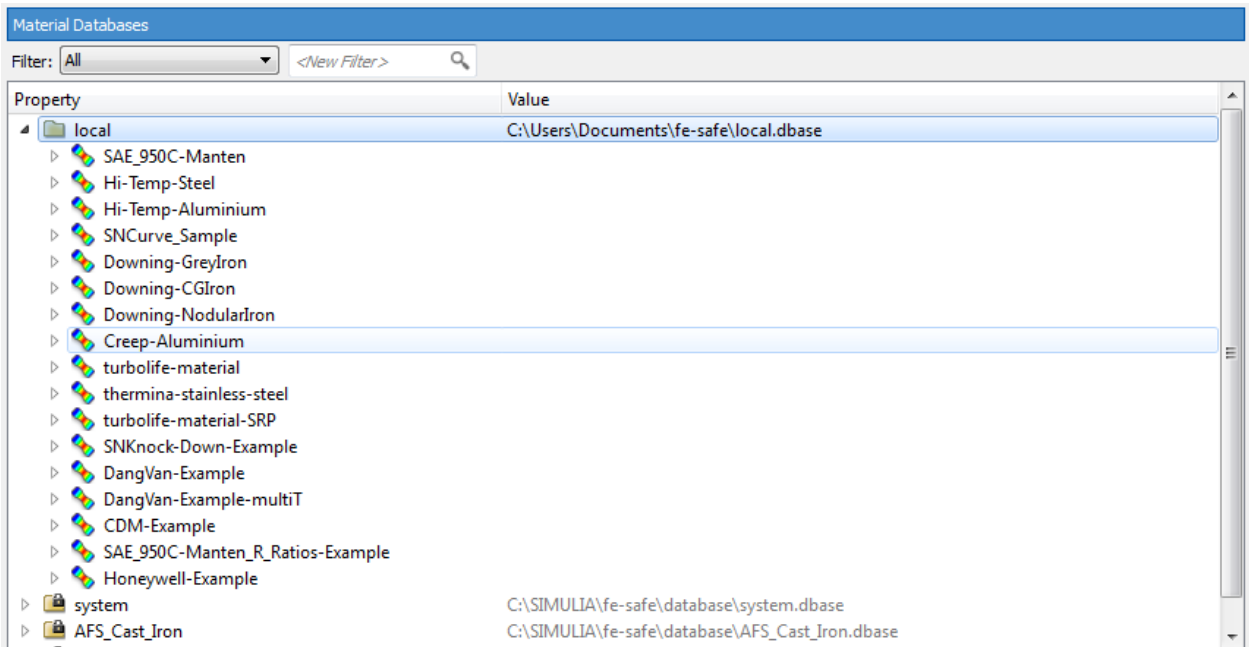


Figure 8-3

Similarly, the material's parameters can be displayed by expanding the material name:

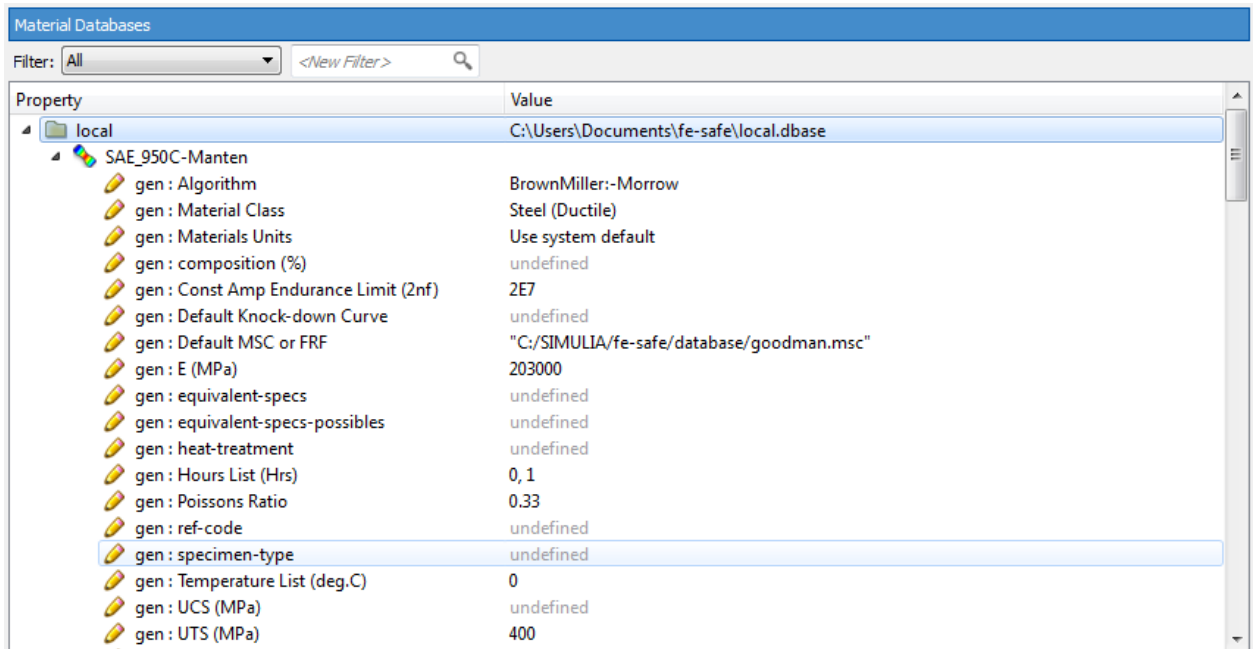



Figure 8-4

8.4.2 Editing material parameters

Material parameters can be edited if the user has write-access to the database, by double-clicking on the parameter value. Alternatively, pressing function key **F2** opens the highlighted field for editing. Read-only databases (locked for editing) are marked with a padlock symbol on their icon: .

8.4.3 Creating new material records

New material records can be created by either:

- copying an existing material, renaming it, and editing its properties, or
- using the material approximation function (see 8.4.7).

8.4.4 Creating a new database

Two steps are required to create a new database: selecting the file location for the database and an associated template. The template file defines properties to be included for each material in the database, and must always be present for the database to work correctly. A template can be stored within its own file or can be embedded in one file with its corresponding material database.

Select **File >> Materials >> Create Materials Database ...**

A **Pick database template** dialogue will appear. Select a source template for the database to be based on. The template can be in a `.template` file or embedded in an existing material database file (`.dbase`). The use of `.template` files is deprecated and it is no longer possible to create new ones.

A **Choose database location** dialogue will follow. Type a name for the new database in the **File name** box, for example: `my_new_database_01.dbase`. If the file already exists, it will be overwritten. If a template file was selected in the first step, *fe-safe* will ask whether to embed a copy of the template inside the new database file

(Figure 8-5). If a database file was selected in the first step, the source template will automatically be copied into the new database and this dialogue will not appear.

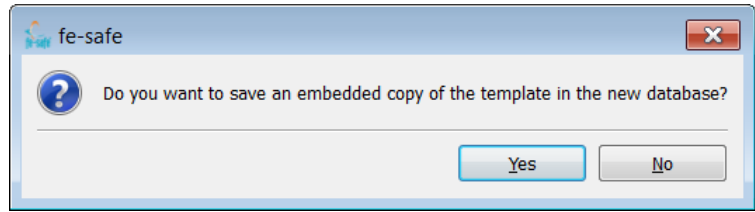


Figure 8-5

The new database is added to the tree-view list in the **Material Databases** window. To add a material to the database, the **Approximate Material** function can be used as described in section 8.4.7, below.

8.4.5 Filtering and sorting material records

Sort criteria can be used to filter the displayed materials. A sort string can be selected from the Filters drop down menu at the top of the Material Database window. The list of materials displayed will now be filtered such that only materials that contain the selected sort string in one of their parameters will be displayed.

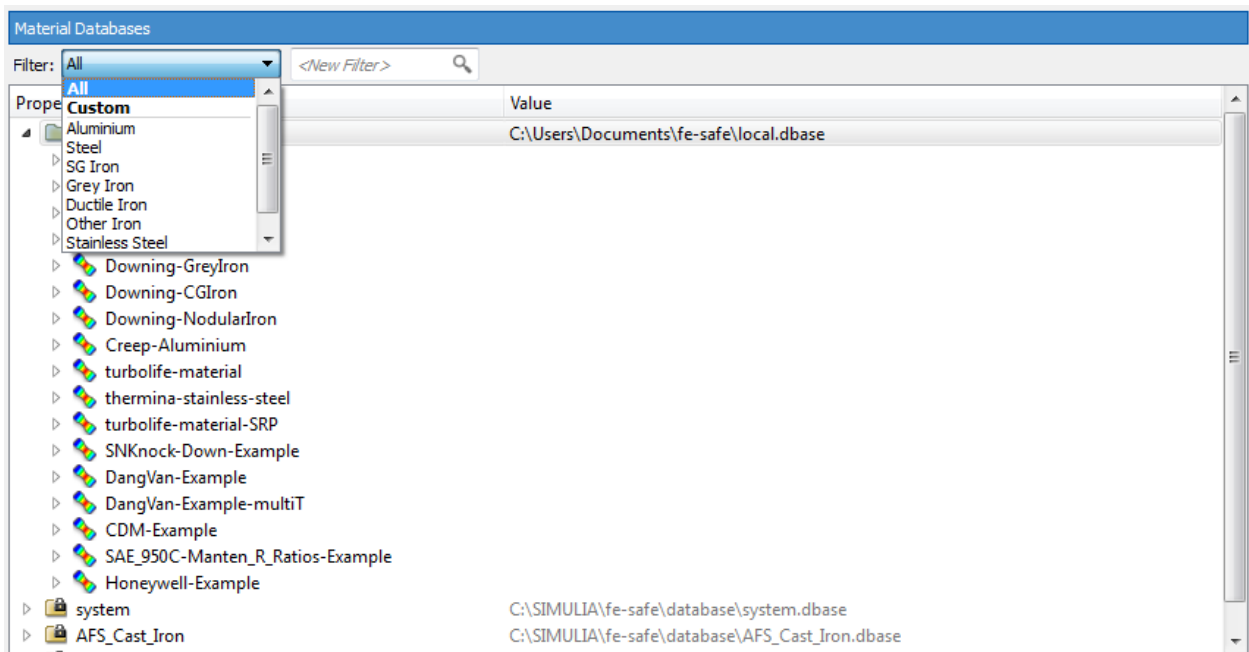


Figure 8-6

To filter using a custom sort string, select the Custom option from the Filters drop down menu and then type the chosen string in the adjacent search box.

To return to showing all materials select the 'All' option from the drop down menu.

8.4.6 Material reference document links


Material record text fields can be used to reference additional related documentation, for example a material test report. Any text field can be used as a link to another document, by typing the path to the file surrounded by quotation marks.

For a file stored locally, the path will be a local path, e.g.:

“c:\my_data\material_reports\Inconel_718.doc”

Alternatively an internet path can be entered, e.g.:

“http://www.<website_name>.com/data/inconel_718.html”

For every document link created, an additional field is added to the end of the material record. Double-clicking on the document link icon, , can display the document in an appropriate third-party viewer for that document type, for example:

- an html file can be displayed using the default browser, e.g. Internet Explorer®;
- a pdf file can be displayed using Adobe Acrobat® viewer;
- ASCII files can be viewed using a text editor;
- *.doc files can be viewed using Microsoft Word®.

Document viewers are not part of the *fe-safe* suite of software. If *fe-safe* cannot make this association the user will be prompted for the application. On Windows platforms, facilities are available for associating file name extensions with a particular viewer.

8.4.7 Approximating material parameters

If test data is not available for a particular material, then fatigue properties can be approximated using the **Approximate Material** function.

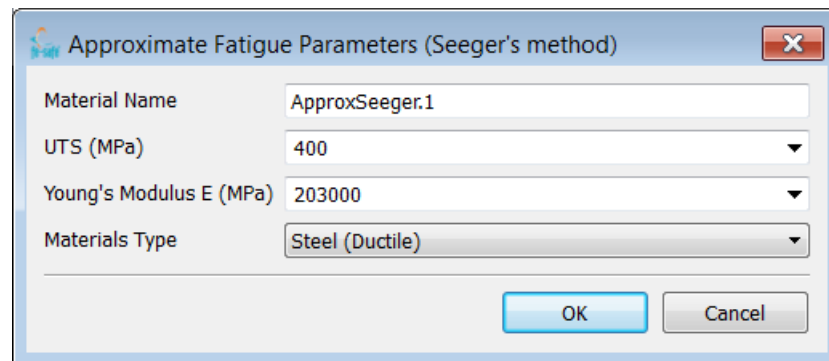


Figure 8-7

This function uses Seeger's method (see the Fatigue Theory Reference Manual) to generate approximate fatigue parameters based on the UTS (tensile strength) and elastic modulus of the material. In this dialogue, the default system units are used for defining E and UTS. S-N data is also generated.

A material type must be selected from the drop-down list:

- For plain carbon and low to medium alloy steels, use either:
 - Steel (Brittle), or
 - Steel (Ductile).
- For aluminium alloys, use either:
 - Aluminium (Brittle), or
 - Aluminium (Ductile).
- For titanium alloys, use:
 - Titanium.

The material type information is used to evaluate the most suitable 'preferred fatigue algorithm' setting for the material.

The approximated material is added to the list in the **Material Databases** window. Parameters can subsequently be edited in the same way as any other material.

8.4.8 Supplementing approximated material data

Approximated material data may be modified so that it corresponds more closely to available test results. A method for supplementing approximated material data with available data is detailed in Technical Note TN-001.

8.4.9 Preferred fatigue algorithm

A preferred fatigue algorithm is embedded in each material record. Using the material approximation algorithm, this is allocated based on the material type selected (see 8.4.7). The preferred fatigue algorithm can be configured by double clicking on the `gen:Algorithm` or `gen:Material Class` fields in a material record. Both these options pop-up the same dialogue:

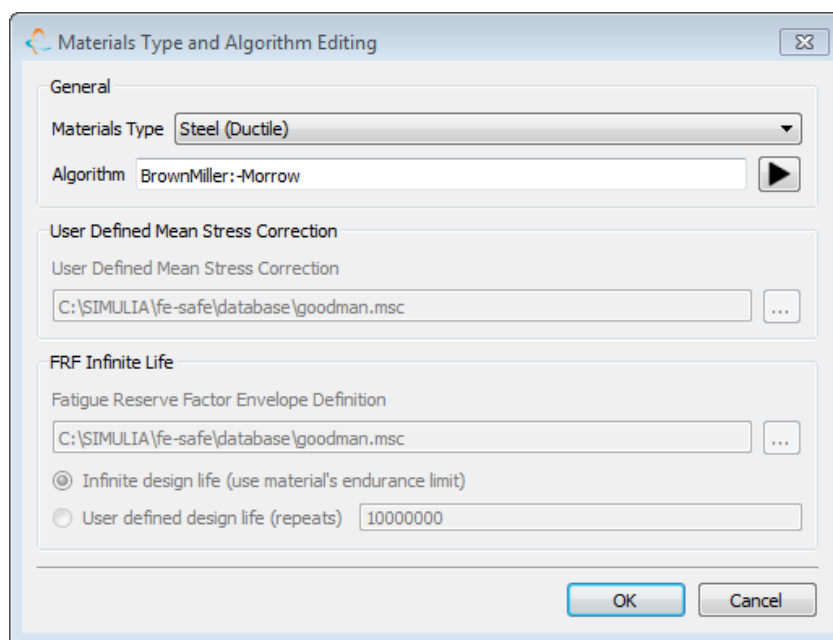


Figure 8-8

The following material type options are available:

- Steel (Brittle)
- Steel (Ductile)
- Aluminium Alloy (Brittle)
- Aluminium Alloy (Ductile)
- Titanium
- SG Iron
- Grey Iron
- Ductile Iron
- Other Iron
- Other

Modifying the material type will suggest a preferred algorithm.

8.4.10 Material data units

Material data is stored in the database in units of MPa and degrees C. It may be entered into the database using either MPa and degrees C, or ksi and degrees F. Material parameters may be listed in the data base in either unit system, and may be plotted in either unit system. This option to set the units is configured by double-clicking the `gen:Material Units` field for the particular material. This opens a drop down menu:

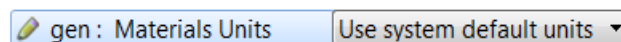


Figure 8-9

The following options are available:

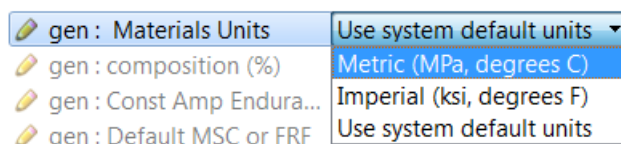


Figure 8-10

The units setting applies only to that material, and applies only to the units used to display and list the material properties. It has no effect on values stored in the material database, which are always stored in units of MPa and degrees C.

8.4.11 Selecting the material used in an analysis

For fatigue methods from measured data

For fatigue methods from measured data from the **Uniaxial Fatigue** menu (see sections 11 and 12), highlight the required database in the **Material Databases** window, before selecting the analysis method from the menu. In the

dialogue box for the selected analysis, the required material can be selected from a drop down list of the materials in the highlighted database.

For fatigue methods from FE data

Highlight the required material in the **Material Databases** window. In the **Fatigue from FEA** dialogue box, double-click on the material field for the required group in the **Group Parameters** table. Confirm whether or not to change the material for the current group to the highlighted material.

8.5 Material parameters

8.5.1 Database file structure

Material database files consist of a tab delimited ASCII file (with extension `.dbase`) containing material parameters, and an associated template file (extension `.template`) containing the metadata which relates the data fields in the database file to associated keywords used by the software.

8.5.2 Table of material parameters

The following table contains the material parameters used by *fe-safe*, and their associated keywords.

Category	Displayed name	Units	Keyword	Definition
General	gen : E (*)	MPa	E	Elastic (Young's) Modulus
	gen : Algorithm		MATL_ALGORITHM	Default fatigue algorithm for this material, e.g. BrownMiller-Morrow
	gen : MaterialClass		Material_Class	Material class, e.g. Steel (Ductile)
	gen : MaterialsUnits		MATL_UNITS	Units used for this material record, (default = Use system default)
	gen : ConstAmpEnduranceLimit (*)	2nf	Const_Amp_Endurance_Limit	Constant amplitude endurance limit
	gen : Default MSC or FRF		DEFAULT_MSC	Default user defined mean stress correction function
	gen : Default Knock-down Curve		DEFAULT_KNOCKDOWN	User defined additional effects curve
	gen : PoissonsRatio (*)		MATL_POISSON	Poisson's ratio
	gen : TemperatureList (*)	°C	TempList	List of temperatures (in °C)
	gen : UCS	MPa	UCS	Ultimate Compressive Strength
	gen : UTS	MPa	UTS	Ultimate Tensile Strength
	gen : HoursList (*)	hours	HoursList	Hours list
gen : Sf	MPa	sf	True fracture strength	
Text	text : Comment1		Comment1	Comment field for text entry
	text : Comment2		Comment2	Comment field for text entry
	text : Data_Quality		Data	Data quality (text entry)
	text : RevisionDate		Revision_Date	Revision date (read-only: updated automatically when any field is updated)
	text : RevisionHistory		Revision_History	Revision history (text entry)
	text : RevisionNumber		Revision_Number	Revision number (read-only: updated automatically when any field is edited)
	text : Source		Source	Data source, e.g. Test report ref 98765
Cyclic strength	css : K' (*)	MPa	K'	Cyclic strain hardening coefficient
	css : n' (*)		n'	Cyclic strain hardening exponent
Strain-life	en : b (*)		b	Basquin's fatigue strength exponent
	en : b2		BASQUIN2	Basquin's fatigue strength exponent at lives above the knee in life curve
	en : c (*)		c	Coffin-Manson fatigue ductility exponent
	en : Ef' (*)		Ef'	Fatigue ductility coefficient
	en : LifeAboveWhichToUseB2	2nf	KNEE_2NF	Knee in life curve (life above which to use b2)
en : sf' (*)	MPa	sf'	Fatigue strength coefficient	

Category	Displayed name	Units	Keyword	Definition
Cast Iron	gi : K'-Comp	MPa	K_C	Compressive cyclic strain hardening coefficient
	gi : m'-CompSecSlope		M_C	Compressive secant slope
	gi : m'-TensSecSlope		M_T	Tensile secant slope
	gi : Mu-ModulusofUnloading		MU	Modulus of unloading
	gi : n'-Comp		N_C	Compressive cyclic strain hardening exponent
	gi : SWTLifeCurveCoeff	MPa	SWT_COEFF	Smith-Watson-Topper life curve coefficient
	gi : SWTLifeCurveExponent		SWT_EXP	Smith-Watson-Topper life curve exponent
S-N curve	sncurve : NValues	nf	SN_Curve_N_Values	Life values for the S-N curve
	sncurve : SValues	MPa	SN_Curve_S_Values	Stress amplitude values for the S-N curve
	sncurve: R ratio		R_Ratio_List	List of R-ratios
T-N curve	tn curve: N Vaues	nf	TN_Curve_N_Values	Life values for the T-N curve
	tn curve : T Values	MPa	TN_Curve_T_Values	Torsional stress amplitude values for the T-N curve
	Note that the T-N curve is similar to the S-N curve but applies to torsional loadings. This can be used in algorithms which combine shear and normal or hydrostatic stress in variable weightings (e.g. Prismatic Hull ,TCD Susmel-Lazzarin), or are intended for torsional loadings (e.g. Weld shear methods).			
	Or alternatively the T-N curve family can be derived from S-N values using a constant conversion factor:			
S-N to T-N conversion factor	TN: k	dimensionless	K_SN_TO_TN	Multiplier to convert S-N curve stress to T-N curve amplitude for the same life.
Probability of failure	weibull : MinQMUF		WeibullMin_QMUF	Weibull minimum parameter for probability based analyses
	weibull : SlopeBF		WeibullSlope_BF	Weibull slope parameter for probability based analyses
Walker	walker : gamma@R<0		WALKER_GAMMA_R_POSITIVE	Walker gamma parameter for R≥0
	walker : gamma@R>=0		WALKER_GAMMA_R_NEGATIVE	Walker gamma parameter for R<0
Buch	buch : ProofStress0.2%	MPa	Rp0.2 (MPa)	0.2% proof stress for Buch analysis
Dangvan	dangvan : EnduranceLimitSMax	MPa	DangVanFLS	Dang Van endurance limit stress
	dangvan : R[SMin/Smax]	R	DangVanR	Dang Van stress ratio, R (=Smin/Smax) at endurance limit stress

8.5.3 Local strain parameters

To perform a local strain analysis the following material parameters must be defined:

Category	Displayed name	Units	Keyword	Definition
General	gen : E	MPa	E	Elastic (Young's) Modulus
	gen : UTS	MPa	UTS	Ultimate Tensile Strength
Cyclic strength	css : K'	MPa	K'	Cyclic strain hardening coefficient
	css : n'		n'	Cyclic strain hardening exponent
Strain-life	en : b		b	Basquin's fatigue strength exponent
	en : b2		BASQUIN2	Basquin's fatigue strength exponent at lives above the knee in life curve
	en : c		c	Coffin-Manson fatigue ductility exponent
	en : Ef'		Ef'	Fatigue ductility coefficient
	en : LifeAboveWhichToUseB2	2nf	KNEE_2NF	Knee in life curve (life above which to use b2)
	en : sf'	MPa	sf'	Fatigue strength coefficient

The parameters E , K' and n' are used to define the cyclic stress-strain curve and the hysteresis loops.

The parameters E , sf' , ef' , b , $b2$, $knee_2nf$ and c are used to define the strain-life curve. For the strain-life curve at lives above the specified knee, $b2$ is used instead of b . This facility is provided to allow for kinks in strain-life curves observed in some materials. If you do not have such a material you can set the knee to $1e15$, then $b2$ will not be used.

The Ultimate Tensile Strength, UTS , is used for:

- normal stress analyses using Goodman or Gerber mean-stress correction;

Material properties

- any analysis using a user-defined mean-stress correction (see E7.2);
- any analysis using a Kt value derived from a curve, (see section 5.5.4).

8.5.4 S-N curve parameters

For S-N curve based analyses the following parameters must be defined:

Category	Displayed name	Units	Keyword	Definition
S-N curve	sncurve : Nvalues	nf	SN_Curve_N_Values	Life values for the S-N curve
	sncurve : Svalues	MPa	SN_Curve_S_Values	Stress amplitude values for the S-N curve
	sncurve: R ratio		R_Ratio_List	List of R-ratios

To define an S-N curve for a material, select the required material in the database. Then double click on either the **sn curve : N Values** field, or the **sn curve : S Values** field. This pops-up an editable table for entering S-N data. If multiple temperatures have been entered in the `Temperature_List` field, then the table will have columns for each defined temperature, for example:

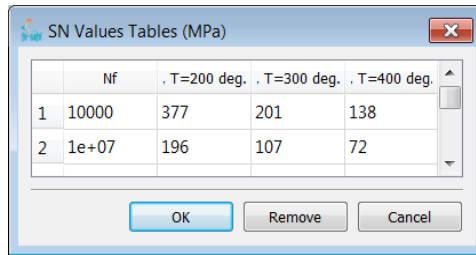


Figure 8-11

Pressing OK, transfers the values to the `sncurve : NValues` and `sncurve : SValues` fields as comma-separated lists. If values are defined for more than one temperature, then the comma separated list of stresses for each temperature are enclosed in brackets. For the above example the following values are transferred:

```
sncurve : NValues (nf) = 1e4, 1e7
sncurve : SValues (MPa) = (377,196) (201,107) (138,72)
```

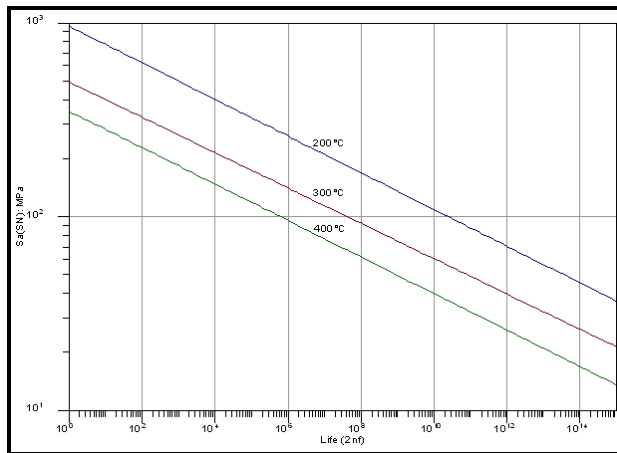


Figure 8-12 S-N curves for three different temperatures.

fe-safe performs a log interpolation or extrapolation so that the curve covers the required life range – see section 8.6.1).

S-N curves can be used for stress-based fatigue analyses (i.e. Stress-life, Normal Stress, Von Mises) by checking the **Use stress-life curve defined using SN datapoints** in the **Analysis Options** dialogue box. If this option is not selected then stress-based analyses will be performed using the elastic-plastic stress-life curve derived from the local strain parameters (see section 8.8)

Multiple S-N curves may be specified for use with different stress ratios (R-ratios). These can then be used to provide a mean stress correction for use with stress-based fatigue analyses (see section 14.11). To define multiple S-N curves for a material first double-click on **s-n curve: R ratio** which will pop-up an editable list of R-ratios.

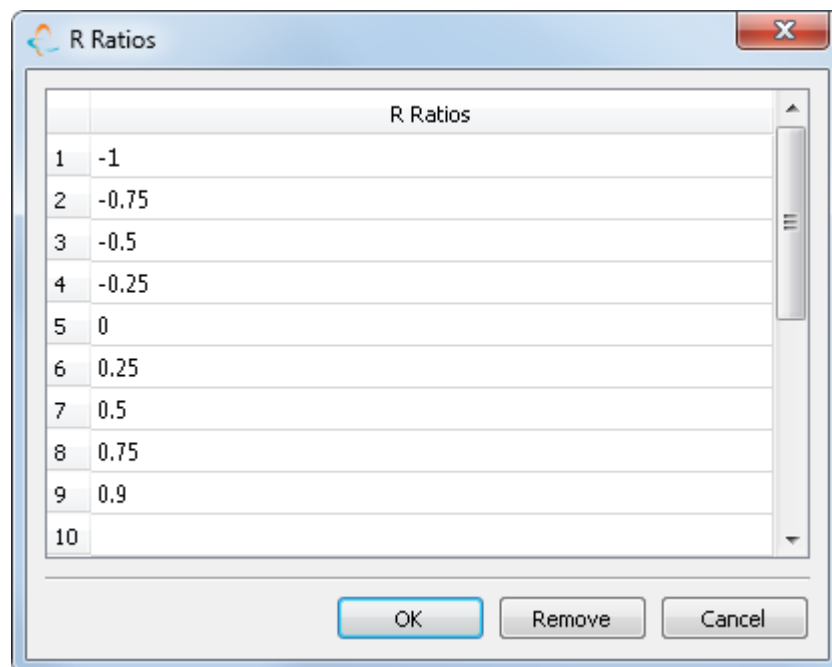


Figure 8-13

Edit the list so that it contains an R-ratio corresponding to each S-N curve to be specified, and then click **OK**.

When either **s-n curve: N Values** or **s-n curve: S Values** is double-clicked an editable table will again appear but now a drop down menu will be available at the top of the window which can be used to select one of the specified R-ratios. Select each R-ratio in turn to specify stress and N values for each, as before it is possible to specify different stress values for different temperatures. When all the values have been entered, click **OK**.

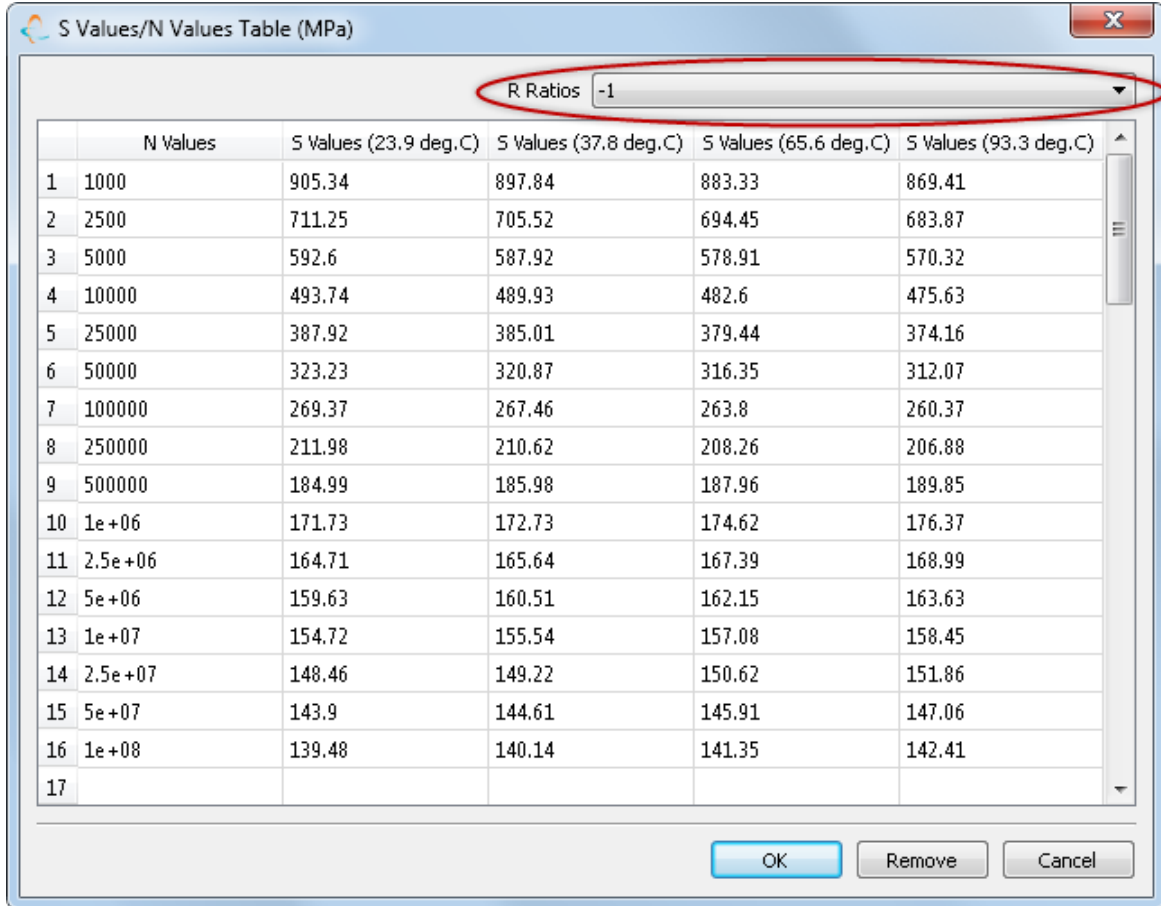


Figure 8-14

When multiple stress ratios have been specified the values will be displayed in the **s-n curve R-ratio** field as a list of comma separated values. Values in the **s-n curve S Values** and **s-n curve: N values** fields are also displayed as comma separated lists with values for each R-ratio contained within square brackets and within those values for different temperatures enclosed in curved brackets (where applicable).

Note that it is possible to specify similar T-N curves derived from torsional loadings. These can be applied in algorithms which combine shear and normal or hydrostatic stress in variable weightings (e.g. Prismatic Hull, or Susmel-Lazzarin within TCD, or Weld shear methods). Some of these only use the torsional endurance limit, not the full T-N curve. The curves may be temperature-dependent in the same way as S-N curves, but it is not possible to specify R-ratio dependent T-N curves, as mean stress effects are usually estimated with respect to normal stress and the S-N curve. Either the full T-N curve may be specified in a similar manner to the S-N curve, or a constant multiplier may be supplied which is used to derive any T-N life value's corresponding torsional stress amplitude by applying this factor to the S-N curve.

T-N curve	tn curve: N Values	nf	TN_Curve_N_Values	Life values for the T-N curve
	tn curve : T Values	MPa	TN_Curve_T_Values	Torsional stress amplitude values for the T-N curve

Or alternatively

T-N to S-N conversion factor	TN:k	dimensionless	K_SN_TO_TN	Multiplier to convert S-N curve stress amplitude to T-N curve amplitude for the same life.
------------------------------	------	---------------	------------	--

If both a T-N curve and a TN:k scaling parameter are specified, then the T-N curve will be used.

8.5.5 Goodman MSC parameters

To perform a Goodman mean-stress correction the material's tensile strength must be defined.

Category	Displayed name	Units	Keyword	Definition
General	gen : UTS	MPa	UTS	Ultimate Tensile Strength

For further details, see section 14.3.

8.5.6 Walker MSC parameters

To perform a Walker mean-stress correction the material's Walker parameter γ must be defined.

Category	Displayed name	Units	Keyword	Definition
Walker	walker : gamma@R<0		WALKER_GAMMA_R_POSITIVE	Walker gamma parameter for R \geq 0
Walker	walker : gamma@R \geq 0		WALKER_GAMMA_R_NEGATIVE	Walker gamma parameter for R<0

As the mean stress effect is usually less prominent in loadings involving compression, separate Walker parameters γ can be defined for tensile (stress ratio R \geq 0) and compressive (stress ratio R<0) loadings.

For further details, see section 14.4.

8.5.7 Alternative Morrow MSC (Morrow B)

When using the alternative Morrow mean stress correction (Morrow B) the true fracture stress must be defined. This is used in place of the fatigue strength coefficient.

Category	Displayed name	Units	Keyword	Definition
General	gen : Sf	MPa	sf	True fracture stress

8.5.8 User defined MSC or FRF curve

The user defined mean stress correction (MSC) function can be used to define a set of correction factors as a function of the mean stress of a cycle, in a similar manner to a Goodman diagram.

Category	Displayed name	Units	Keyword	Definition
General	gen : Default MSC or FRF		DEFAULT_MSC	Default user defined mean stress correction function

Material properties

For further details, see section 14.9.

8.5.9 SN data knock-down curve

The knock-down curve can be used to define scale factors to be applied to scale all stress data points in the SN curve of a material. This option applies to stress-based analyses only.

Category	Displayed name	Units	Keyword	Definition
General	gen : Default Knock-down Curve			User defined knock-down curve

For further details, see section 14.10.

8.5.10 Buch analysis parameters

To perform a Buch analysis the material's 0.2 % proof stress must be defined.

Category	Displayed name	Units	Keyword	Definition
Buch	buch : ProofStress0.2%	MPa	Rp0.2 (MPa)	0.2% proof stress for Buch analysis

For further details, see section 17.2.1 and 14.12.

8.5.11 Cast iron analysis parameters

To perform a cast iron analysis the following material properties must be defined (for both Smith-Watson-Topper and strain-life analyses). These define the tensile and compressive stress-strain and hysteresis loop shapes:

Category	Displayed name	Units	Keyword	Definition
General	gen : E	MPa	E	Elastic (Young's) Modulus
Cyclic strength	css : K'	MPa	K'	Cyclic strain hardening coefficient
	css : n'		n'	Cyclic strain hardening exponent
Cast Iron	gi : K'-Comp	MPa	K_C	Compressive cyclic strain hardening coefficient
	gi : m'-CompSecSlope		M_C	Compressive secant slope
	gi : m'-TensSecSlope		M_T	Tensile secant slope
	gi : Mu-ModulusofUnloading		MU	Modulus of unloading
	gi : n'-Comp		N_C	Compressive cyclic strain hardening exponent

For a local strain analysis the following strain-life parameters must also be defined:

Category	Displayed name	Units	Keyword	Definition
Strain-life	en : b		B	Basquin's fatigue strength exponent
	en : b2		BASQUIN2	Basquin's fatigue strength exponent at lives above the knee in life curve
	en : c		C	Coffin-Manson fatigue ductility exponent
	en : Ef'		Ef'	Fatigue ductility coefficient
	en : LifeAboveWhichToUseB2	2nf	KNEE_2NF	Knee in life curve (life above which to use b2)
	en : sf'	MPa	Sf'	Fatigue strength coefficient

For a Smith-Watson-Topper life analysis the following parameters must also be defined:

Category	Displayed name	Units	Keyword	Definition
	gi : SWTLifeCurveCoeff	MPa	SWT_COEFF	Smith-Watson-Topper life curve coefficient
	gi : SWTLifeCurveExponent		SWT_EXP	Smith-Watson-Topper life curve exponent

For further details, see section 14.19.

8.5.12 Probability based analysis parameters

To perform a probability based analysis (i.e. failure rate or reliability rate for target lives), the following parameters are required:

Category	Displayed name	Units	Keyword	Definition
Probability of failure	weibull : MinQMUF		WeibullMin_QMUF	Weibull minimum parameter for probability based analyses
	weibull : SlopeBF		WeibullSlope_BF	Weibull slope parameter for probability based analyses

For further details, see section 17.4.

8.5.13 Dang Van analysis parameters

To perform a Dang Van analysis, the following parameters are required:

Category	Displayed name	Units	Keyword	Definition
Dangvan	dangvan : EnduranceLimitSMax	MPa	DangVanFLS	Dang Van endurance limit stress
	dangvan : R[SMin/Smax]	R	DangVanR	Dang Van stress ratio, R (=Smin/Smax) at endurance limit stress

Double-clicking on one of these fields displays an editable table for entering pairs of values, for example:

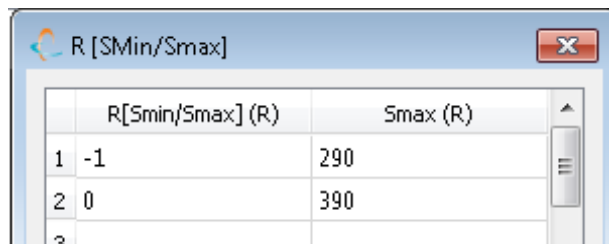


Figure 8-15

These parameters allow a list of Endurance Limit stresses (as maximum stress) and corresponding R values to be defined. For the above example, the endurance stress is 390 MPa for constant amplitude testing at R=0 and the endurance stress is 290 MPa for R=-1.

Pressing **OK** transfers the values to the database fields as a comma-separated list. For the above example the following values are transferred:

```
dang van : Endurance Limit Smax (MPa) = 290, 390
dang van : R:SMin/Smax = -1, 0
```

8.5.14 High temperature fatigue parameters

To define material properties that vary with temperature, local strain material parameters (as defined in section 8.5.3) and S-N curve parameters (as defined in 8.5.4) can be defined for a number of different temperatures.

First define a list of temperatures in the parameter `Temperature_List`. Double clicking on the `Temperature_List` field displays an editable table. Enter the list of temperature values in the table, as shown in the example below:

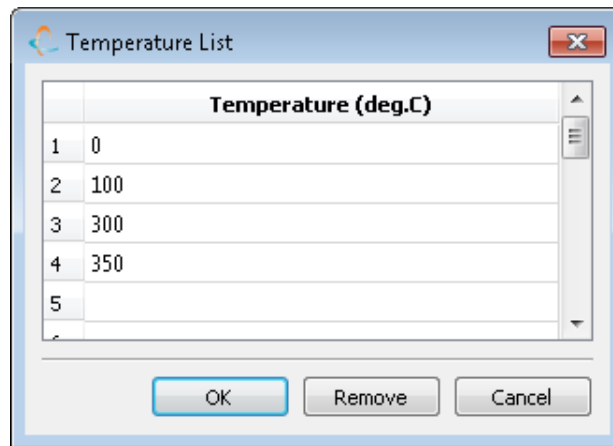


Figure 8-16

Pressing **OK** transfers the values to the `Temperature_List` field as a comma-separated list, i.e.:

0, 100, 300, 350

Once a temperature list has been entered for a material, each of the fatigue variables defined in 8.5.3 and 8.5.4 require multiple values - one for each temperature. Double clicking on one of these fields displays an editable table with the correct number of columns. By default, each value is the same, but these can then be edited where multiple temperature values are known, for example:

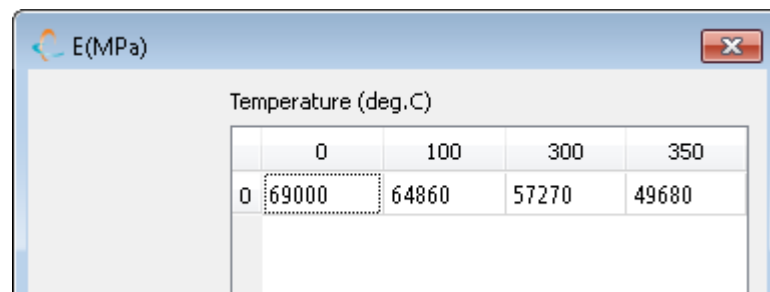


Figure 8-17

Pressing **OK** transfers the values from the table to the selected field, (in this example the Elastic (Young's) Modulus field), as a comma-separated list:

69000, 64860, 57270, 49680

These values correspond to the temperatures defined in the temperature list.

Where multiple temperature data is used, each material parameter is linearly interpolated between data points – see 8.6.3.

To define an S-N curve for each temperature, see section 8.5.4.

8.5.15 BM combined direct and shear stress material parameters

This algorithm requires the SN curve parameters outlined in section 8.5.4, and Youngs modulus.

8.5.16 Critical Distance Method material parameters

The Critical Distance Method (Taylor method) analysis requires one of two extra material parameters. The critical distance (L) may be specified directly using `taylor: L`. If the critical distance is not known it may be calculated from ΔK_{th} (`taylor: Kthreshold@R:-1`), see section 26.6 for more details. Where both parameters have been specified the value for `taylor:L` will be used for the Critical Distance Method.

Category	Displayed name	Units	Keyword	Definition
Taylor	<code>taylor: Kthreshold@R:-1</code>	MPa*m ^{1/2}	KTH	Threshold stress intensity factor at R=-1
	<code>taylor: L</code>	mm	CRIT_DIST	Critical distance

8.6 Interpolation and extrapolation of material data

8.6.1 S-N curve extrapolation

fe-safe allows S-N curves to be defined as pairs of data points, each consisting of stress amplitude Sa, and endurance N cycles.

To calculate fatigue lives, *fe-safe* fits a straight line, on log(Sa) – log(N) axes, between each pair of data points.

The S-N curve is extrapolated to N = 1 cycle using the slope between the two lowest-N pairs of data points, and extrapolated to N = 10¹⁵ cycles using the slope between the two highest-N pairs of data points.

In *Figure 8-18*, the defined S-N curve covers the range from 100 to 1e10 cycles (full line) and the dotted line shows the extrapolation to 1 cycle, and to 1e15 cycles.



Figure 8-18 S-N curve extrapolation

8.6.2 Strain-life data extrapolation

The equations used to calculate the relationship between true stress and endurance

$$\frac{\Delta\sigma}{2} = \sigma'_f (2N_f)^b \quad \text{(equation 3.3 in the Fatigue theory Reference Manual)}$$

and between true strain and endurance,

$$\frac{\Delta\varepsilon}{2} = \frac{\sigma'_f}{E} (2N_f)^b + \varepsilon'_f (2N_f)^c \quad \text{(equation 3.4 in the Fatigue Theory Reference Manual)}$$

cover values of $2N_f$ from 1 to the specified endurance limit endurance, so no extrapolation is necessary.

8.6.3 Material data extrapolation warnings

A warning similar to the following will be shown and logged in the analysis log file if any material data extrapolation past the SN or temperature limits defined in the material properties was necessary for the analysis run:

`* WARNING * : While processing Element 1.1 data has extrapolated to a lower life than defined in the material's SN data.`

When using a knock-down curve the extrapolation warning limits may be adjusted to the knock-down curve life limits if they extend further. To ensure the warning is shown when extrapolating beyond the original SN curve data points, care should be taken to define the knock-down curve within the same limits.

8.6.4 Interpolation of endurance data for temperature-based analyses

S-N Curves

To construct a S-N curve for a specific temperature, *fe-safe* takes S-N curves from the material database for the two temperatures which bracket the required temperature. Using stress amplitude, a new curve is constructed by linear interpolation.

In *Figure 8-19*, if the lower S-N curve represents data at 300 °C and the upper S-N curve represents data at 200 °C, the S-N curve at a temperature of 250 °C will be as shown (dotted line).

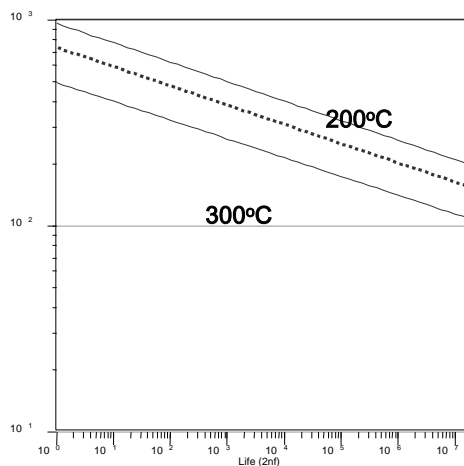


Figure 8-19 S-N curve temperature interpolation

Strain-life data

fe-safe interpolates each of the local stress-strain parameters that are specified as temperature-dependent in the material database. These are

E	the elastic modulus (Young's Modulus)
K'	the strain hardening coefficient
n'	the strain hardening exponent
b	the fatigue strength exponent (Basquin's exponent)
σ'_f	the fatigue strength coefficient
c	the fatigue ductility exponent (the Coffin-Manson exponent)
ϵ'_f	the fatigue ductility coefficient

fe-safe also interpolates the yield stress, the ultimate tensile stress, and the endurance-limit endurance.

The interpolation is linear on each parameter. Beyond the extremes of the lowest and highest temperature the values at the lowest and highest temperatures are used respectively. Each parameter is interpolated independently.

For example, if values of σ'_f are defined for 100°C and 300°C:

the value of σ'_f at 200°C is the (linear) average of the two specified values;

the value of σ'_f at 350°C is the same as the value for 300°C.

8.7 Plotting material properties

The **Material Plot** dialogue is accessed by highlighting the required material database in the **Material Databases** window, then selecting **Generate Material Plot Data...** from the **Material** menu.

Material properties

Materials from the highlighted database are displayed in the **Material Type** drop-down list. A number of different plot options are available. Some plot options are not applicable to all materials – if an option is not applicable it is automatically disabled. Any options, which are checked but disabled, are ignored.

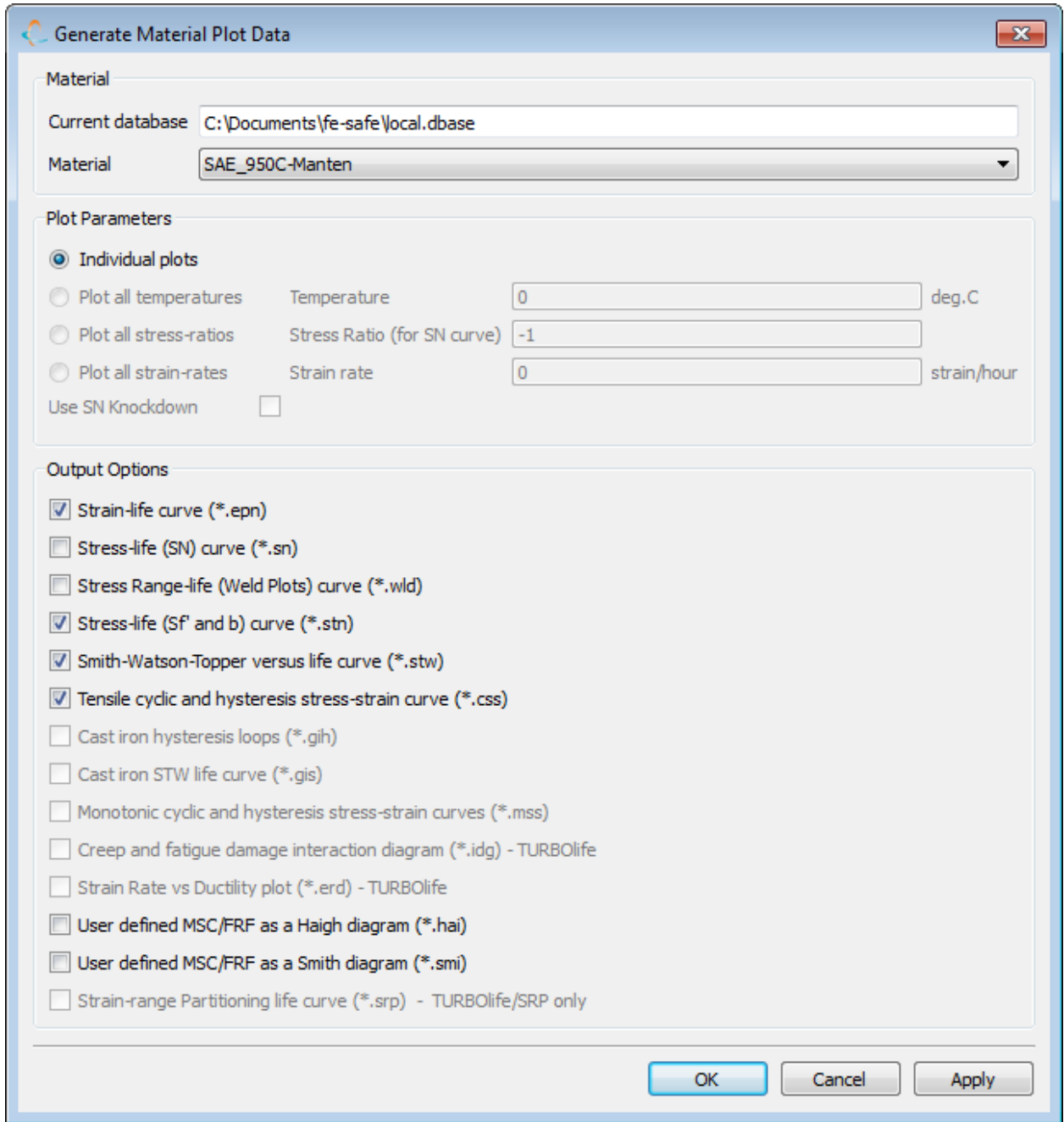


Figure 8-20

For material data defined at multiple temperatures a plot temperature can be defined.

The plot files are added to the **Loaded Data Files** window and can be plotted and overlaid using the plot functions described in section 7.5.

8.7.1 Strain-life curve

This is a plot of strain amplitude versus endurance (as half cycles $2N_f$) to crack initiation.

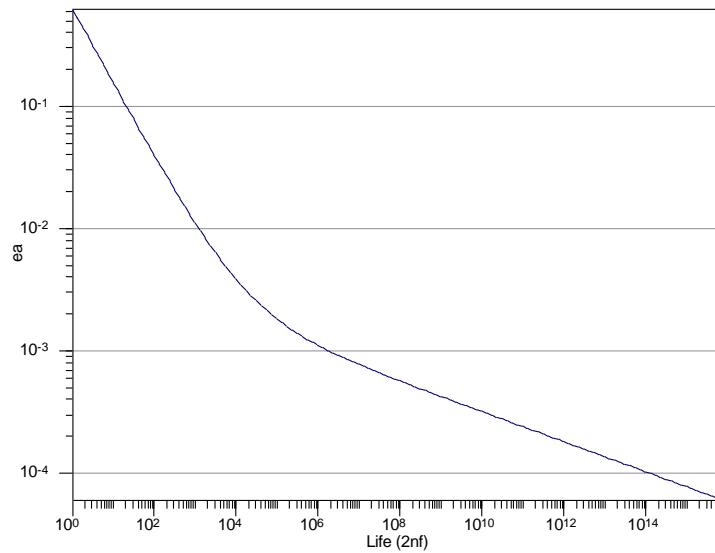


Figure 8-21

The equation defining this curve is:

$$\frac{\Delta \epsilon}{2} = \frac{\sigma_f'}{E} (2N_f)^b + \epsilon_f' (2N_f)^c$$

The strain-life curve can be modified to allow b and hence σ_f' to have different values above a specified life. This is accomplished by defining the life at the knee ($knee=2n.f$), and the value of b above the knee ($b2$). See 8.5.3.

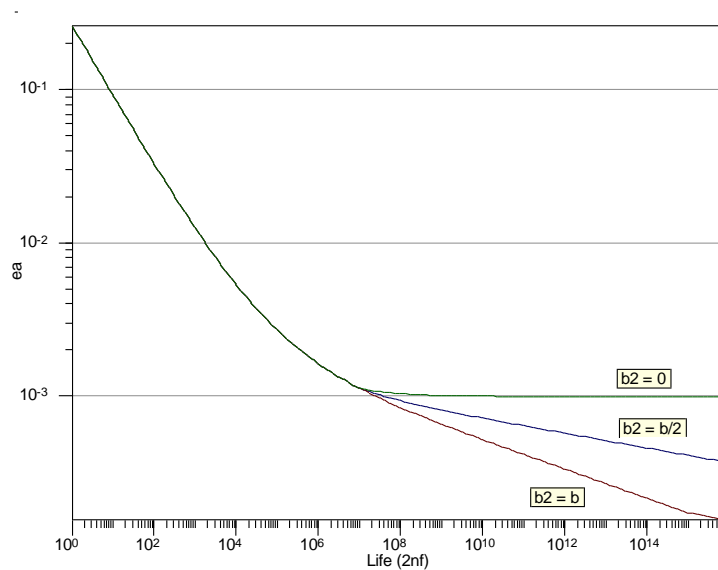


Figure 8-22

In the above figure, the strain-life curves for various settings of $b2$ are shown for a knee in the strain-life curves at an endurance of $2N_f = 10^7$ reversals.

8.7.2 Stress-life curve

This is a plot of stress amplitude versus endurance (as half cycles $2N_f$) to crack initiation. If a knock-down curve is defined for the material its effect on the SN curve can be plotted by selecting the **Use SN Knockdown** checkbox.

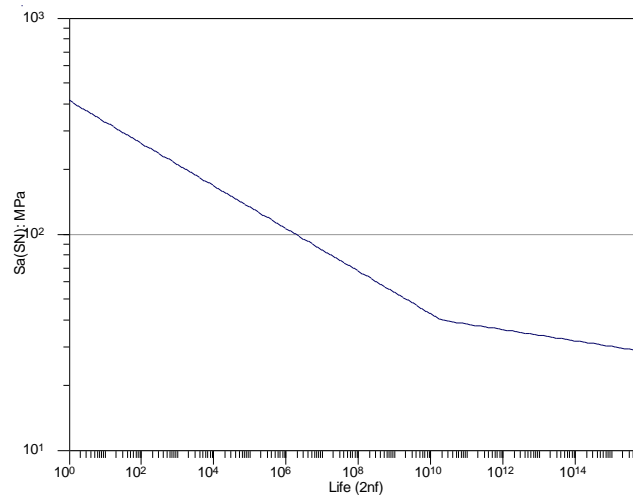


Figure 8-23

The source of this data can be derived from the specified S-N curve, or alternatively from the local strain parameters using the equation:

$$\frac{\Delta\sigma}{2} = \sigma_f' (2N_f)^b$$

fe-safe will select an S-N curve or a $\sigma - 2N_f$ depending on the selecting two options;

FEA Fatigue>>Analysis Options...>>Use stress-life curve defined using SN datapoints

FEA Fatigue>>Analysis Options...>>Use sf' and b if no SN datapoints

An S-N curve will be selected if the **Use stress-life curve defined using SN datapoints** option *is* selected, and an S-N curve is present. This is the only condition for which an S-N curve will be used.

A $\sigma - 2N_f$ curve will be selected if **Use stress-life curve defined using SN datapoints** option *is not* selected.

A $\sigma - 2N_f$ curve will be selected if **Use stress-life curve defined using SN datapoints** option *is* selected, but there is no S-N curve present, and the **Use sf' and b if no SN datapoints** check box *is* selected. This means that the user requested an S-N analysis, but as no S-N curve was present *fe-safe* selected a $\sigma - 2N_f$ curve instead.

If the **Use stress-life curve defined using SN datapoints** option is selected, and the **Use sf' and b if no SN datapoints** option *is not* selected, and there is no S-N data present, *fe-safe* will not start the analysis, and will display a warning.

If S-N data is used then the label on the material's data plot is **sa (SN)** (as in the above figure). If local strain data is used the label is **sa (Mat)**.

Note: S-N data is entered in the material database as Stress amplitude (S) versus endurance N_f cycles. It is always plotted as Stress amplitude (S) versus endurance $2N_f$ half-cycles.

8.7.3 Smith-Watson-Topper life curve

This is a plot of the Smith-Watson-Topper parameter versus endurance (as half cycles $2N_f$).

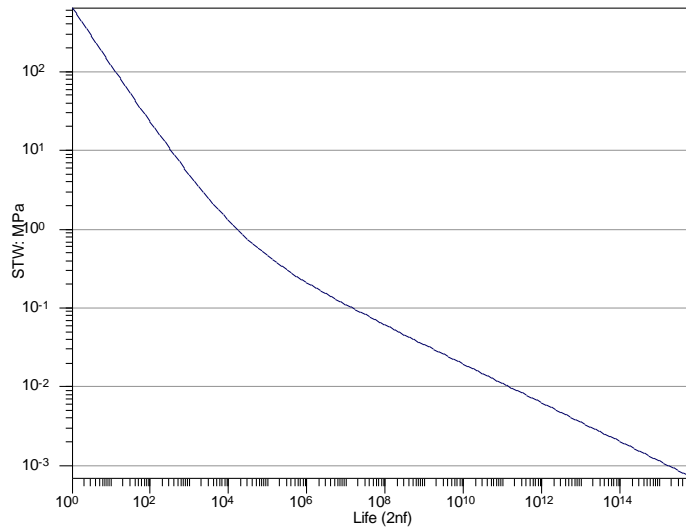


Figure 8-24

The equation defining the SWT parameter life curve is:

$$\frac{\Delta \varepsilon}{2} \sigma_{\max} = \frac{(\sigma_f')^2}{E} (2N_f)^{2b} + \sigma_f' \varepsilon_f' (2N_f)^{b+c}$$

8.7.4 Cyclic and hysteresis loop ('Twice cyclic stress-strain') curves

These are plots of the stable cyclic stress-strain curve and the stable hysteresis loop curve.

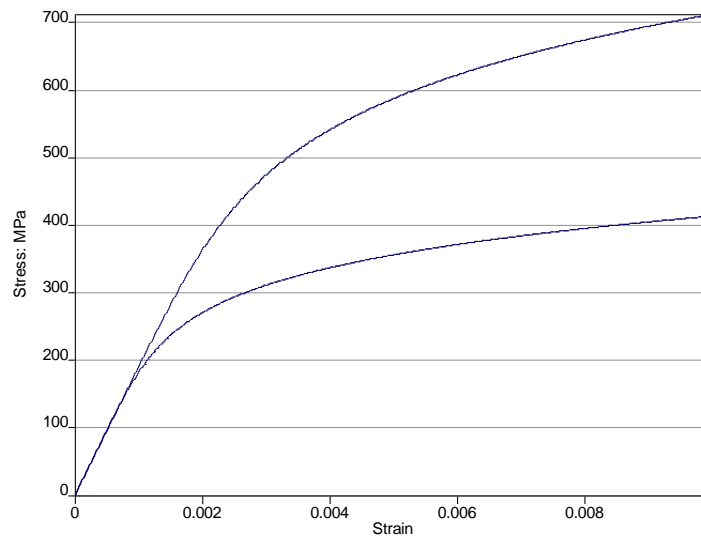


Figure 8-25

The equations for these curves are:

Cyclic stress-strain

$$\varepsilon = \frac{\sigma}{E} + \left(\frac{\sigma}{K'}\right)^{\frac{1}{n'}}$$

Hysteresis loop

$$\Delta\varepsilon = \frac{\Delta\sigma}{E} + 2\left(\frac{\Delta\sigma}{2K'}\right)^{\frac{1}{n'}}$$

8.7.5 Cast iron hysteresis loops

Hysteresis loops and cyclic stress-strain curves for cast iron are not symmetrical. To show the hysteresis loop shape, together with the individual contributions of the bulk matrix, the graphite morphology and surface crack closure, *fe-safe* provides plots as shown below. The hysteresis loop is always plotted for a cycle of +/- 3000 micro-strain.

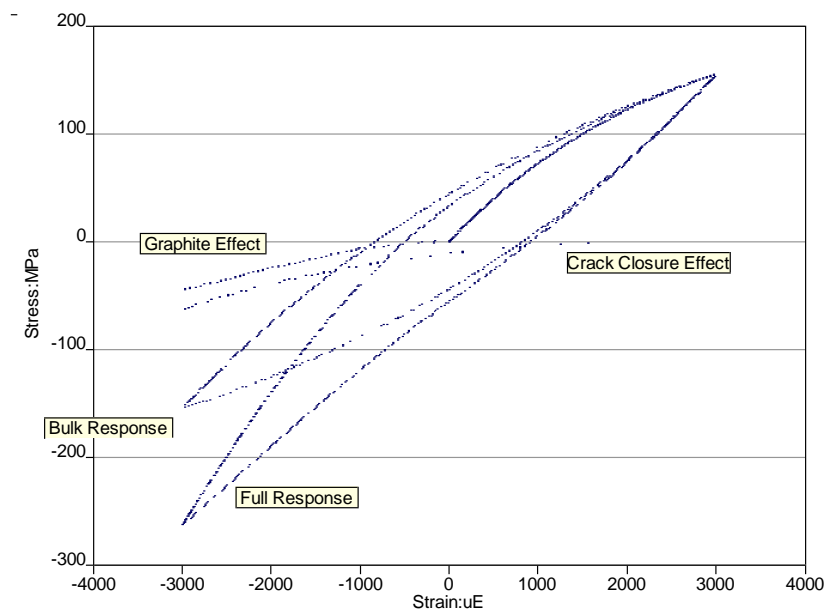


Figure 8-26

More details of the equations used in this calculation are provided in the cast iron technical background in section 14.19

8.7.6 Cast iron SWT life curve

This is a plot of the relationship between the SWT parameter and endurance (as $2N_f$ half-cycles)

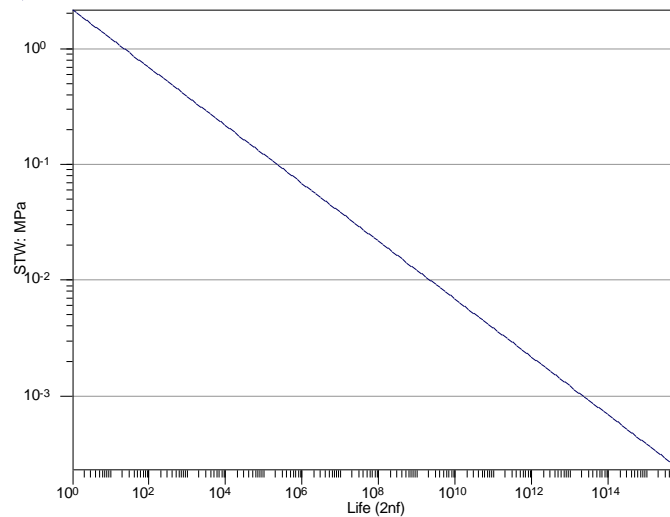


Figure 8-27

The equation for this curve is:

$$\sigma_{\max} \varepsilon_a = STW_coeff (N_f)^{SWT_Exp}$$

See section 14.19 for more details of the equations used for fatigue analysis of cast irons.

8.8 Importing and exporting material data using text files.

A material can be exported to a text file using the **Material** menu item **Export Material to Text File**. The user is prompted for the name of the text file to create.

The text file contains a header marker indicating that this is a material file.

All comments are marked with a #.

An empty line indicates the end of a table.

Where a parameter has the value -9999 it is not defined.

An example of an exported material file is shown below. Each parameter has a name (in italics) followed by the data, editing the parameter name will prevent *fe-safe* associating the data with the correct parameter on re-importing.

```
DASSAULT SYSTEMES MATERIAL DEFINITION FILE
```

```
# NOTES-9999 indicates a parameter is not set!
```

```
#           All items after a # are comments that will be ignored.
```

```
DEFAULT_MSC
```

Material properties

```
# Default MSC or FRF
"C:/SIMULIA/fe-safe/2017_232/database/goodman.msc"

STANDARD_&_GRADE
# BSName
SAE_950C-Manten

Material_Class
# Material Class
Steel (Ductile)

MATL_ALGORITHM
# Algorithm
BrownMiller:-Morrow

MATL_UNITS
# Materials Units
Use system default

Data
# Data_Quality
Use only as an example; Kth; Sbw and Tw have notional values

Comment1
# Comment1
c:\material_data\manten_ref1.html

Revision_Number
# Revision Number
2

Revision_Date
# Revision Date
Wed Jun 10 08:24:28 2015

Revision_History
# Revision History
SN curve modified at v5.01-01

WeibullSlope_BF
# Slope BF
3

WeibullMin_QMUF
# Min QMUF
0.25

TAYLOR_KTH
# Kthreshold@R
5

gi_index
# Grey Iron Index
None

TempList
```

```
# Temperature List
0

StrainRateList
# StrainRateList
0

HoursList
# Hours List
0 1

SN_Curve_N_Values
# N Values
1e4 1e7

CPF_TW
# Tw
325

CPF_SBW
# Sbw
325

Const_Amp_Endurance_Limit
# Const Amp Endurance Limit
2.00E+07

MATL_POISSON
# Poissons Ratio
0.33

E
# E
203000

Rp0.2 (MPa)
# Proof Stress 0.2%
325

UTS
# UTS
400

K'
# K'
1190

n'
# n'
0.193

Ef'
# Ef'
0.26

c
```

Material properties

```
# c
-0.47

sf'
# sf'
930

b
# b
-0.095

PreSoakFactor
# Pre Soak Factor
1

SN_Curve_S_Values
# S Values
363.0 188.3
```

A material can be imported from a text file using the **Material** menu item **Import Material from Text File**. The user is prompted for the name of the text file to import. The material's name is extracted from the MATERIAL-NAME field. If a material of the name already exists the opened material test file will be archived with the time and date as shown in *Figure 8-28*.

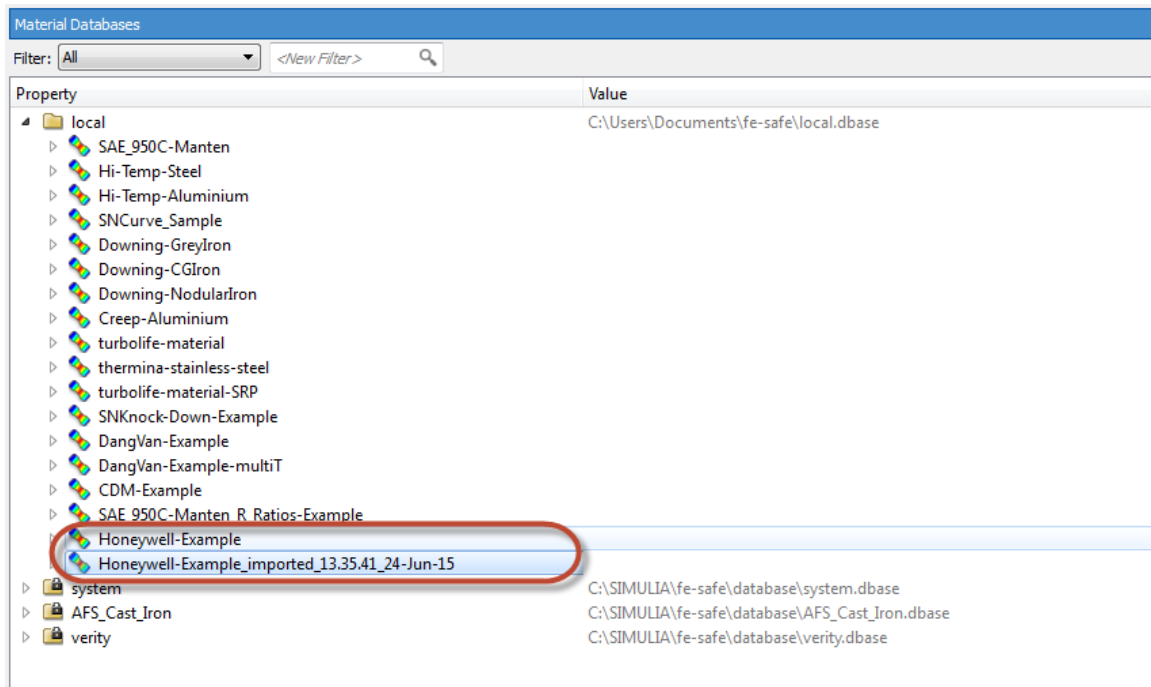


Figure 8-28

8.9 Importing and exporting material data using .dbase files.

From *fe-safe* version 6.5 onwards, the syntax used to define material data when importing and exporting individual materials as a text file was changed in a way that was not compatible with previous releases of *fe-safe*.

To import these older defined materials it is suggested that they are imported as part of the .dbase file that they are contained within. Additional unwanted material entries resulting from this process can then be removed within the *fe-safe* Material Database manager.

1. Locate the .dbase file that the old version of *fe-safe* was using to define the material (typically found in an *fe-safe.** folder, which is called the user directory or <UserDir>. By default this is located within the user's My Documents or home directory.).
2. Create a copy of this file, in a location associated with the newer version of *fe-safe*, and rename it so that it is identifiable (e.g. \Documents\fe-safe.version.6.4\local.dbase becomes \Documents\fe-safe.2016\custom.dbase)
3. Within the new version of *fe-safe*, right-click in the Material Databases window and choose **Open Database**

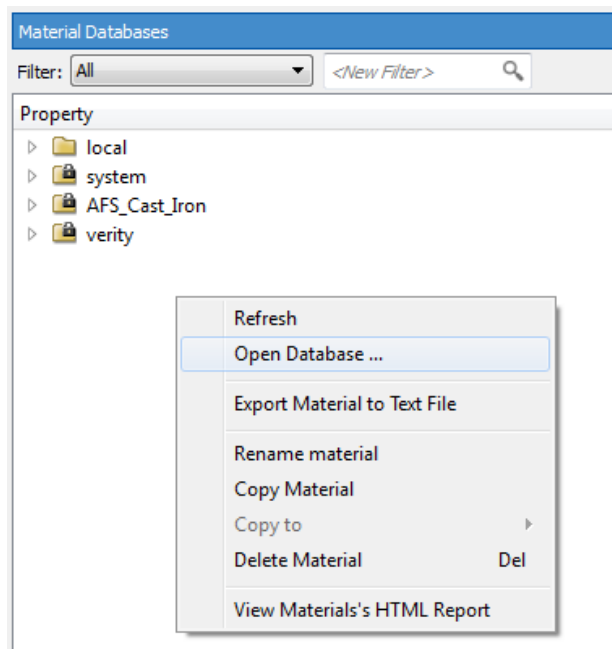


Figure 8-29

4. Browse to the new .dbase file that was prepared in step 2, and click the Open button

Material properties

- The .dbase file is now shown in the Material Databases window, click the arrow to the left of the folder to expand the display to show all of the materials

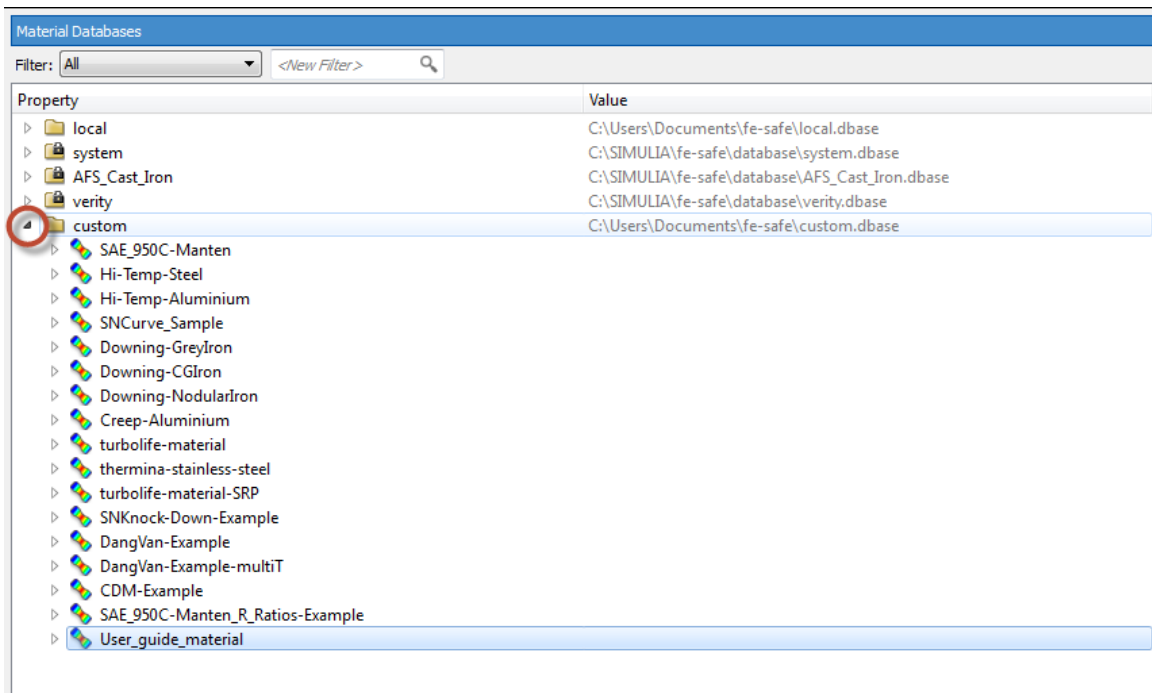


Figure 8-30

6. A database may be imported with additional materials that are not required to be retained (e.g. duplicates of materials provided by the local.dbase packaged with the installation of the new version of *fe-safe*). The surplus materials can be removed by selecting them in the Material Database window and either right-clicking on them and selecting Delete, or by pressing the Delete key. As the process cannot be undone, ensure that you have the correct material/database selected before confirming the delete operation with the Yes button on the prompt.

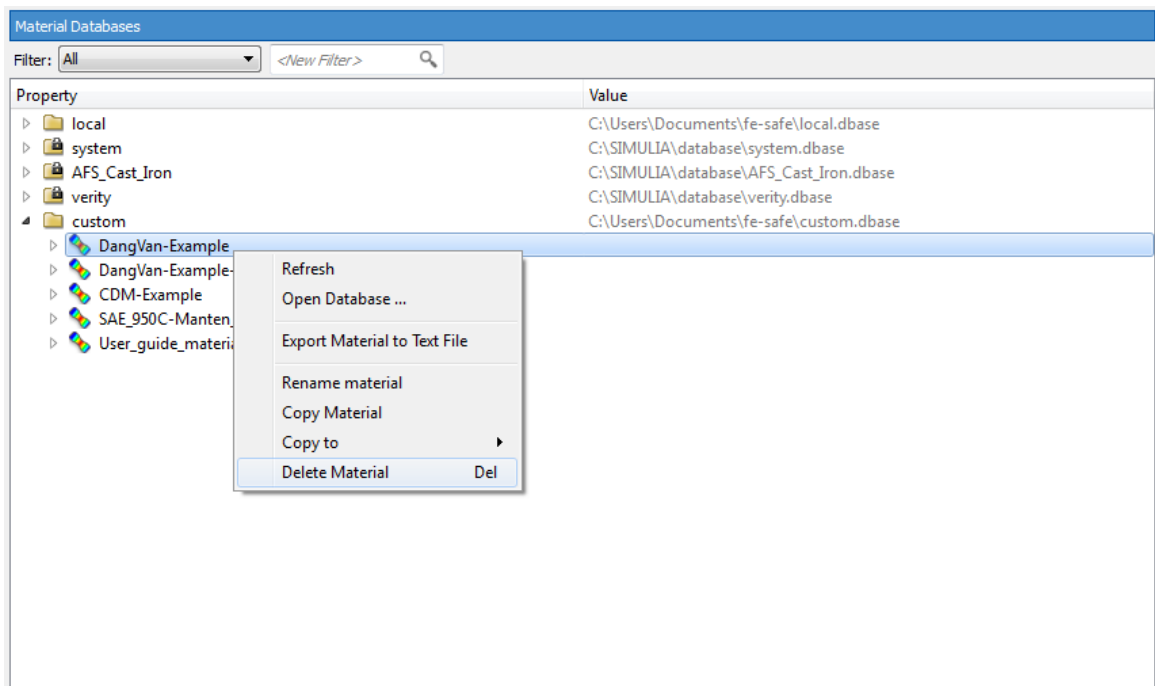


Figure 8-31

8.10 References

- 8.1 ASME NH, ASME Boiler and Pressure Vessel Code, Division 1, Subsection NH, Class 1 Components in Elevated Temperature Service, 2001.
- 8.2 Halford, G, R and Manson, S, S, Application of a Method of Estimating High-Temperature Low-Cycle Fatigue Behaviour of Materials, Transactions of the ASME, Vol. 61, 1968.
- 8.3 Ainsworth, R, A, Budden, P, J, O'Donnell, M, P, Tipping, D, J, Goodall, I, W and Hooton, D, J, 2001. Creep-Fatigue Crack Initiation Assessment Procedures, SMIRT 16, 2001, F04/2.

9 Signal generation methods

The following functions are available from the **Generation** menu option.

9.1 Signal Generation (Sine Wave, White Noise)...

This function generates time history files by superimposing defined sine waves and Gaussian white noise signals. Selecting **Generation >> Signal Generation (Sine Wave, White Noise)...** displays the dialogue box in *Figure 9.1-1*, below:

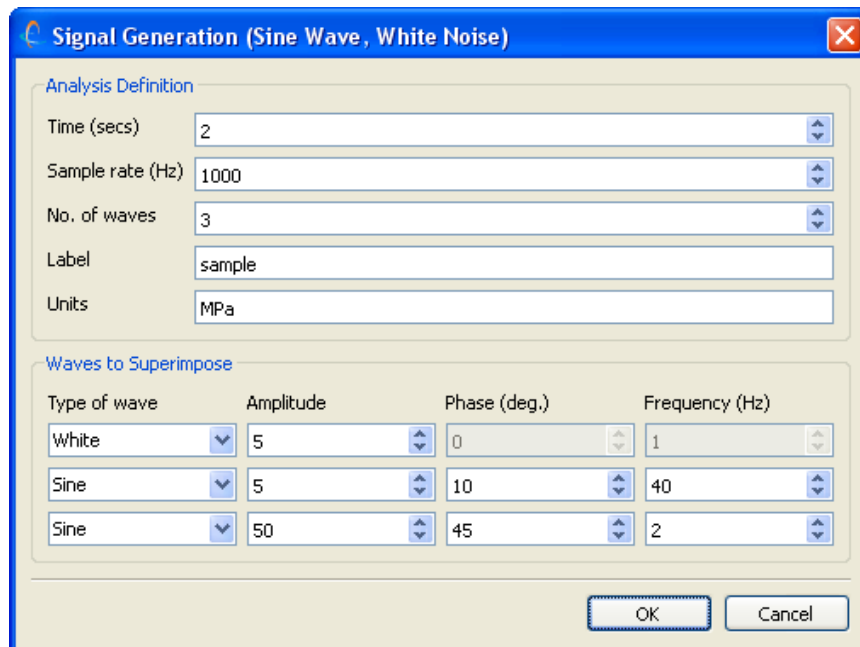


Figure 9.1-1

Example:

Setting the parameters shown in *Figure 9.1-1* superimposes two sine waves and one white noise signal, as defined. The resultant signal is shown in *Figure 9.1-2*, below:

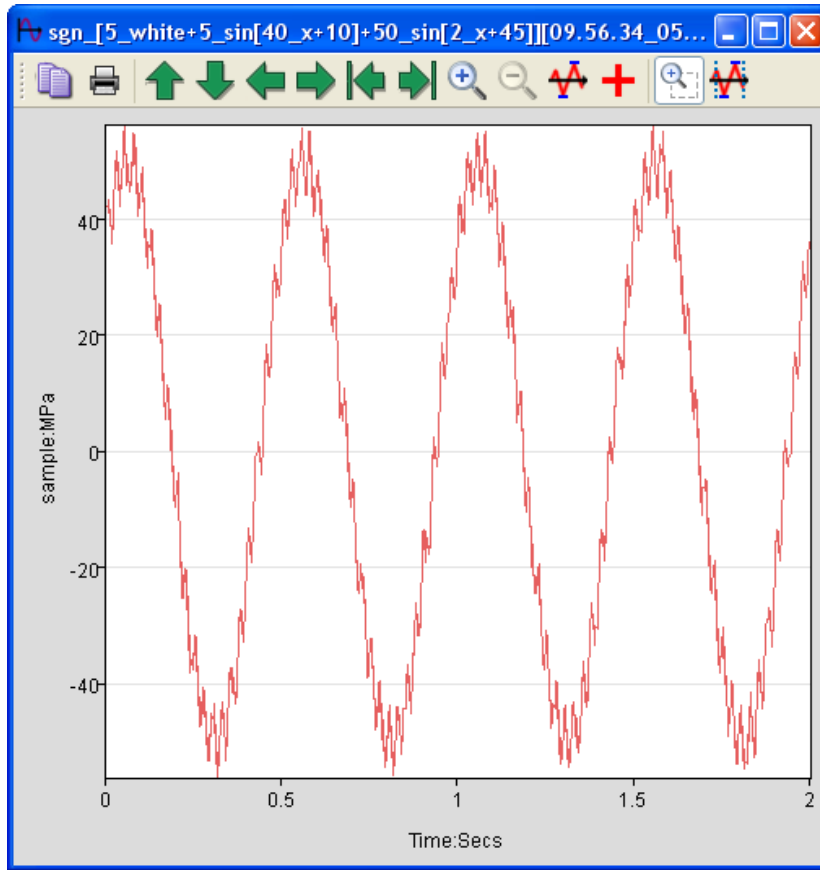


Figure 9.1-2

Note that for the sine wave function, the specified amplitude is the amplitude of the generated sine wave, whilst for the white noise function the amplitude refers to the rms. amplitude of the generated Gaussian white noise.

The output signal is written to a DAC format file, and the results added to the **Loaded Data Files** list. Subsequent handling of the file (for example plotting, analysis, saving the results as an ASCII file) is discussed in section 7.

9.2 Reconstruction of time histories from peak-valley data

This function reconstructs time history data from peak-valley files, and is particularly useful for generating drive signals appropriate for the dynamics of a particular test rig. Selecting **Generation >> Generate Time History from Peak-Valley...** displays the dialogue box in *Figure 9.2-1*, below:

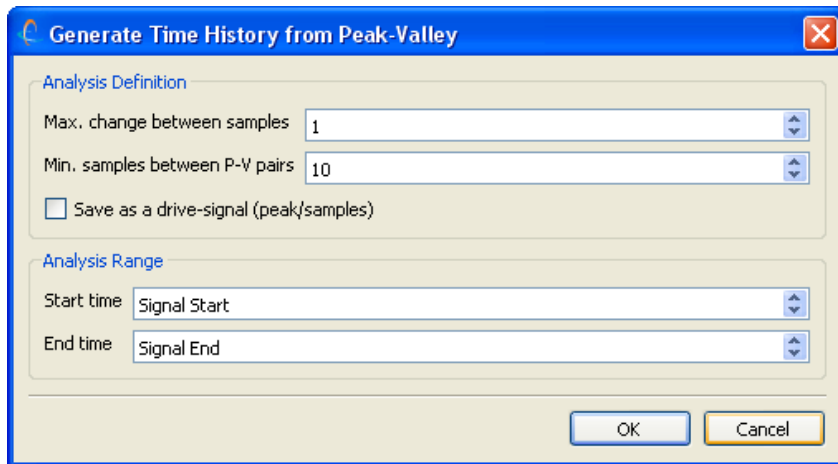


Figure 9.2-1

The function takes a sequence of peak/valleys. A half-cosine is fitted between each peak and valley, by inserting intermediate data points.

The following parameters can be defined:

- the maximum change in value between any two data points (to control the ramp rate);
- the minimum number of data points to be inserted between each peak-valley pair – (to maintain the shape of the cosine curve).

The output signal is written to a DAC format file, and the results added to the **Loaded Data Files** list. Subsequent handling of the file (for example plotting, analysis, saving the results as an ASCII file) is discussed in section 7.

10 Signal processing methods

10.1 Signal processing in *fe-safe* and *safe4fatigue*

fe-safe includes a comprehensive set of signal processing functions. These are available in both *fe-safe* and *safe4fatigue*. In addition to general-purpose signal manipulation and signal processing tools, *fe-safe* includes functions of particular use in preparing fatigue loading data, for example the multi-channel peak-valley analysis functions.

The Signal Processing functions discussed in this section are all accessed from the **Amplitude** and **Frequency** menu options. Functions operate on one or more input signals from the **Loaded Data Files** list. A full discussion on signal handling, including file handling, signal editing, signal plotting, etc. can be found in section 7.

10.2 Signal processing function summary

Function	Section	Function input			Input Options			Function output				
		Sequential data format	Histogram data format	Number of input signals (see Note 2)	Analysis start/end time can be specified	Data label and units can be specified	Other options	Output is a single file pertaining to all inputs	Output is a point-by-point function of input	Output	File extension (see Note 1)	File format
Amplitude												
Differentiate	10.3.1	any sequential file type		multi	•	•	Polynomial order (between data points): - 1 st order - 3 rd order		•	Differentiation of the input.	.dif	DAC (S)
Integrate	10.3.2	any sequential file type		multi	•	•	Integration order: - #1 order (Trapezoidal rule) - #2 order (Simpson's rule) - #3 order (3/8th rule) Optional integration constant		•	Definite integral of the input.	.int	DAC (S)
Mathematical functions	10.3.3	any sequential file type	range-mean histogram	multi	•	•	Mathematical function: - SIN (sine) - COS (cosine) - TAN (tangent) - ASIN (inverse sine) - ACOS (inverse cosine) - ATAN (inverse tangent) - LOG (common logarithm, ie. log10) - 10^X (exponential function base 10) - LN (natural logarithm, i.e. loge) - EXP or e^X (exponential function) - PI (multiplies input by π (p))		•	Result of the selected mathematical function.	.mth	DAC (S) or DAC (H)
Scale and offset	10.3.4	any sequential file type	range-mean histogram (see Note 3)	multi	•	•	Input constants m, c ₁ , c ₂ and r must be specified		•	Linear and non-linear scaling of the input – see 10.3.4, below.	.dac	DAC (S) or DAC (H)
Multiply, divide, add or subtract two signals	10.3.5	any sequential file type		2	•	•	Operator: - add (+) - subtract (-) - multiply (x) - divide (÷)	• see Note 4	•	Result of the selected operation.	.dac	DAC (S)
Concatenate multiple signals	10.3.6	any sequential file type		multi	•	•		•		Concatenation of all selected input files in the order they were selected.	.dac	DAC (S)

Signal processing methods

Function	Section	Function input			Input Options			Function output				
		Sequential data format	Histogram data format	Number of input signals (see Note 2)	Analysis start/end time can be specified	Data label and units can be specified	Other options	Output is a single file pertaining to all inputs	Output is a point-by-point function of input	Output	File extension (see Note 1)	File format
Spike analysis	10.3.7	any sequential file type		multi	•		Number of bins			Spike content as a rise-time distribution histogram.	.rtd	DAC (S)
Spike removal	10.3.8	any sequential file type		multi	•		Maximum permissible rise			Spike filtered signal – see 10.3.8, below.	.dac	DAC (S)
Time at level	10.3.9	any sequential file type		multi	•		Number of bins Lower bin edge Upper bin edge			Time-at-level diagram. Normalised probability density diagram.	.tal .pdd	DAC (S) DAC (S)
Joint time at level	10.3.10	any two sequential files of any file type		2	•		Number of bins Lower bin edge - signal 1 Upper bin edge - signal 1 Lower bin edge - signal 2 Upper bin edge - signal 2	•		Time-at-level 3D histogram. Probability density 3D histogram.	.tah .pdh	DAC (S) DAC (H)
Level crossing from time histories	10.3.11	any sequential file type		multi	•		Number of bins Lower bin edge Upper bin edge			Level-crossing distribution diagram.	.lca	DAC (S)
Level crossing from range-mean histograms	10.3.12		range-mean histogram (see Note 3)	multi			(No dialogue)			Level-crossing distribution diagram.	.lca	DAC (S)
Rainflow (and cycle exceedence) from time histories	10.3.13	any sequential file type		multi	•		Gate (as % of range) Number of bins: - 2 to 64 Lower bin edge Upper bin edge Output options: - range-mean histogram - range-only histogram - cycle exceedence diagram - cycle density diagram			Range-mean rainflow cycle histogram. Range-only rainflow cycle histogram. Cycle exceedence diagram. Cycle density diagram.	.cyh .cyr .cyx .cyd	DAC (S) DAC (S) DAC (XY) DAC (S)
Rainflow histogram from PSD	10.3.14	any sequential file type containing PSD information (see Note 5)		multi			Length of signal (seconds) Gate level (as % of range) – disabled in this release			Range-mean rainflow cycle histogram.	.cyh	DAC (H)
Range-pair and cycle density from level crossing	10.3.15	any sequential file type containing level crossing information (see Note 6)					(No dialogue)			Range-pair (cycle) distribution diagram. Cycle density distribution diagram.	.rp .cyd	DAC (XY) DAC (XY)
Convert rainflow to LDF for FEA fatigue	10.3.16		range-mean histogram (see Note 3)	multi			Bin Edge - centre - upper Dataset number	•		A fatigue loading definition (LDF file), for the defined dataset number, for use in an FE-based fatigue analysis – see 10.3.16, below.	.ldf	LDF

Function	Section	Function input			Input Options			Function output				
		Sequential data format	Histogram data format	Number of input signals (see Note 2)	Analysis start/end time can be specified	Data label and units can be specified	Other options	Output is a single file pertaining to all inputs	Output is a point-by-point function of input	Output	File extension (see Note 1)	File format
Peak-valley (and P-V exceedence)	10.3.17	any sequential file type		multi	•		Gate level (as engineering units) Add time information as an additional results signal Output options: - save peak-valley results - save peak-valley exceedence plot			Peak-valley file. Peak-valley exceedence diagram.	.pv .xcd	DAC (S) DAC (Hyst)
Multi-channel peak-valley	10.3.18	two or more sequential files of any sequential file type (see Note 4)		2 or more	•		Gate level : - as engineering units - as % of range Add time information as an additional results signal Add inverse time increment (drive signal) as an additional results signal			One peak-valley file for each input file. Time stamp channel (if requested). Drive signal (if requested).	.pv .pv .pv	DAC (S) DAC (S) DAC (S)
Markov from-to matrix	10.3.19	any sequential file type			•		Gate (as % of range) Number of bins: - 2 to 64 Lower bin edge Upper bin edge Output options: - range-mean histogram - range-only histogram - to-from matrix diagram			Markov range-mean cycle histogram. Markov range-only cycle histogram. Markov "from-to" matrix.	.rgm .rvc .t.fm	DAC (H) DAC (S) DAC (H)
Transform 2D strain tensor to 45° rosette data	10.3.20	any 3 sequential signals of any file type, containing 2D tensor data (xx, yy, xy) (see Note 4)		3	•				•	Three sequential output files corresponding point-by-point with the input channels, containing: - Rosette data 0° - Rosette data 45° - Rosettes data 90°	.dac .dac .dac	DAC (S) DAC (S) DAC (S)
Transform 3D tensor to new orthogonal axes	10.3.21	any 6 sequential signals of any file type containing a 3D tensor (xx, yy, zz, xy, yz, zx) (see Note 4)		6	•		Angle theta (degrees) - angle of rotation of the z-axis towards the x-axis, in the x-y plane. Angle phi (degrees) - angle of rotation of the x-axis towards the y-axis in the x-y plane.		•	Six sequential output files corresponding point-by-point with the input channels, containing: - xx → xx' - yy → yy' - zz → zz' - xy → xy' - yz → yz' - zx → zx'	.dac .dac .dac .dac .dac .dac	DAC (S) DAC (S) DAC (S) DAC (S) DAC (S) DAC (S)
Transform XYZ vectors onto new axis	10.3.22	any 3 sequential signals of any file type, containing an		3	•		Transformation method: absolute: i.e. {x,y,z} → {x',y',z'}, using Euler angles; relative: i.e. pitch → roll → yaw		•	Three sequential output files corresponding point-by-point with the input channels, containing: - x → x'	.dac .dac	DAC (S)

Signal processing methods

Function	Section	Function input			Input Options			Function output				
		Sequential data format	Histogram data format	Number of input signals (see Note 2)	Analysis start/end time can be specified	Data label and units can be specified	Other options	Output is a single file pertaining to all inputs	Output is a point-by-point function of input	Output	File extension (see Note 1)	File format
		orthogonal set of vectors in x, y and z. (see Note 4)					Forward or reverse transform. Transformation angles: θ , ϕ and φ , in degrees.			- $y \rightarrow y'$ - $z \rightarrow z'$.dac	DAC (S) DAC (S)
Statistics	10.3.23	any sequential file type		multi	•					A statistical summary for all selected signals is written to an ASCII text file and is also displayed in a dialogue box. The name of the text file is displayed in the message log window.	.txt	ASCII
Frequency												
Power spectral density	10.4.2	any sequential file type		multi	•		FFT buffer size – a whole power of 2, between 32 and 2048. Buffer overlap (%). Normalise analysis. Peak hold.			Power spectral density (PSD) distribution.	.psd	DAC (S)
Cross-spectral density	10.4.3	any two sequential signals of any sequential file type		2	•		FFT buffer size – a whole power of 2, between 32 and 2048. Buffer overlap (%). Normalise analysis.			Power spectral density (PSD) distribution. Cross-spectral density (CSD) distribution. Gain diagram. Phase diagram. Coherence diagram.	.psd .csd .gai .pha .coh	DAC (S) DAC (S) DAC (S) DAC (S) DAC (S)
Transfer function	10.4.4	any two sequential signals of any sequential file type		2	•		FFT buffer size (a whole power of 2, between 32 and 2048). Buffer overlap (%). Normalise analysis.					
Filtering												
Butterworth filtering	10.5.2	any sequential file type		multi	•		Filter type: - low-pass - high-pass - band-pass Lower cut-off frequency (Hz) Upper cut-off frequency (Hz) Filter-order: - #1 order - 6dB/Octave - #2 order - 12dB/Octave - #3 order - 18 dB/Octave Pass-region gain (dB)			Filtered signal.	.dac	DAC (S)
FFT filtering	10.5.3	any sequential file type		multi	•		Definition of up to ten sets of filter coefficients, where each set includes: - passband region gain (dB) - lower cut-off frequency (Hz)			Filtered signal	.dac	DAC (S)

Function	Section	Function input			Input Options			Function output				
		Sequential data format	Histogram data format	Number of input signals (see Note 2)	Analysis start/end time can be specified	Data label and units can be specified	Other options	Output is a single file pertaining to all inputs	Output is a point-by-point function of input	Output	File extension (see Note 1)	File format
							- upper cut-off frequency (Hz) Filter order. Filter definitions can be saved, loaded and plotted.					

Note 1: The following descriptors refer to files using the industry standard DAC format - see Appendix E, 205.2.1.

- DAC (S) : a single channel sequential file;
- DAC (H) : a histogram file;
- DAC (XY) : an XY data file;
- DAC (Hyst) : an XY data file containing hysteresis loops.

Note 2: Some files require a specified number of input channels. "multi" implies that the function can be applied to multiple sequential input files of mixed formats.

Note 3: The function can be applied to multiple histogram input files.

Note 4: Input files must be of the same length (i.e. contain the same number of data points). If the number of data points is different, then all input signals are cropped to the same length as the shortest signal.

Note 5: The input file can be of any sequential file type, but must contain PSD information. PSD information produced using one of the frequency-domain algorithms (see 10.4) will be in DAC[S] format, and have the extension `.psd`.

Note 6: The input file can be of any sequential file type, but must contain level crossing information. Level crossing information produced using one of the Level Crossing analysis functions (see 10.3.11 and 10.3.12) will be in DAC[S] format, and have the extension `.lca`.

10.3 Amplitude methods

10.3.1 Differentiate

This function calculates the derivative of the input using a first or third-order polynomial.

10.3.2 Integrate

This function calculates the definite integral of the input using one of the following methods:

a) Trapezoidal rule (1st order): $\int x_k \cdot dt = \frac{x_{k+1} + x_k}{2} \cdot dt$

b) Simpson's rule (2nd order): $\int x_k \cdot dt = \frac{x_{k+2} + 4x_{k+1} + x_k}{3} \cdot dt$

c) Simpson's 3/8th rule (3rd order): $\int x_k \cdot dt = \frac{x_{k+3} + 3x_{k+2} + 3x_{k+1} + x_k}{8} \cdot 3dt$

where

x_k is the k^{th} input data point, and

dt is the interval between data points:

Limitation

The integration of long data files should be avoided, as even a very small non-zero mean value will cause the output values to diverge. This effect can be minimised by calculating the mean value of the input file (using the Statistical Analysis module – see 10.2.23), and subtracting it from each data point (using the Scale and Offset module – see 10.2.4) to produce a mean value which is close to zero.

10.3.3 Mathematical functions

This function applies one of the following mathematical functions to a sequential data file:

- SIN * - sine
- COS * - cosine
- TAN * - tangent
- ASIN * - inverse sine
- ACOS * - inverse cosine
- ATAN* - inverse tangent
- LOG - common logarithm, i.e. log10
- 10^X - exponential function base 10
- LN - natural logarithm, i.e. loge
- EXP or e^X - exponential function
- PI - multiplies input by π (pi)

* input/output may be in degrees or radians

Limitations

The limitations inherent in the individual mathematical functions apply to the software. For example, log10 values cannot be obtained for negative numbers in signals or results files.

10.3.4 Scale and offset

This function performs a linear or non-linear scaling of the input, using:

$$y = m (x + c_1)^r + c_2$$

where:

x is the input

y is the output

m , c_1 , c_2 and r are input constants.

Checks are made to ensure the integrity of the scaling operation. An initial check verifies that the scaling parameters will not cause output values to overflow (i.e. become numerically too large for the computer to manipulate). A second check prevents negative values being raised to non-integer powers (an operation which is mathematically undefined).

To distinguish integers from real values in the exponent, r , only the first three decimal places are considered significant. This avoids unnecessary restrictions caused by rounding.

10.3.5 Multiply, divide, add or subtract two signals

This function multiplies, divides, adds or subtracts two sequential input signals point by point. (For example, in the 'add' function, the first output data point is formed by adding the first point in each of the two input signals.)

10.3.6 Concatenate multiple signals

This function concatenates selected sequential input files in the order they were selected, to produce a single sequential output file.

10.3.7 Spike analysis

This function analyses the spike content of a signal by evaluating the difference, or rise, between successive data points and formulating them into a rise-time distribution histogram.

The analysis may be used to determine a threshold above which rises are considered to form spikes. This value can then be used as an input parameter in the spike removal function - see 10.3.8, below.

10.3.8 Spike removal

This function filters spikes from a signal, based on a maximum permissible rise threshold.

The filtering process is as follows:

- The first point in the signal is copied to the output file and becomes the current point, P(n).
- The signal is read point-by-point and the rise, R, between the current point, P(n), and the next point, P(n+1), is evaluated.

- If the difference, R, is less than the specified maximum permissible rise, Rmax, then the next point becomes the current point and is written to the output file. Processing continues with the new current value.
- If the difference, R, is greater than Rmax then the point is considered to form either a part or the whole of a spike. The current point is held and the next point is incremented to P(n+2).
- The rise between the two points, P(n) and P(n+2), is evaluated and compared with twice the maximum permissible rise value (2×Rmax). If the rise is greater than (2×Rmax) then the next point is incremented again.
- A new rise is evaluated and compared with (3×Rmax).
- This process continues until the rise falls below the permitted multiple of the maximum rise. This point is considered to be the end of the spike.
- Assume that a spike is detected between two points P(n) and P(n+m). The module now linearly interpolates between these two values over (m-1) points and the interpolated values are written to the output file. The point P(n+m), becomes the current point and is also written to the output file.
- The whole process continues with the new current point.

Note:

If the beginning of a spike is detected at point p(n), but the end of the signal is reached (at a point P(n+m)), before the end of the spike has been determined, then the current data point P(n) is copied to the output file (m) times. This avoids any inconsistency between the number of data points in the input file and the number of data points in the output file.

10.3.9 Time at level

This function determines the length of time the signal spends within any amplitude band.

The results are presented as a time-at-level diagram and as a normalised probability density diagram.

Time-at-level diagram:

This diagram shows the length of time the signal spends within any amplitude band (bin). The total area is equivalent to the length of the signal in units of time.

A time at level results matrix is defined by specifying the number of bins, an upper limit and a lower limit. The range of each amplitude band, or bin width, is defined by:

$$\text{bin width} = \frac{\text{upper limit} - \text{lower limit}}{\text{number of bins}}$$

Conventional methods approximate time-at-level by counting the data values that fall within any amplitude band, assuming that the time spent in the band is given by the time between samples. However, such methods tend to give poor results for short signals.

Instead, this program determines the bins passed through between each data point in the signal, and performs a linear interpolation to find the time spent traversing each bin.

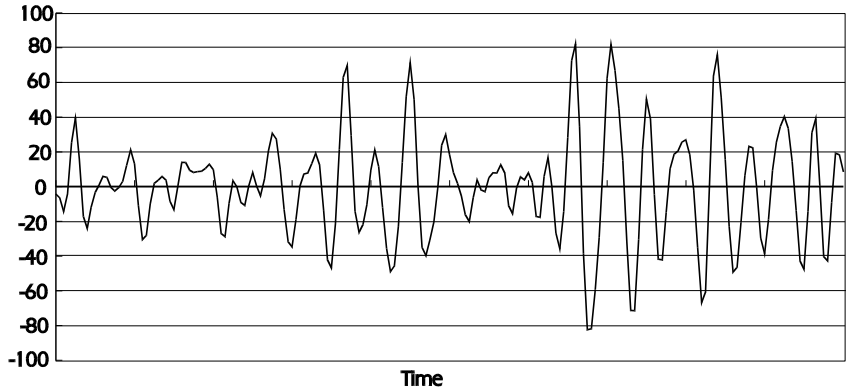


Figure 10.3.9-1

The time taken for a cycle to cross a particular amplitude band is Δt . The time spent within a particular amplitude band for the complete signal is calculated by summing Δt for all cycles that cross the band.

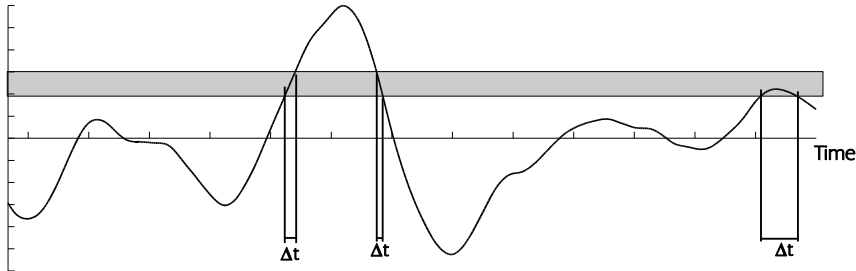


Figure 10.3.9-2

Because the time spent in a band is dependant on the width of the band, the program produces a time-at-level density diagram by dividing the time in each band by the width of the band.

The time-at-level result is a distribution whose area represents the total time of the signal (assuming the amplitude limits encompass the whole signal). The time spent between any two limits is represented by the area between these limits.

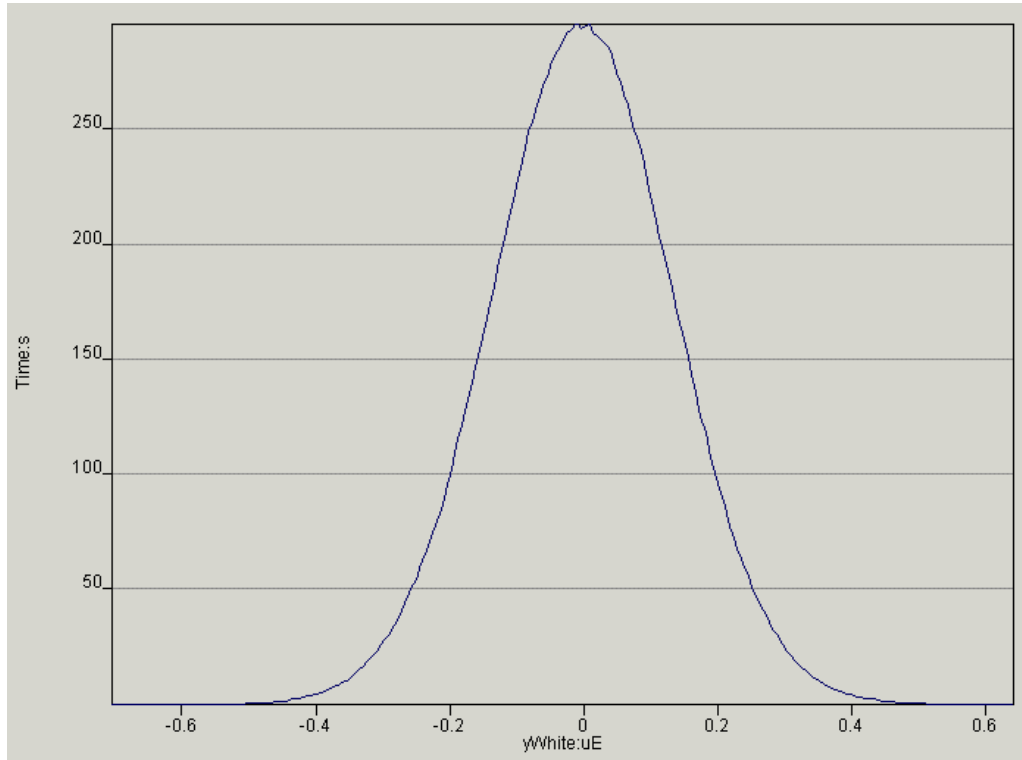


Figure 10.3.9-3

Normalised probability density diagram

This diagram is similar to the time-at-level diagram, except that the total area is normalised to give an area of unity. Its shape is therefore the same as the time-at-level diagram. The area between any two amplitude limits represents the proportion of time the signal spends between these limits, and hence the probability of a given data point falling between these two limits.

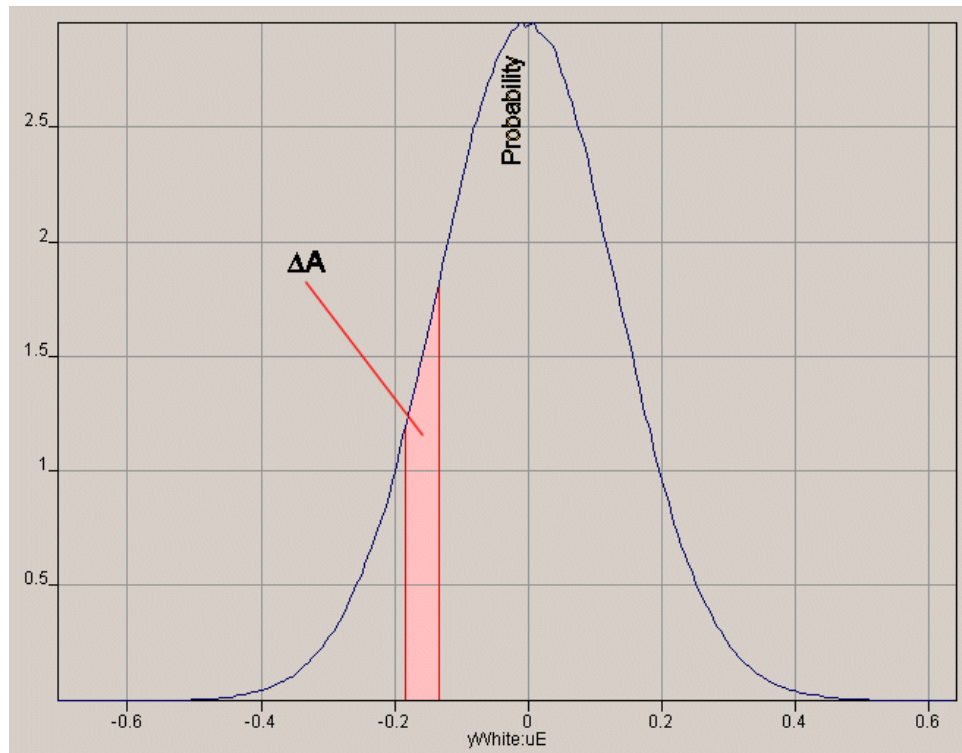


Figure 10.3.9-4

10.3.10 Joint time-at-level

This function uses the same techniques as the time-at-level function (see 10.3.9) to determine the length of time one signal spends within an amplitude band at the same time as a second signal lies within an amplitude band.

The results are presented as a time-at-level histogram and as a normalised probability density histogram.

Time-at-level histogram

The histogram shows the proportion of time that a data point in the first signal lies within an amplitude band at the same time as a data point in the second signal lies within a band.

The sum of all bins in the histogram is equivalent to the length of the signal in units of time.

Normalised probability density histogram

This diagram is similar to the time-at-level histogram, except that the sum of all bin 'volumes', (i.e. bin width (signal 1) × bin width (signal 2) × bin count) is normalised to have unit volume.

The profile of the histogram will be the same as that for the time-at-level histogram. However, in the probability density histogram the value of any histogram bin represents the probability that a data point in the first signal lies within an amplitude band at the same time as a data point in the second signal lies within a band.

The following examples show the joint time-at-level histogram and the normalised probability density histogram for the two white noise signals.

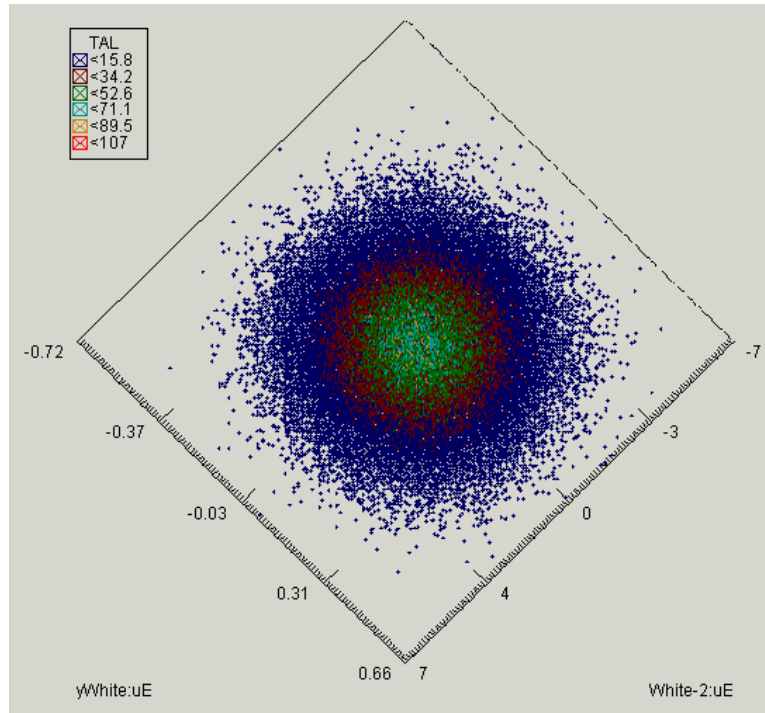


Figure 10.3.10-1 Time-at-level histogram

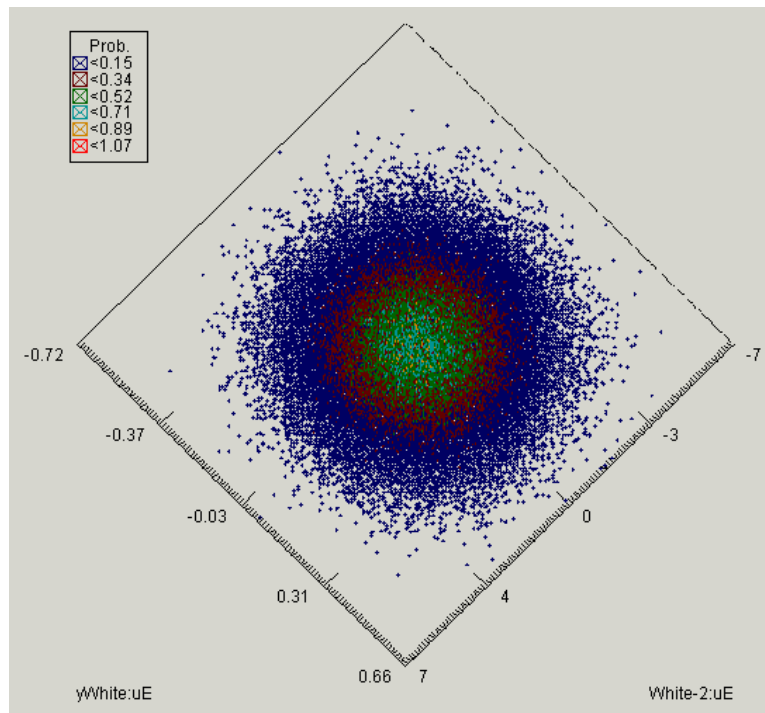


Figure 10.3.10-2 Normalised probability density distribution histogram

10.3.11 Level crossing from time histories

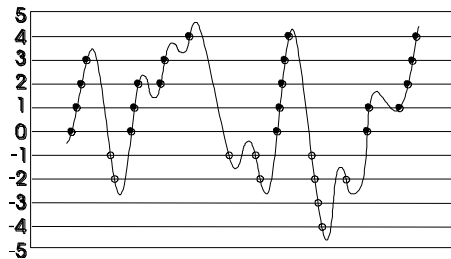
This function calculates the level crossing distribution for a sequential signal as defined in DIN45667.

A level crossing results matrix is defined by specifying the number of bins, an upper limit and a lower limit. The range of each amplitude band, or bin width, is defined by:

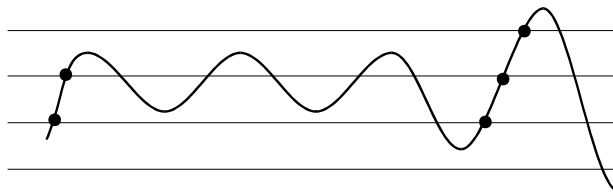
$$\text{bin width} = \frac{\text{upper limit} - \text{lower limit}}{\text{number of bins}}$$

The limits, which can be rounded to give a specific bin width, can be greater than the limits of the input signal or need not fully encompass the signal. In the latter case, any crossings outside the limits will be ignored.

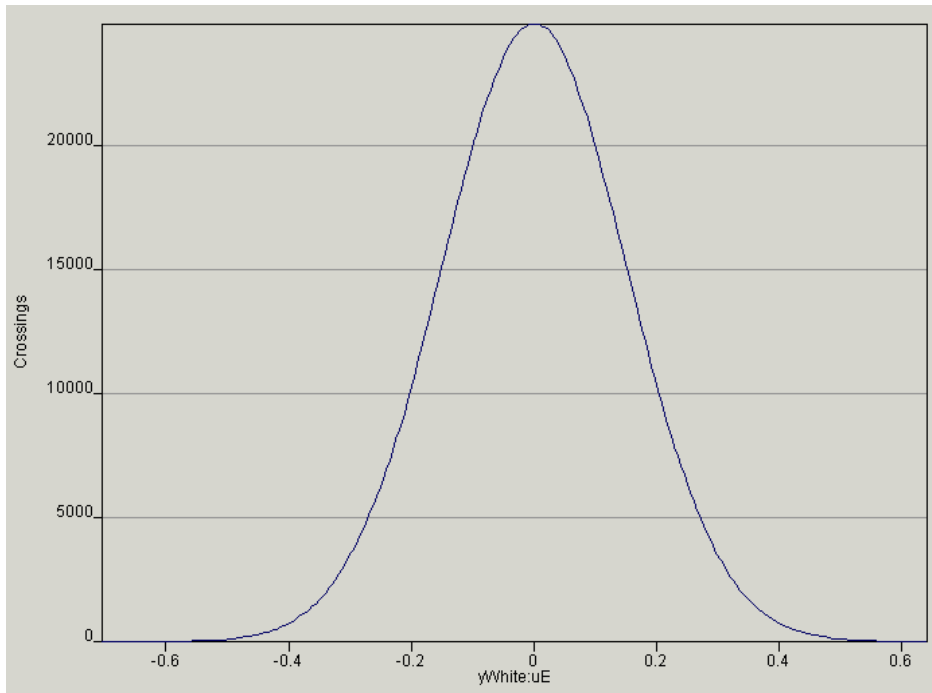
The program counts the number of times the signal crosses each band in a positive direction. This is equivalent to DIN45667 which specifies counting positive slope crossings for positive signal values, and negative slope crossings for negative signal values.



A threshold gate level may be set to reduce the effect of noise in the signal. If noise coincides with a bin boundary, many crossings may be counted. However, if a gate value is defined, the signal must cross an adjacent bin boundary for a repeat crossing to be counted.



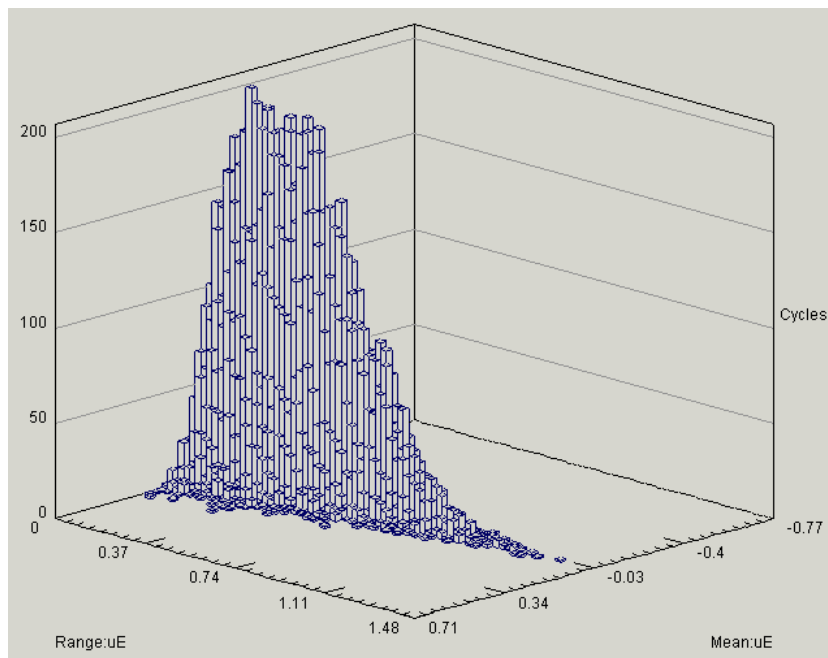
The figure below shows a level crossing distribution for a Gaussian white noise signal.



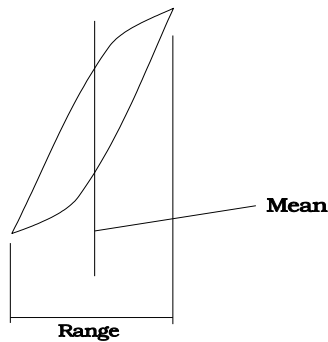
10.3.12 Level crossing from range-mean histograms

This function calculates a level crossing distribution from a Rainflow cycle histogram.

A cycle histogram contains a description of the signal in terms of cycle range and cycle mean.



From the range and mean, the maximum and minimum values for the cycle can be calculated.



$$max = mean + \frac{range}{2}$$

$$min = mean - \frac{range}{2}$$

The levels crossed between the cycle maximum and minimum are determined for each cycle in the histogram, to produce a level crossing distribution.

A level crossing distribution for a cycle histogram will be very similar to that obtained from the original (sequential) signal – see 10.3.11.

10.3.13 Rainflow (and cycle exceedence) from time histories

This function produces a range-mean cycle histogram using a rainflow cycle counting algorithm. The method used is described in section 4 of the Fatigue Theory Reference Manual.

Rainflow cycle counting uses the peaks and valleys in a signal to determine the fatigue cycles (closed stress-strain hysteresis loops) present. The input to the function can be the original signal, or the result of a peak-valley analysis.

Rainflow cycle counting can be used to show a concise summary of a signal. However, only histograms produced from signals which had units of microstrain may be used as the input to local strain-based fatigue analysis algorithms. Stress and strain ($\mu\epsilon$) signals may be used as input to the 'fatigue of welded joints' programs.

The output of this function may be one or more of the following (user-selectable) options:

Range-mean rainflow cycle histogram

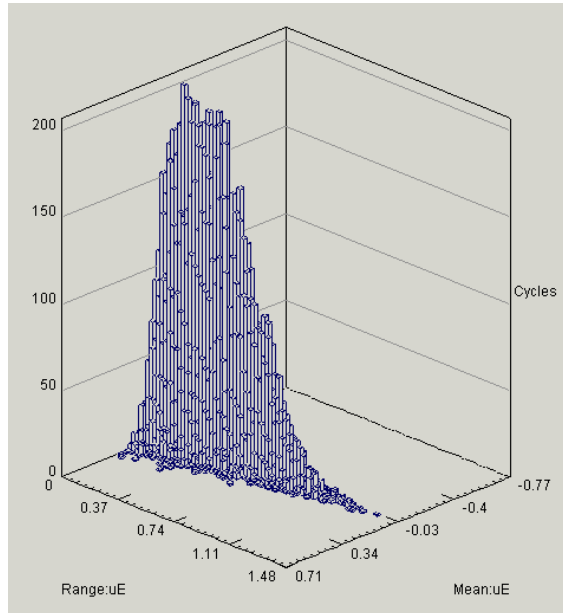
A range-mean matrix is defined by specifying the number of bins (between 2 and 64), an upper limit and a lower limit. The width of each bin is defined by:

$$\text{bin width} = \frac{\text{upper limit} - \text{lower limit}}{\text{number of bins}}$$

The limits, which can be rounded to give a specific bin width, can be greater than the limits of the input signal or need not fully encompass the signal. In the latter case, any cycles outside the limits will be ignored. Each range

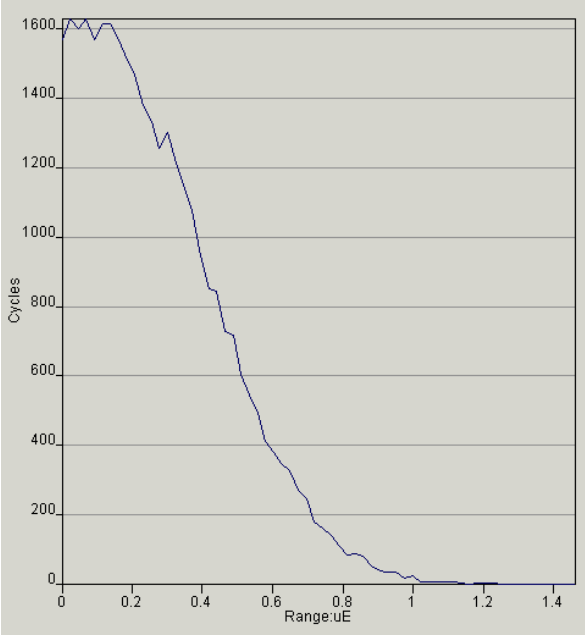
and mean bin represents the same increment in engineering units. Note that there may be rounding issues around the bin limits, so care should be taken if specifying limits close to the signal minimum and maximum; for example, if the exact anticipated maximum is given, it may be that the maximum range cycle is missed, because of rounding errors in the way that the rainflow counting is performed on a discretized version of the signal for efficiency. If this happens then a warning message is given of the number of cycles which overflowed the maximum range. If the lower and upper values are explicitly specified then it is recommended that a small extra allowance of order $R/4096$ is made at each bound if it is desired to always count all cycles in the signal (where R is the signal min to max range). The range axis will be between 0 and $U-L$ (where U and L are the upper and lower limits), and the mean axis will be between L and U . If the values for U and L are not specified explicitly but left at the default Signal Min and Signal Max, then note that the actual applied signal bounds used in the histogram are extended by 5% of the range at each end. This guarantees that all cycles are counted but results in bin sizes 10% greater than R/N and N is the number of bins. The range axis will extend to $1.1R$ and the mean axis will be $\text{Signal Min}-0.05R$ to $\text{Signal Max}+0.05R$.

The range and mean of each closed cycle are determined, and used to position the cycle in the range-mean histogram.



Range-only rainflow cycle histogram

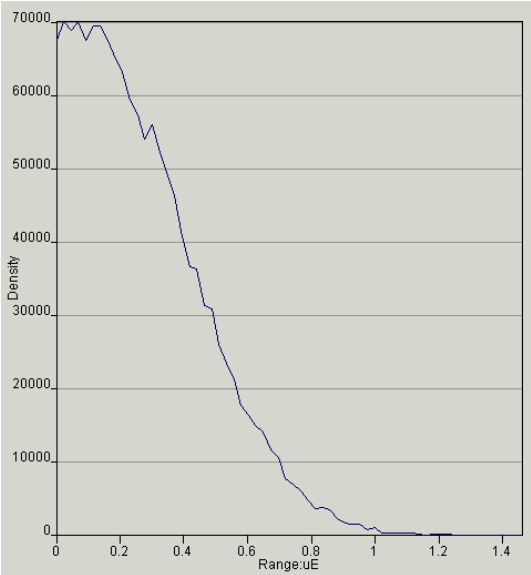
The range-only histogram is a 2D histogram showing the distribution of cycle ranges in the signal.



Cycle density diagram

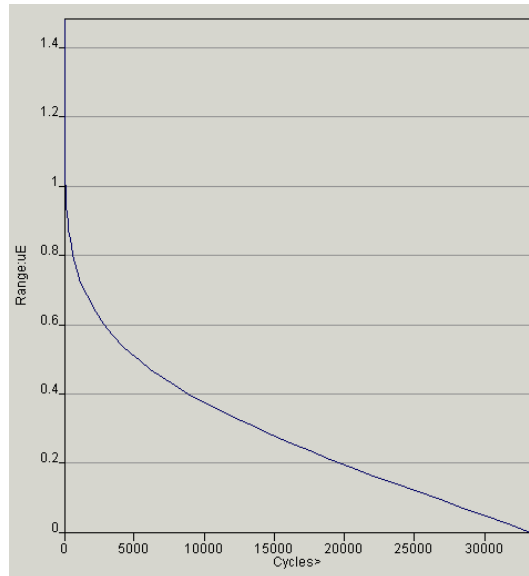
Histograms produced from different analyses may be difficult to compare, since the number of cycles in a bin depends on the bin width.

However, if the number of cycles in each bin is divided by the bin width, a cycle density distribution diagram is produced, where the area between any two range values represents the number of cycles with ranges between these values. Since this distribution is independent of bin width, it can be used to compare cycle distributions for different signals.



Cycle exceedence diagram

A cycle exceedence diagram is produced by integrating the cycle density diagram from the right hand side. This distribution shows the number of cycles which exceed a given range.



10.3.14 Rainflow histogram from PSD

The power spectral density, or PSD, describes the frequency content of a signal. Certain properties of the original signal can be determined by analysing the PSD, making the PSD a very efficient method for describing a signal. For a PSD to be used as the input to a fatigue analysis, a method is required for determining the fatigue cycle information contained in the signal – see section 12 of the Fatigue Theory Reference Manual, *“Fatigue Analysis from the PSD”*.

This function produces a range-mean rainflow cycle distribution histogram from a PSD, using an enhanced algorithm based on the work of Sherratt and Dirlik – see section 12.4 of the Fatigue Theory Reference Manual.

10.3.15 Range-pair and cycle density from level crossing

This function determines the range-pair distribution and the range-pair density distribution of a signal from level crossing data.

10.3.16 Convert rainflow to LDF for FEA fatigue

A fatigue loading definition (LDF file), for the defined dataset number, for use in an FE-based fatigue analysis.

A loading block is created for each non-empty range-mean bin in the input range-mean rainflow histogram.

Each loading block contains a cycle from:

(MEAN – RANGE)

to

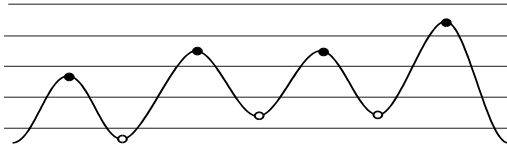
(MEAN + RANGE)

repeated n times, (where n is the number of cycles in the bin).

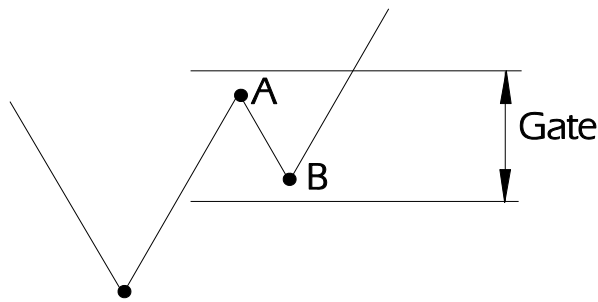
The MEAN and RANGE values can be based on either the upper bin edge (most conservative) or the centre of the bin.

10.3.17 Peak-valley (and P-V exceedence)

This function extracts sequential peaks and valleys from a signal, and discards data points between the peaks and valleys, to produce a peak-valley file.

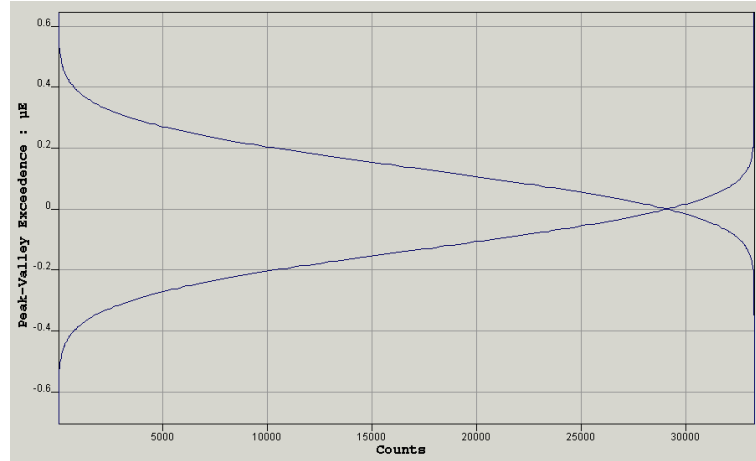


A gate level may be set to exclude small signal fluctuations. For example, in the following signal extract, the range A to B is smaller than the gate value, so the peak-valley pair A-B would not be written to the output file.



Gating may be used to reduce the size of a peak/valley file. For a signal from a digital source, gating can be used to exclude the effects of quantisation noise (which can make almost every point a peak or valley). However, if the file is to be used in a fatigue analysis, care must be taken not to exclude potentially damaging events. The constant amplitude endurance limit is *not* a guide to gate selection, since cycles much smaller than the endurance limit can cause fatigue damage. For the same reason, it is also potentially dangerous to use gating to produce command signals for accelerated fatigue tests.

The results of a peak-valley analysis may also be displayed as a peak-valley exceedence diagram. This shows the number of peaks or valleys which exceed any specified value:



10.3.18 Multi-channel peak-valley

This function extracts sequential peaks and valleys from all selected input signals, and discards data points between the peaks and valleys. To maintain synchronisation between channels, if a peak or valley is found in one channel, the corresponding data point from all other channels is also extracted.

Time information can be added as an additional results signal. The extracted peaks and valleys can be plotted on the same time axis as the original signal by cross-plotting the peak-valley results with the time information signal. See section 8.8.1 of the Fatigue Theory Reference Manual for a description of multi-channel peak-valley operations. See section 5.7.2 for a discussion of the application of multi-channel peak-valley analysis to the analysis of FEA models.

The inverse time increment can also be exported for use as a drive signal.

10.3.19 Markov ‘from-to’ matrix

This function analyses the ranges between successive peak-valley pairs in a signal. The outputs available are a from-to matrix, a range-mean histogram and a range-only histogram.

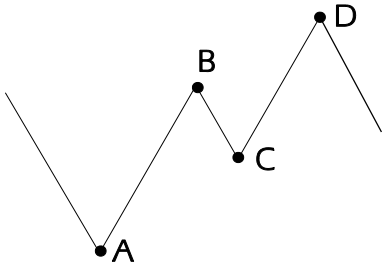
Markov ‘from-to’ matrix

A ‘from-to’ matrix is defined by specifying the number of bins (between 2 and 64), an upper limit and a lower limit. The width of each bin is defined by:

$$\text{bin width} = \frac{\text{upper limit} - \text{lower limit}}{\text{number of bins}}$$

The limits, which can be rounded to give a specific bin width, can be greater than the limits of the input signal or need not fully encompass the signal. In the latter case, any data-points outside the limits will be ignored. Each range and mean bin represents the same increment in engineering units.

For the from-to matrix, the algorithm extracts peaks and valleys from the signal. As each turning point is extracted, the peak and valley are binned in the output matrix.



Peak-valley pair	'from' bin	'to' bin
A-B	A	B
B-C	B	C
C-D	C	D
etc...		

In the above example, points A and B will be binned with point A in the *from* bin, and point B in the *to* bin. Then points B and C are binned with point B in the *from* bin, and point C in the *to* bin. The complete signal is analysed in this way. The result is a 'from-to' matrix, as shown in *Figure 10.3.19-1*. A 'from-to' matrix is sometimes referred to as a Markov matrix.

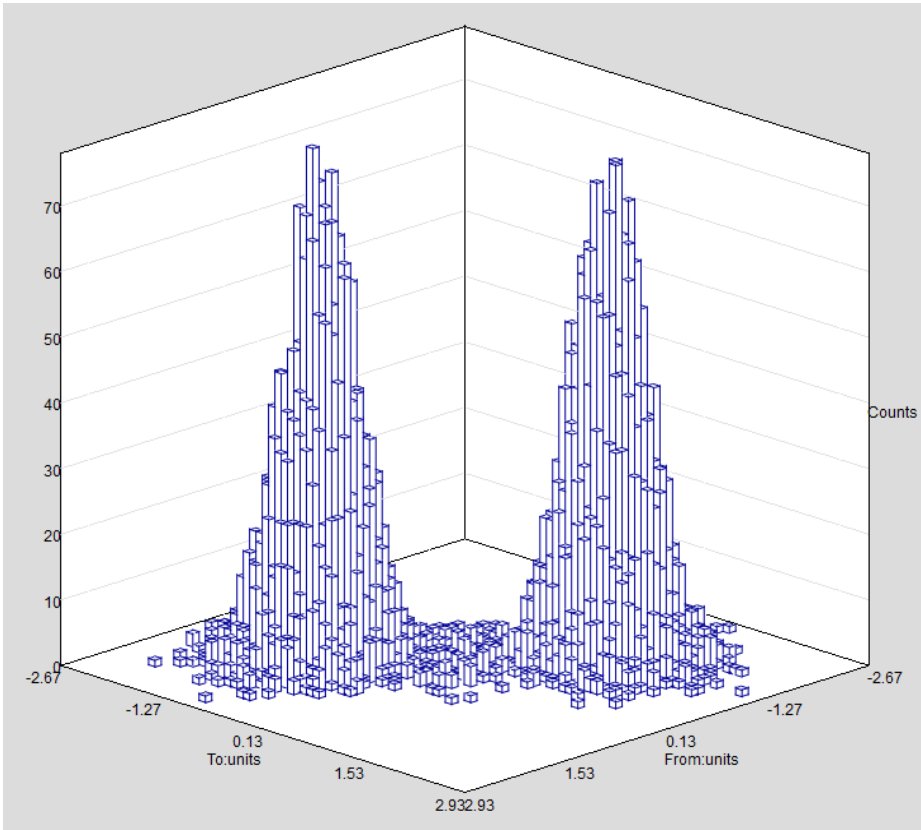


Figure 10.3.19-1 : Markov from-to distribution as a 3D histogram.

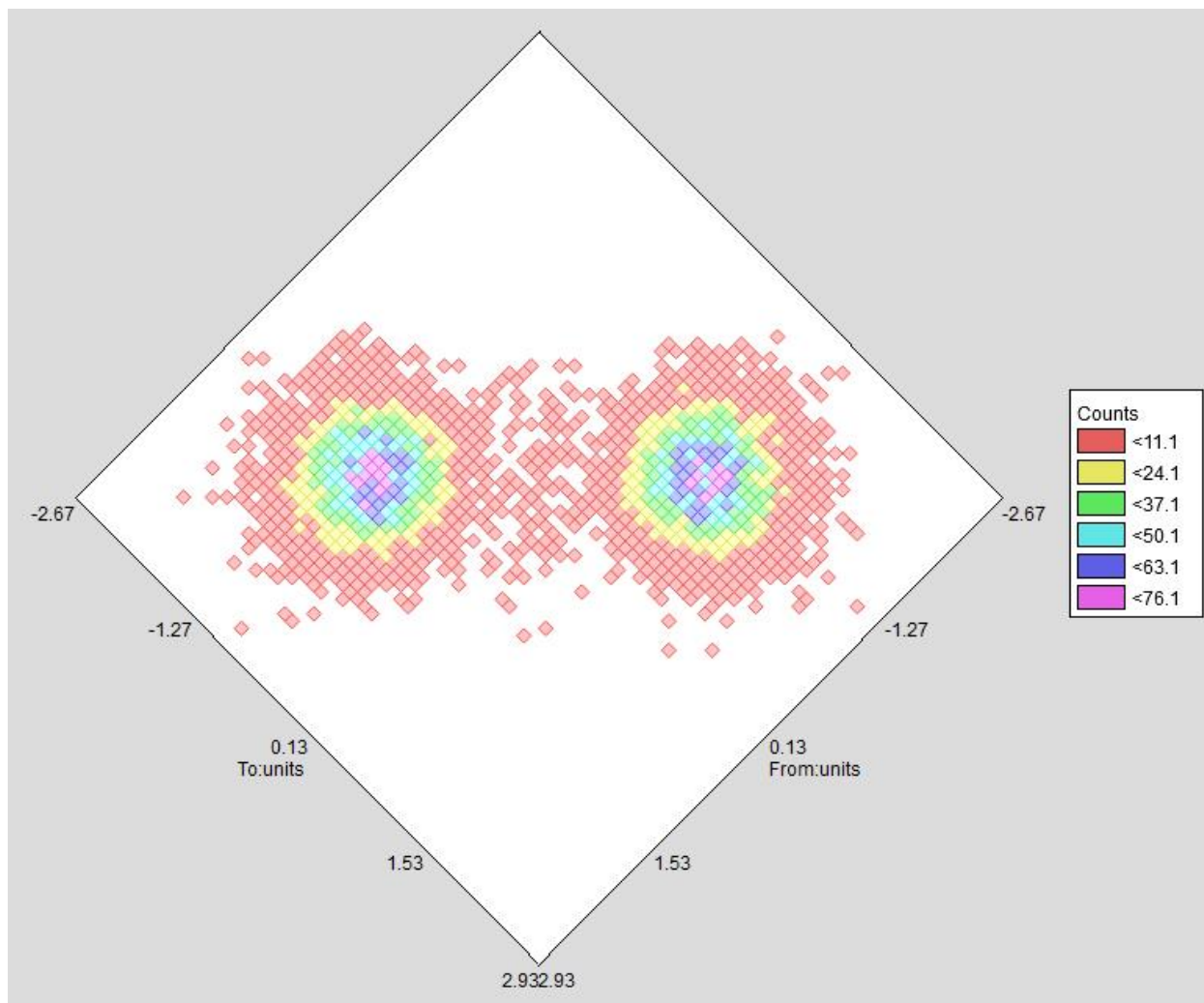
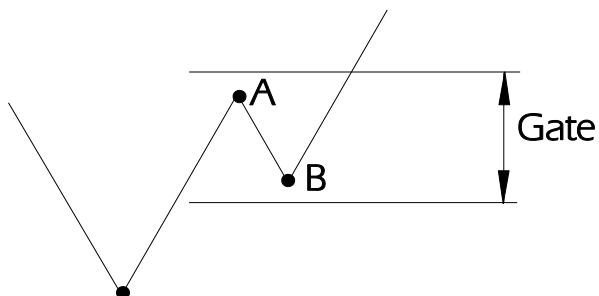


Figure 10.3.19-2 : Markov from-to distribution as a contour plot.

A gate level may be set to exclude small signal fluctuations. For example, in the following signal extract, the range A to B is smaller than the gate value, so the peak-valley pair A-B would not be written to the output file.

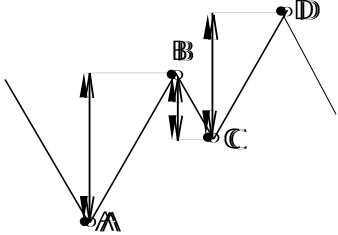


Range-mean histogram

The matrix for the range-mean histogram is defined in the same way as the matrix for the 'from-to' histogram, above.

For each peak-valley pair, the peak-valley range and mean are calculated, and the peak-valley pair is binned in the histogram.

For example:



Peak-valley pair	Range bin	Mean bin
A-B	$ A - B $	$(A+B) / 2$
B-C	$ B - C $	$(B + C) / 2$
C-D	$ C - D $	$(C + D) / 2$
etc...		

A value is binned for every peak-valley pair in the signal, to produce the range-mean histogram.

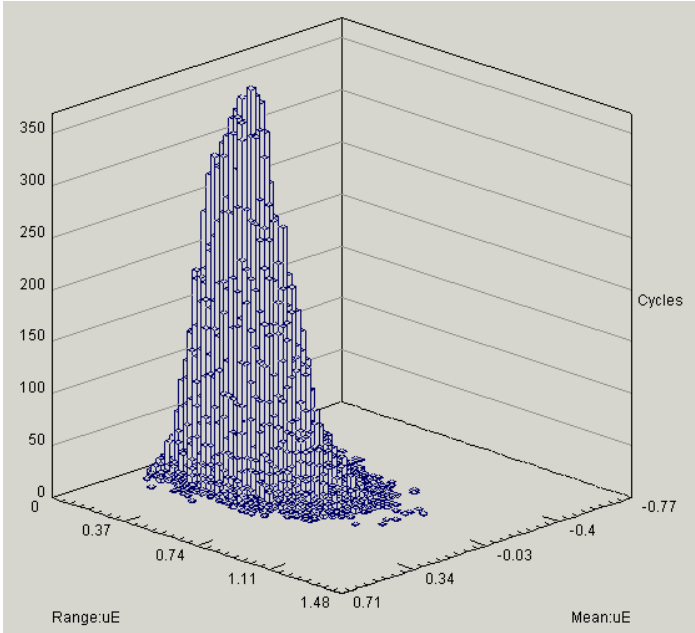


Figure 10.3.19-3 : Range-mean histogram

Range-only histogram

The range-only histogram is a 2D histogram showing the distribution of peak-valley pair ranges for every peak-valley pair in the signal.

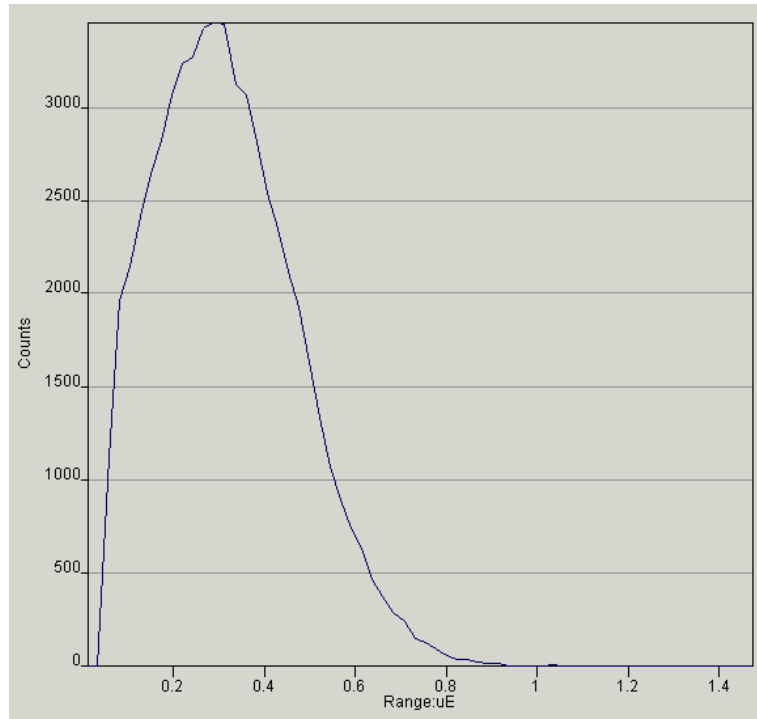


Figure 10.3.19-2 Range-only histogram

10.3.20 Transform 2D strain tensor to 45° rosette data

This function transforms 2D tensor data (i.e. three channels of data containing ϵ_{xx} , ϵ_{yy} and γ_{xy} data) into equivalent data for a 45° strain gauge rosette. The function performs a point-by-point transformation on the three input signals. Units must be microstrain ($\mu\epsilon$). Three output files are created, containing:

- Strain gauge rosette data - **0°**
- Strain gauge rosette data - **45°**
- Strain gauge rosette data - **90°**

10.3.21 Transform 3D tensor to new orthogonal axes

This function transforms 3D tensor data (i.e. six channels of data containing xx, yy, zz, xy, yz, zx, data) onto a new set of orthogonal axes. The transformation is defined by the two transformation angles:

- θ° - angle of rotation of the z-axis towards the x-axis, in the x-y plane.
- ϕ° - angle of rotation of the x-axis towards the y-axis in the x-y plane.

The function performs a point-by-point transformation on the six input signals, to create a transformed 3D tensor, (i.e. six channels of transformed data containing **xx**, **yy**, **zz**, **xy**, **yz** and **zx** data).

10.3.22 Transform XYZ vectors onto new axis

This function transforms a set of orthogonal vectors in X, Y and Z onto a new orthogonal axis system.

Two transformation methods are available:

- Absolute transformation – this method transforms input vectors $\{x,y,z\}$ to output vectors $\{x',y',z'\}$, using Euler angles, θ , ϕ and φ .
- Relative transformation – this method transforms input vectors $\{x,y,z\}$ to output vectors $\{x',y',z'\}$, using pitch, roll and yaw angles, θ , ϕ and φ .

A forward or reverse transform can be specified for both methods.

10.3.23 Statistics

This function produces a statistical summary for all selected signals. The information is written to an ASCII text file in a tabular format, and is also displayed in a dialogue box. The name of the text file is displayed in the message log window.

The following information is produced for each signal (for the selected analysis range):

Max	-	Maximum amplitude
Max Pos-		Position of maximum amplitude (in x-axis units)
Min	-	Minimum amplitude
Min Pos	-	Position of minimum amplitude (in x-axis units)
Mean	-	Arithmetic mean of all data points
SD	-	Standard deviation of all data points

If y is a data point value and N is the number of points in the signal, then:

$$\text{mean } \bar{y} = \frac{\sum(y)}{N}$$

$$\text{std dev} = \sqrt{\frac{\sum(y-\bar{y})^2}{N}}$$

10.4 Frequency

10.4.1 Background theory for frequency analysis techniques

The theory supporting the frequency analysis and transform methods used in *fe-safe* is discussed in detail in sections 4 and 5 of the Signal Processing Reference Manual.

Some key definitions are:

- The power spectral density (PSD) distribution is a frequency domain description of the amplitude of each frequency present in a signal.
- The cross-spectral density (CSD) distribution is a frequency domain description of the relationship between two signals for different frequencies, and can be used to establish the extent to which the amplitudes and frequencies in the signals are common.
- The Gain is the ratio of the output and input amplitudes at each frequency.
- The Phase angle is a measure of how much an output signal is time-shifted with respect to the input signal, at each frequency.
- The Coherence is the extent to which an output is the function of a specified input. A coherence of 1 means that the system is linear, i.e. that the output is a linear response to the input, and that the output was produced only from the one input, with no contributions from other inputs.

10.4.2 Power spectral density

This function calculates the Power Spectral Density (PSD) of a signal. A FFT function is used. The program takes a number of data points (the 'FFT buffer'). A 10% cosine taper function can be applied to the data. The FFT coefficients are calculated. The next buffer is then taken (with an overlap between buffers). The FFT coefficients for this buffer are averaged with those of the first buffer. This process continues until the end of the signal. Finally the PSD ordinates are calculated.

For a signal with a non-zero mean, the PSD analysis may show a very high ordinate at zero Hz. This may be removed by selecting 'Normalised analysis', in which case the mean value of the signal is calculated and subtracted from each data point before the FFT coefficients are calculated.

A PSD should be calculated for signals which are statistically stationary. For a non-stationary signal, frequencies present in only a small part of the signal can be 'lost' in the averaging process. For these signals the user may select a 'Peak hold' PSD. In this case, at each frequency the highest ordinate, rather than the average ordinate, is retained.

The PSD dialogue is shown in *Figure 10.4.2-1*.

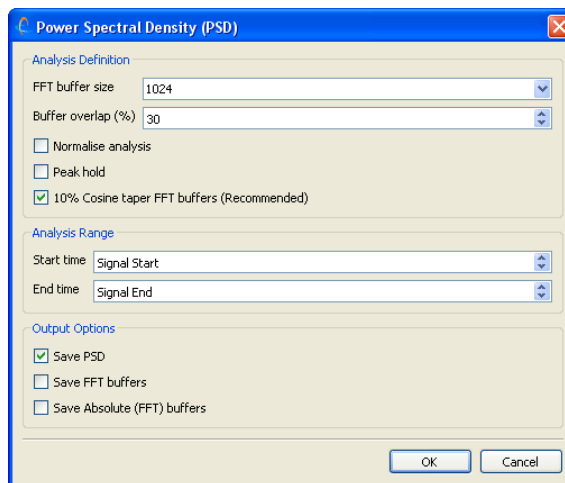


Figure 10.4.2-1

The user may define

- The FFT buffer size in data points (an integer power of 2)
- The Buffer overlap (this should be set to 10%)
- Normalised analysis
- A peak hold PSD instead of an averaged PSD
- Include 10% cosine taper on each buffer or not.

The following plots can be created:

- PSD – the PSD diagram is created.
- FFT buffers – the average or peak hold real and imaginary FFT buffers are saved.
- Absolute FFT buffers – the average or peak hold value of the absolute real and imaginary FFT buffers are saved.

All plots are shown in *Figure 10.4.2-2*. The input history was 4 superimposed sine waves of various amplitude and phase.

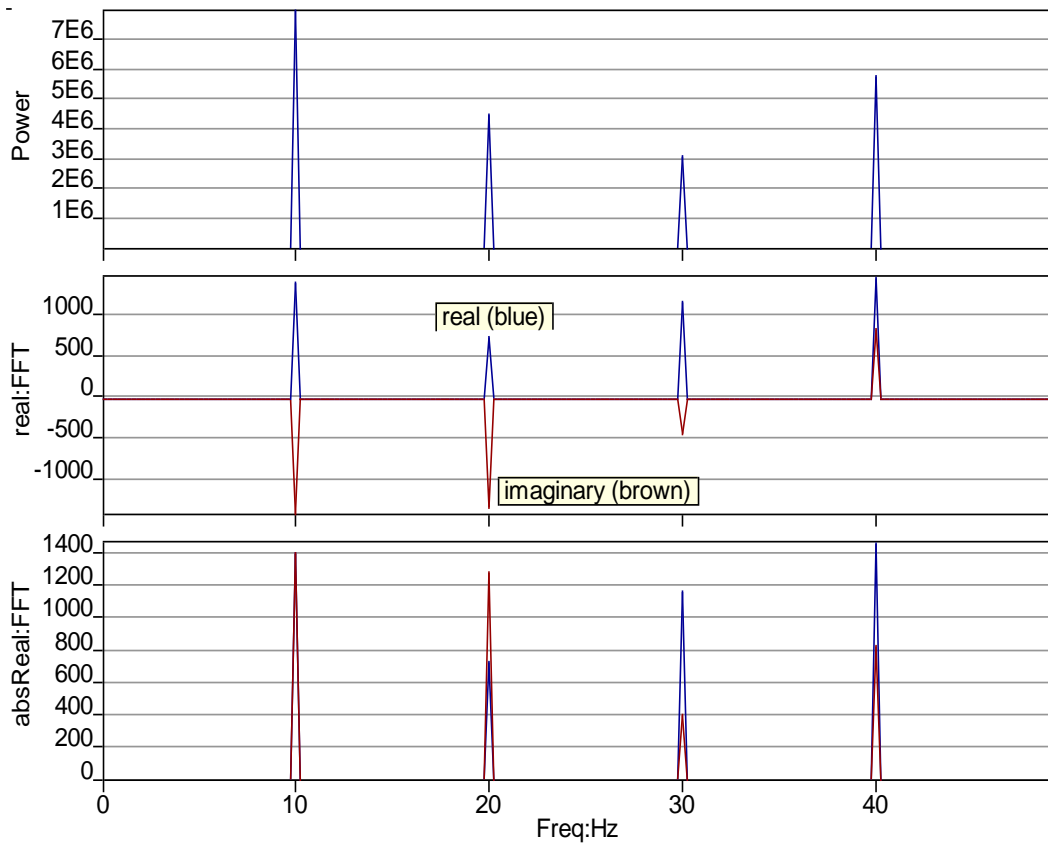


Figure 10.4.2-2

10.4.3 Cross-spectral density (CSD)

This option selects the Transfer Function/CSD function – see section 10.4.4

10.4.4 Transfer function

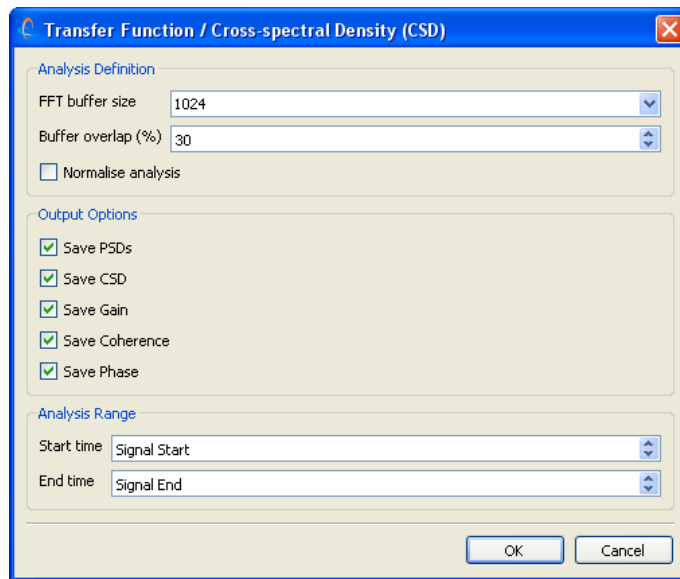
For a pair of signals this function calculates

- the PSD of each signal
- the cross spectral density (CSD) of the signals
- the gain, phase and coherence diagrams

The user highlights first the 'input' signal and then the 'output' signal. Note : if the gain diagram tends to infinity, the files have probably been selected in the wrong order.

FFT buffer size and overlap are defined as in section 10.4.2.

The required output files are then selected.



10.5 Filtering

10.5.1 Background theory for filtering techniques

The theory supporting the filtering methods used in *fe-safe* is discussed in detail in section 2 of the Signal Processing Reference Manual.

10.5.2 Butterworth filtering

Butterworth filters have a smooth frequency response in both the pass band and the stop band, and their effect on the phase relationships in the signal is reasonably linear over much of the pass band range.

This function uses either a low-pass, high-pass or band-pass Butterworth algorithm to remove frequency components from a signal. The filter cut-off frequency (or frequencies) and the filter order (or 'roll-off' rate) can be specified.

A gain can be applied to the signal in the pass region. (Gain=1 gives an output amplitude equal to the input amplitude)

Gain diagrams for the three filter orders (i.e. roll-off rates) for each filter type are shown below:

Butterworth low-pass filter

The gain diagram for the three filter orders (i.e. roll-off rates) for the low-pass Butterworth filter are shown in *Figure 10.5.2.1*:

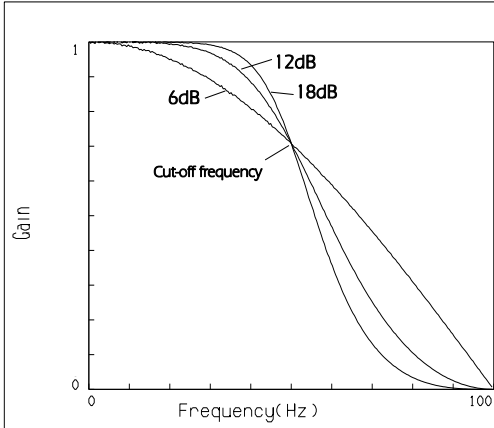


Figure 10.5.2-1 – Gain of low-pass Butterworth filter

Butterworth high-pass filter

The gain diagram for the three filter orders (i.e. roll-off rates) for the high-pass Butterworth filter are shown in *Figure 10.5.2.2*:

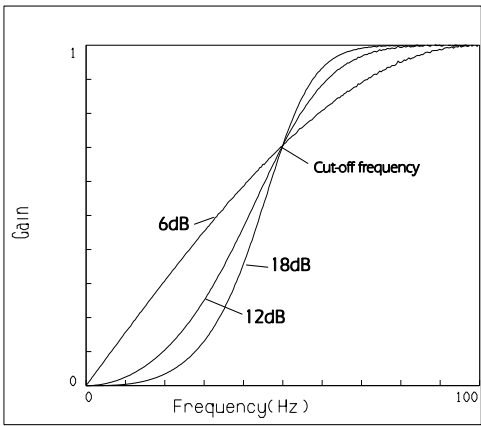


Figure 10.5.2-2 – Gain of high-pass Butterworth filter

Butterworth band-pass filter

The gain diagram for the three filter orders (i.e. roll-off rates) for the high-pass Butterworth filter are shown in *Figure 10.5.2.2*.

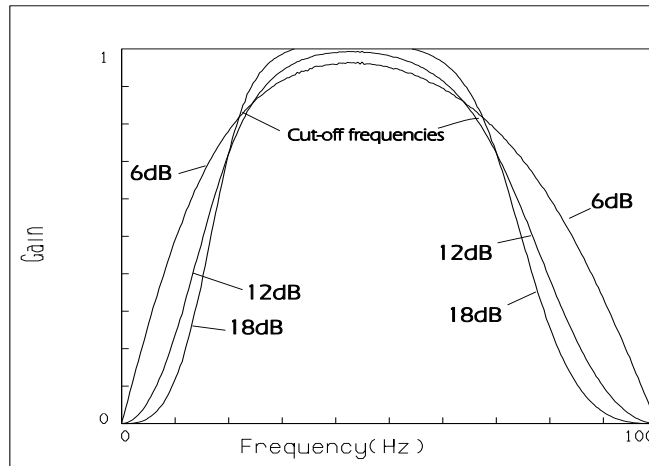


Figure 10.5.2-3 – Gain of band-pass Butterworth filter

10.5.3 FFT filtering

This function allows a digital FFT filter definition to be created from up to ten sets of filter coefficients. Each set of coefficients consists of:

- passband region gain (dB) (Note: 0dB is equivalent to a gain = 1)
- lower cut-off frequency (Hz)
- upper cut-off frequency (Hz)

The order of the filter can also be specified.

Filter definitions can be saved, loaded and plotted.

To filter a signal or signals using an FFT filter, highlight the signal or signals to be filtered in the **Loaded Data Files** window, and select **Frequency >> FFT Filtering...**

This displays the following dialogue:

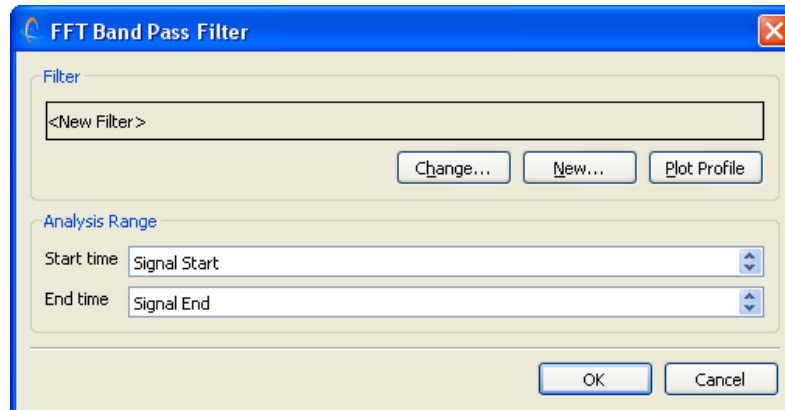


Figure 10.5.3-1

The current FFT filter is displayed. Clicking **Plot Profile** creates a gain diagram for the filter, which is added to the file list in the **Loaded Data Files** window. The analysis range can also be specified. To filter the signal click **OK**.

The filter definition can be changed by selecting either **Change...** (to modify the definition of the current filter) or **New...** (to create a new filter definition). These options display the **FFT Band Pass Filter Definition** dialogue box, as shown in *Figure 10.5.3-2*.

Defining an FFT filter

An FFT filter can be defined by:

- Opening an existing filter profile definition from an FPD file. The FPD file format is used to save filter coefficients for the FFT filter.
- Opening an existing filter profile definition from a GEN file. This file format is similar to the FPD format, and is provided for backward compatibility with some earlier *fe-safe* software.
- Opening a gain diagram from a file with extension GAI. A gain diagram can be defined using other signal processing functions (for example the transfer function) and the file saved (using the **Save Data File As** option) as a DAC file with extension *.gai.
- Defining new filter coefficients.

Filter coefficients are defined in the **FFT Band Pass Filter Definition** dialogue box:

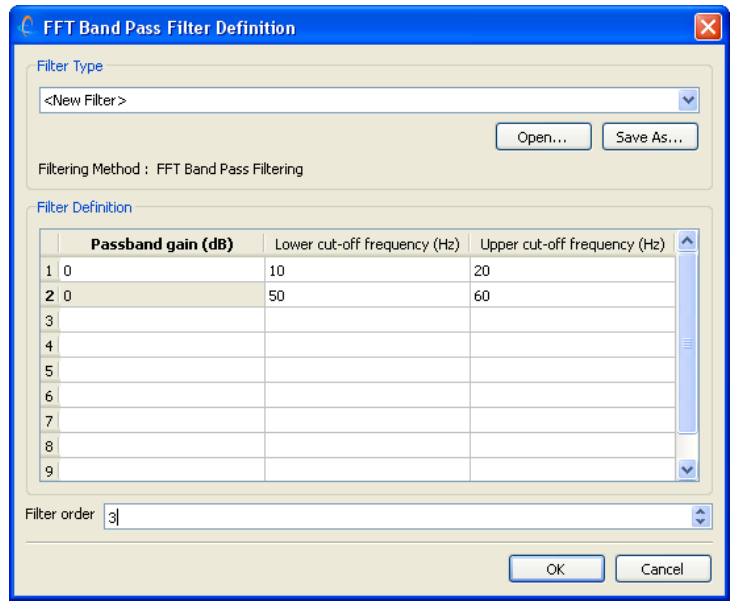


Figure 10.5.3-2

The definition is saved using the **Save As...** option. Saved filter definitions are recalled using the **Open...** option.

The filter coefficients define the passband, so the gain diagram for the coefficients shown in *Figure 10.5.3-2* is as shown in *Figure 10.5.3-3*.

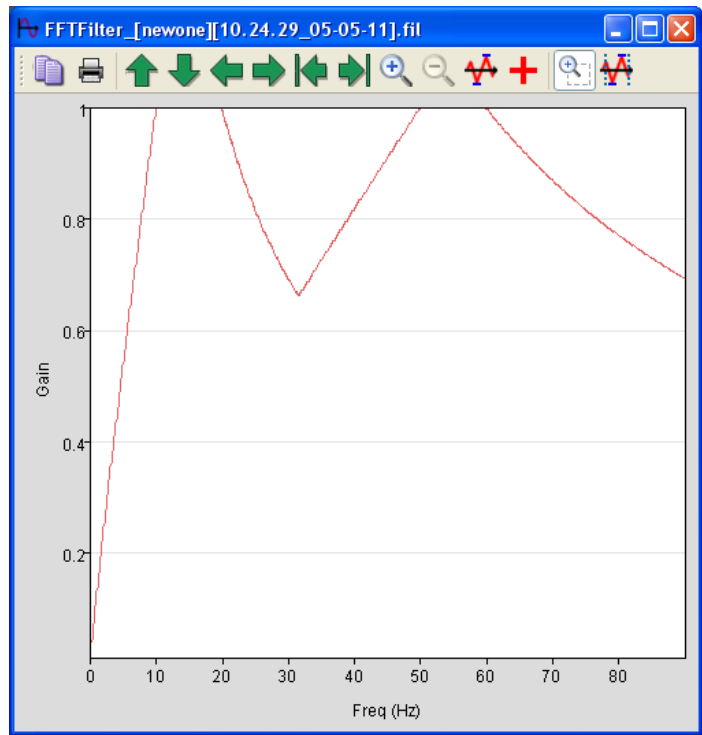


Figure 10.5.3-3

11 Fatigue analysis from measured signals [1] : using S-N curves

This section discusses the classical methods for evaluating fatigue life from measured stresses or strains using the S-N (stress-life) curve for a material. See section 5 of the Fatigue Theory Reference Manual for the background to stress-life fatigue analysis. Analysis of the fatigue life of welded joints to BS5400/BS7608 is also included – see section 11 of the Fatigue Theory Reference Manual. This section applies to the analysis of stress-time and strain-time signals, and to the analysis of cycle histograms.

11.1 General analysis properties

These functions are accessed from the **Gauge Fatigue** menu.

Input stresses or strains can be a time history, or a Rainflow cycle histogram.

Analysis can use a Goodman mean stress correction, or no mean stress correction.

Sensitivity analysis can be carried out to investigate the effect of different stress concentrations or signal scale factors.

Cycle histograms produced by the fatigue programs in this section can be used as input to the histogram analysis functions, as can cycle histograms from the Rainflow cycle counting program (section 10.3.13). A peak-picked signal can be used as input instead of the full signal. Analysis will be quicker, but the time-correlated damage file will not have a true time axis. Other parameters can be used – for example a load-time history may be analysed with a load-life fatigue curve.

User-defined fatigue damage curves - S-N curves or other relationships - can be entered and saved in the material database – see section 8.

The strain gauge rosette program calculates time histories of principal strains and stresses. The stresses in the output file can be used in the BS5400 welded joint programs, and may also be used in the S-N curve analysis programs, provided the user is confident that such input (with possible biaxial stresses) will produce a valid result.

All programs allow entry of a stress concentration factor if nominal stresses have been measured.

Results are displayed on the screen and written to the Generated Results.

11.2 S-N Curve Analysis from Time Histories

11.2.1 Function

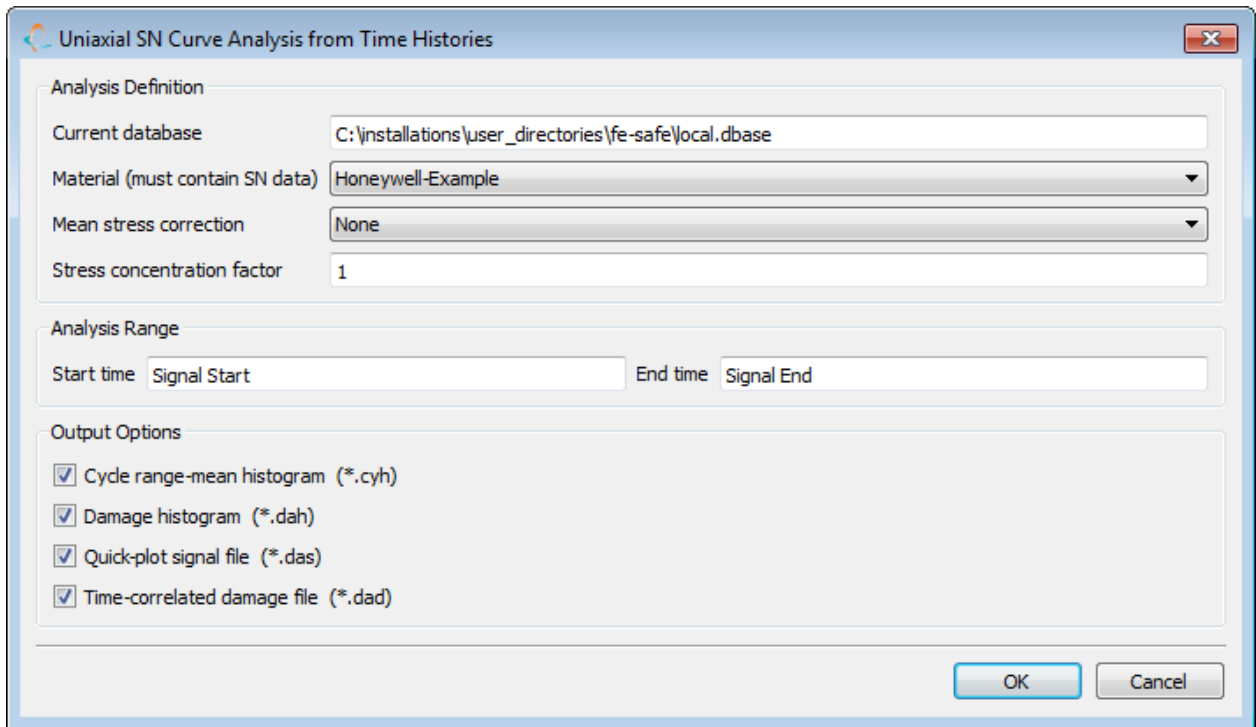
Calculates fatigue lives from a time history, using a material's stress-life (S-N) curve. Input signals may be a stress-time signal or a peak-picked signal.

11.2.2 Operation

Select:

Gauge Fatigue >> Uniaxial SN Curve Analysis from Time Histories...

This displays the following dialogue:



Goodman mean stress correction or no mean stress correction can be specified, and a stress concentration factor and analysis range can be entered.

11.2.3 Output

The following results are created:

- a Rainflow cycle histogram (extension `.cyh`)
- the damage histogram (extension `.dah`)
- a file of time-correlated fatigue damage (`.dad`)
- a file of the signal in a fast-plot format (`.das`)

The cycles and damage histograms are cycle range-mean histograms, in the same units as the signal, 32 bins x 32 bins, scaled to include all cycles. The cycle histogram may be used as an input file for the programs which provide fatigue analysis of cycle histograms

The time-correlated damage file gives an indication of whereabouts in time the fatigue damage occurs.

The fast-plot signal file contains 2048 data points which provide the same plot display (if not zoomed) as the full signal file.

11.2.4 Technical data

The program first searches for the absolute maximum value in the selected section of the signal (positive or negative).

The program then takes each data point and checks if it is a turning point (a peak or valley). For each turning point, the program checks if it has closed a cycle. For each closed cycle the endurance N_f cycles is calculated. The cycle and its damage are added to the output histograms.

At the end of the selected section of the signal, the program returns to the start point of the section, and carries on the analysis until the absolute maximum data point is reached again.

The calculated fatigue damage for each cycle is summed and used to calculate the fatigue life.

To form the time-correlated damage file, as each cycle is closed, the times for the three points which form the cycle are used to position the fatigue damage in time. Half the damage for the cycle is presumed to occur mid-way between the first two points, and the other half of the damage is presumed to occur mid-way between the 2nd and 3rd points. The damage is added to any previously calculated damage at these points.

Note that if the input signal is a peak/valley file, the time axis of the time-correlated damage file has no meaning.

The program calculates fatigue endurance using the stress-life curve. Each endurance is obtained by linear interpolation of the log stress amplitude and log endurance values.

If a stress concentration factor not equal to 1.0 is being applied, the program uses a relationship defined by Peterson:

$$K_{fn} = 1 + \frac{K_f - 1}{0.915 + 200/(\log N)^4}$$

where

K_f is the stress concentration at 10^7 cycles

K_{fn} is the value of K_f at endurance N cycles

The Goodman mean stress correction is

$$\frac{S_a}{S_{ao}} + \frac{S_m}{f_t} = 1.0$$

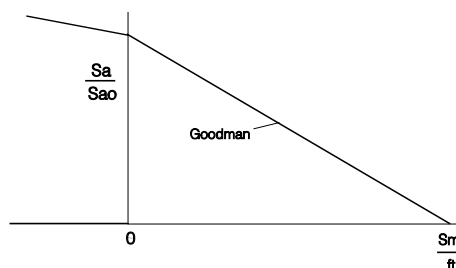
where

S_a is the stress amplitude, and S_m the mean stress

S_{ao} is the stress amplitude at zero mean which gives the same endurance

f_t is the material ultimate tensile stress.

This gives a linear relationship between a range at a given mean stress, and the range at zero mean stress that would give the same endurance. For compressive mean stresses the Goodman line has been extended with half the slope of the original line.



Fatigue analysis from measured signals [1] : using S-N curves

The value of the material UTS, f_t , is read from the materials data base.

Although the Goodman correction is defined for stress, the program does allow the user to use any other measured parameter, providing that an appropriate equivalent of f_t can be obtained.

If a strain history has been measured, and an S-N curve with a stress-based Goodman mean stress correction is required, the strain history should be converted to stress. Care should be taken to ensure that, if a linear conversion is being used, the values do not exceed the elastic limit.

Fatigue lives are calculated using Miner's rule, that for each cycle

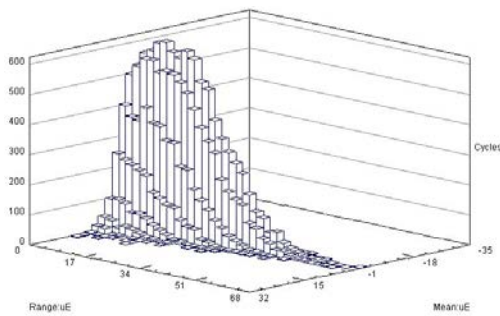
$$total\ damage = \sum \frac{1}{N_f}$$

and that fatigue failure occurs when total damage = 1.0,

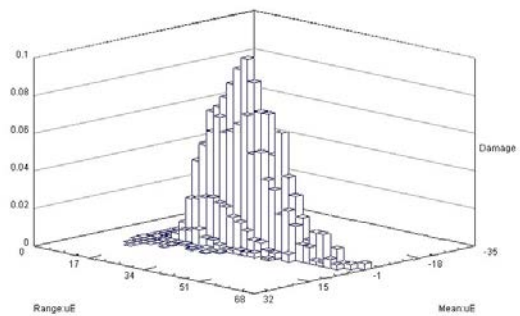
so that the life in repeats of the signal is

$$life = \frac{1.0}{\sum \frac{n}{N_f}}$$

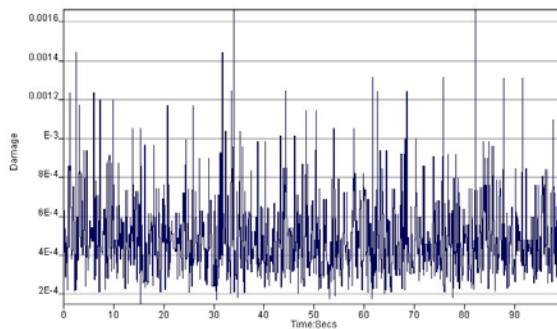
Fatigue failure is to be interpreted using the same criteria as was used to define the endurance values on the S-N curve. If these were lives to crack initiation, then the life calculated by the program will be a calculated life to crack initiation. If they were lives to component failure, then the life calculated by the program will be a calculated life to component failure.

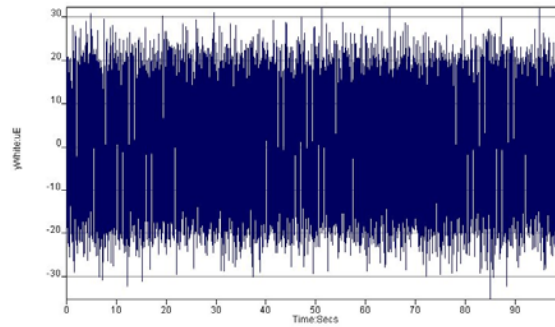


Cycles histogram



Damage histogram





Time-correlated fatigue damage (upper graph) with the loading history (lower graph)

11.3 S-N Curve Analysis from Histograms

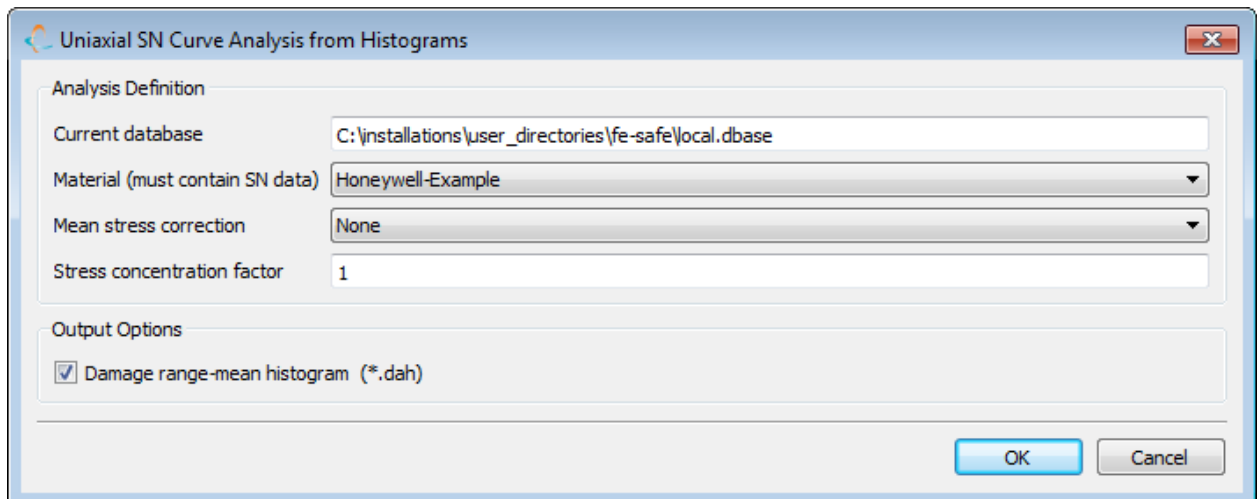
11.3.1 Function

Calculates fatigue lives from a Rainflow cycles histogram, using the stress-life (S-N) curve.

11.3.2 Operation

Select:

Gauge Fatigue >> Uniaxial SN Curve Analysis from Histograms...



Goodman mean stress correction or no mean stress correction can be specified, and a stress concentration factor can be entered.

11.3.3 Output

The screen display shows:

- the number of cycles in the histogram;
- the fatigue life as repeats of the histogram.

The following results are created:

Fatigue analysis from measured signals [1] : using S-N curves

- the name of the cycle histogram;
- the name of the stress-life curve;
- the stress concentration factor;
- the name of the results file;
- the number of cycles in the histogram;
- the fatigue life as repeats of the histogram.

The output files is a damage histogram (extension [.dah](#))

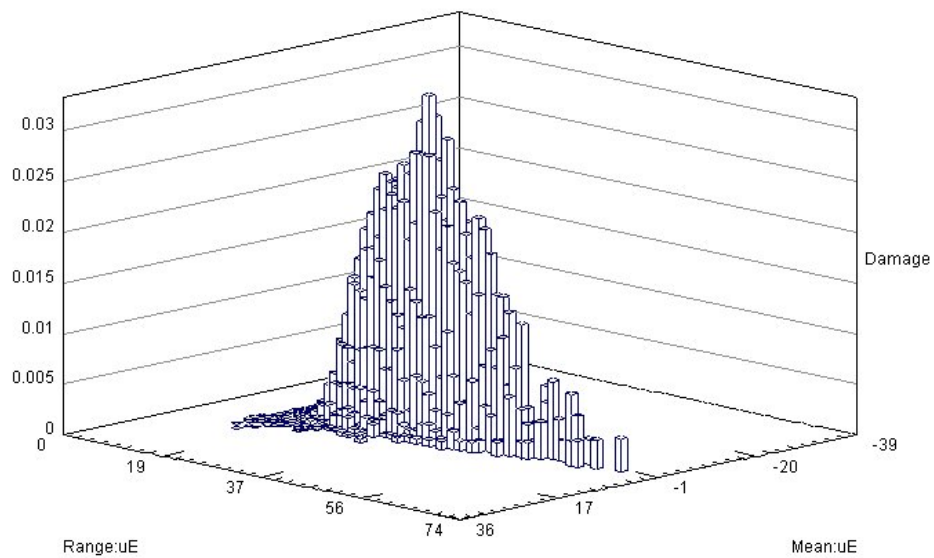
11.3.4 Technical data

The input cycles histogram will have been produced by a cycle counting operation, either in software or by a hardware Rainflow counter.

It must be a matrix of cycle ranges and cycle mean values. The default file extension is [.cyh](#).

The technical background is as defined in section 11.2.4

The fatigue damage is calculated using the mean value of stress range and mean value of mean stress from each bin of the histogram. This is written to the output damage histogram, extension [.dah](#).



11.4 BS5400 Welded Joints from Time Histories

11.4.1 Function

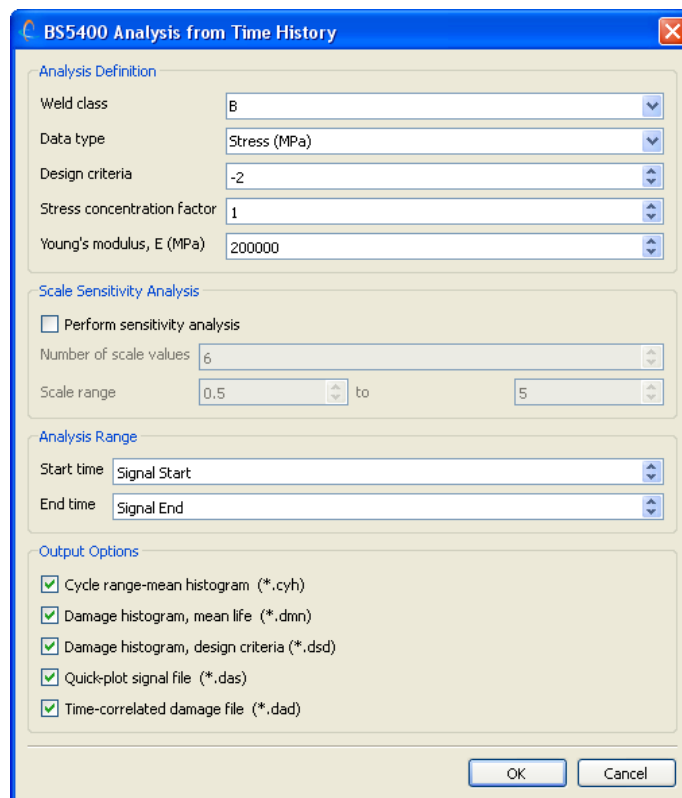
Calculates fatigue lives from a time history, using the stress-life relationships defined in BS5400 part10:1980 for welded joints. Input signals may be a strain-time or stress-time signal or a peak-picked signal. BS5400 allows use of histories measured using a strain-gauge rosette. A sensitivity analysis can also be performed.

11.4.2 Operation

Select:

Gauge Fatigue >> BS5400 Welded Joints from Time Histories...

This displays the following dialogue:



The analysis definition can be configured, including scale sensitivity analysis parameters if required.

11.4.3 Output

The screen display shows:

- the number of cycles in the signal
- the fatigue life as repeats of the signal (mean life)
- the fatigue life as repeats of the signal (design criteria)

The output files are:

- a Rainflow cycles histogram (.cyh)
- the damage histogram, mean life (.dmn)
- the damage histogram, design criterion (.dsd)
- a file of time-correlated fatigue damage, design criterion (.dad)
- a file of the signal in a fast-plot format (.das)
- a sensitivity file of factor vs life (.ssd)

The cycles and damage histograms are cycle range-mean histograms, in the same units as the signal, 32 bins x 32 bins, scaled to include all cycles. The cycle histogram may be used as an input file for the programs which provide fatigue analysis of cycle histograms

The time-correlated damage file gives an indication of whereabouts in time the fatigue damage occurs.

The fast-plot signal file contains 2048 data points which provide the same plot display (if not zoomed) as the full signal file.

11.4.4 Technical Data

Input signals may be uniaxial strain or stress time histories.

Rosette gauge data may be input, using the **Strain Gauge Rosette Analysis** module to produce a time history of the principal stress or strain which lies between $\pm 45^\circ$ of a line perpendicular to the weld. Note that this file must be produced from the rosette gauge channels before they are peak picked, but the resulting output file from Strain Gauge Rosette Analysis may be peak picked before being input to this program.

The program first searches for the absolute maximum value in the selected section of the signal (positive or negative).

The program then takes each data point and checks if it is a turning point (a peak or valley). For each turning point, the program checks if it has closed a cycle. For each closed cycle the endurance N_f cycles is calculated. The cycle and its damage are added to the output histograms.

At the end of the selected section of the signal, the program returns to the start point of the section, and carries on the analysis until the absolute maximum data point is reached again.

The calculated fatigue damage for each cycle is summed and used to calculate the fatigue life.

To form the time-correlated damage file, as each cycle is closed, the times for the three points which form the cycle are used to position the fatigue damage in time. Half the damage for the cycle is presumed to occur mid-way between the first two points, and the other half of the damage is presumed to occur mid-way between the 2nd and 3rd points. The damage is added to any previously calculated damage at these points.

Note that if the input signal is a peak/valley file, the time axis of the time-correlated damage file has no meaning.

The program calculates fatigue endurance using the BS5400 stress-life curves. These are equivalent to the endurance curves in BS7608. Each endurance is calculated from the equation for the curve.

Fatigue lives are calculated using Miner's rule, that

$$damage = \sum \frac{1}{N_f}$$

for each cycle

and that fatigue failure occurs when $total\ damage = 1.0$,

so that the life in repeats of the signal is

$$life = \frac{1.0}{\sum \frac{1}{N_f}}$$

Fatigue lives are calculated for two criteria - the mean life defined by the S-N curve, and the curve corrected to the specified design criteria. The fatigue data in BS5400 normally allows for the stress concentration produced at the weld, and so the stress concentration factor K_t used in the analysis will normally be 1.0. Some component geometry details or other factors may produce an additional stress concentration at the weld, in which case a factor greater than 1 should be used. The stresses are multiplied by the value of the stress concentration that is entered.

The weld classification is defined by a letter - B,C,D,E,F,F2 or G. (see section 11 of the Fatigue Theory Reference Manual for details of the weld classification procedure.)

The design criteria is defined as the number of standard deviations from the mean life. Any value will be accepted by the program. Examples are:

Design criteria	Probability of failure before the calculated life (%)
0	50
-2	2.3
-3	0.14

11.5 BS5400 Welded Joints from Histograms

11.5.1 Function

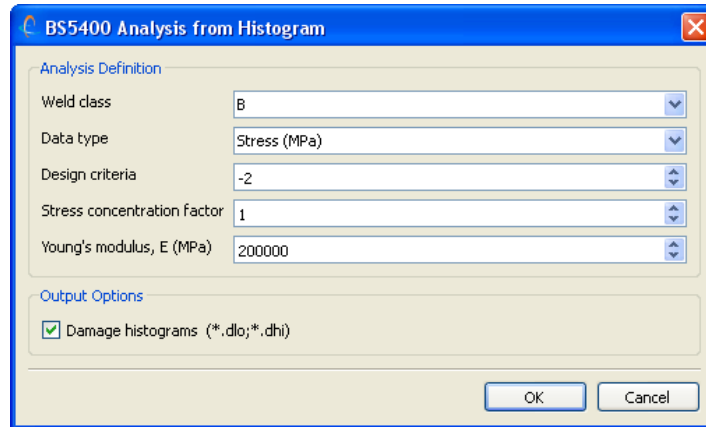
Calculates fatigue lives from a Rainflow cycle histogram, using the fatigue life data for welded joints in BS5400 Part 10:1980.

11.5.2 Operation

Select:

Gauge Fatigue >> BS5400 Welded Joints from Histograms...

This displays the following dialogue:



The analysis definition can be configured.

11.5.3 Output

The screen display shows:

- the number of cycles in the histogram
- the fatigue life as repeats of the histogram (upper estimate)
- the fatigue life as repeats of the histogram (lower estimate)

The output files written to the Generated Results contain:

- the name of the cycle histogram
- the weld classification
- the design criteria
- the stress concentration factor
- the elastic modulus (Young's modulus) (Note : this is only used for strain input data)
- the name of the results file
- the number of cycles in the histogram
- the fatigue life as repeats of the histogram (upper estimate)
- the fatigue life as repeats of the histogram (lower estimate)

Two fatigue lives are calculated. The first uses the largest strain range represented by each bin in the histogram, and provides the most conservative life estimate. This is written to the output damage. histogram, extension `.dhi`. The second estimate uses the smallest strain range represented by each bin in the histogram, and provides the least conservative life estimate. This is written to the output damage histogram, extension `.dlo`.

11.5.4 Technical Data

The input cycles histogram will have been produced by a cycle counting operation, either in software or by a hardware Rainflow counter. It must be a matrix of cycle ranges and cycle means, and be a strain cycle histogram in units of micro-strain, or a stress cycle histogram in units of MPa. The default file extension is `.cyh`.

A strain histogram is converted into stresses using the elastic modulus.

$$S = E e$$

where S is the stress

e is the strain

E is the elastic modulus

The remainder of the technical background is defined in section 11.4.4.

11.6 Strain Gauge Rosette Analysis

11.6.1 Function

The program takes 3 channels of strain gauge rosette data and calculates the principal strains or stresses and the angle between the first strain gauge and the first principal strain or stress. Output is 4-channels of data. For strain output, the 4th channel contains the principal strain of numerically largest magnitude. For stress output, the 4th channel contains the value of the principal stress within $\pm 45^\circ$ of the first strain gauge arm. This stress can be used as input to the welded joint fatigue programs. The principal values and angles can be plotted or cross-plotted in *fe-safe* (see section 7)

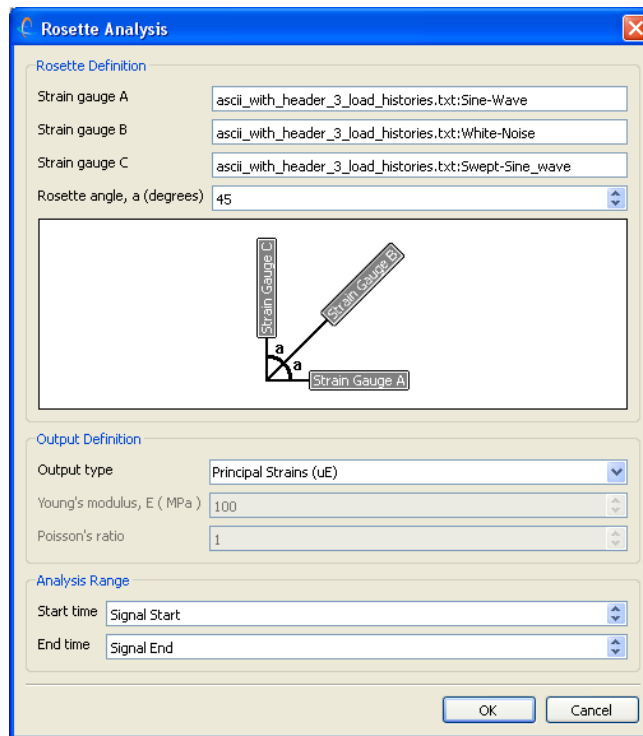
11.6.2 Operation

Select three channels from the **Loaded Data Files** window. Then select:

Gauge Fatigue >> Strain Gauge Rosette Analysis

This displays the following dialogue:

Fatigue analysis from measured signals [1] : using S-N curves



Select the Rosette Angle (45 or 120 degrees), the output type, either Principal Strains or Principal Stresses. If Principal Stresses is selected both Young's Module and Poisson's Ratio can be defined.

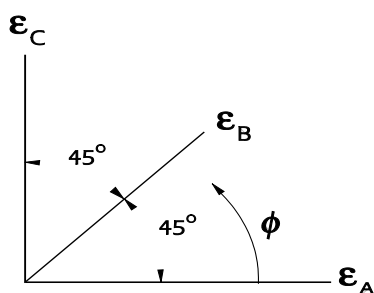
11.6.3 Output

Four output files with extension .DAC, containing:

- channel 1: the maximum principal strain or stress
- channel 2: the minimum principal strain or stress
- channel 3: the angle between the maximum principal strain and the first arm of the strain gauge (positive anti-clockwise)
- channel 4: for strain output, the numerically largest value of strain channels 1 and 2
- channel 4: for stress output, the value of the principal stress within $\pm 45^\circ$ of the first strain gauge arm.

11.6.4 Technical data

For a 45° rosette, the principal strains ϵ_1 and ϵ_2 are given by



$$\varepsilon_1 = \frac{1}{2}(\varepsilon_A + \varepsilon_C) + \frac{1}{2}\sqrt{(\varepsilon_A - \varepsilon_C)^2 + (2\varepsilon_B - \varepsilon_A - \varepsilon_C)^2}$$

$$\varepsilon_2 = \frac{1}{2}(\varepsilon_A + \varepsilon_C) - \frac{1}{2}\sqrt{(\varepsilon_A - \varepsilon_C)^2 + (2\varepsilon_B - \varepsilon_A - \varepsilon_C)^2}$$

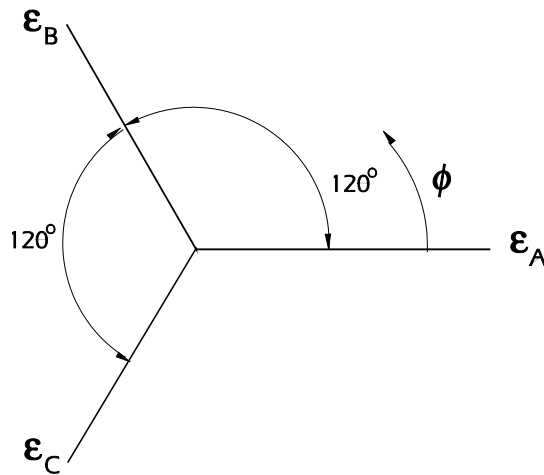
$$\tan 2\phi = \frac{2\varepsilon_B - \varepsilon_A - \varepsilon_C}{\varepsilon_A - \varepsilon_C}$$

For a 120° rosette, the principal strains ε_1 and ε_2 are given by

$$\varepsilon_1 = \frac{1}{3}(\varepsilon_A + \varepsilon_B + \varepsilon_C) + \frac{\sqrt{2}}{3}\sqrt{(\varepsilon_A - \varepsilon_B)^2 + (\varepsilon_B - \varepsilon_C)^2 + (\varepsilon_C - \varepsilon_A)^2}$$

$$\varepsilon_2 = \frac{1}{3}(\varepsilon_A + \varepsilon_B + \varepsilon_C) - \frac{\sqrt{2}}{3}\sqrt{(\varepsilon_A - \varepsilon_B)^2 + (\varepsilon_B - \varepsilon_C)^2 + (\varepsilon_C - \varepsilon_A)^2}$$

$$\tan 2\phi = \frac{\sqrt{3}(\varepsilon_C - \varepsilon_B)}{2\varepsilon_A - \varepsilon_B - \varepsilon_C}$$



The principal stresses are calculated from the principal strains using

$$\sigma_1 = \frac{E}{1 - \nu^2} (\varepsilon_1 + \nu\varepsilon_2)$$

Fatigue analysis from measured signals [1] : using S-N curves

$$\sigma_2 = \frac{E}{1 - \nu^2} (\varepsilon_2 + \nu\varepsilon_1)$$

where E is the elastic modulus

ν is Poisson's ratio,

ε_1 and ε_2 are in units of microstrain ($\mu\varepsilon$),

and the units of σ_1 and σ_2 are in the same units as E .

BS5400 analysis of welded joints allows input of multiaxial stresses, and recommends using the largest value of principal stress which is within $\phi = \pm 45^\circ$ of a line perpendicular to the weld. The Strain Gauge Rosette Analysis module can calculate this value, assuming that e_A is the strain gauge arm perpendicular to the weld.

12 Fatigue analysis from measured signals [2] : uniaxial strain-life and multiaxial methods

This section discusses the strain-life methods for evaluating fatigue life from measured strains. See the Fatigue Theory Reference Manual section 2 for the technical background to strain-life fatigue analysis.

12.1 Technical background

The following fatigue life relationships are used.

Strain-life
$$\frac{\Delta\varepsilon}{2} = \frac{\sigma'_f}{E} (2N_f)^b + \varepsilon'_f (2N_f)^c$$

Smith-Watson-Topper
$$\frac{\Delta\varepsilon}{2} \sigma_{\max} = \frac{(\sigma'_f)^2}{E} (2N_f)^{2b} + \sigma'_f \varepsilon'_f (2N_f)^{b+c}$$

Morrow
$$\frac{\Delta\varepsilon}{2} = \frac{(\sigma'_f - \sigma_m)}{E} (2N_f)^b + \varepsilon'_f (2N_f)^c$$

where

- $\Delta\varepsilon$ is the strain range for the cycle
- σ_{\max} is the maximum stress in the cycle
- σ_m is the mean stress in the cycle
- $2N_f$ is the endurance in reversals (half cycles)

and the materials properties are

- σ'_f the fatigue strength coefficient
- ε'_f the fatigue ductility coefficient
- n' the strain hardening exponent
- K' the strain hardening coefficient
- b Basquin's exponent
- c the Coffin-Manson exponent

Local strains are calculated from the nominal strains using Neuber's rule and the stress concentration factor K_t

$$\Delta\sigma\Delta\varepsilon = K_t^2 \Delta S \Delta e$$

where $\Delta\varepsilon$ is the local strain range

- $\Delta\sigma$ is the local stress range
- Δe is the nominal strain range
- ΔS is the nominal stress range

The cyclic stress-strain equation

$$\varepsilon = \frac{\sigma}{E} + \left(\frac{\sigma}{K'}\right)^{\frac{1}{n'}}$$

and the hysteresis loop equation (from Masing's hypothesis)

$$\Delta\varepsilon = \frac{\Delta\sigma}{E} + 2\left(\frac{\Delta\sigma}{2K'}\right)^{\frac{1}{n'}}$$

Fatigue lives are calculated using Miner's rule, that for each cycle, damage = $\frac{1}{N_f}$

and that fatigue crack initiation occurs when total damage = 1.0

so that the life in repeats of the signal is $\frac{1}{\sum \frac{1}{N_f}}$

12.2 General analysis properties

These functions are accessed from the **Gauge Fatigue** menu.

Input strains can be a micro-strain history, or a micro-strain-based cycle histogram.

Analysis can use a Smith-Watson-Topper, Morrow or no mean stress correction.

Sensitivity analysis can be carried out to investigate the effect of different stress concentrations or signal scale factors.

Cycle histograms produced by the signal functions in this section can be used as input to the histogram analysis functions, as can cycle histograms from the Rainflow cycle counting program. It may be quicker for 'what-if' analysis to use a histogram input, then confirm the results with analysis of the full signal.

A peak-picked strain signal can be used as input instead of a strain signal. Analysis will be quicker, but the time-correlated damage file will not have a true time axis.

If nominal strains have been measured a stress concentration factor can be entered.

Local measured strains can be converted from one material to another.

12.3 Local Strain Analysis from Time Histories

12.3.1 Function

Calculates fatigue lives from a micro-strain-time history, using either the uniaxial strain-life relationship or the uniaxial Smith-Watson-Topper life relationship. The sensitivity of the analysis to stress concentration and signal scale factor can also be calculated. Input signals may be a strain-time signal or a peak-picked strain history.

12.3.2 Operation

Select: **Gauge Fatigue >> Uniaxial Strain Life from Time Histories...**

This displays the following dialogue:

The dialog box 'Uniaxial Strain Life from Time Histories' is shown with the following configuration:

- Material:**
 - Current database: C:\installations\user_directories\fe-safe\local.dbase
 - Material: Honeywell-Example
- Analysis Definition:**
 - Stress concentration factor: 1
 - Mean stress correction: None
- Scale Sensitivity Analysis:**
 - Perform sensitivity analysis
 - Number of scale values: 6
 - Scale range: 0.5 to 5
- Analysis Range:**
 - Start time: Signal Start
 - End time: Signal End
- Output Options:**
 - Cycle range-mean histogram (*.cyh)
 - Damage range-mean histogram (*.dah)
 - Stress-strain hysteresis loop (*.evs)
 - Quick-plot signal file (*.das)
 - Time-correlated damage file (*.dad)

Figure 12.3.2-1

The analysis definition can be configured, including the scale sensitivity analysis parameters if required.

12.3.3 Output

The screen display shows:

- the fatigue life as repeats of the signal
- the number of cycles in the signal
- the name of the signal

The output written to the Generated Results contains:

- the signal name
- the analysis type
- the material
- the start time
- the end time
- the stress concentration factor
- the sensitivity range
- the name of the results file
- the number of cycles in the signal
- the fatigue life as repeats of the signal

The output files are:

- a Rainflow cycles histogram (extension [.cyh](#))
- the damage histogram (extension [.dah](#))
- a file containing the 10 largest stress-strain hysteresis loops ([.evs](#))
- a file of the signal in a fast-plot format ([.das](#))
- a file of time-correlated fatigue damage ([.dad](#))
- a file of sensitivity factor vs. life ([.smn](#))

The cycles and damage histograms are cycle range-mean histograms, in terms of nominal strain, 32 bins x 32 bins, scaled to include all cycles. The cycle histogram may be used as an input file for the programs which provide fatigue analysis of cycle histograms

The time-correlated damage file gives an indication of whereabouts in time the fatigue damage occurs.

The fast-plot signal file contains 2048 data points which provide the same plot display (if not zoomed) as the full signal file.

12.3.4 Technical data

The program first searches for the absolute maximum value in the selected section of the signal (positive or negative). This data point is converted into local stress and strain using the cyclic stress-strain curve, the stress concentration factor, and Neuber's rule.

The program then takes each data point and checks if it is a turning point (a peak or valley). For each turning point, the program checks if it has closed a cycle. For each closed cycle the endurance is calculated. The cycle and its damage are added to the output histograms. Once all the cycles closed by the data point have been analysed, the data point is converted into local stress and strain using the hysteresis loop curve, the stress concentration factor, and Neuber's rule.

At the end of the selected section of the signal, the program returns to the start point of the section, and carries on the analysis until the absolute maximum data point is reached again.

The calculated fatigue damage for each cycle is summed and used to calculate the life to crack initiation.

To form the time-correlated damage file, as each cycle is closed, the times for the three points which form the cycle are used to position the fatigue damage in time. Half the damage for the cycle is presumed to occur mid-way between the first two points, and half of the damage is presumed to occur mid-way between the 2nd and 3rd points. The damage is added to any previously calculated damage at these points.

Note that if the input signal is a peak/valley file, the time axis of the time-correlated damage file has no meaning.

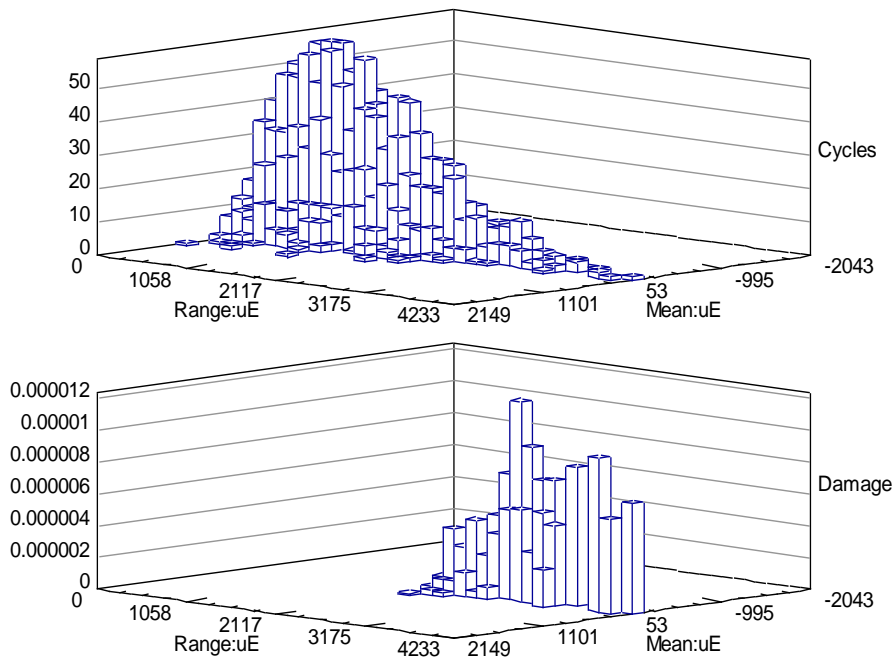


Figure 12.3.4-1 Cycle and damage histograms

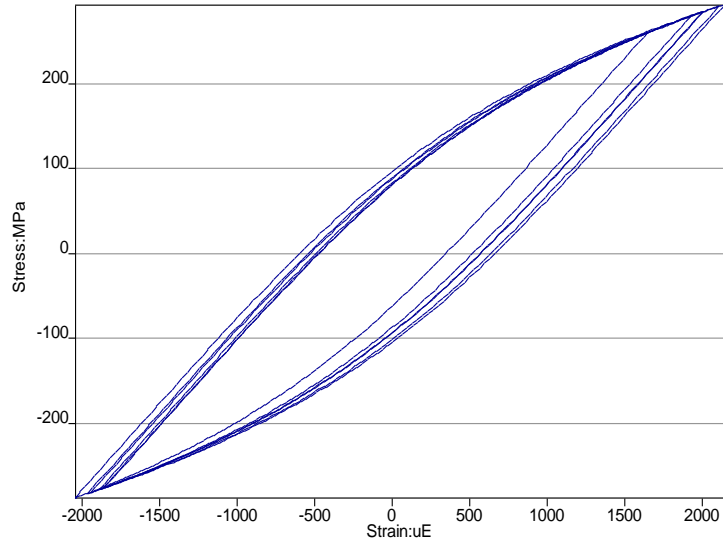


Figure 12.3.4-1 The 10 largest hysteresis loops

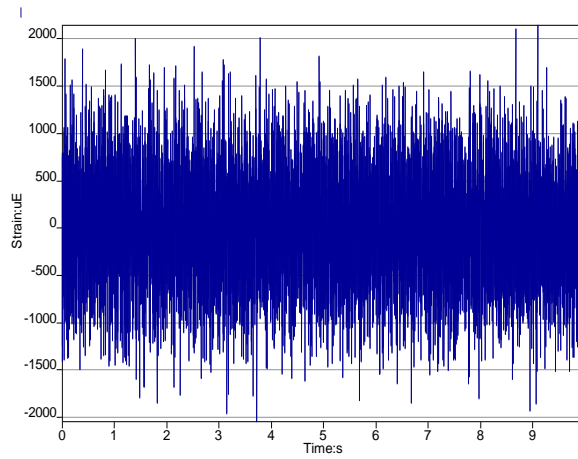


Figure 12.3.4-2 Pictorial summary of input data

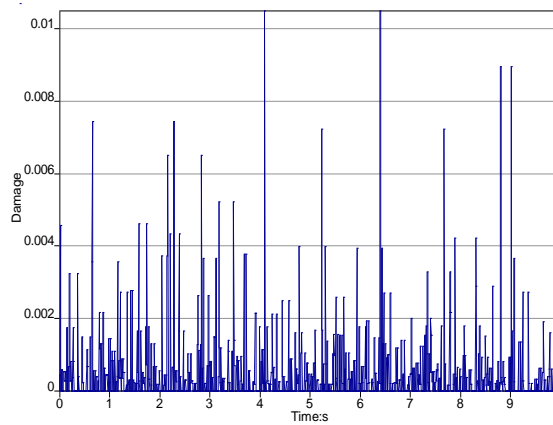


Figure 12.3.4-3 Time-correlated fatigue damage

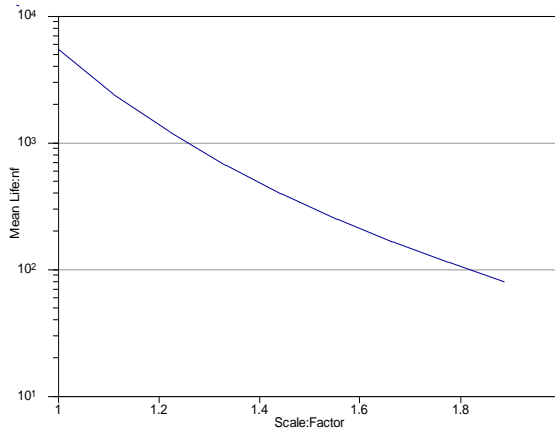


Figure 12.3.4-4 Scale factor sensitivity

12.4 Local Strain Analysis from Cycle Histograms

12.4.1 Function

Calculates fatigue lives from a micro-strain Rainflow cycles histogram, using the strain-life relationship. Analysis can use a Smith-Watson-Topper, Morrow or no mean stress correction.

12.4.2 Operation

Select: **Gauge Fatigue >> Uniaxial Strain Life from Histograms...**

This displays the following dialogue:

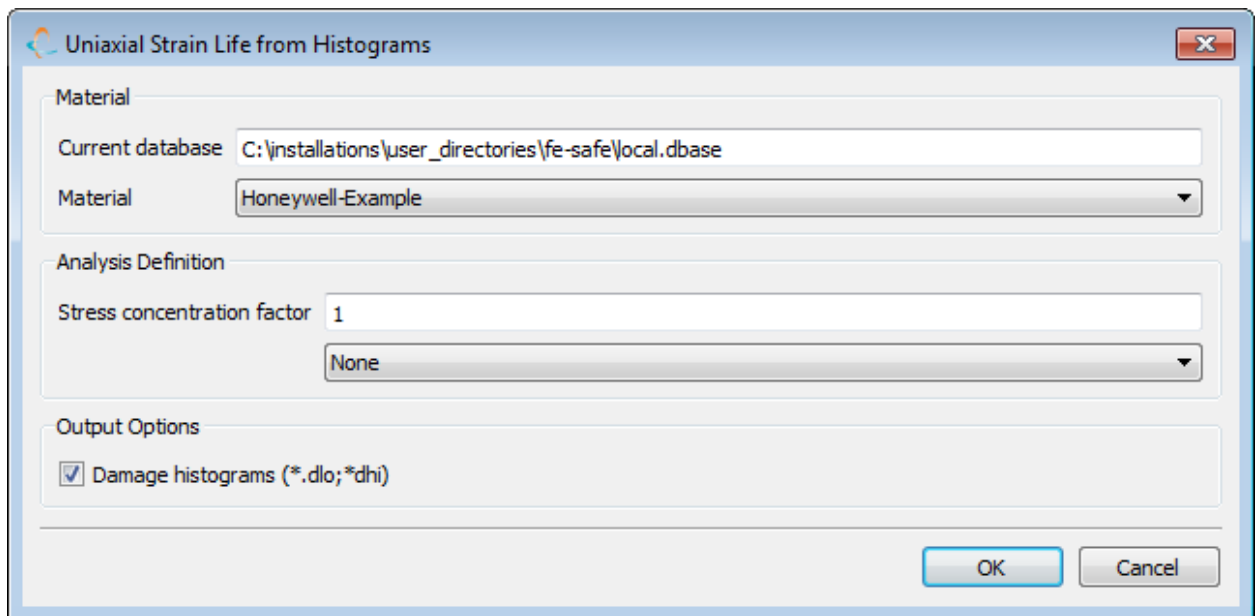


Figure 12.4.2-1

The analysis definition can be configured.

12.4.3 Output

The screen display shows:

- the most conservative and least conservative estimates of the fatigue life as repeats of the histogram;
- the number of cycles in the histogram;
- the name of the signal.

The output written to the Generated Results contains:

- the name of the cycle histogram;
- the analysis type;
- the material;
- the stress concentration factor;
- the name of the results files;
- the number of cycles in the histogram;
- the most conservative and least conservative estimates of the fatigue life as repeats of the histogram.

The following files are created:

- two damage histograms, containing the upper (most conservative) and lower (least conservative) estimates of fatigue damage (extensions `.dhi` and `.dlo`.)

12.4.4 Technical data

The input cycles histogram will have been produced by a cycle counting operation, either in software (rainflow cycle counting or the fatigue programs which perform fatigue analysis of signals) or by a hardware Rainflow counter. It must be a matrix of cycle ranges and cycle means. The default file extension is `.cyh`.

Two fatigue lives are calculated. The first uses the largest strain range represented by each bin in the histogram, and assumes that the upper tip of each cycle touches the outside loop. This provides the most conservative life estimate. This is written to the output damage histogram, extension `.dhi`. The second estimate uses the smallest strain range represented by each bin in the histogram, and assumes that the lower tip of each cycle touches the outside loop. This provides the least conservative life estimate. This is written to the output damage histogram, extension `.dlo`.

The Fatigue Theory Reference Manual, Section 2.11 describes the algorithm in detail.

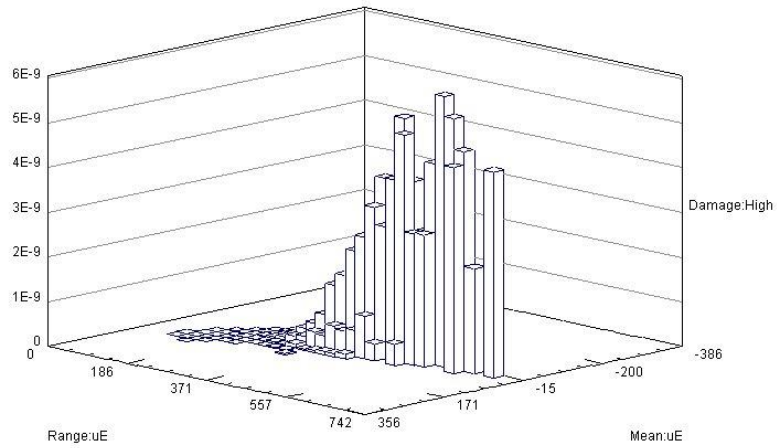


Figure 12.4.4-1 Damage histogram (.dhi)

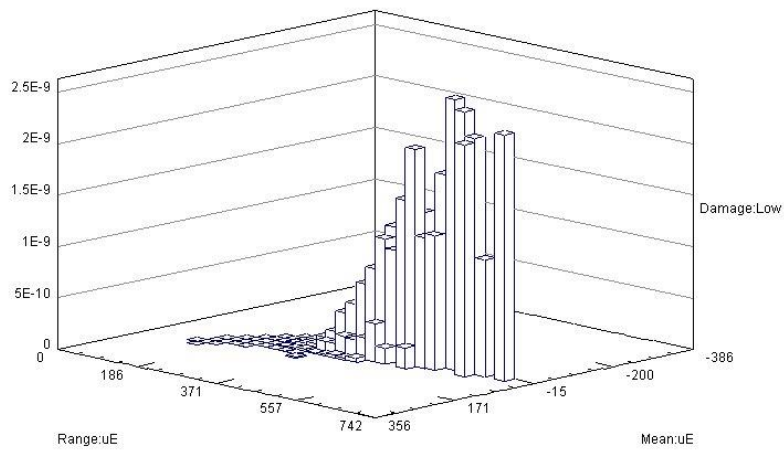


Figure 12.4.4-2 Damage histogram (.dlo)

12.5 Quick Look Local Strain Analysis

12.5.1 Function

For a peak/valley pair of nominal strains and an optional stress concentration factor, the program calculates the local stress and strain for the peak and valley, and the endurance of the cycle using the strain-life and Smith-Watson-Topper relationships.

12.5.2 Operation

Select:

Gauge Fatigue >> 'Quick Look' Strain Life...

This displays the following dialogue:

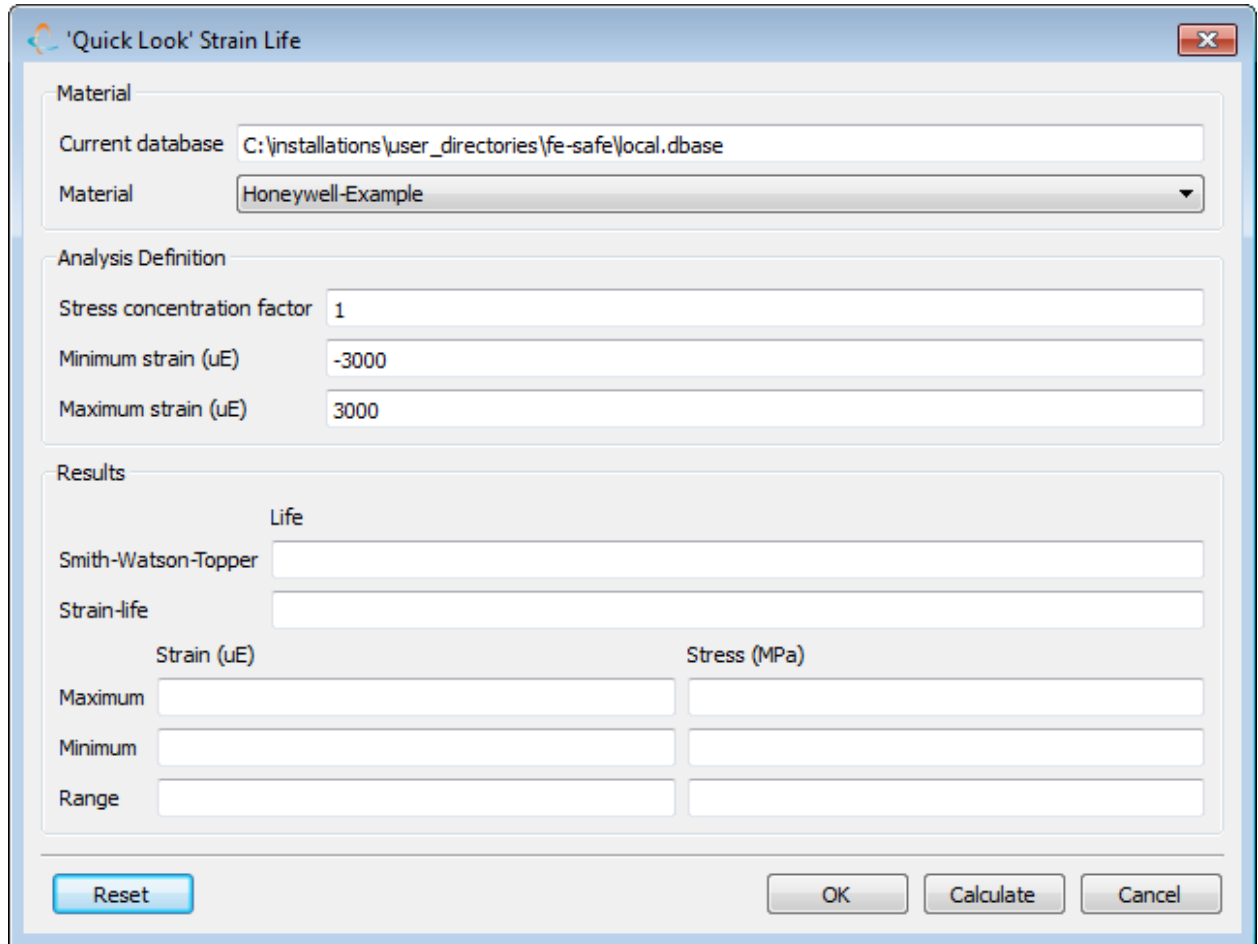


Figure 12.5.2-1

12.5.3 Output

The results are displayed in the Results area at the bottom of the dialogue box.

12.5.4 Technical data

The program calculates fatigue damage using the Smith-Watson-Topper relationship, and the strain-life relationship. The strain of the largest absolute magnitude is converted into local stress and strain using the cyclic stress-strain curve, the stress concentration factor, and Neuber's rule. The remaining strain is converted using the hysteresis loop curve, the stress concentration factor and Neuber's rule.

12.6 Local Strain Material Conversion

12.6.1 Function

Converts a time history of local strains measured on one material, into the equivalent local strain history for another material. Input signals may be a strain-time signal or a peak-picked strain history. Strain histories measured using a strain-gauge rosette should not be analysed by this program. This operation is essential if local strains have been measured in a notch and the user requires calculating fatigue lives for the same geometry in a different material. It would be prudent to use this program with similar types of material, for example two steels, or two aluminium alloys, rather than with two very dissimilar materials

12.6.2 Operation

Highlight a time history signal in the **Loaded Data Files** window, then select:

Gauge Fatigue >> Uniaxial Local Strain Material Conversion...

The strain history **MUST** have units of μE .

This displays the following dialogue:

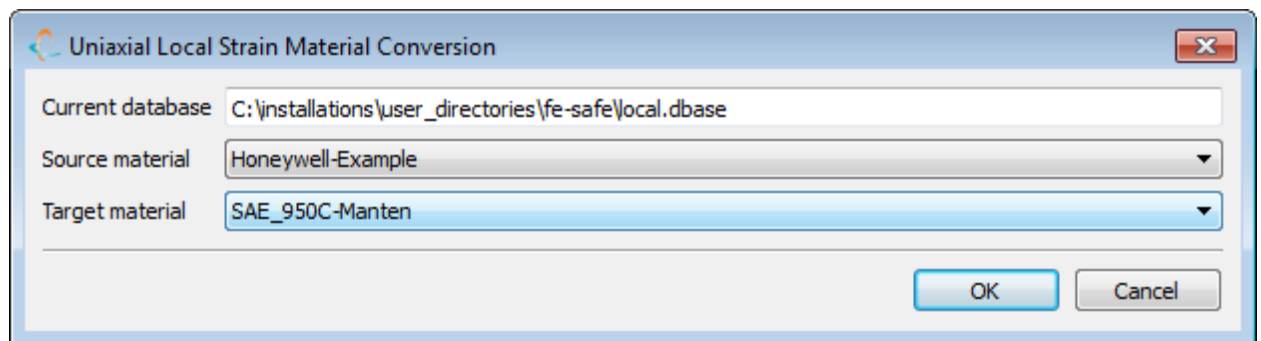


Figure 12.6.2-1

Select a source material and target material (both must be from the current database).

12.6.3 Output

A time history file containing the converted signal.

12.6.4 Technical data

Because local measured strains may include the effects of cyclic plasticity, they are particular to the material on which they were measured, and cannot be used to investigate alternative materials. The program can be used to convert local strain histories from one material to another. The strain-time signals will normally have units of micro-strain.

Equivalent elastic stress/strains are calculated from the local strains using Neuber's rule implemented as

$$\sigma^e \varepsilon^e = \sigma \varepsilon$$

where ε, σ are the measured strain and associated stress in the first material

and σ^e, ε^e are the stress and strain in an elastic material (the 'nominal stress and strain')

Local strains for the second material are then calculated from the nominal stress/strains using Neuber's rule implemented as

$$\sigma \varepsilon = \sigma^e \varepsilon^e$$

where ε, σ are now the strain and associated stress in the new material.

The program first searches for the absolute maximum value in the selected section of the signal (positive or negative). This data point is converted into nominal stress/strain using Neuber's rule and the cyclic stress-strain curve for material 1, and then converted into stress and strain for material 2.

The program then takes each data point and checks if it has closed a cycle. If not, the data point is converted into nominal stress/strain using the hysteresis loop curve for material 1, and then into local stress/strain using the hysteresis loop for material 2.

If a cycle has been closed, material memory is used to position the data point on a new hysteresis loop, and the nominal strain calculated.

At the end of the selected section of the signal, the program returns to the start point of the section, and carries on the conversion until the absolute maximum data point is reached.

See the Fatigue Theory Reference Manual, section 2.9.4 (and particularly *Figure 2.43* in the Fatigue Theory Reference Manual) for further details of this method.

12.7 Multiaxial Fatigue from Time Histories

12.7.1 Function

Calculates fatigue lives from 3 channels of strain or micro-strain strain gauge rosette data. The available algorithms are normal strain or Brown Miller with the Morrow or the user-defined mean stress corrections, and the stress-life algorithm for S-N curves with the Goodman, Gerber or user-defined mean stress corrections

12.7.2 Operation

Select three channels from the **Loaded Data Files** window. Then select:

Gauge Fatigue >> Multiaxial Fatigue from Time Histories...

This displays the following dialogue:

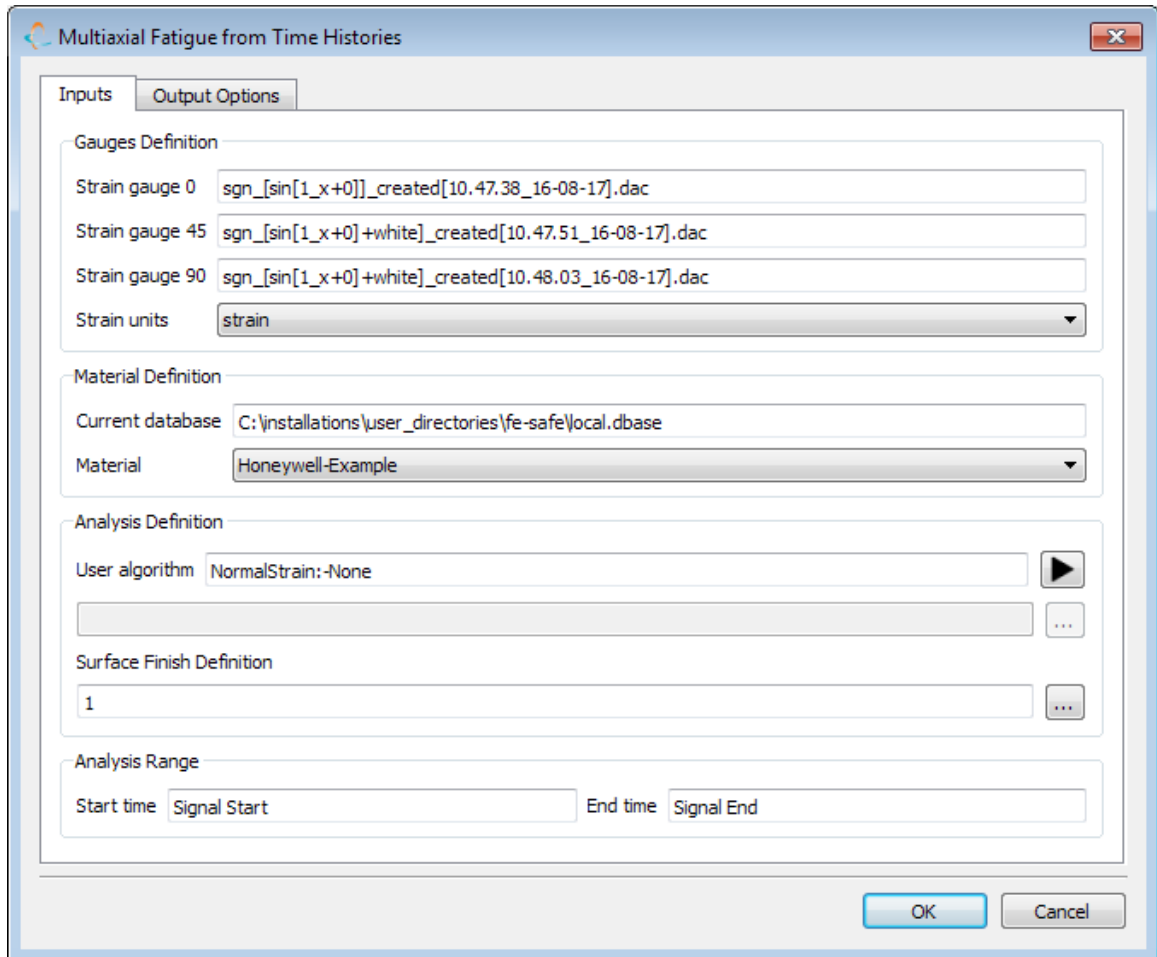


Figure 12.7.2-1

In the **Gauges Definition** group select the units used in the time histories.

Select the required outputs in the **Output Options** tab. The x-axis of Histogram plots can either be the mean of the damage parameter or the mean stress. See figure 12.6.3-1.

Select the desired algorithm by clicking on the **User algorithm** browse button, which displays the following menu:

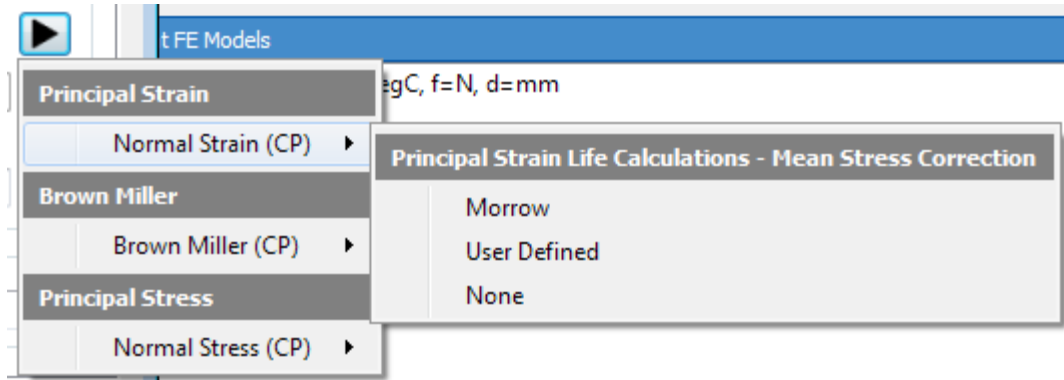


Figure 12.7.2-2

Details on Normal Strain, Brown Miller and Normal Stress analyses can be found in sections 14.14, 14.16 and 14.7 respectively. Note that in this module the Normal Stress algorithm uses S-N curve data to evaluate fatigue life.

If a user-defined mean stress correction is chosen, the **User Defined Mean Stress Correction** browse button can be used to select a file. See section 14.9 for an explanation and Appendix E for the file syntax. For all other mean stress corrections, see the sections for the main algorithms noted above.

Press the **Surface Finish Definition** browse button to select a surface finish. This is the same as the dialogue as described in 5.5.4.

12.7.3 Output

The screen display shows:

- the sources for the time history inputs;
- the critical plane angle at which the most damage occurs. This is measured from the first input channel;
- the number of cycles on the critical plane;
- the life on the critical plane. This is the number of repeats of the time histories.

The output written to the `results.lst` contains:

- the name of the strain gauge time histories;
- the analysis type;
- the mean stress correction used;
- the material;
- the stress concentration factor;
- the angle of the critical plane from the first input;
- the number of cycles on the critical plane;
- the life on the critical plane as repeats of the strain gauge time histories;
- the name of the results files.

An example:

```
Signal name (0 deg) : C:\safeResultsArchive\test49.dac, 1
```

```

Signal name (45 deg)      : C:\safeResultsArchive\test49.dac, 2
Signal name (90 deg)     : C:\safeResultsArchive\test49.dac, 3
Analysis type            : Principle Strain Life Analysis from Strain Gauge Histories
Material                 : SAE_950C-Manten
                        (from P:\data\safe4test\test.dbase)

Analysis start time     : Signal Start
Analysis end time       : Signal End
Stress concentration    : 1
Critical Plane Angle    : 0 degrees
Mean life               : 3736.4 repeats
Number of cycles        : 794
Results files : c:\safeResultsArchive\MultiFatigue_Cycle_Histogram_[test49.dac_0_Deg].cyh
                c:\safeResultsArchive\MultiFatigue_Damage_Histogram_[test49.dac_0_Deg].dah
                c:\safeResultsArchive\MultiFatigue_Damage_vs_Time_[test49.dac_0_Deg].dad
                c:\safeResultsArchive\MultiFatigue_Damage_vs_Angle_[test49.dac_0_deg].dva
    
```

The following files are created (when enabled):

- a Rainflow cycles histogram (extension `.cyh`) See Figure 12.6.4-1.
- a Rainflow damage histogram (extension `.dah`) See Figure 12.6.4-1.
- a damage versus angle plot (extension `.dav`) See Figure 12.6.4-2.
- a damage versus time plot (extension `.dad`) See Figure 12.6.4-4.

For a Brown Miller analysis two damage and two cycle histograms will be produced. They will contain the direct and shear strains on the critical plane. See Figure 12.7.3-1. There will be three angle and three time plots, one for the 1-2, 2-3 and 1-3 planes. See Figure 12.7.3-2.

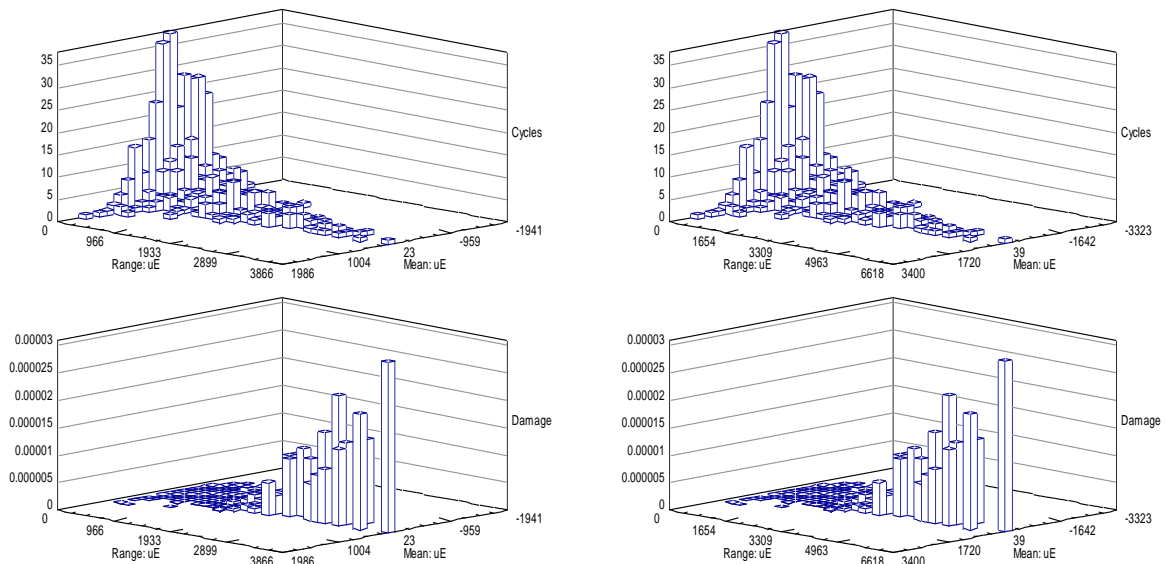


Figure 12.7.3-1 Brown Miller cycle and damage histograms for direct (left) and shear (right).

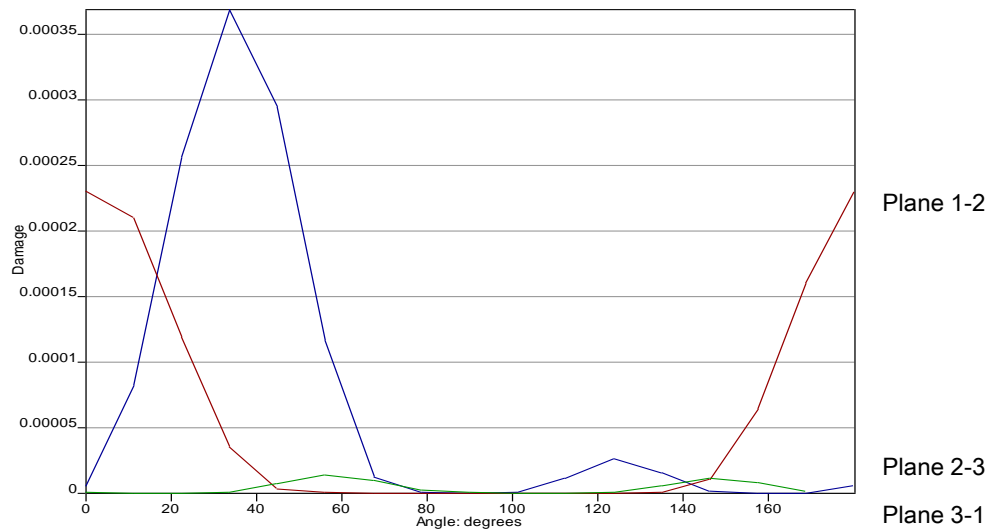


Figure 12.7.3-2 Brown Miller damage vs. angle overlay of all 3 planes.

The x-axis of histograms is either the mean normal stress of the cycle (in MPa) or the mean of the damage parameter (in units of MPa for the Normal Stress analysis and in units of micro-strain for the Normal Strain and Brown Miller analyses) as shown in Figure 12.7.3-3.

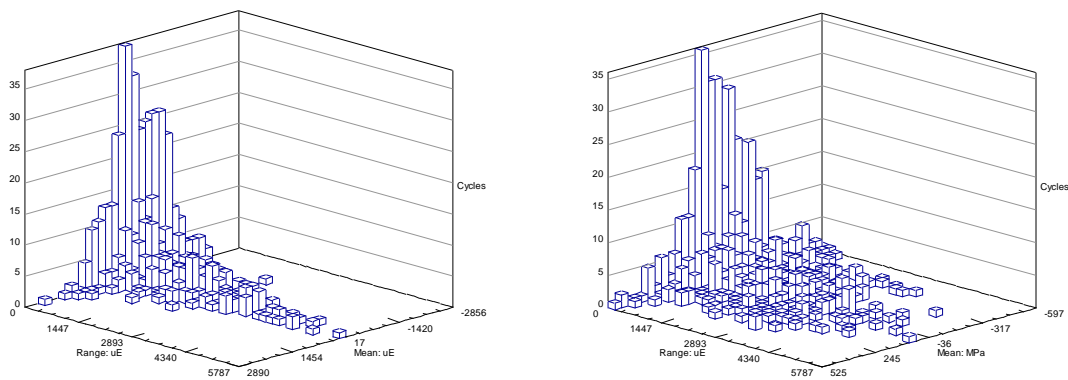


Figure 12.7.3-3 Cycle histograms from the same source. Left with mean damage parameter, right with mean stress.

12.7.4 Technical data

The input time histories must contain 45° strain gauge rosette data measured in micro-strain or strain. Due to the use of the critical plane procedure peak/valley and cycle omission gating should not be applied to strain gauge rosette signals.

Details on Normal Strain, Brown Miller and Normal Stress technical data can be found in sections 14.14, 14.16 and 14.7 respectively. In this module the Normal Stress algorithm uses S-N curve data to evaluate fatigue life.

Details on mean stress corrections can be found as follows: for Goodman and Gerber, section 14.3; for user-defined mean stress corrections; section 14.9; for Morrow applied to the Normal Strain and Brown Miller algorithms; sections 14.14 and 14.16.

The module calculates the principal strains at each point in time. Stresses are calculated using a kinematic hardening model. A critical plane procedure resolves the stresses and strains onto 16 planes (48 planes with Brown Miller algorithm), at 11.25° increments. For each plane, the calculated fatigue damage for each cycle is summed and used to calculate the life to crack initiation. The plane with the highest calculated fatigue damage is the critical plane. This defines the fatigue life. Output files for plotting are written for this plane.

Critical plane procedures are described in section 7.5 of the Fatigue Theory Reference Manual. Kinematic hardening models are described in section 7.7.2 of the Fatigue Theory Reference Manual.

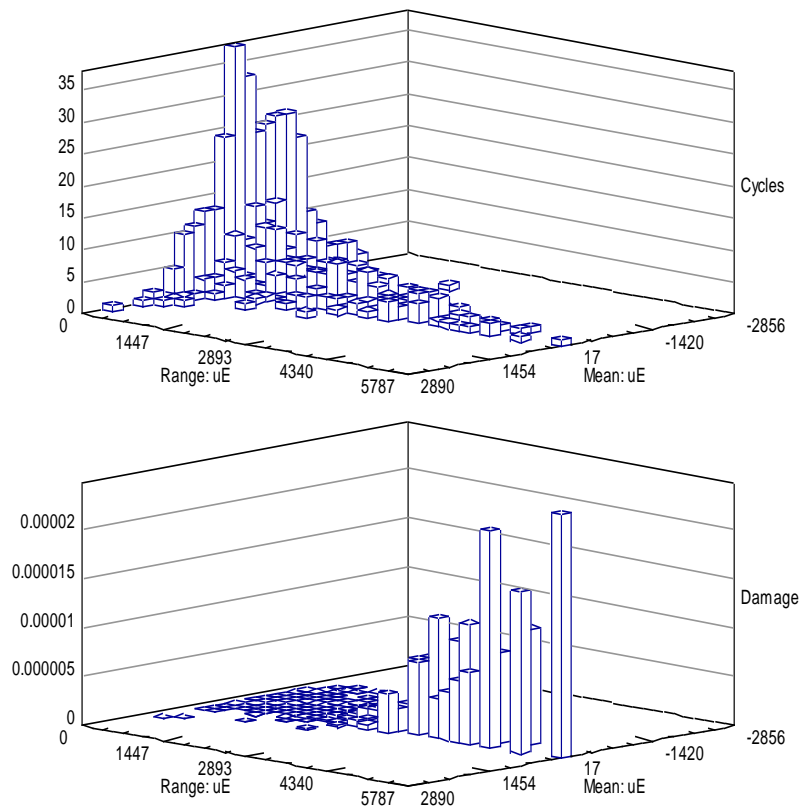


Figure 12.7.4-1 Cycle and damage histograms.

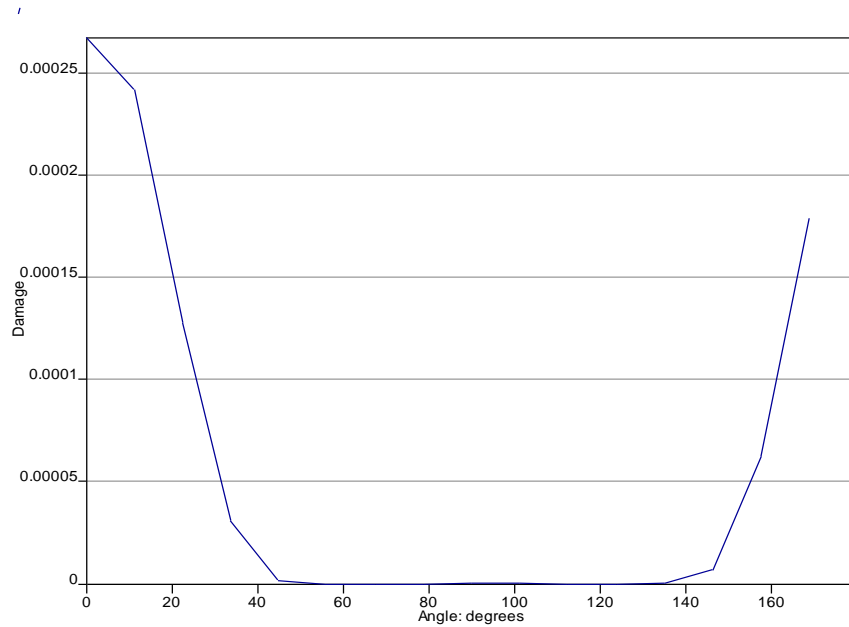


Figure 12.7.4-2 Damage versus angle plot.

To form the time-correlated damage file, as each cycle is closed, the times for the three points which form the cycle are used to position the fatigue damage in time. Half the damage for the cycle is presumed to occur mid-way between the first two points, and the other half of the damage is presumed to occur mid-way between the 2nd and 3rd points. The damage is added to any previously calculated damage at these points.

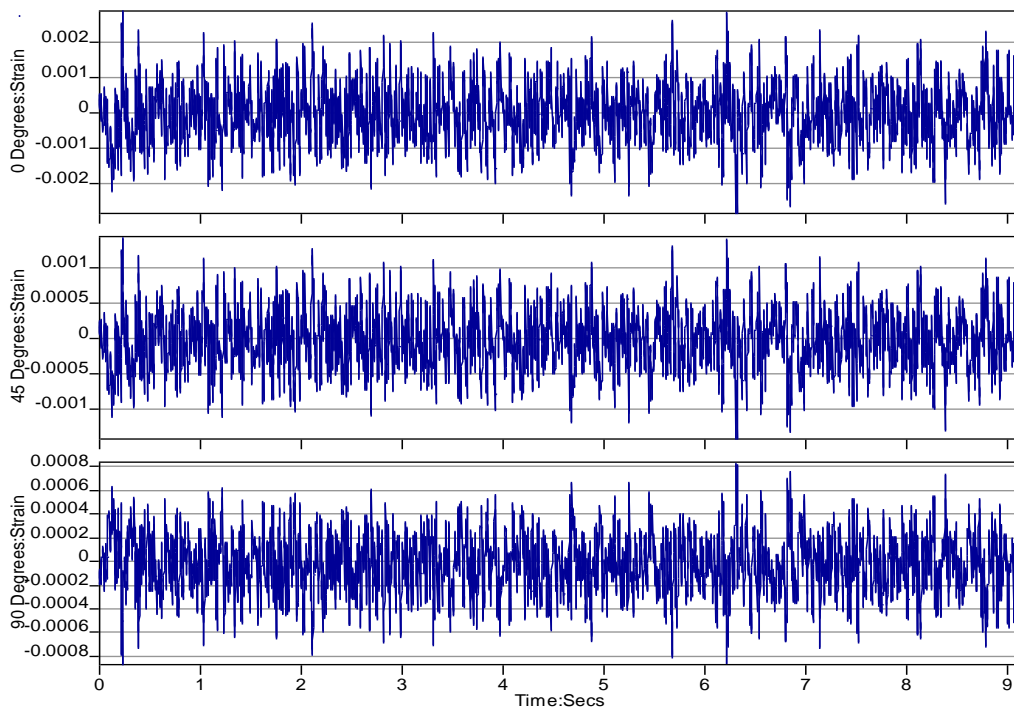


Figure 12.7.4-3 Pictorial summary of input data

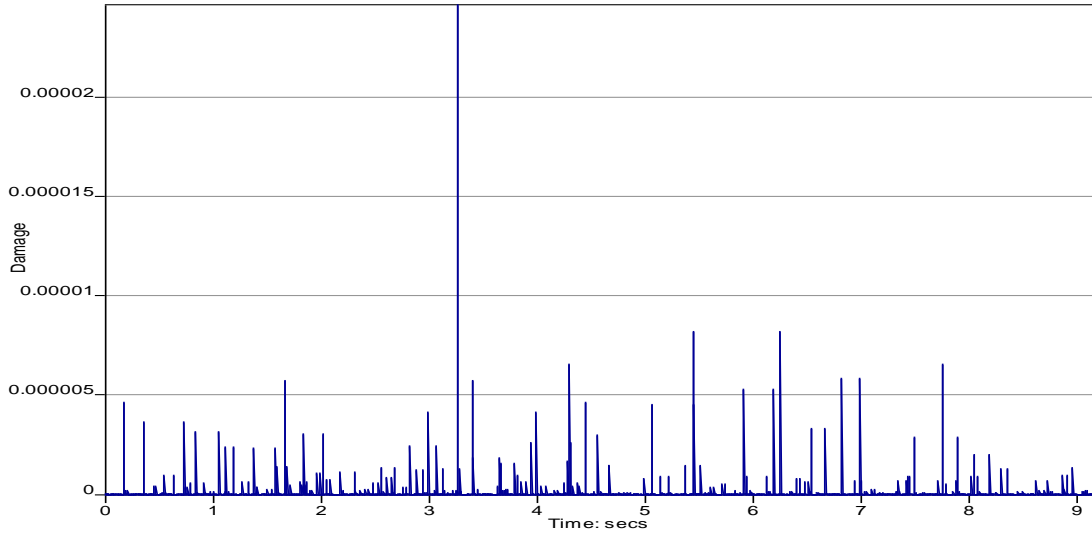


Figure 12.7.4-4 Time correlated damage.

13 Defining fatigue loadings

13.1 Loading methods

fe-safe offers comprehensive, flexible options for defining fatigue loading conditions. Very complex loading conditions can be defined. The following fatigue loading methods are supported:

- the time history of a component load can be applied to the results of an FEA analysis;
- time histories of multi-axis loading can be superimposed to produce a time history of the stress tensor at each location on the model (*fe-safe* supports over 4000 load histories of unlimited length);
- a sequence of FEA stresses, for example: the results of a transient analysis, the analysis of several rotations of an engine crankshaft or models which undergo several discrete loading conditions;
- block loading programmes, consisting of a number of blocks of constant amplitude or more complex cycles;
- complex test programmes and 'proving ground' sequences;
- high and low frequency loading can be superimposed with automatic sample rate matching by interpolation.

These fatigue loading conditions can be combined and superimposed with great flexibility. PSD's, dynamics, Rainflow matrices and other capabilities are also supported.

Reference should be made to section 5 for a description of the user interface and to section 7 for general file handling.

13.1.1 Simple loading

Constant amplitude loading. Each stress tensor in the FEA dataset is scaled by a simple cycle of constant amplitude. The load definition may contain a single constant amplitude cycle, or many cycles.

A constant amplitude loading can be defined directly as a series of numbers in the user interface, or using the load definition (LDF) file (section 13.9).

A single load history. A time history or loading can be applied to each stress tensor in the FEA dataset. The FEA results would represent the stresses for a 'unit' of the applied loading. The load history may be contained in a data file, or may be entered directly as a series of numbers in the user interface.

Rainflow cycle histograms can be exported directly to an LDF file (see 19.3.2), and a PSD of loading can be exported to an LDF file via a Rainflow cycle matrix (see 19.3.1).

The FEA stresses are scaled by the applied loading. In practice the stresses could be calculated for any value of applied load. However, two often-used conditions are:

- (a) the FEA stresses are calculated for a unit load, and the loading history contains load values.
- (b) the FEA stresses are calculated for the maximum load, and the load history represents each load as a proportion of the maximum load.

The fatigue life is calculated as a number of repeats of the defined loading. Optionally this may be converted into user-defined units (miles, hours, etc) – see section 13.2

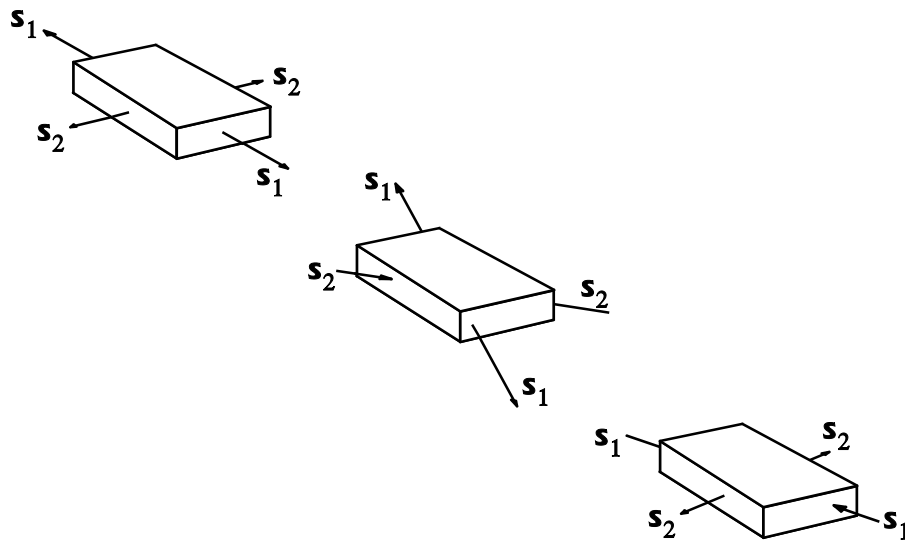
13.1.2 Multiple load history (scale-and-combine) loading

Imported FE data results files may contain several stress datasets, where each dataset contains the results of a 'unit' load analysis for a particular load direction. A load history can be applied to each stress dataset. For each node, the stress tensor from each stress dataset is scaled by the appropriate load history, and the results summed to provide a stress history for the combined loading. Principal stresses are then calculated for each data point from the time history of the stress tensor, and are resolved onto the surface of the component.

Load history (scale-and-combine) loadings can be defined directly (in the user interface), or using the load definition (LDF) file (section 13.9).

13.1.3 Dataset sequences

A dataset sequence is a sequence of stress datasets that define the variation in loading over time.



For example: a vehicle engine may have been analysed to provide FE results at 5° intervals of crank angle, through three revolutions of the crank shaft. The stresses will be contained in 216 stress datasets. These 216 sets of results can be chained together in sequence. *fe-safe* can analyse the sequence of datasets to calculate the fatigue life at each node.

The stress datasets can be applied in any order; can occur more than once in the sequence; and can have scale factors applied to them.

Complex sequences of stresses can be built up by superimposing load history (scale-and-combine) loadings and dataset sequences, providing that the sampling frequencies are the same. Additional scale factors, repeat counts and multiple sequences can also be incorporated.

Dataset sequence loading can be defined directly (in the user interface), or using the load definition (LDF) file (section 13.9).

13.1.4 Complex (block sequence) loading

A typical complex loading will comprise of a sequence of loading blocks. Each loading block can define a dataset sequence and/or a set of scale and combine operations between stress datasets and their associated load histories. High frequency loading cycles can be superimposed onto any block.

For a sequence of blocks, the damage resulting from the transitions between blocks can also be included (see 13.9.8). For the transitions between blocks, a critical plane algorithm is used.

Complex (block sequence) loading can be defined directly (in the user interface), or using the load definition (LDF) file.

13.2 Defining loading equivalence

Fatigue lives in *fe-safe*, are based on the number of repeats of the complete fatigue loading cycle up to the point where a crack initiates. A conversion factor is provided for converting the fatigue life in repeats to fatigue life with respect to some other quantity, for example hours or miles.

The conversion dialogue is displayed (*Figure 13.2-1*):

- click the **Loading...** button in the **Fatigue from FEA** dialogue, this will display the loading section.
- double click **Loading is equivalent to...** tree item in the **Settings** section, see *Figure 13.7-2*.

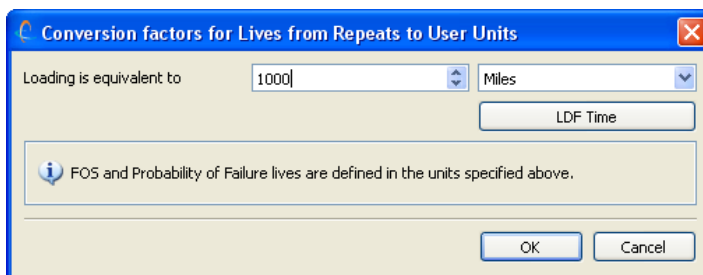


Figure 13.2-1

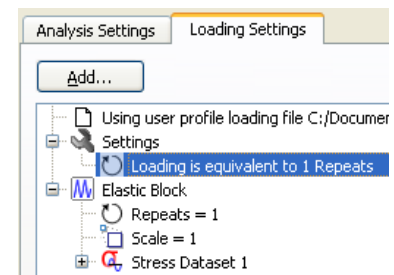


Figure 13.2-2

If the loading units are repeats, then the loading is always equivalent to one repeat of the complete loading.

If any other unit is specified, for example miles, then the fatigue loading can be equivalent to any number in that unit. For example, the above dialogue is defining one repeat of the fatigue loading cycle to be equivalent to 1000 miles.

This setting applies to all lives, including the Factor of Strength (FOS) design life, Probability of Failure target life and Traffic Light Export life range thresholds.

13.3 Importing load histories

Load histories can be imported from many industry-standard file formats.

The following proprietary file formats are supported:

- industry standard binary DAC file (*.dac)
- analogue multi-channel AMC file (*.amc)
- ASCII single and multi-channel data files (*.txt, *.asc, etc.)
- ASCII histogram files (*.txt, *.asc, etc.)

The following third-party data file types are also supported:

- Servotest SBF and SBR files (*.sbf, *.sbr)

- Snap-Master file (*.sm?)
- MTS RPCIII binary data file (*.rsp)
- Adams multi-column ASCII tabular data (*.tab)
- ANSYS Modal Coordinates File (*.mcf)
- ASAM MDF4 binary data file (*.mf4)

Full details of the data structure and syntax of these file types can be found in sections 2 and 3 of Appendix E.

13.4 Pre-processing time histories

By default, *fe-safe* does not assume that peaks and valleys in the damage parameter (the shear strain on the critical plane, the normal stress, etc.) will coincide with peaks and valleys in the loading history. Full loading histories, not peak/valleyed load histories, are used. This is the most rigorous method.

To reduce the analysis time, the time history can be pre-processed using the peak-picking function (see 10.3.17), which extracts the peaks and valleys from a history. A cycle-omission gate level can be set to reduce the number of small cycles in the peak-picked history. For multiple-channel time history data, the multi-channel peak-picking function (see 10.3.18) should be used on all the channels together, as this maintains the phase relationship between channels. The user should be aware that this procedure can lead to inaccuracies in the calculated lives, and should check whether there are significant differences in the fatigue lives by comparing the results from the peak-picked and the full load histories. A significant number of the most damaged elements, plus other less damaged elements, should be used for this comparison.

fe-safe can perform this operation automatically, using the **Pre-gate load histories** option in **FEA Fatigue>>Analysis Options**.

If full histories (i.e. histories that have not been peak-picked) are used, then *fe-safe* will automatically perform a peak-valley analysis on the time history of (for example) the shear strain on the critical plane, for each node. The cycle omission criteria, or 'gate', can be set using **Gate tensors** in **FEA Fatigue>>Analysis Options**. This gate is pre-configured to a range which is 5% of the maximum range. The user should always assess the sensitivity of the fatigue results to this gate setting.

The following table compares the advantages and disadvantages of the different approaches.

Technique	Using original time histories in the fatigue loading definition.	Using histories that have been individually peak-valley picked in the fatigue loading definition.	Using multi-channel histories that have been peak-valley picked.
Process	<p><i>For each node:</i></p> <ul style="list-style-type: none"> - time histories of the principal stresses are calculated from the original histories. - the automatic peak-picking routine (part of the fatigue algorithm) extracts fatigue cycles from the time histories of (for example) the shear strain on the plane, the normal stress on the plane.. 	<p><i>For the whole model:</i></p> <ul style="list-style-type: none"> - each original history is pre-processed individually using the peak-picking function to extract fatigue cycles. 	<p><i>For the whole model:</i></p> <ul style="list-style-type: none"> - the original histories are pre-processed using the multi-channel peak-picking function to extract peaks and valleys, whilst maintaining the phase relationship between the cycles.
		<p><i>For each node:</i></p> <ul style="list-style-type: none"> - time histories of the principal stresses are calculated from the peak-picked histories - the automatic peak-picking routine (part of the fatigue algorithm) extracts fatigue cycles from the time histories of (for example) the shear strain on the plane, the normal stress on the plane. 	
Gating	In the fatigue analysis, small cycles are removed using a cycle-omission gate automatically set.	<p>Small cycles are removed from the pre-processed history using a user-defined cycle-omission gate level. The level of gating affects the length of the pre-processed history, which has an impact on the speed of the fatigue analysis.</p> <p>In the fatigue algorithm, small cycles are removed using a cycle-omission gate automatically set</p>	
Advantages	The preferred method of analysis. No risk of missing peaks or valleys due to the orientation of the principals.	The fastest method.	The phase-relationship between channels is maintained.
Disadvantages	Slower.	<p>For multi-channel histories. The phase relationship between channels may be lost.</p> <p>If the cycle-omission gate level is set too high, damaging cycles may be missed.</p> <p>Assumes peaks and valleys in the damage parameter will coincide with peaks and valleys in the loading history.</p>	Multi-channel peak-picked histories are longer than histories that are peak-picked individually, since additional points are inserted to maintain phase relationship between channels.
Application	Very much the preferred method of analysis, and the default <i>fe-safe</i> setting.	Should only be used for a single channel of loading. The 'gate' should be chosen to ensure all damaging cycles are retained.	This method may be used to obtain 'quick-look' results for multi-channel histories, but fatigue hot-spots may be missed and fatigue lives may be in error.

13.5 Importing FE datasets

Imported FE datasets are listed in the **Current FE Models** window - see section 5.

13.6 Referencing FE datasets

In all cases, the index used to reference stress and strain datasets is the one displayed in the **Current FE Models** window in *fe-safe*. This may not be the same as the step number in the source FE model file. Note also that the numbering of stress datasets in the **Current FE Models** window may change. A number of factors can affect the numbering of datasets including:

- changing the order in which source FE data files are appended;
- changing the FE data type (e.g. importing data at integration points instead of at element nodes);
- re-importing the model after the status of the **Read strains from FE models** option (in the **Analysis Options** dialogue) has been changed.

If the current loading or an LDF file from a previous analysis is being used, always check that the dataset numbering is compatible.

When referencing FE datasets for use in a fatigue analysis from an elastic-plastic FE analysis (see section 13.10 below), care must be exercised when defining dataset numbers to ensure that the defined stress and strain datasets are an elastic-plastic stress-strain pair. For example, a file containing five steps of stress and strain data may be imported. In *fe-safe* the stress data from each step may be listed as datasets 1 to 5, and the strain data from each step may be listed as datasets 6 to 10, so the matching pairs would be 1 and 6, 2 and 7, etc..

13.7 Selecting the loading method

Simple scale-and-combine loading including constant amplitude loading, can be defined directly in the user interface.

Select the **Loading Settings** tab in the **Fatigue from FEA** dialogue. This displays the loading section as in *Figure 13.7-1*.

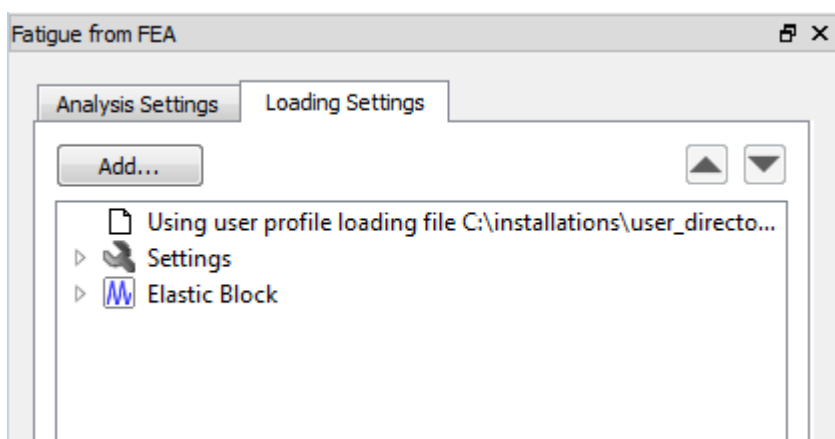


Figure 13.7-1

The current loading configuration is summarised in the **Loading Details...** tree control.

Note: Creation and modification of the loading is performed within the user interface by default, and can be performed using LDF and HLDF files in version 6.0-00 and later.

Note: The LDF file has replaced the dataset sequence (LCD) file format and the block loading (SPC) format.

13.8 Simple load definition using the user-interface

Click on the **Add...** button to display the menu for defining the loading or click the right mouse button on an item in the loading tree to display a context sensitive menu. The top section of the menu contains the properties of the selected item, or possibly its parent.

To edit an item double click on it or select it and press F2.

Press the delete key to delete the selected item. In the case of properties the value will be reset to its default. An example of a property is 'block repeats' and its default is 1.

13.8.1 Add a Block.

Select **Add... >> Block** to add and select an empty block. The block type will be unknown as shown in *Figure 13.8.1-1*. The buttons **Move Up** and **Move Down** reorder the blocks when you have more than one block.

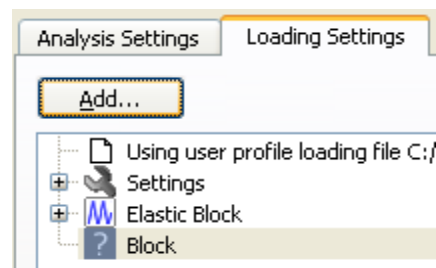


Figure 13.8.1-1

If a block contains no datasets or high frequency blocks there will be an option **Make Block Modal** in the context menu. Selecting this will force all stress datasets that are added to be real or imaginary stresses. If no datasets are added pre-analysis setup will include all modal datasets from the currently loaded model(s).

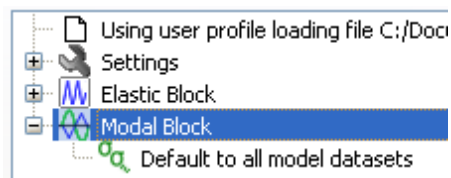


Figure 13.8.1-2

To delete a block select a block or one of its child items and then select the context menu option **Delete Block**. If a high frequency block is selected (or one of its child items) then only the high frequency block will be deleted.

13.8.2 Add a High Frequency Block.

The **Add... >> High Freq. Block** menu item adds and selects an empty high frequency block to the current block. If no block is selected an empty block will be appended to the current loading as shown in *Figure 13.8.2-1*. If there is more than one high frequency block in the same main block the buttons **Move Up** and **Move Down** are used to reorder the high frequency blocks.

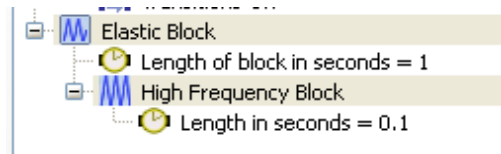


Figure 13.8.2-1

When a high frequency block or one of its child items is selected the context menu option **Delete Block** can be used to delete the high frequency block only.

13.8.3 Add a Load History to a Dataset.

Select **Add... >> Load History to Dataset** to add the selected load histories in the **Loaded Data Files** windows to the selected dataset. If no dataset is selected or the dataset is inappropriate (i.e. part of a stress – strain pair, or modal) the dataset selected in the **Current FE Models** window will be added to the current block. If no block is selected a block will be appended with the dataset and loading(s) as shown in *Figure 13.8.3-1*.

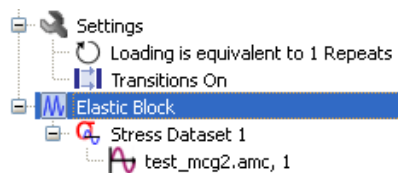


Figure 13.8.3-1

If the selected dataset already has a loading it will be replaced, unless the loading is user defined, in which case the loading will have to be deleted first. If multiple load histories are selected, the selected dataset list will be duplicated for each loading.

When a load history is selected the context menu option **Delete History** will delete the load history from the loading definition.

13.8.4 Add a User Loading to a Dataset.

Select **Add... >> User Loading to Dataset** to add a user defined loading to the selected dataset. If no dataset is selected or the dataset is inappropriate (i.e. part of a stress – strain pair, or modal) the dataset selected in the **Current FE Models** window will be added to the current block. If no block is selected a block will be appended with the dataset and loading. The history editing **Dataset Embedded Load History** dialogue will be shown as in *Figure 13.8.4-1*.

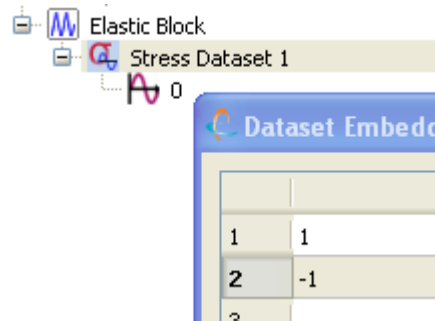


Figure 13.8.4-1

If the selected dataset already has a loading a prompt to replace the loading will be displayed.

When a user loading is selected the context menu option **Delete History** will delete the loading from the loading definition.

13.8.5 Add a Time History to a Block.

Select **Add... >> Time History to Block** to add the selected history to the current block as a time definition. A time definition allows the time in seconds to be applied to each sample in the loading block. If no block is selected a block will be appended with the time history as shown in *Figure 13.8.5-1*.

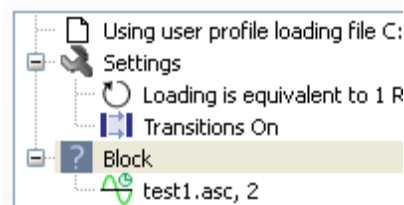


Figure 13.8.5-1

When a time history is selected, the context menu option **Delete History** will delete the time history from the loading definition.

13.8.6 Add a User Time History to a Block

Select **Add... >> User Time to Block** to add a user-defined sequence to the current block as a time definition. A time definition allows the time in seconds to be applied to each sample in the loading block. If no block is selected a block will be appended with the time history. The **Dataset Embedded Time History** dialogue is displayed as shown in *Figure 13.8.6-1*. The dataset time intervals must be ascending.

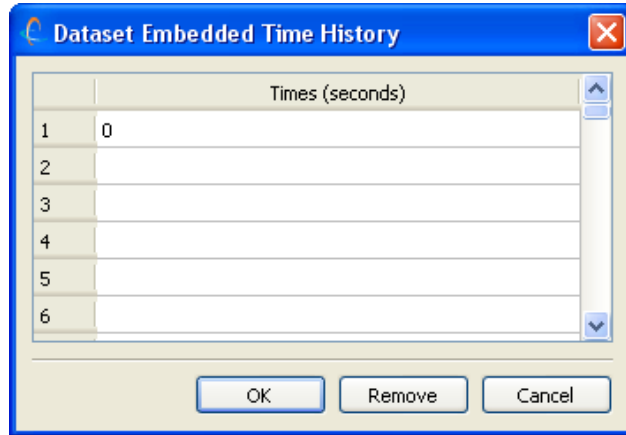


Figure 13.8.6-1

When a user time history is selected the context menu option **Delete History** will delete the user time history from the loading definition.

13.8.7 Add a Dataset

Select **Add... >> Dataset** to add the dataset selected in the **Current FE Models** window to the current block. If no block is selected a block will be appended with the dataset as shown in *Figure 13.8.7-1*. Use the buttons **Move Up** and **Move Down** to reorder the datasets when you have more than one dataset.

If the added dataset is modal and is not being appended to a dataset list the corresponding real or imaginary stress dataset of the same frequency is automatically added to make a real and imaginary pair.

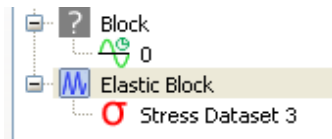


Figure 13.8.7-1

If a dataset list (the target) was selected in the **Fatigue from FEA** dialogue then the source dataset will be added in different ways:

- If the source and target are the of the same type, the source dataset is added to the target list e.g. adding stress dataset 6 to stress datasets '1-4, 7-9' will become '1-4, 6-9'.

- If the source and target dataset types can be paired (i.e. stress with strain, real with imaginary stress) then the source dataset is added to the pair dataset list of the correct type, or one is created if there is not one present. *Figure 13.8.7-2* shows the loading in *Figure 13.8.7.1* after a stress dataset is added.

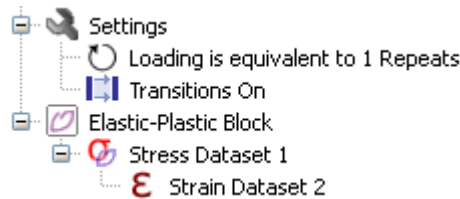


Figure 13.8.7-2

As noted above at start of section 13.8, dataset lists can be edited via double clicking the item or pressing F2. When editing dataset sequences in this manner, a continuous list of datasets can be specified with a hyphen e.g. datasets 1 through 10 would be '1-10'. A list of datasets incrementing or decrementing by a fixed amount can be specified. This is done by adding the increment within parenthesis after the end dataset number e.g. datasets 1, 4, 7 and 10 would be '1-10(3)'.

Note: Even if the increment of the sequence would not include the last dataset specified, it is always included e.g. Datasets from 20 decreasing by 3 have the sequence 20, 17, 14, 11, 8, 5 and 2, but 20-1(3) would produce the sequence 20, 17, 14, 11, 8, 5, 2 and 1.

When a dataset is selected the context menu option **Delete Dataset** will delete the dataset list from the loading definition. A message box will ask if any associated datasets should also be deleted i.e. strains.

13.8.8 Add a Load * Dataset

Select **Add... >> A Load * Dataset** to add the stress dataset selected in the **Current FE Models** window and the loading history selected in the **Loaded Data Files** window to the current block. If no block is selected a block will be appended with the dataset and loading. If multiple load histories are selected, the selected dataset list will be duplicated for each loading.

13.8.9 Add a User Defined Load * Dataset

Select **Add... >> A User Defined Load * Dataset** to add the stress dataset selected in the **Current FE Models** window to the current block. If no block is selected a block will be appended with the dataset and loading. The history editing **Dataset Embedded Load History** dialogue will be shown as in *Figure 13.8.9-1*.

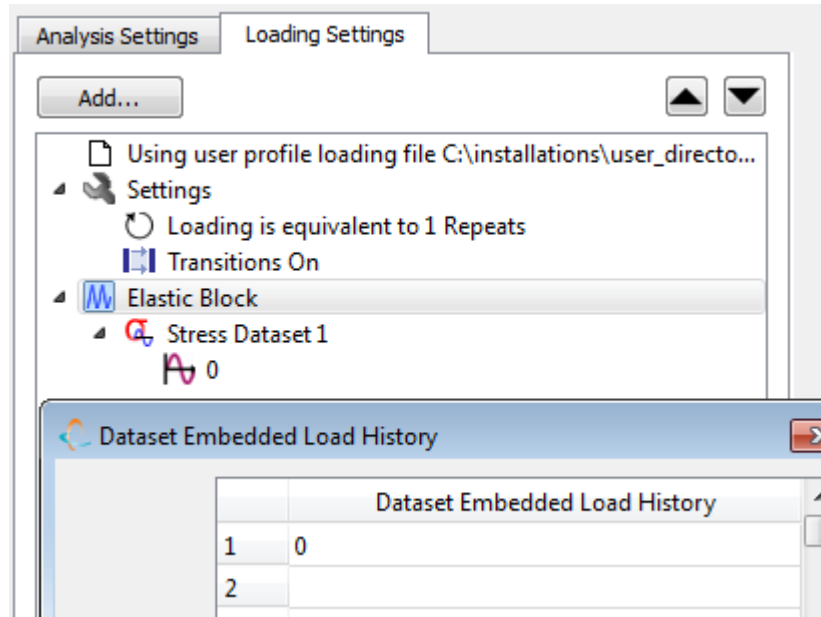


Figure 13.8.9-1

If the selected dataset already has a loading, embedded histories are edited while normal histories cause a message box to be displayed asking if the loading should be replaced.

13.8.10 Add a Temperature Dataset

Select **Add... >> Temperature Dataset** to add the temperature dataset selected in the **Current FE Models** window to the current block. If no block is selected a block will be appended with the dataset. If no temperature dataset is selected, temperature dataset 1 is used.

When a temperature dataset is selected, the context menu option **Delete Dataset** will delete the temperature dataset list from the loading definition.

13.8.11 Add a Residual Dataset

Select **Add >> Residual Dataset** to add a residual stress or strain dataset to a block. It behaves similar to adding a dataset (see section 13.8.7) with some exceptions. Only modal blocks and the loadings as a whole can have residuals. There must be one stress and one strain dataset as shown in *Figure 13.8.11-1*.

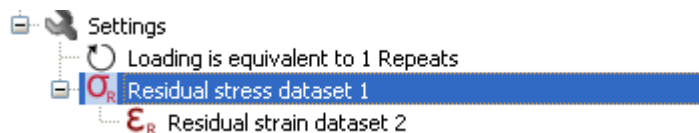


Figure 13.8.11-1

When a residual dataset is selected the context menu option **Delete Dataset** will delete both residual datasets from the loading definition.

13.8.12 Embedding histories

Normally load and time histories are stored in separate disk based files. Short histories can be embedded using the loading context menu option **Embed History**. The original source file will no longer be required for the analysis as the data is stored with the loading. Unlike user defined histories, all embedded histories from the same source history share the same data which makes storing and reading the loadings quicker compared with many separate user defined histories. These embedded histories can be identified by the file name being appended to the history data as shown in *Figure 13.8.12-1*.

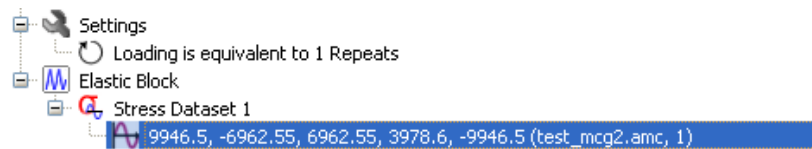


Figure 13.8.12-1

Successful editing of an embedded history causes the history to behave like a normal user defined history and no longer shares the data.

13.8.13 Refresh the loading

Select either the **Refresh** option in the loading context menu or an item in the loading tree and press F5 to refresh the view. If the source file has changed the tree will be updated and any loadings will be validated e.g. global residuals will be removed if there are elastic and elastic-plastic blocks. There will also be a prompt to validate the datasets used in the loadings with the currently opened datasets in the **Current FE Models** window.

13.8.14 Profile loading definition

Normally the loadings are stored with the current user's profile, and thus will be different for each user. The loadings can however be saved for later analysis by selecting **File >> Loadings >> Save Current FEA Loadings As...** or the **Save As...** option in the loading context menu. For more information on the loading definition file (.LDF) and the loading options and constraints see section 13.9. When not using the current profile a child in the settings item will appear indicating the loading source file as shown in *Figure 13.8.14-1*.

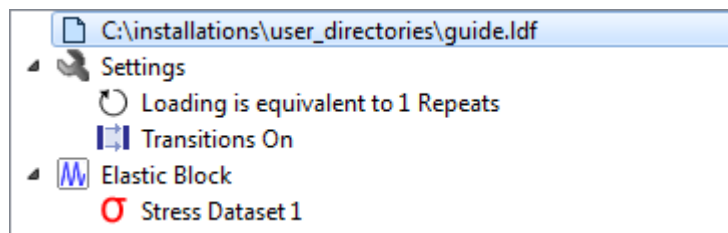


Figure 13.8.14-1

To open a .LDF file select **File >> Loadings >> Open FEA Loadings File...** or alternatively select the **Open Loadings...** from the loading context menu. Then select a file from the **Open a Loading Definition File (*.ldf)** dialogue and click **Open**.

To save the loading to the current profile select the **Save to Profile** option from the loading context menu or **File >> Loadings >> Save FEA Loadings to Current Profile**.

13.8.15 Clearing the loading

Select **Clear All Loadings** from the loading context menu to remove all loadings.

13.9 The Load Definition (LDF) file

IMPORTANT NOTE - Supporting legacy file definition formats.

From version 5.00, onwards

The LDF file has replaced the block loading (SPC) format (see Appendix 205.7.4) and the data set sequence (LCD) file format (see Appendix 205.7.5). From version 5.00, onwards, support for the LCD and SPC file formats is disabled by default. New users should always use the LDF file.

From version 5.2, onwards

fe-safe v5.2-00 saw the introduction of an enhanced GUI-based method for defining loading. Underlying the GUI method is the existing LDF file format, and an LDF file called "`current.ldf`" is maintained in the user directory – see 13.8.

Existing users can continue to edit LDF files using a text editor. However, it is anticipated that all new users and most existing users will use the GUI-based method for defining the loading.

From version 6, onwards

fe-safe v6.0-00 saw the introduction of an enhanced high-level loading definition (HLDF) method. Underlying the HLDF method is a means to support generating loadings based on reference to the original FE model instead of to the dataset number in *fe-safe*.

The load definition (LDF) file is a versatile file structure that can be used to define simple and complex loading situations. In its simplest form, the LDF file can define a constant amplitude loading block. Complex loadings can be defined as a series of loading blocks.

Each loading block can define a dataset sequence and a set of scale and combine operations between stress datasets and their associated load histories.

For a sequence of blocks, fatigue cycles resulting from the transitions between blocks can also be included (see 13.9.8).

The LDF format also supports:

- superimposition of high frequency loading cycles onto any block;
- analysis of datasets from an elastic-plastic FE analyses;

- temperature variation - for use in conventional high temperature fatigue analysis (see section 18);
-

In all cases, the index used to reference stress and strain datasets is the one displayed in *fe-safe*, (see 13.5, above).

The loading is defined using a combination of the following definitions:

- the BLOCK definition;
- the dataset sequence definition;
- the load history (scale-and-combine) definition;
- the high frequency loading definition;
- the temperature variation definition;
- the time definition.

When an LDF file is selected, a summary is displayed in a pop-up dialogue box.

13.9.1 General syntax

All LDF files must adhere to the following general syntax rules:

- An LDF file must contain at least one block.
- Each block must begin with the start of block definition statement: `BLOCK`
- Each block must end with the end of block definition statement: `END`
- Block definition statements must start at the first character in a line.
- Each block can contain any number of dataset sequence and/or scale-and-combine definitions.
- Comment lines must include a # (hash) character in column 1. If you edit an .LDF file using the usual interface only the whole file and block comments are retained.
 - The whole file comments are the first set of consecutive comment lines prior to the first `BLOCK` statement.
 - The block comments are the last set of consecutive comment lines prior to each `BLOCK` statement.

e.g.

```
# This is a file comment

# This comment will be lost when saved
INIT
END

# This is the comment for the 1st block
BLOCK n=1, scale=1
ds=1-2
END

# This comment will be lost when saved

# This is the comment for the 2nd block
# This is the 2nd line of the comment for the 2nd block
BLOCK n=2, scale=1
ds=3-4
END

# This comment will be lost when saved
```

- Anything outside a block is ignored.
- The last block in an LDF file must always be followed by a line termination character – see the general note on ASCII formats, and their portability between platforms, in Appendix E, section 205.1.2.
- All definition parameters must have an argument, e.g. `scale=1.0`
- Some definition parameters are optional. If an optional parameter (and its argument) is omitted then a default value is used. If a required parameter is omitted, a syntax error (`default=NA`) is displayed.
- Some definition parameter names can be omitted, in which case the position of the value in the line is used to associate it with the corresponding parameter.

13.9.2 The BLOCK definition

The BLOCK definition begins with the start of block statement, `BLOCK`, and ends with the end of block statement, `END`.

There is a designed limit of 10240 blocks per LDF file.

The BLOCK definition has the format:

```
BLOCK n=10, scale=1.0, dt = 3600, temp=100
```

Loading definition for the block, which may contain:

- any number of datasets as sequences, within the limitation of a maximum of 256000 unique datasets across all blocks;
- up to 4096 scale-and-combines per block;
- up to 20 high frequency block definitions;
- a time definition (which overrides the block parameter, `dt`);
- a temperature variation definition.

```
END
```

where the parameters have the following properties:

Parameter	Parameter position	Default value	Optional parameter?	Property
n	1	1*	Yes	Number of repeats of block. The default for modal blocks is calculated, for other blocks it defaults to 1. The value of n must be at least 1 (as it is not possible to run the block less than once) and must be an integer value for TURBOlife and plugin algorithms.
scale	2	1.0	Yes	Scale factor for the block. This is multiplied by the scale factor for any individual items in the block.
dt	3	0.0	Yes	Block length, (the time, in seconds, that the block is equivalent to). This time is equivalent to the n repeats of the block NOT 1 repeat of the block, as is shown in the Loading Settings window.
temp	4	-300	Yes	Temperature of the block, (in °C). If a value of less than -273 is specified, then the temperature data will be extracted from the FE model.

13.9.3 Dataset sequence definition

The dataset sequence definition has the format:

```
ds=1, es=2, scale=1.0, escale=2.0
```

where the parameters have the following properties:

Parameter	Parameter position	Default value	Optional parameter?	Property
ds	1	NA	No	Stress dataset number reference for the open FE model. This parameter must be defined and it must be the first item on the line.
es	2	NA	Yes	Strain dataset reference number for the open FE model. This parameter must be defined for elastic-plastic FEA results – see 13.9.9. It is not required for elastic FEA results, as <i>fe-safe</i> always uses FEA stresses.
scale	3	1.0	Yes	A scale factor for the stress dataset.
escale	4	1.0	Yes	Scale factor for the strain datasets. After scaling, strain values should be in units of strain (m/m), not microstrain ($\mu\epsilon$).

13.9.4 Load history (scale and combine) definition

The load history definition has the format:

```
lh=<load_history_filename>, signum=1, ds=7, scale=45.0
```

where the parameters have the following properties:

Parameter	Parameter position	Default value	Optional parameter?	Property
lh	1	NA	No	Load history filename. This parameter must be defined and must be the first item on the line.
signum	2	NA	No	Data file history number, where 1 is the first history. This parameter must be defined.
ds	3	NA	No	Stress dataset number reference for the open FE model. This parameter must be defined.
Scale	4	1.0	Yes	Datasets scale factor.

This definition is used to combine a stress dataset with the time history of a loading to create a LOAD*DATA set. Multiple LOAD*DATA sets can be defined and may be added to the loading definition in any order, since they are combined by superimposing (adding) the time histories of the stress tensors at each point in time, to produce a history of the stresses for the combined loading.

Load histories can be imported from any supported file format – see section 13.3.

13.9.5 Time definition

A time must be defined if a high frequency block is superimposed onto a lower frequency sequence (see section 13.9.6).

The time can be defined by the dt parameter or by a time for each item in the sequence using the lhtime parameter. All time position values must be in seconds.

If the defined time position series is shorter than the loading, a warning will be written to the diagnostics log and the last defined time position will be used for all subsequent sequence items.

The lhtime definition overrides the block parameter dt, in this case the time for 1 repeat of the block will be the last value in the lhtime sequence.

If items in the sequence are to be equally spaced in time then the dt block parameter will suffice, and the lhtime parameter is unnecessary. In this case the time associated with the first sample is 0 seconds and the time associated with the last sample is dt/n seconds.

A zero time increment between the last sample and the zero block time on repeating the block is assumed. So to define a time difference between the last sample and the first sample on repeating the block the first sample must have a non zero time associated with it.

The `lhtime` time positions can be defined directly as the argument to the `lhtime` parameter (see example 1, below), or alternatively, they can be extracted from an ASCII text file, that contains a series of time positions (see example 2, below). Both of the following examples yield the same total loading for the block:

Example 1	Example 2
<pre># Each load history has ten samples BLOCK n=100, scale=1.0 lh=/data/test.txt, signum=1, ds=4 lh=/data/test.txt, signum=1, ds=4 lh=/data/test.txt, signum=1, ds=4 lh=/data/test.txt, signum=1, ds=4 lhtime=0 5 7 9 10 11 25 27 30 31 END</pre>	<pre># Each load history has ten samples BLOCK n=100, scale=1.0 lh=/data/test.txt, signum=1, ds=4 lh=/data/test.txt, signum=1, ds=4 lh=/data/test.txt, signum=1, ds=4 lh=/data/test.txt, signum=1, ds=4 lhtime=/data/test.txt, signum=1 END</pre> <p>test.txt, column 1 defines the times for each sample in the time history loading, in seconds, for example:</p> <pre>0 5 7 9 10 11 25 27 30 31</pre>

Figure 13.9.5-1 indicates the difference caused by using the `dt` parameter to define the time for a block and using the `lhtime` parameter. The block in both cases is 20 seconds long. For the `lhtime` parameter the 5 datasets are spaced at 4 second intervals.

Using dt	Using lhtime
<pre>BLOCK n=1, dt=20 ds=1-5 END</pre>	<pre>BLOCK n=1 ds=1-5 lhtime=4, 8, 12, 16, 20 END</pre>

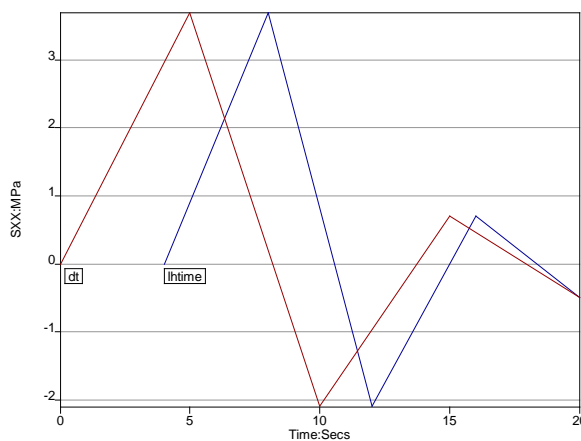


Figure 13.9-5.1

Generally the effect caused by this difference would have no effect on your fatigue analysis. It does become important if a HFBLOCK loading is used.

13.9.6 High frequency loading definition (superimposition of a high frequency load blocks)

A block containing high frequency cycles can be superimposed on the defined loading in any block. Up to 20 high frequency load blocks can be superimposed on each main block. Each high frequency block can be built up from dataset sequences and load history scale-and-combine loads. The high frequency cycle is repeated from the start of the block to the end of the block.

The definition statements HFBLOCK and HFEND are used to indicate the start and end of the high frequency block definition.

The length of each high frequency block (HFBLOCK) is defined using the dt parameter. If a high frequency block is used, the main block must also have its length defined using either the dt or lhtime parameters. The repeat frequency of the high frequency block is a function of the main block time and the high frequency block time. The amplitude of the loading is interpolated so that at each point in the main block and the high frequency block, a data sample is evaluated.

The high frequency block can contain a dataset definition (see example 1, below) or a scale-and-combine definition (see example 2, below). In both examples, the low frequency block lasts 100 seconds and the high frequency block lasts 1 second so there are 100 repeats of the high frequency block.

Example 1 – using a dataset sequence definition in the high frequency block

```
# Each load history has ten samples
BLOCK n=1, scale=1.0, dt=100
lh=/data/test.txt, signum=1, ds=4
lh=/data/test.txt, signum=1, ds=4
lh=/data/test.txt, signum=1, ds=4
lh=/data/test.txt, signum=1, ds=4
lhtime=/data/test.txt, signum=2
HFBLOCK dt=1.0
lhtime=0 0.1 0.3 0.8
ds=1-2
ds=7-8
HFEND
END
```

Example 2 - using a scale-and-combine definition in the high frequency block

```
# Each load history has ten samples
BLOCK n=1, scale=1.0, dt=100
lh=/data/test.txt, signum=1, ds=4
lh=/data/test.txt, signum=1, ds=4
lh=/data/test.txt, signum=1, ds=4
lh=/data/test.txt, signum=1, ds=4
lhtime=/data/test.txt, signum=2
HFBLOCK dt=1.0
lhtime=0 0.1 0.3 0.8
lh=/test/lh.txt, signum=1, ds=1
HFEND
END
```

The `lhtime` parameter does not override the `dt` parameter for high frequency blocks, this allows a time between the last and first samples in the high frequency block to be defined. If `lhtime` is not defined the samples within the high frequency block are located at the times:

$$0, dt/nSamples, \dots dt*(nSamples-1)/nSamples.$$

Where `nSamples` is the length of the data set sequence or load histories within the hf block.

It should be noted that *fe-safe* expands the high frequency blocks into a full loading definition for each node prior to analysis. This is done within the computer's memory. Hence the limitation that a very long block and a very short high frequency block will require very large amounts of memory, in some cases far more than is available. For example a block of 6 months and a high frequency block of 1000 rpm would require in excess of 20 Gbytes of memory. This limitation should be considered when using the high frequency block facility.

Care should be taken in defining the time for the main block to achieve the required effect. If no data amplitude is defined at $t=0$ in the main block (as for the `lhtime` example in figure 13.9.5-1) then the last amplitude in the block is wrapped around to take the place of the missing start amplitude. This allows the high frequency amplitudes to be superimposed upon an amplitude history over the complete loading time of the main block. Figure 13.9.6-1 shows the same loading as figure 13.9.5-1 with and without a high frequency block superimposed.

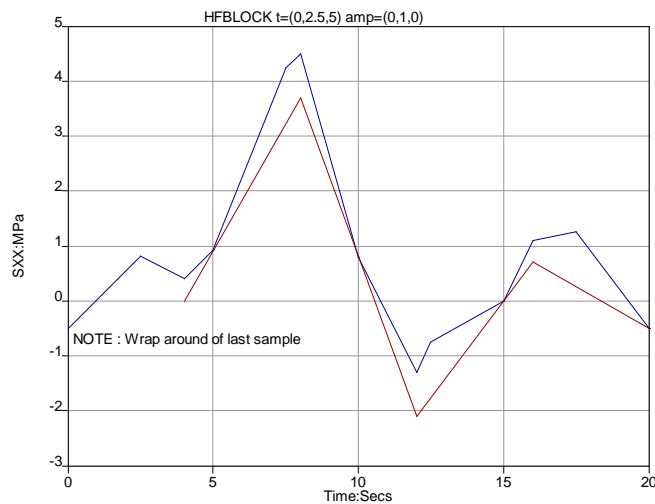


Figure 13.9.6-1.

Defining fatigue loadings

Multi block complex loading can be built up using this technique. If a section of the analysis contains long flat plateaus with a high frequency content then these should be reduced to as short a time as possible with a repeat factor. The two examples below will give identical fatigue lives but the left hand example would generate a tensor history of 3511 samples and the right hand one would only generate 3 samples. 3 samples will analyse much quicker than 3511.

```
# time = 1 repeat of 35100 secs
BLOCK n=1, dt=35100
ds=1, 0.784
ds=1, 0.784
HFBLOCK dt=20.0
ds=2-4
HFEND
END
```

```
# time = 1755 repeats of 20 secs
BLOCK n=1755
ds=1, 0.784
ds=1, 0.784
lhtime = 0, 20
HFBLOCK dt=20.0
ds=2-4
HFEND
END
```

An example of a multi-block loading simulating a number of flight missions is shown below. The left-hand side shows the mission simulated correctly. The right-hand side shows what would happen due to wrap-around if the samples were not defined at $t=0$ and $t=dt$:

```

INIT
Transitions=YES
END

#####
BLOCK, n=1, dt=144
ds=1, scale=0.0
ds=1, scale=3.70
ds=1, scale=0.0
END

#####
BLOCK, n=1, dt=144
ds=1, scale=0.0
ds=1, scale=1.765
lhtime=0,144
HFBLOCK, dt=20
ds=1, lh=p:\data\hf.txt, signum=1
HFEND
END

#####
BLOCK, n=1
ds=1, scale=1.765
ds=1, scale=1.765
lhtime=0, 144
HFBLOCK, dt=20
ds=1, lh=p:\data\hf.txt, signum=1
HFEND
END

#####
BLOCK, n=1
ds=1, scale=1.765
ds=1, scale=0.0
lhtime=0,90
HFBLOCK, dt=20
ds=1, lh=p:\data\hf.txt, signum=1
HFEND
END

#####
BLOCK, n=1
ds=1, scale=0.0
ds=1, scale=0.784
lhtime=0,90
HFBLOCK, dt=20
ds=1, lh=p:\data\hf.txt, signum=1
HFEND
END

#####
BLOCK, n=1755
Ds=1, scale=0.784
ds=1, scale=0.784
lhtime=0,20
HFBLOCK, dt=20
Ds=1, lh=p:\data\hf.txt, signum=1
HFEND
END

#####
BLOCK, n=1
Ds=1, scale=0.784
ds=1, scale=0.517
ds=1, scale=2.086
ds=1, scale=0.0
lhtime=0,283,285.5,288
HFBLOCK, dt=20
Ds=1, lh=p:\data\hf.txt, signum=1
HFEND
END

```

```

INIT
Transitions=YES
END

#####
BLOCK, n=1, dt=144
ds=1, scale=0.0
ds=1, scale=3.70
ds=1, scale=0.0
END

#####
BLOCK, n=1, dt=144
ds=1, scale=0.0
ds=1, scale=1.765
lhtime=1,144
HFBLOCK, dt=20
ds=1, lh=p:\data\hf.txt, signum=1
HFEND
END

#####
BLOCK, n=1
ds=1, scale=1.765
ds=1, scale=1.765
lhtime=1, 144
HFBLOCK, dt=20
ds=1, lh=p:\data\hf.txt, signum=1
HFEND
END

#####
BLOCK, n=1
ds=1, scale=1.765
ds=1, scale=0.0
lhtime=1,90
HFBLOCK, dt=20
ds=1, lh=p:\data\hf.txt, signum=1
HFEND
END

#####
BLOCK, n=1
ds=1, scale=0.0
ds=1, scale=0.784
lhtime=1,90
HFBLOCK, dt=20
ds=1, lh=p:\data\hf.txt, signum=1
HFEND
END

#####
BLOCK, n=1755
ds=1, scale=0.784
ds=1, scale=0.784
lhtime=1,20
HFBLOCK, dt=20
ds=1, lh=p:\data\hf.txt, signum=1
HFEND
END

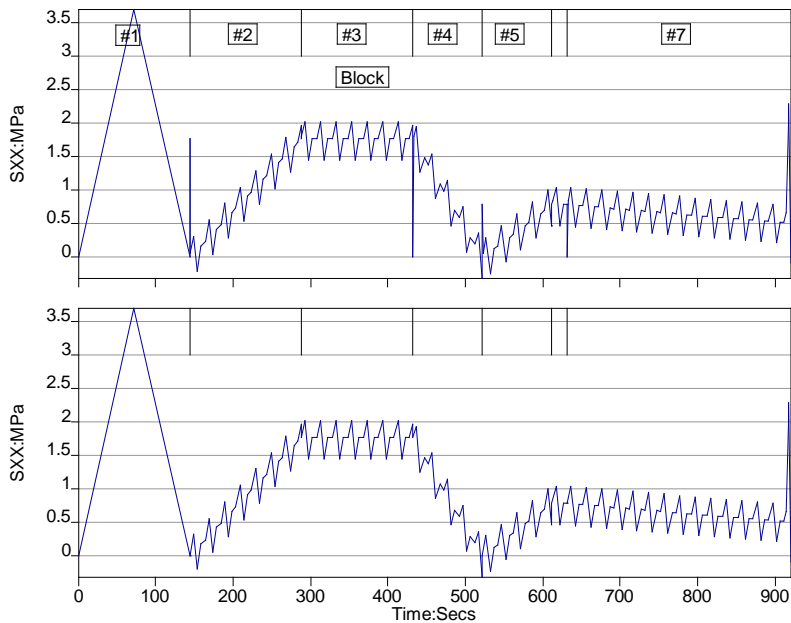
#####
BLOCK, n=1
ds=1, scale=0.784
ds=1, scale=0.517
ds=1, scale=2.086
ds=1, scale=0.0
lhtime=1,283,285.5,288
HFBLOCK, dt=20
ds=1, lh=p:\data\hf.txt, signum=1
HFEND
END

```

The text file p:\data\hf.txt is shown below :

```
0
0.259
-0.316
0
```

The resulting plots of the Sxx transitions for a unit Sxx stress tensors are shown in the figure 13.9.7-1. In the upper plot the spikes at the block edges are caused by the wrap-around technique used when a main block sample is not defined at t=0.



Upper Plot = Right Hand .ldf

Figure 13.9.7-1.

13.9.7 Temperature variation definition

Within a loading block a temperature variation can be defined. This is a list of temperature datasets extracted from the FE model - one for each time point in the loading definition. The temperature dataset is defined in a similar manner to the stress datasets, but uses the keyword `dtemp` instead of `ds`. A range of temperature datasets can be defined on a single line or each dataset can be on a separate line:

In the following example, the temperature dataset sequence is built up from datasets 6 to 11 followed by datasets 17 to 20.

```
# Each load history has ten samples
BLOCK n=100, scale=1.0
lh=/data/test.txt, signum=1, ds=1
lh=/data/test.txt, signum=1, ds=2
lh=/data/test.txt, signum=1, ds=3
lh=/data/test.txt, signum=1, ds=4
dtemp=6-11
dtemp=17-20
END
```

If the defined temperature history is shorter than the loading a warning will be written to the diagnostics log and the last defined temperature will be used for all subsequent temperatures.

For high frequency blocks, the temperature across the block is not defined. Instead, it is calculated for each repeat of the block from the temperature definition of the main blocks.

Note:

The definition of fatigue loading for varying temperature, as discussed in this section, is not required for conventional high temperature fatigue.

13.9.8 Including block transitions in the loading definition

Instructs *fe-safe* to include the fatigue damage that may be caused by cycles that have a peak in one block and a valley in another block. Block transitions are enabled by default where all loading blocks are elastic or all loading blocks are elastic-plastic. Block transitions are disabled by default where loading blocks are a combination of elastic and elastic-plastic blocks, in which case a warning is shown that transitions have been switched off.

The option to include block transitions is available within the settings block of the LDF file, as in the following example:

```
INIT
Transitions=Yes
END
```

The settings block is normally placed at the beginning of the LDF file.

13.9.9 Defining elastic-plastic residual stresses

This is an initial loading condition. For each node, the stress tensor is resolved onto the current plane, and the resultant stress is used as an addition to the mean-stress of the cycle. For an elastic-plastic FE analysis an elastic-plastic stress-strain pair can be defined. The same residual stress tensor is used for all blocks (including the transitions block, if enabled – see 13.9.8).

Residual stresses datasets (or stress-strain dataset pairs from elastic-plastic FE analyses) are defined within the initialisation block of the LDF file, as in the following example:

```
INIT
ds=1, es=2
END
```

Note that:

- no elastic-plastic correction is applied to the residual tensors;
- the residual stress is not relaxed for thermo-mechanical analyses;
- the residual stress is not scaled during a Factor of Strength (FOS) analysis;
- since the residuals are applied as an addition to the mean stress of the cycle, residuals will not be 'washed-out' by large cycles.

A diagnostics option is available (**Export elastic-plastic residuals**), which allows the resolved residual stresses to be exported – see section 22.

13.10 Defining loads for analyses from elastic-plastic FE datasets

By default, *fe-safe* assumes that stress datasets contain elastic stresses and so, where necessary, performs an elastic to elastic-plastic correction (using the biaxial “Neuber Rule”).

When stress data is from an elastic-plastic FE analysis, stress and strain datasets must be read from the FEA results file as elastic-plastic stress-strain pairs. The method for reading strain datasets is described in section 5.5.7, and section 15.

For *fe-safe* to analyse an elastic-plastic stress-strain pair, either the loading interface or the load definition (LDF) file must be used.

To create an elastic plastic loading in the loading interface:

- add a stress dataset;
- select the new dataset item and add a strain dataset, as described in section 13.8.7;
- edit the stress dataset list and change it to the required datasets, which in this example is datasets 1-4;
- do the same with the strain dataset list, this time with datasets 5-8;
- to repeat the block 10 times, edit the **Repeats** property of the block item by selecting the block and accessing the context menu option **Repeats**;
- change to value to 10 and press enter.

After which the loadings should match *Figure 13.10-1*.

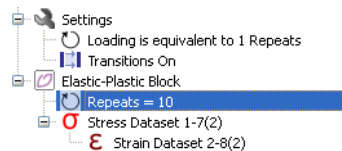


Figure 13.10-1

In a LDF the use of the `es` keyword in a dataset sequence definition (see 13.9.3) turns off the elastic to elastic-plastic correction function (i.e. the biaxial “Neuber Rule”) and treats the defined stress and strain datasets as a stress-strain pair. For the above example:

Example 1 – a stress-strain sequence defined using keywords

```
# Sample LDF file
# Block with elastic-plastic
# dataset sequence
# using keywords
BLOCK n=10, scale=1.0
ds=1,es=5
ds=2,es=6
ds=3,es=7
ds=4,es=8
END
```

Example 2 - a stress-strain sequence defined using position

```
# Sample LDF file
# Block with elastic-plastic
# dataset sequence
# using keywords
BLOCK n=10, scale=1.0
1,5
2,6
3,7
4,8
END
```

(Desktop spreadsheet software can make entering long sequences much easier).

Scale factors must not be applied to elastic-plastic FEA results, unless they are used to convert non-standard stress units to Pa, and strain units to m/m. See section 13.9.3 for the stress and strain scale factors.

Normal Strain, Brown Miller and Maximum Shear Strain analysis methods may be used with elastic-plastic FEA results.

A range of datasets for both stresses and strains can be used to simplify the definition of the .ldf file.

Example 3 – a stress-strain sequence defined
using keywords

```
# Sample LDF file
# Block with elastic-plastic
# dataset sequence
# using keywords
BLOCK n=10, scale=1.0
ds=1-4,es=5-8
END
```

13.11 Defining loads for analyses from steady state dynamic FE datasets (modal)

Chapter 25 outlines the use of *fe-safe* with steady state dynamics FEA results. When a model containing frequency response data is read 2 *fe-safe* datasets are created for each mode stored in the file.

The loading is defined in terms of these datasets.

To define a block as modal use the **Make Block Modal** option or add a real/imaginary stress to a new block using the visual loading interface (see 13.8.1). In an LDF the parameter `modal=steady` is used. A block time must be defined for this type of block. An optional `n` parameter can be defined in the LDF (or the repeats property in the interface).

This follows the rules for standard blocks, that it must have a value of at least 1 and, for TURBOlife and plugin algorithms, be a positive integer, and the time defined by `n` is the time for the entire block, inclusive of repeats, but the value displayed in the **Loading Settings** GUI will be the length per repeat, e.g. `dt/n`

If this is omitted then *fe-safe* will evaluate a suitable number of repeats based upon the frequency content of the modes.

In the loading interface each real and imaginary dataset pair can have optional frequency and scale properties. In the loading definition file each mode to be used is specified using a single line:

```
rds=1, ids=2, freq=26.7, scale=1e6
```

where the parameters have the following properties:

Parameter	Default value	Optional parameter?	Property
rds	NA	No	The dataset containing the real coefficients of the stresses for this mode.
ids	NA	No	The dataset containing the imaginary coefficients of the stresses for this mode.
freq	Extracted from file	Yes	The frequency in Hz to associate with the mode.
Scale	1.0	Yes	Datasets scale factor.

In the case where all of the steady state data read in from the FEA results are to be used then leave the block empty e.g. for an LDF file just the BLOCK definition and END can be defined.

i.e.

```
# 100 seconds of data
# let fe-safe work out how many repeats and which datasets
BLOCK modal=steady, dt=100
# Leave it empty so fe-safe works out what to use from loaded models
END
```

If a selection of the modes is to be used then they must be specified. The `freq` parameter would generally not be defined as this will be extracted from the FEA results file.

e.g.

```
# Just use first 7 modes and extract frequency for each from loaded FE model
# Force fe-safe to generate 10 seconds of data and repeat it 10 time to make
# up the 100 seconds of data
BLOCK modal=steady, dt=100, n=10
rds=1, ids=2
rds=3, ids=4
rds=5, ids=6
rds=7, ids=8
rds=9, ids=10
rds=11, ids=12
rds=13, ids=14
END
```

One set of residuals can be used as an offset for the steady state dynamic data. This is defined in the LDF using the `datums` and `datume` parameters, for the loading interface see section 13.8.11. They refer to the datasets containing the elastic-plastic stress and strains.

e.g.

```
# Use first two datasets as residuals
# 100 seconds of data
# let fe-safe work out how many repeats and which datasets are mmodal
BLOCK modal=steady, dt=100
datume=2, datums=1
END
```

If global residuals are defined they are overridden by any defined within a modal block.

Steady state blocks can be mixed with other elastic and elastic-plastic blocks within a loading definition.

13.12 Mixing elastic and elastic-plastic blocks within an analysis

Elastic and elastic-plastic blocks can be added to the same analysis using the block syntax defined in the previous sections.

The only limitations on analyses which contain both blocks is that the transitions option is not supported and that gauge output is not supported.

13.13 Example loading definitions using the LDF file

13.13.1 Using the BLOCK definition, the dataset sequence definition and the load history (scale-and-combine) definition types

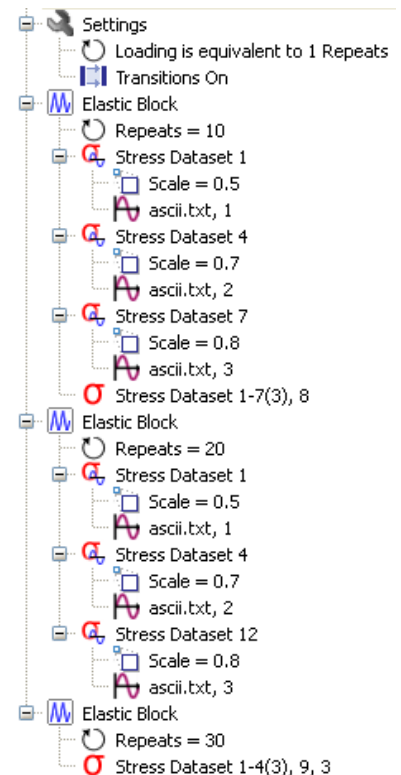
Equivalent loading.

```
# Sample LDF file
# Block with dataset sequence and load history combines
# using definition parameters

BLOCK n=10, scale=1.0
lh=d:\prod\ascii.txt, signum=1, ds=1, scale=0.5
lh=d:\prod\ascii.txt, signum=2, ds=4, scale=0.7
lh=d:\prod\ascii.txt, signum=3, ds=7, scale=0.8
ds=1
ds=4
ds=7
ds=8
END

# Block with load history combines using positional
# parameters
BLOCK n=20, scale=1.0
d:\prod\ascii.txt, 1, 1, 0.5
d:\prod\ascii.txt, 2, 4, 0.7
d:\prod\ascii.txt, 3, 12, 0.8
END

# Block with dataset sequence using positional
# parameters
BLOCK n=30, scale=1.0
1
4
9
3
END
# End Sample LDF file
```



13.132 Three superimposed load histories

Consider three unit load FEA solutions, with the three sets of stresses contained in datasets (steps) 3, 4, and 5.

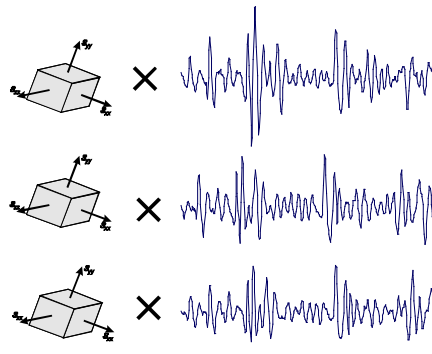
Load histories are contained in a multi-column ASCII file. It is required to apply

Channel 1 to dataset 3

Channel 3 to dataset 4

Channel 5 to dataset 5

All the histories are to be applied without additional scaling, i.e. with scale factors equal to 1.0



The LDF file would be:

```
BLOCK n=1, scale=1.0
lh=\myfiles\testloading.txt, signum=1, ds=3, scale=1.0
lh=\myfiles\testloading.txt, signum=3, ds=4, scale=1.0
lh=\myfiles\testloading.txt, signum=5, ds=5, scale=1.0
END
```

Lines beginning with # are comment lines.

If the keywords are omitted the parameters must be in the correct positions, and the file would be

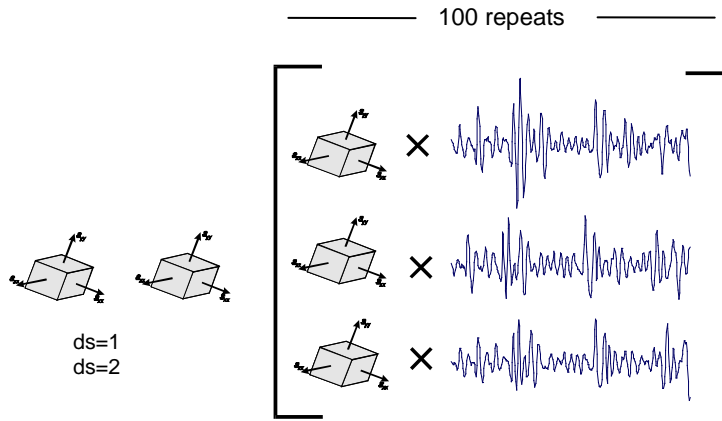
```
BLOCK n=1, scale=1.0
\myfiles\testloading.txt, 1, 3, 1.0
\myfiles\testloading.txt, 3, 4, 1.0
\myfiles\testloading.txt, 5, 5, 1.0
END
```

If the three histories are in three separate files (say .dac files), the .ldf file will be

```
BLOCK n=1, scale=1.0
lh=\myfiles\filex.dac, signum=1, ds=3, scale=1.0
lh=\myfiles\filey.dac, signum=1, ds=4, scale=1.0
lh=\myfiles\filez.dac, signum=1, ds=5, scale=1.0
END
```

13.13.3 Three superimposed load histories with a repeat count specified, and two initial stress datasets.

In this example, the same load histories as in example 2 are applied. Now, two additional datasets 1 and 2 are to be inserted at the beginning of the load history, and the section in brackets [] is repeated 100 times.



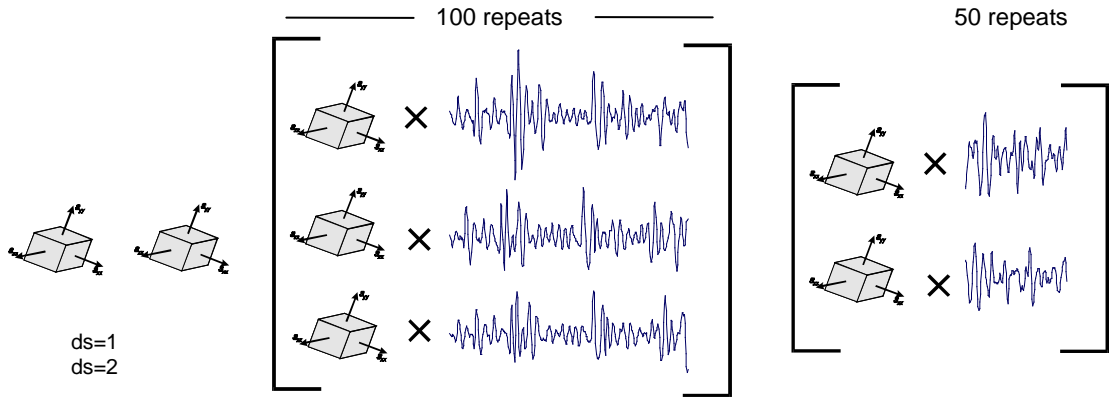
The LDF file would be:

```
BLOCK n=1, scale=1.0
ds=1
ds=2
END
BLOCK n=100, scale=1.0
lh=\myfiles\testloading.txt, signum=1, ds=3, scale=1.0
lh=\myfiles\testloading.txt, signum=3, ds=4, scale=1.0
lh=\myfiles\testloading.txt, signum=5, ds=5, scale=1.0
END
```

The fatigue life is calculated in repeats of this complete sequence then optionally converted into user-defined units (miles, hours, etc)

13.13.4 The above example with additional loading appended.

In this example, the same load histories as above are applied. Now, two more unit load solutions are multiplied by their appropriate load histories with different scale factors applied, to be repeated 50 times.



```

BLOCK n=1, scale=1.0
ds=1
ds=2
END
BLOCK n=100, scale=1.0
lh=\myfiles\testloading.txt, signum=1, ds=3, scale=1.0
lh=\myfiles\testloading.txt, signum=3, ds=4, scale=1.0
lh=\myfiles\testloading.txt, signum=5, ds=5, scale=1.0
END
BLOCK n=50, scale=2.0
lh=\myfiles\moreloading.txt, signum=4, ds=11, scale=1.5
lh=\myfiles\moreloading.txt, signum=2, ds=12, scale=1.0
ds=6
END
    
```

Note:

If a superimposed history is shorter than the other histories in the block, it will be padded to the length of the longest history, using zeros.

If no history is specified, the dataset is applied only once. In the following example, if each history in moreloading.txt contains 250 data points, dataset 6 will be superimposed on the first data point only.

```

BLOCK n=50, scale=2.0
lh=\myfiles\moreloading.txt, signum=4, ds=11, scale=1.5
lh=\myfiles\moreloading.txt, signum=2, ds=12, scale=1.0
ds=6
END
    
```

13.13.5 Long sequence of datasets

Long sequences of datasets can now be specified easily. In the following example, the first 65 datasets are applied in sequence.

```
BLOCK n=1, scale=1.0
ds=1-65
END
```

This facility is an alternative to listing each dataset, i.e.

```
BLOCK n=1, scale=1.0
ds=1
ds=2
ds=3
...
...
ds=64
ds=65
END
```

The datasets do not need to start at 1. `ds=5-65` would be acceptable, as would

```
BLOCK n=1, scale=1.0
ds=4-20, scale=1.5
ds=21-30, scale=-1.2
ds=15-25, scale=1.1
END
```

13.13.6 Adding temperature information for thermo-mechanical analysis

In this example the datasets containing the temperatures are also specified. The number of seconds over which this dataset sequence occurs is also specified, using the `dt` parameter.

```
BLOCK n=1, scale=1.0, dt=360000
ds=1-65
dtemp=1-65
END
```

Note:

The `dtemp` datasets need not have the same numbers as the corresponding stress datasets.

For all input files except UNV files, only the maximum temperature will be extracted.

For simple high temperature analysis the temperature datasets do not need to be specified. The analysis options are used to select or de-select temperature effects.

13.13.7 Specifying elastic-plastic FEA results

These must be specified as pairs of stresses and strain datasets. `es` denotes a strain dataset.

```
BLOCK n=1, scale=1.0
ds=1, es=6
ds=2, es=7
END
```

Scale factors must not be used to re-scale elastic-plastic results. However, scale factors can be used to change units. Stresses must be in Pascals, strain in units of m/m (not micro-strain).

Defining fatigue loadings

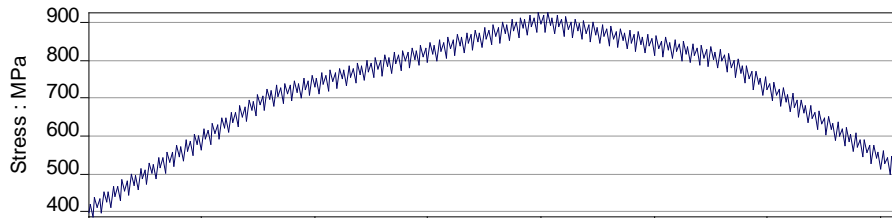
The scale factor for stresses is defined by `scale=`, and the scale factor for strain is defined by `escale=`

For example, if the stresses are in MPa, and the strains in micro-strain :

```
BLOCK n=1, scale=1.0
ds=1, es=6, scale=1e6, escale=1e-6
ds=2, es=7, scale=1e6, escale=1e-6
END
```

13.13.8 Superimposing a high frequency loading block

This may be required when a high frequency loading is superimposed on thermal stresses.



In the following example, the high frequency block is defined by

```
HFBLOCK dt=0.5
ds=5-6, scale=0.1
ds=7, scale=-0.1
lhtime=0.0 0.2 0.3
HFEND
```

This block takes 0.5 seconds (`dt=0.5`)

Three stress datasets are applied in sequence (`ds=5-6` and `ds=7`)

The times at which these datasets occur are given in seconds,

```
lhtime=0.0 0.2 0.3
```

and in this example the times are unequally spaced.

These time values could be contained in a history:

```
lhtime=\myfiles\datafile.txt, signum=3
```

If the time values are equally spaced, only the length of time for the block need be specified.

```
HFBLOCK dt=0.5
ds=5-6, scale=0.1
ds=7, scale=-0.1
HFEND
```

The specification of the outer block follows the syntax described in examples 1- 5. The parameter `lhtime=` is used to specify the time values for each dataset.

```
BLOCK n=10, scale=1.0
ds=1
ds=2, scale=-1.0
ds=3
ds=4, scale=-1.0
lhtime=0 3 5 10
HFBLOCK dt=0.5
ds=5-6, scale=0.1
ds=7, scale=-0.1
lhtime=0.0 0.2 0.3
HFEND
END
```

fe-safe repeats the high frequency block the required number of times. In the above example, the high frequency datasets would be applied at times of

0.0 0.2 0.3 0.5 0.7 0.8 1.0 1.2 1.3 and so on.

To superimpose these datasets on the low frequency block, the values in the low frequency block are interpolated to give a value at each time in the high frequency block.

Note that this form of superimposition can produce very long analysis times. Users should experiment with small groups of elements.

13.13.9 Example LDF file for thermomechanical fatigue analysis including a high frequency block

Consider a node in an FE Model with its stresses and temperatures calculated at 5 increments in time (0, 20, 50, 70, and 90 seconds) as shown below:

Time	Sxx	Syy	Szz	Sxy	Sxz	Syz	Temp
0	40	0	0	0	0	0	20
20	70	0	0	0	0	0	100
50	90	0	0	0	0	0	250
70	80	0	0	0	0	0	250
90	50	0	0	0	0	0	50

And assume that a unit load analysis provided a sixth load case with the stress tensor

1 0 0 0 0 0 0

To define a loading for the five time increments, and also superimpose the unit load dataset (sixth dataset) scaled by a load of (0, 2, -2, 3), where the load history is repeated each second: then the LDF file would be:

```
# Main block lasting 100 seconds (NOTE dt is overridden by lhtime)
BLOCK dt=100
lhtime=0, 20, 50, 70, 90
ds=1-5
dtemp=1-5
# High frequency block repeating once per second
```

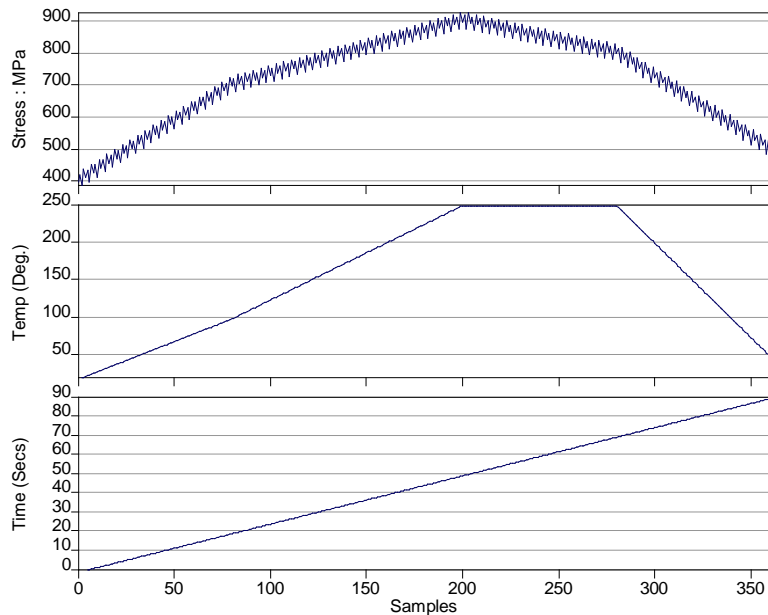
Defining fatigue loadings

```
HFBLOCK dt=1
lh=lhfl.txt, signum=1, ds=6
HFEND
END
```

where the file `lhfl.txt` would contain the following lines representing the loading applied to the sixth dataset:

```
0
2
-2
3
```

The stress (Sxx), temperature and time for the loading would be:



13.14 The High-Level Loading Definition (HLDF) file

High Level loading in *fe-safe* provides an optional alternative to using LDF files (see section 13.9 above) to define fatigue loadings. This allows the user to base the references in the loading definitions on the actual FEA results files instead of referring to the numbered datasets in the *fe-safe* **Current FE Models** window. For instance:

- Refer to FEA Model filenames directly
- Pick steps and increments from the filenames by number
- Pick steps by using `all` or `last` description
- Define many blocks in a loading with separate repeats, etc by using a loop definition
- Repeat similar loading definitions quickly, if the number of increments in a solution changes

Future developments will expand this functionality to read required increments from the FEA results files at the time of analysis. At *fe-safe* version 6.0-00, the datasets are still opened in *fe-safe* previous to running an FEA fatigue analysis. Features supported at this release are as follows:

- Dataset Sequence loading is supported (see section 13.1.3 above)
- Use of Elastic and/or Elastic-plastic loading are supported (see section 13.10 above)

- Temperature-dependent load definitions are supported (see section 13.9.7 above)
- Time-dependent load definitions are supported (see section 13.9.5 above)
- Scale-and-combine loading is not supported at this release (see section 13.1.2 above)

13.14.1 The High-Level Loading Definition (HLDF) process

The format of an HLDF file is defined in section 3 below. The process of applying loadings from an HLDF file in *fe-safe* is as follows:

- The HLDF file is created in a third-party spreadsheet software or a text editor
- The FEA model is Opened in *fe-safe*
- The HLDF file is Opened in *fe-safe*
- A conventional LDF file is generated automatically based on the HLDF file and the loaded FEA model
- Configuration of fatigue analysis is completed, and the analysis started
- Validation of loading to ensure the referenced solutions are available is completed at analysis time.

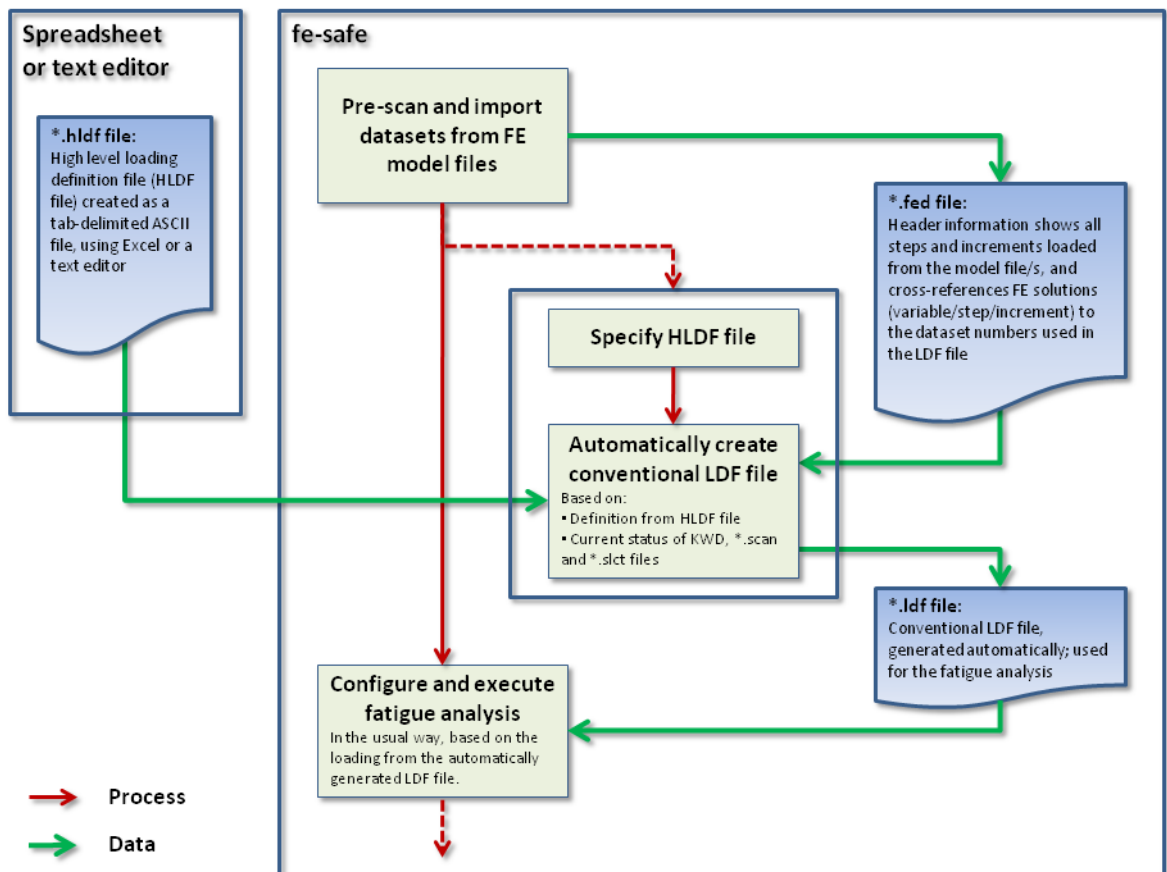


Figure 13.14-1 Process flow using HLDF file

13.14.2 Using an HLDF file in *fe-safe*

Three methods can be used to Open the HLDF file in *fe-safe*: File Menu, Loading Settings tab, and Macro files.

1. An HLDF file can be opened from the File menu, using:

File >> Loadings >> Open FEA Loadings File... and adjusting the file filter accordingly

As shown in *Figure 13.14-2* below.

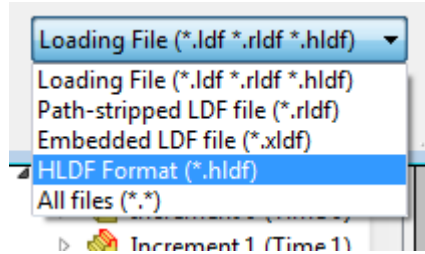


Figure 13.14-2

2. The Open Loadings dialogue can be accessed from the context-sensitive menu in the **Loading Settings** tab as shown in *Figure 13.14-3* below:

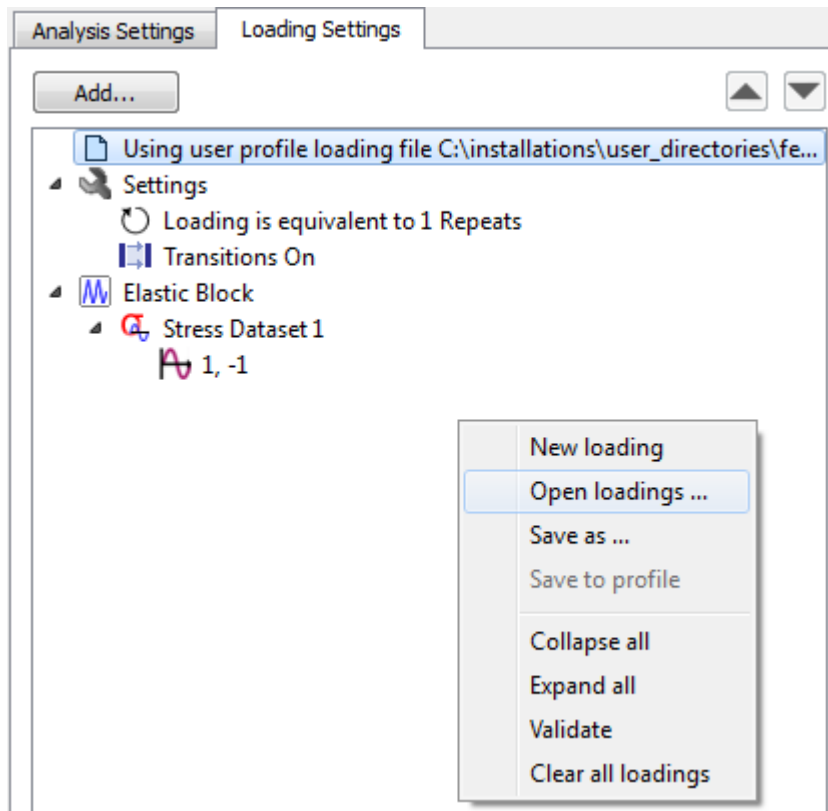


Figure 13.14-3

3. An HLDF file can be accessed from a macro

The macro command to import a HLDF file is:

```
HLDF <hldf_file_path>
```

Optionally, the user may specify the name of the LDF file generated from the HLDF file:

```
HLDF <hldf_file_path>, <ldf_file_path>
```

If the LDF path name is not specified, it defaults to *_hldf.ldf, where * is the root of the original HLDF file.

Example 1: the LDF file generated from the HLDF command:

```
HLDF myload.hldf
```

will be:

```
myload_hldf.ldf
```

Example 2: the LDF file generated from the HLDF command:

```
HLDF myload.hldf, newload.ldf
```

will be:

```
newload.ldf
```

the HLDF token can be specified as the only command in a macro, or combined with other tokens to process signals, pre-scan, manage groups, or run FEA fatigue analyses as needed. Macros can be run from within the *fe-safe* GUI or on the command line. See section 23 for details.

13.14.3 Formatting and Syntax of an HLDF file

This section describes the syntax and format of the HLDF file. However, the format is quite self explanatory, and a good starting point to understand the format may be to analyse some of the examples later in this section.

The HLDF file is a tab-delimited ASCII file. The file can be created either in a text editor, or a spreadsheet. Since the file must be tab-delimited, it will often be easier to use a spreadsheet software to generate the content of the file, and then save this to a tab-delimited ASCII file.

Notes on using A spreadsheet software to generate an HLDF file:

1. Create the tabulated content of the HLDF in a spreadsheet software.
2. Save the file as a tab-delimited .txt (ASCII) file.
3. Change the extension of the file to *.hldf.

Note: when A spreadsheet software exports to a tab-delimited file, it puts any cell containing a comma in double-quotes. This looks a bit odd if you then view the text file in an editor, because comments without commas are not in quotes, whereas comments with commas are in quotes. This doesn't pose any problem for the HLDF reader in *fe-safe*, since this will ignore the double-quotes.

Lines in the HLDF file can be:

- metadata
- a header line
- comments
and
- data

Metadata lines:

- Metadata lines are used to describe the fields in subsequent lines that usually begin with the same token.
- Metadata lines begin with the “*” character (an asterisk, without the quotes) followed by an identifier.
- The metadata line defines the order of the fields on the subsequent data line. The fields may appear in any order, except for the defining field which must always come first.
- Fields documented below as optional may be omitted from the metadata line if they are not required but all other fields must appear.
- Any unrecognised metadata field will be ignored.
- The file header is a special case of metadata.

The file-header:

The header line is used to indicate that the file is an HLDF file and provide the version number of the minimum supported HLDF file format/syntax, and it is required. See usage below.

Note: The first line of an HLDF file should always be either a comment line or a file-header line

Comment lines:

- A hash character (#) is used to precede a comment.
- A line beginning with a # character indicates that the remainder of the line (until CR or LF/CR) is a comment.
- A # character mid-line indicates that the text before the # is not a comment, but the remainder of the line (until CR, LF or CR/LF) is a comment.

Note: The first line of an HLDF file should always be either a comment line or a file-header line

Data lines:

- Data lines must contain the same number of fields as the associated metadata.
- The first token must take a value determined by the metadata type (which is usually the same as the metadata identifier).
- Subsequent fields are interpreted according to the field names listed in the preceding metadata line.

Types of Metadata Block and Data line combinations:

Below is a description of the syntax for each type of data line. Examples of ASCII data will contain <t> to illustrate a tab character

The file header:

The metadata for the file header line always has the following syntax:

*HEADER_TYPE<t>version

The syntax of the header line is:

HLDF<t>{HLDF_file_version_number}

Spreadsheet Example:

*HEADER	version
HLDF	1.0

ASCII Example:

*HEADER_TYPE<t>version

HLDF<t>1.0

The version described in this document is 1.0.

Note: Only *HEADER and HLDF are case sensitive.

Redirection:

Redirection is used to represent a path or a part of a path using a user-defined string variable.

The metadata for the redirection lines always has the following syntax:

*REDIRECT<t>token<t>path

The syntax of the redirection line is:

REDIRECT<t>token<t>path

Spreadsheet Example:

*REDIRECT	token	path
REDIRECT	my_data	"c:\model_data\my_data\"
REDIRECT	<my_data1>	<my_data>/part1

ASCII Example:

*REDIRECT<t>token<t>path

REDIRECT<t>my_data<t>"c:\model_data\my_data\"

REDIRECT<t><my_data1><t><my_data>/part1

- Optionally tokens may be enclosed in angle brackets (< and >), which will be removed. These must be used when referring to the token in a DATADEF item (see spreadsheet example above).
- Paths are case sensitive in Linux but not Windows Operating Systems.
- Paths may be described by:
 - Combinations of forward- and back-slashes in the same path.
 - Paths with spaces – these need not be surrounded by quotes.
 - Duplicate slashes are tolerated. This is useful when defining redirections in terms of other redirections. For instance:
 - C:*
 - C:/*
 - *
 - *
 - /*
 - /**
 - Environment variables and Macro redirections
 - Redirection already defined in a macro is inherited, without needing to be redefined in the HLDF file. If a token that is already defined in a macro is redefined in the HLDF file, the definition in the HLDF file takes precedence.
 - Previously-defined HLDF redirections may be used as shown above.

Note: Only *REDIRECT and REDIRECT are case sensitive

Loading definition

The optional loading definition data line is used to specify block transitions (see section 13.9.8 above) or residual dataset pairs (see section 13.9.9 above).

The metadata for the loading definition line always starts with *LOADINGDEF. It may contain an optional field for enabling transitions, and/or a pair of optional fields for defining a residual dataset pair.:

*LOADINGDEF<t>transitions<t>residual_xref<t>residual_data_type

The syntax for the loading definition line starts with LOADINGDEF (no asterisk) and is also tab delimited to match those fields specified in the metadata as follows:

transitions	If undefined, the transitions <i>are</i> considered (equivalent to 'Transitions=yes' in the LDF file).	
	on	Transitions are considered.
	yes	
	1	
	true	
	off	Transitions are not considered
	no	
	0	
false		

The value of this field determines whether transitions are considered between the loading blocks.

residual_xref	These two fields must be used together.	This is a reference to a single DATADEF item to define a residual elastic-plastic stress/strain pair.
residual_data_type		A string value defining the type of data in the block, which must be ep meaning Elastic-plastic. No other values are currently supported.

Note: DATADEF items are covered in detail below.

Spreadsheet Example:

*LOADINGDEF	transitions	residual_xref	residual_data_type		
LOADINGDEF	No	my_residuals	ep		
*DATADEF	data_xref	model_path	model_filename	steps	increments
DATADEF	my_residuals	"d:/temp"	keyhole69.odt	1	1

ASCII Example:

```
*LOADINGDEF<t>transitions<t>residual_xref<t>residual_data_type  
LOADINGDEF<t>no<t>resid_13<t>ep
```

```
*DATADEF<t>data_xref<t>model_path<t>model_filename<t>steps<t>increments  
DATADEF<t>resid_13<t>"d:/temp"<t>keyhole69.odt<t>1<t>1
```

Note: DATADEF and LOADINGDEF are case sensitive

Block definition

For top-level definition of the block type, number of repeats etc. The order of the defined blocks is important, since this determines the order in which the loading is applied.

The metadata line has the following syntax (where <t> is a tab character):

```
*BLOCKDEF<t>block_type<t>block_label<t>block_data_type<t>  
block_repeats<t>block_length_lrep<t>block_length_nrep<t>  
block_temperature<t>block_scale<t>block_data_xref
```

The syntax for the block definition lines start with BLOCKDEF and is described as follows:

Metadata field name	Optional field?	Case-sensitive?	Data Description	
*BLOCKDEF	No	No	The definition type. Must take the value BLOCKDEF.	
block_type	No	No	A string value defining the type of loading block, which can be only one value at version 6.0-00: <i>sequence</i> for a Dataset Sequence block.	
block_label	No	Yes	A user-defined string for referring to a block. Each block must have a unique value. The label appears as a comment preceding the corresponding block in the LDF output.	
block_data_type	No	No	A string value defining the type of data in the block, which can be one of the following:	
			e	Elastic
			ep	Elastic-plastic
			te	Elastic with temperature datasets defined
			tep	Elastic-plastic with temperature datasets defined
block_repeats	Yes	N/A	Number of repeats, $n (\geq 1)$. If not defined, this defaults to 1. For modal blocks this is a calculated value (see user guide, 13.9.2).	
block_length_1rep	Yes	N/A	Block length, in seconds. This is the time for 1 repeat	

Defining fatigue loadings

			of the block.
block_length_nrep	Yes	N/A	Block length, in seconds. This is the time for n repeats of the block (where the value of n is that defined by block_repeats).

Metadata field name	Optional field?	Case-sensitive?	Data Description
block_temperature	Yes	N/A	Temperature of the block, in °C (floating point or integer). Note: If a temperature value of less than {-273°C} is specified, then the temperature data will be extracted from the FE Model (see user guide, 13.9.2).
block_scale	Yes	N/A	Cumulative scale factor for stresses in the block.
data_xref	No	Yes	A user-defined string used as a cross-reference between the block definition in BLOCKDEF and the data that the block will contain. In other words, all DATADEF lines beginning with the value of data_xref will be included in the block.

Spreadsheet Example:

*BLOCKDEF	block_label	block_type	block_data_type	block_repeats	data_xref	
BLOCKDEF	x-then-y	Sequence	ep	7000	x-y	
*DATADEF	data_xref	model_path	model_filename	steps	increments	scale_stress
DATADEF	x-y	<my_path>	keyhole69.odt	1	1	0
DATADEF	x-y	<my_path>	keyhole69.odt	1	1	1

ASCII Example:

```
*BLOCKDEF<t>block_label<t>block_type<t>block_data_type<t>block_repeats
<t>data_xref
BLOCKDEF<t>x-then-y<t>sequence<t>ep<t>7000<t>x-y
*DATADEF<t>data_xref<t>model_path<t>model_filename<t>steps<t>increments
<t>scale_stress<t>
DATADEF<t>x-y<t><my_path><t>keyhole69.odt<t>1<t>1<t>0
DATADEF<t>x-y<t><my_path><t>keyhole69.odt<t>1<t>1<t>1
```

Note: *BLOCKDEF, BLOCKDEF, block_label and data_xref are case sensitive.

Data definition

The data definition lines assign of the loading which is referenced by the block definition lines. Each data line with the same data_xref value in order represents one point in the loading sequence.

A loading block is created for each block defined using **BLOCKDEF**. Each loading block is populated according to the data definitions in all data lines with the same value of **data_xref** as the value of **data_xref** defined in the **BLOCKDEF** definition.

Multiple data lines can have the same **data_xref** reference, which can be cross-referenced from more than one block.

For example: if a block definition with **block_label=engine_1000rpm** includes **data_xref=1000revs**, all instances of data lines with **data_xref=1000revs** will contribute to the loading block labelled **engine_1000rpm** in the LDF file. An additional block can also reference **data_xref=1000revs**.

The nature of the contribution of a data line to the loading block will depend on the context – for example in a dataset-sequence (**sequence**) block, FE solutions will be treated sequentially.

The metadata line has the following syntax (where <t> is a tab character):

```
*DATADEF<t>data_xref<t>model_path<t>model_filename<t>steps<t>increments<t>
scale_stress<t>scale_strain
```

The syntax for the data definition lines start with **DATADEF** and is described as follows:

Metadata field name	Optional field?	Case-sensitive?	Data Description
DATADEF	No	Yes	Indicates that this line contains a definition of data. Must take the value DATADEF .
data_xref	No	Yes	A user-defined string for blocks to refer to the DATADEF . This value must be referred to by one or more BLOCKDEFs , otherwise the data will not form part of the fatigue loading. No warning is given if the HLDF file contains unused DATADEFs .
model_path	No	Sensitivity to case is platform-dependent	The FE model's directory. The path may make use of tokens defined previously as redirections using REDIRECT items.
model_filename	No	Sensitivity to case is	The file-name of the FE model, without its path.

		platform-dependent	Can include wildcards and regular expressions. When multiple files are matched, their data is applied in order of their file-names (rather than ,for example, the order in which they were loaded by <i>fe-safe</i>).
Steps	Yes; if not specified, defaults to all_loaded .	N/A	The step or steps to include at this point in the loading sequence. The syntax options described below are supported.
increments	Yes; if not specified, defaults to all_loaded .	N/A	The increment or increments to include at this point in the loading sequence. The syntax options described below are supported.
scale_stress	Yes	N/A	Scale factor for stresses in this line in the data definition.
scale_strain	Yes	N/A	Scale factor for strains in this line in the data definition.

- The content of a **DATADef** line is only checked if it is referred to in a **BLOCKDef** definition. So, for example, if a **DATADef** specifies a file that does not exist, but the **DATADef** is not referred to in a **BLOCKDef**, then no error is reported.
- Where a wildcard is used to refer to a file, an error message occurs if no file matches it.
- For step and increment definitions, the following syntax options are supported. Note that spaces are permitted next to the delimiters (commas):
- A step [or increment] number, e.g.:
1
- A series of step [or increment] numbers, separated by commas, e.g.:
1, 2, 5, 6, 12, 14
- An increasing or decreasing contiguous sequence of step [or increment] numbers defined using a hyphen, e.g.:
1-12
25-18

- An increasing or decreasing sequence defined using a hyphen as above, but incorporating in brackets an increment, e.g.:

24-32(2)

31-37(2)

Note that the last increment in the sequence is always included, regardless of the increment size. For example:

1-11(3)

will yield the sequence:

1, 4, 7, 10, 11

- Reversed sequences, e.g.:

1-12

25-18

32-24(2)

37-31(2)

- Specify that all steps in the *loaded* model should be used, using:

all_loaded

Note that **all_loaded**, if specified, is exclusive.

- Specify that only the last step in the model should be used, using:

last

- With the exception of “all” and “all_loaded”, which are exclusive, the above syntax can be used to define complex sequences of steps or increments, by dividing each part of the definition using a comma. For example:

1-12, 24-32(2), last

Spreadsheet Example

*DATADEF	data_xref	model_path	model_filename	steps	increments	
DATADEF	x_a	<my_path>	keyhole69.odb	1	0	
DATADEF	x_a	<my_path>	keyhole69.odb	1	1	
*DATADEF	data_xref	model_path	model_filename	steps	increments	scale_stress
DATADEF	x-yb	<my_path>	keyhole69.odb	1	1	0
DATADEF	x-yb	<my_path>	keyhole69.odb	1	1	1
*DATADEF	data_xref	model_path	model_filename	steps	increments	
DATADEF	all-c	<my_path>	keyhole69.odb	all_loaded	all_loaded	

ASCII Example

```
*DATADEF<t>data_xref<t>model_path<t>model_filename<t>steps<t>increments
DATADEF<t>x_a<t><my_path><t>keyhole69.odb<t>1<t>0
DATADEF<t>x_a<t><my_path><t>keyhole69.odb<t>1<t>1

*DATADEF<t>data_xref<t>model_path<t>model_filename<t>steps<t>increments
<t>scale_stress
DATADEF<t>x-yb<t><my_path><t>keyhole69.odb<t>1<t>1<t>0
DATADEF<t>x-yb<t><my_path><t>keyhole69.odb<t>1<t>1<t>1

*DATADEF<t>data_xref<t>model_path<t>model_filename<t>steps<t>increments
DATADEF<t>all-c<t><my_path><t>keyhole69.odb<t>all_loaded<t>all_loaded
```

Note: only *DATADEF, DATADEF, data_xref are case sensitive. Model_path and model_filename are case sensitive in Linux platforms but not in Windows.

Notes on formatting and syntax:

- The first line of an HLDF file should always be either a comment line or a file-header line
- Blank lines are ignored, so can be used to clarify layout.
- Quotation marks (" ") are ignored.
- Line termination characters CR or CR+LF are supported.
- The concept of a continuation character is not supported for any line type

Example:

Defining fatigue loadings

To apply the HLDF file below the user must reference the path to the <DataDir>/Abaqus/keyhole69.odb file used below, (where <t> is a tab character)

```
*HEADER_TYPE&ltt>version
HLDF&ltt>1.0

*LOADINGDEF&ltt>transitions
LOADINGDEF&ltt>no

*REDIRECT&ltt>token&ltt>path
REDIRECT&ltt>my_path&ltt>"d://temp"

*BLOCKDEF&ltt>block_label&ltt>block_type&ltt>block_data_type&ltt>block_repeats
&ltt>block_scale&ltt>data_xref
BLOCKDEF&ltt>Block-A_xload&ltt>sequence&ltt>tep&ltt>5000&ltt>1&ltt>x_a

*BLOCKDEF&ltt>block_label&ltt>block_type&ltt>block_data_type&ltt>block_repeats
&ltt>data_xref
BLOCKDEF&ltt>Block-B_yload&ltt>sequence&ltt>ep&ltt>7000&ltt>x-yb

*BLOCKDEF&ltt>block_label&ltt>block_type&ltt>block_data_type&ltt>block_repeats
&ltt>block_scale&ltt>data_xref
BLOCKDEF&ltt>Block-C_all&ltt>sequence&ltt>e&ltt>52&ltt>3&ltt>all-c
*BLOCKDEF&ltt>block_label&ltt>block_type&ltt>block_data_type&ltt>block_repeats
&ltt>block_scale&ltt>data_xref
BLOCKDEF&ltt>Block-D_FL&ltt>sequence&ltt>e&ltt>100&ltt>5&ltt>firstlast-c

*DATADEF&ltt>data_xref&ltt>model_path&ltt>model_filename&ltt>steps&ltt>increment
s
DATADEF&ltt>x_a&ltt><my_path>&ltt>keyhole69.odb&ltt>1&ltt>0
DATADEF&ltt>x_a&ltt><my_path>&ltt>keyhole69.odb&ltt>1&ltt>1

*DATADEF&ltt>data_xref&ltt>model_path&ltt>model_filename&ltt>steps&ltt>increment
s&ltt>scale_stress&ltt>scale_strain
DATADEF&ltt>x-yb&ltt><my_path>&ltt>keyhole69.odb&ltt>2&ltt>1&ltt>0&ltt>0
DATADEF&ltt>x-yb&ltt><my_path>&ltt>keyhole69.odb&ltt>2&ltt>1&ltt>1&ltt>1

*DATADEF&ltt>data_xref&ltt>model_path&ltt>model_filename&ltt>steps&ltt>increment
s
DATADEF&ltt>all-c&ltt><my_path>&ltt>keyhole69.odb&ltt>all_loaded&ltt>all_loaded

*DATADEF&ltt>data_xref&ltt>model_path&ltt>model_filename&ltt>steps&ltt>increment
s
DATADEF&ltt>firstlast-c&ltt><my_path>&ltt>keyhole69.odb&ltt>1&ltt>0
DATADEF&ltt>firstlast-c&ltt><my_path>&ltt>keyhole69.odb&ltt>last&ltt>last
```

The ODB file was opened in *fe-safe* (all variables, steps, and increments) and then the High Level Loading Definition HLDF file was opened in *fe-safe*. The loadings GUI showed transitions=off, and four load blocks as shown in Figure 13.14-4 below:

The blocks include:

- Block A: A sequence of elastic and plastic datasets with temperature datasets, with repeats
- Block B: A sequence of elastic and plastic datasets scaled by 1 and 0, with repeats.

Note: it is not realistic to scale an elastic-plastic pair by a number other than 0, as the stress strain response comes from the Finite Element solution.

- Block C: A sequence of all of the elastic stresses in the Current FE Models window
- Block D: A sequence of the first step, 0 increment, followed by the last step, last increment.

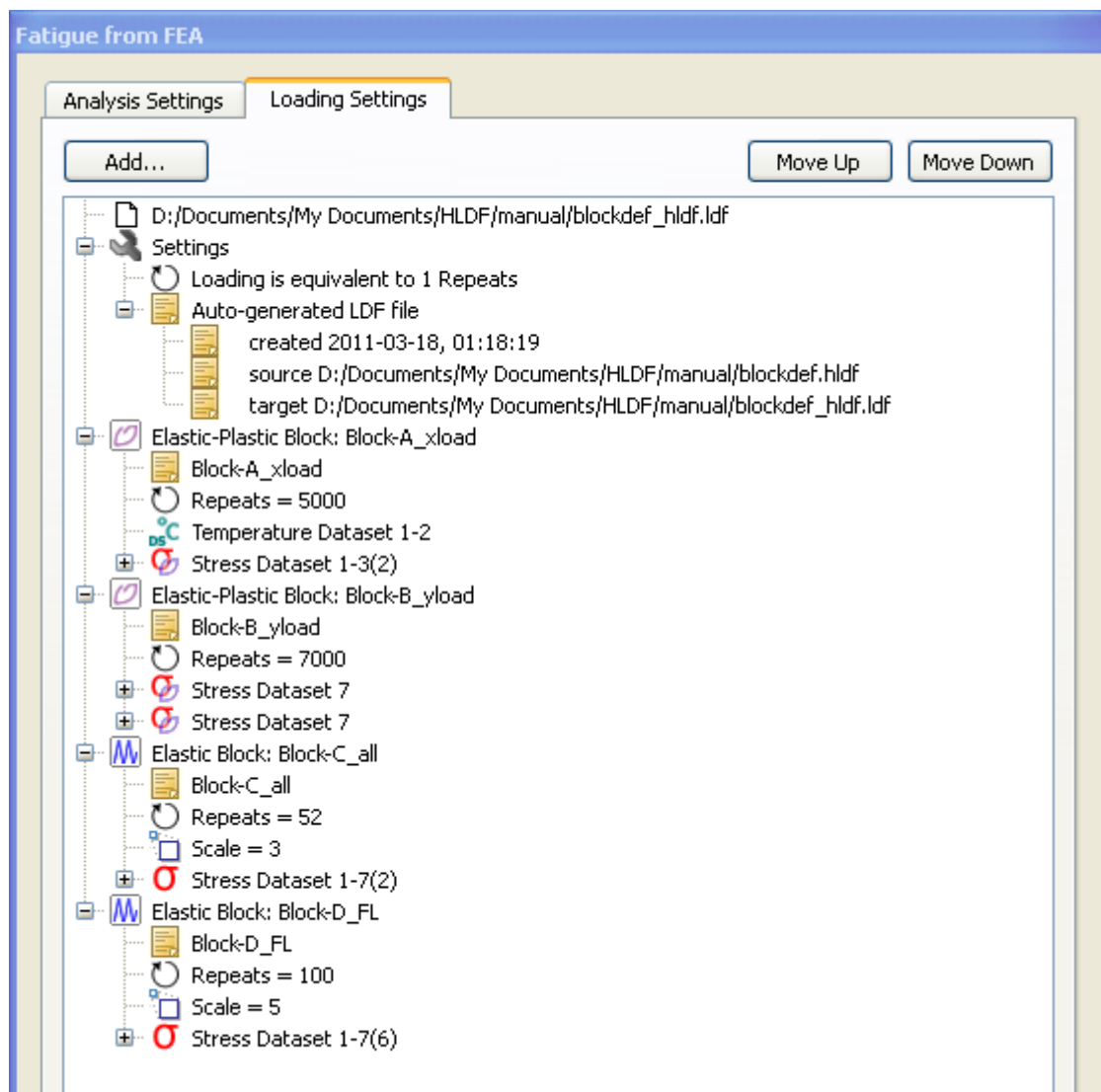


Figure 13.14-4

14 Fatigue analysis of elastic FEA results

14.1 Uniaxial strain-life

This algorithm is provided for analysing uniaxial stresses. Elastic FEA stresses are required. Uniaxial stresses rarely occur in practice, and the multi-axial algorithms are strongly recommended. The theoretical background to strain-life analysis for uniaxial stresses is described in the Fatigue Theory Reference Manual, section 2.

The elastic-plastic strain amplitude is used to calculate the fatigue life. Morrow, Smith-Watson-Topper, Walker (see section 14.4) or no mean stress correction can be selected. The strain-life equations for these mean stress corrections are:

No MSC:

$$\frac{\Delta \varepsilon}{2} = \frac{\sigma_f'}{E} (2N_f)^b + \varepsilon_f' (2N_f)^c$$

Morrow:

$$\frac{\Delta \varepsilon}{2} = \frac{(\sigma_f' - \sigma_m)}{E} (2N_f)^b + \varepsilon_f' (2N_f)^c$$

SWT:

$$\frac{\Delta \varepsilon}{2} \sigma_{\max} = \frac{(\sigma_f')^2}{E} (2N_f)^{2b} + \sigma_f' \varepsilon_f' (2N_f)^{b+c}$$

Although these strain-life algorithms are intended for uniaxial stress states, *fe-safe* uses multiaxial methods to calculate elastic strains from elastic FEA stresses, and a multiaxial elastic-plastic correction to derive the strain amplitudes and stress values used in these equations.

14.2 Uniaxial stress life

This algorithm is provided for analysing uniaxial stresses. Elastic FEA stresses are required. Uniaxial stresses rarely occur in practice, and the multi-axial algorithms are strongly recommended.

For this analysis the stress amplitude is used to calculate the fatigue life. The fatigue life curve can be an S-N curve or a stress-life curve derived from local strain materials data. This is configured from the **Analysis Options** dialog.

When using the local strain materials data the life curve is defined by the equation:

$$\frac{\Delta \sigma}{2} = \sigma_f' (2N_f)^b$$

and a multiaxial cyclic plasticity correction is used to convert the elastic FEA stresses to elastic-plastic stress-strain.

Otherwise the life curve is defined by the S-N values defined in the material database, and the plasticity correction can be optionally performed depending on settings in **Analysis Options** dialogue [**FEA Fatigue** >> **Analysis Options...**], **Stress Analysis** tab (see section 5).

Goodman, Gerber, Walker or no mean stress correction can be selected – see sections 14.3 and 14.4.

For the theoretical background to S-N curve analysis for uniaxial stresses, see the Fatigue Theory Reference Manual.

14.3 Goodman and Gerber mean stress corrections

As each stress cycle is extracted, the stress range and mean stress for the cycle are calculated.

If S_{max} is the maximum stress in the cycle, and S_{min} is the minimum stress in the cycle

The stress range is $S_{max} - S_{min}$

the stress amplitude is $(S_{max} - S_{min})/2$

and the mean stress is $(S_{max} + S_{min})/2$

For Goodman and Gerber mean stress corrections, the stress amplitude and mean stress are used to calculate the equivalent stress amplitude S_{a0} at zero mean stress. The endurance for the cycle is calculated using this stress amplitude. This method allows the Goodman and Gerber mean stress corrections to be used for endurance other than the endurance limit.

In *fe-safe*, the Goodman diagram is implemented as shown by the full line in *Figure 14.3-1*. This means that no allowance is made for possible beneficial effects of low compressive mean stresses, nor is any allowance made for possible detrimental effects of high compressive mean stresses.

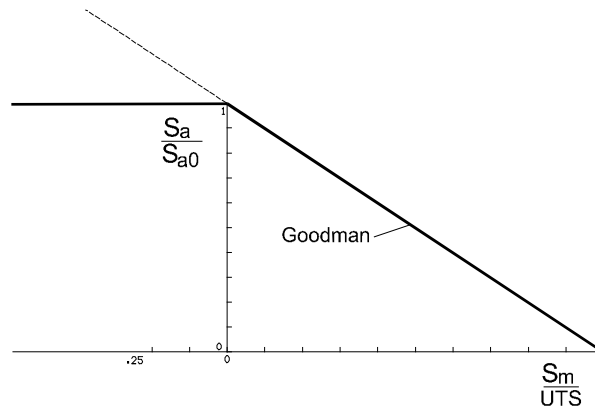


Figure 14.3-1. Goodman mean stress correction

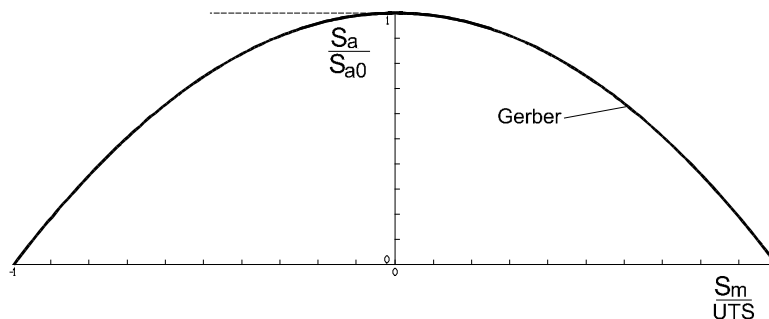


Figure 14.3-2. Gerber mean stress correction

See the Fatigue Theory Reference Manual, Section 5 for the background to (and limitations of) Goodman and Gerber mean stress corrections.

14.4 Walker mean stress correction

The Walker mean stress correction is widely used in crack growth calculations. For crack initiation calculations it has been expressed as:

$$\frac{\Delta\sigma_r}{2} = \sigma_{\max}^{1-\gamma} \left(\frac{\Delta\sigma}{2} \right)^\gamma = \sigma_{\max} \left(\frac{1-R}{2} \right)^\gamma = \frac{\Delta\sigma}{2} \left(\frac{2}{1-R} \right)^{1-\gamma}$$

Where:

$\frac{\Delta\sigma}{2}$ is the stress amplitude

$\frac{\Delta\sigma_r}{2}$ is the effective stress amplitude at mean stress = 0

γ is a material constant (the 'Walker parameter')

$R = \frac{\sigma_{\min}}{\sigma_{\max}}$ is the stress ratio

This is similar to the Smith–Watson–Topper parameter with the additional materials property γ . (putting $\gamma = 0.5$ gives the Smith–Watson–Topper parameter).

The following graphs show examples of the correlation obtained using the Walker parameter.

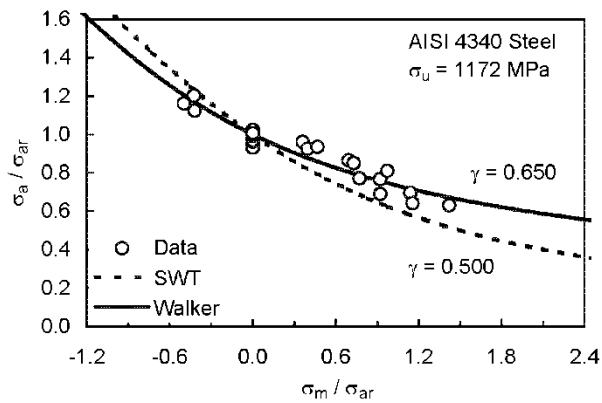


Figure 14.4-1 AISI 4340 steel

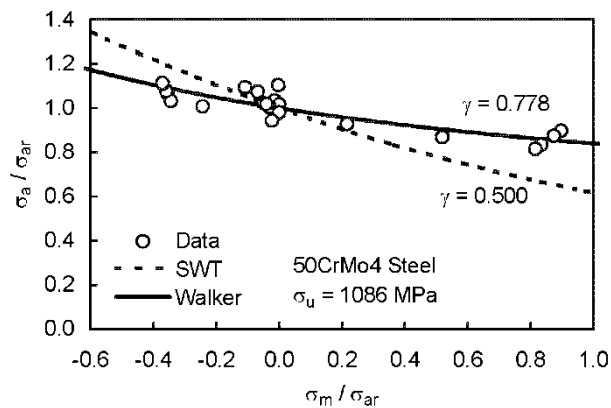


Figure 14.4-2 50CrMo4 steel

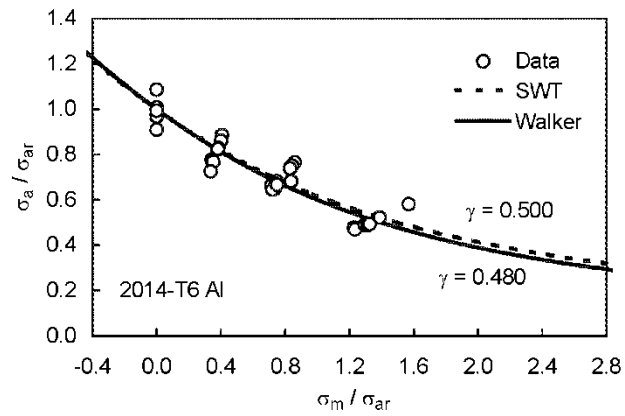


Figure 14.4-3 Aluminium-copper alloy 2014-T6

For steels the following approximation for the Walker parameter has been suggested:

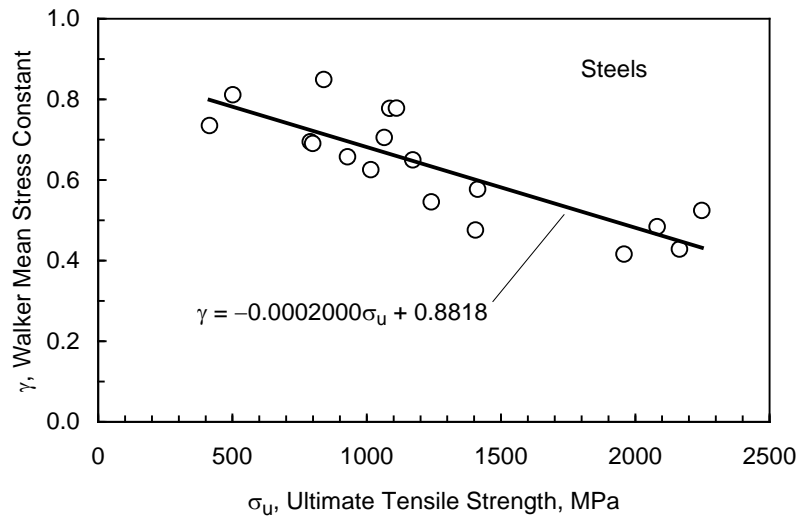


Figure 14.4-4 Trend of Walker parameter γ with Ultimate Tensile Strength for steels

No such trend has been determined for aluminium alloys:

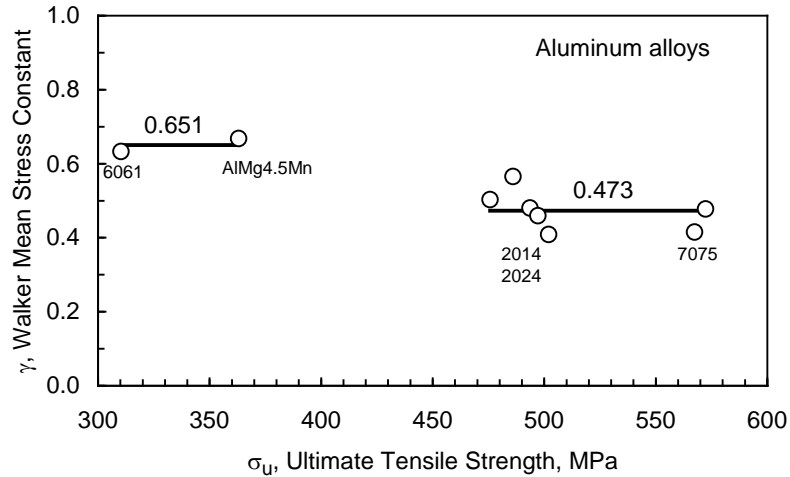


Figure 14.4-5 Trend of Walker parameter γ with Ultimate Tensile Strength for aluminium alloys

References:

N. E. Dowling, C. A. Calhoun, and A. Arcari, "Mean Stress Effects in Stress–Life Fatigue and the Walker Equation," *Fatigue and Fracture of Engineering Materials and Structures*, Vol. 32, No. 3, March 2009, pp. 163–179. Also, *Erratum*, Vol. 32, October 2009, p. 866.

N. E. Dowling, "Mean Stress Effects in Strain–Life Fatigue," *Fatigue and Fracture of Engineering Materials and Structures*, Vol. 32, No. 12, December 2009, pp. 1004–1019.

14.5 von Mises life

This calculation uses the von Mises stress as the damage parameter.

Because the von Mises stress is always positive the sign must be attributed. This can be done by applying the same sign as either (a) the Hydrostatic Stress (average of principals) or (b) the absolute maximum principal stress. This is also configured from the **Analysis Options** dialogue.

Cycles of von Mises stress are extracted. The endurance is calculated from an S–N curve or from a stress–life curved derived from local strain materials data. This is also configured from the **Analysis Options** dialogue.

When using the local strain materials data the life curve is defined by the equation:

$$\frac{\Delta\sigma}{2} = \sigma_f '(2N_f)^b$$

and a cyclic plasticity correction is used to convert the elastic FEA stresses to elastic–plastic stress–strain.

Otherwise the life curve is defined by the S–N values defined in the material database, and the plasticity correction can be optionally performed depending on settings in **Analysis Options** dialogue (**FEA Fatigue >> Analysis Options...**, **Stress Analysis** tab).

The von Mises Stress algorithm is not recommended for general fatigue analysis. See the Fatigue Theory Reference Manual, Section 7 for a discussion of this algorithm.

For finite life calculations Goodman, Gerber, Walker, User Defined, R ratio SN curves or no mean stress correction can be selected. See sections 14.3, 14.4., 14.9 and 14.11.

For infinite life calculations (FRF) a user defined, R ratio SN curves, Goodman or Gerber infinite life envelope analysis can be performed. See section 17.

This algorithm is not recommended because as with all ‘representative’ stress variables that have their sign defined by some criteria there is the possibility of sign oscillation. For the von Mises stress this occurs when the Hydrostatic stress is close to zero (i.e. the major two principal stresses are similar in magnitude and opposite). This is why using such ‘representative’ stress values for fatigue analysis can cause spurious hot spots.

14.6 Modified Manson–McKnight analysis

This algorithm has been superseded by the Manson–McKnight Octahedral analysis described below and is no longer available in *fe-safe*.

14.7 Manson–McKnight Octahedral analysis

This compliance algorithm may be required by national regulatory bodies e.g. in the aviation sector. It is a partial implementation of the NASALife fatigue prediction software referenced below, but is limited to the most simple Manson–McKnight algorithm and does not address creep at all. It replaces the Modified Manson–McKnight NASALife and Fillipini algorithms in previous versions of *fe-safe*.

The Manson–McKnight algorithm is a multiaxial fatigue model which allows for a multiaxial stress state and mean–stress effects to be accounted for. The algorithm is based on the concept of a signed von Mises mean stress. The Manson–McKnight scalar mean stress σ_m is expressed as the product of the von Mises yield criterion and the sign of the first stress tensor invariant I_1 (i.e. hydrostatic component) of a mean–stress tensor, which is simply the mean of the two tensors which define a damage cycle:

$$\sigma_m = \sqrt{\frac{1}{2}[(\sigma_{xm} - \sigma_{ym})^2 + (\sigma_{ym} - \sigma_{zm})^2 + (\sigma_{xm} - \sigma_{zm})^2 + 6(\tau_{xym}^2 + \tau_{yzm}^2 + \tau_{xzm}^2)]} \times \text{sgn}(I_{1,d})$$

$$I_{1,d} = \sigma_{xm} + \sigma_{ym} + \sigma_{zm}$$

Similarly, the scalar amplitude σ_a of a damage cycle is derived from a tensor amplitude which is half the difference of the two stress tensors:

$$\sigma_a = \sqrt{\frac{1}{2}[(\sigma_{xa} - \sigma_{ya})^2 + (\sigma_{ya} - \sigma_{za})^2 + (\sigma_{xa} - \sigma_{za})^2 + 6(\tau_{xya}^2 + \tau_{yza}^2 + \tau_{xza}^2)]}$$

The above parameters are used to calculate the damage of each potential cycle, i.e. every pair of tensors in the stress history, using the Walker mean–stress correction with two limitations:

1. If $\sigma_m < 0$, i.e. the stress ratio $R < -1$, then a value of $R = -1$ is used. This limits the reduction in damage attributed to compressive cycles.
2. If $\sigma_a > \sigma_y$, where σ_y is the 0.2% proof stress, an adjustment is made to cycles which are partly compressive ($R < 0$) so that their amplitudes are corrected as if they were fully tensile.

The highest damage thus obtained defines the Most Damaging Major Cycle (MDMC). This is then used to define a coordinate system for rainflow cycles as follows:

- The principal stress directions are computed for the MDMC;
- An octahedral plane, whose normal is denoted \underline{n}_{oct} , is defined as the normalised sum of the principal vectors. It can be shown that the normal stress on this plane is proportional to the hydrostatic stress;

- The shear component of the traction on this plane is calculated. It can be shown that this is proportional to the von Mises stress;
- The normalised direction of this shear stress is then denoted \underline{n}_τ .

Now for each stress tensor $\underline{\underline{\sigma}}$ in the loading history, the rainflow parameter is given by $\underline{n}_\tau^T \underline{\underline{\sigma}} \underline{n}_\tau$, which is the normal component in the direction of the maximum octahedral shear stress.

Once rainflow cycles have been defined in this way, their damage is calculated using the Manson-McKnight formulation above.

Note that the most damaging cycle thus identified may not be the same as the Most Damaging Major Cycle defined above, since the damage parameter differs from the rainflow parameter. In this case, the MDMC replaces the worst rainflow cycle in the Miner's rule summation for the whole stress history and this is reflected in *fe-safe's* standard Life contour. A second life contour is output by this algorithm which takes no account of the MDMC.

References

J. Z. Gyekensi, P. L. Murthy and S. K. Mital, "NASALIFE- Component Fatigue and Creep Life Prediction Program", *National Aeronautics and Space Administration*, Cleveland, 2005.

14.8 FKM Guideline analysis

This stress-based compliance algorithm is an implementation of the FKM Guideline, 'Analytical Strength Assessment of Components, 6th Edition'. In particular the method of chapter 4 of the Guideline document, 'Assessment of the fatigue strength using local stresses' is considered for non-welded components. Stress amplitudes extracted by rainflow counting the loading history are applied to the consistent version of Miner's rule (discussed in chapter 4 of the FKM guideline) to produce the variable amplitude fatigue strength. The elementary version of Miner's rule is also available as an option via the group properties.

Compliance is assessed by the degrees of utilization, based on the ratio of the largest stress amplitude to the variable amplitude fatigue strength. Assessment of the component fatigue strength is achieved if the largest degree of utilisation is not greater than 1. The results reported by *fe-safe* are the individual utilization for each principal direction, and the total combined utilization.

Datasets within a loading block are combined by superposition before cycle extraction by rainflow counting. To comply with the FKM Guideline, only proportional loading is valid within a loading block, i.e. the direction of the principal directions does not change. For non-proportional loading, the datasets should be applied in separate loading blocks, each with a single superposition to ensure proportionality. For each principal direction, the largest individual utilization over all loading blocks is reported. A combined degree of utilization ($a_{BK,I}$, $a_{BK,II}$, ...) will be calculated for each loading block (I, II, ...) and summed to give the overall degree of utilization (Reference [1] page 109).

$$a_{BK} = a_{BK,I} + a_{BK,II} + \dots$$

The degree of utilization is calculated according to the number of repeats specified in the loading definition. The block repeats property should be combined with the loading block history to specify the number of cycles required for the analysis. For example, to assess the degree of utilization at the knee point for a steel component with fully reversed loading, apply 1e6 repeats to the default loading.

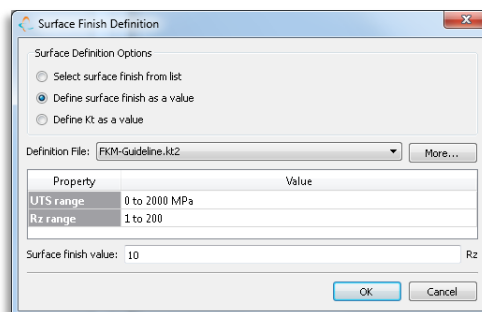
Materials can be selected from either the 'FKM_Fe.dbase' or 'FKM_Al.dbase' material databases for steel/iron and aluminium materials respectively. Please note: the databases are delivered in the fe-safe product installation directory under the /database sub-directory. Open the database to access the materials (see section 8 for more details).

Alternatively, materials from the existing databases can be used with the FKM Guideline algorithm by adding the necessary properties:

- fkm : Material Type – Required for use with this algorithm
- fkm : Grey Iron Index – Required for GJL material type
- fkm : Elongation (%) – Required for Wrought aluminium alloy material type

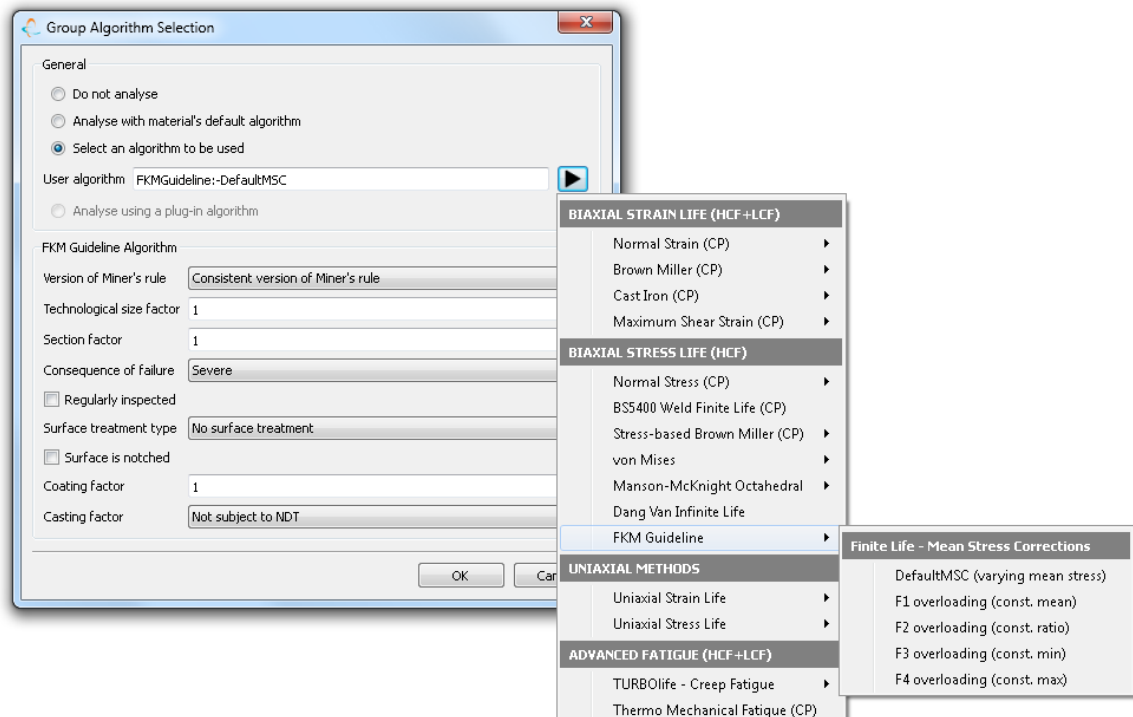
The relative stress gradient, in the direction normal to the component surface is calculated automatically for 3-dimensional element types. A maximum search depth is set by the 'taylor : L (mm)' material parameter.

Surface roughness for groups analysed with this algorithm must be set using the 'FKM-Guideline.kt2' definition file available when 'Define surface finish as a value' is selected in the Surface Finish Definition dialog. The surface roughness value Rz is valid in the range 1 to 200 microns.



Other group properties are set through the Group Algorithm Selection dialog. The FKM Guideline options are only visible when the algorithm is selected or has been specified as the material default algorithm in

the database. The properties should be set according to the guideline document. Note that the coating factor is only applied to aluminium alloys and that the casting factor is only relevant to cast iron material types.



The guideline considers four separate types of overloading which are accessed via the algorithm selection dialog of *fe-safe* as methods of mean stress correction, along with the default method (described below). The characteristics of the loading history for each method are

- Type of overloading F1: constant mean stress
- Type of overloading F2: constant stress ratio
- Type of overloading F3: constant minimum stress
- Type of overloading F4: constant maximum stress

In the case where none of the above conditions apply, the default mean stress correction option for varying mean stress should be used. In this case the stress ratio of each cycle is made equivalent to that of the largest cycle by adjusting the stress amplitude according to type of overloading F2.

References

- [1] Forschungskuratorium Maschinenbau, "Analytical Strength Assessment of Components in Mechanical Engineering", FKM-Guideline, 6th Ed., 2012
- [2] Forschungskuratorium Maschinenbau, "Rechnerischer Festigkeitsnachweis für Maschinenbauteile", FKM-Richtlinie, 6. Auflage, 2012

14.9 Maximum principal stress analysis (Normal Stress)

This is a critical plane multi-axial fatigue algorithm, using planes perpendicular to the surface. Fatigue lives are calculated on eighteen planes, spaced at 10 degree increments. On each plane

- the principal stresses are used to calculate the time history of the stress normal to the plane;
- the cycles are extracted and corrected for mean-stress;

- the fatigue life calculated.

The fatigue life is the shortest life calculated for the series of planes

The fatigue life curve can be an S–N curve or a stress–life curve derived from local strain materials data. This is configured from the **Analysis Options** dialog.

When using the local strain materials data the life curve is defined by the equation:

$$\frac{\Delta\sigma}{2} = \sigma_f' (2N_f)^b$$

and a cyclic plasticity correction is used to convert the elastic FEA stresses to elastic–plastic stress–strain.

Otherwise the life curve is defined by the S–N values defined in the material database, and the plasticity correction can be optionally performed depending on settings in **Analysis Options** dialogue [**FEA Fatigue >> Analysis Options...**], **Stress Analysis** tab (see section 5).

For finite life calculations Goodman, Gerber, Walker, Morrow, Morrow B, Smith–Watson–Topper, R ratio SN curves, User Defined or no mean stress correction can be selected. See sections 14.1, 14.3, 14.4, 14.8, 14.9 and 14.11.

For infinite life calculations (FRF) a user defined, R ratio SN curves, Goodman or Gerber infinite life envelope analysis are supported, see section 17.

Two non–standard fatigue analysis are also supported. To enable these options check on the **Enable non standard fatigue modules** on the **Legacy** tab of the **Analysis Options** dialogue.

The Buch analysis is a hybrid finite and infinite life calculation, see section 17.

The Haigh diagram creation module (see 14.15.) has now been superseded by the diagnostic option for creating Haigh and Smith diagrams for all analysis algorithms.

This algorithm can give very non–conservative results for most ductile metals – see the Fatigue Theory Reference Manual, section 7.

14.10 Alternative Morrow mean stress correction (Morrow B)

This is identical to the Morrow mean stress correction as defined in section 14.1 except that σ_r' is replaced by σ_r ,

As with the standard Morrow mean stress correction this option is only available for finite life calculations.

14.11 User defined mean stress correction

The user defined mean stress correction (MSC) function can be used to define a set of correction factors as a function of the mean stress of a cycle, in a similar manner to a Goodman diagram.

The vertical axis is made non-dimensional, by expressing the stress amplitude, S_a , as a ratio:

$$\frac{S_a}{S_{a0}}$$

where S_{a0} is the stress amplitude at zero mean stress.

This ratio is the correction factor which converts the stress amplitude at zero mean stress, to the stress amplitude at any specified mean stress. This ratio has a value of 1.0 at a mean stress of zero.

The mean stress axis is made non-dimensional by dividing each mean stress by the material ultimate tensile strength, UTS. For compressive mean stresses, the ultimate compressive strength, UCS, can be used, provided that the UCS is defined in the material database.

At a mean stress equal to the material UTS, the allowable stress amplitude is zero, as the material is on the point of fracture. The mean stress axis therefore has a value of 1.0 at $S_a = 0$.

For a cycle (S_a , S_m) the value of the *MSC* factor is extracted for the value of S_m , and the equivalent stress amplitude at zero mean stress is:

$$S_{a0} = \frac{S_a}{MSC}$$

or, if the fatigue algorithm uses strain amplitudes then:

$$e_{a0} = \frac{e_a}{MSC}$$

This can also be defined as a Smith diagram.

Each material can have a default user defined MSC. This will be used as the default MSC when the material is selected for an analysis and also as the infinite life envelope for Haigh and Smith diagram diagnostics.

For details of how to define a mean stress correction curve in *fe-safe*, see appendix E 205.2.

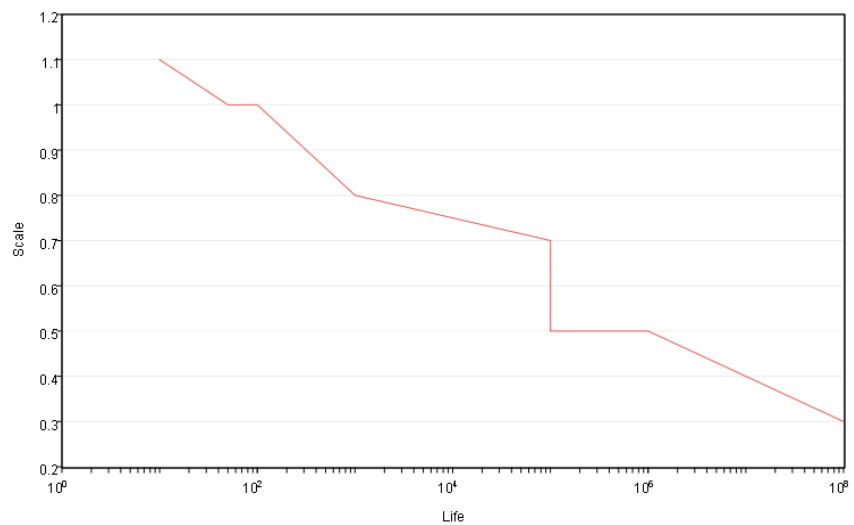
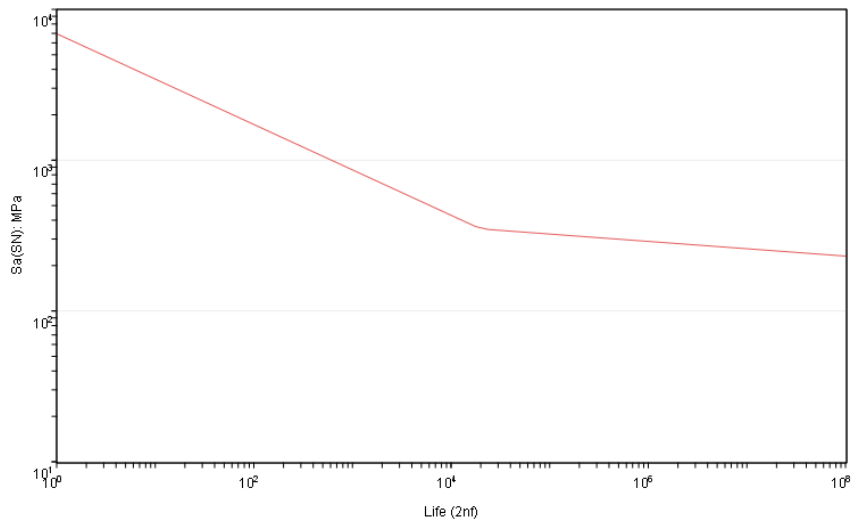
14.12 S-N data knock-down curve

A series of additional S-N data scale factors in the form of a knock-down curve, if defined for the selected material (see section 8), can be applied to the defined S-N data by clicking on **Knock-Down** in the **Group Parameters** region of the **Fatigue from FEA** dialogue. Scale factors will be extracted from the curve and applied to modify all stress data points in the defined material S-N curve, interpolating and extrapolating the available data points as necessary.

The following example will illustrate the application of the knock-down curve. Data points in black are defined; data points in blue are calculated.

Fatigue analysis of elastic FEA results

Original S-N curve		Knockdown curve		Modified S-N curve	
S	Nf	Scale	Nf	S	Nf
2800	10	1.1	10	3080	10
1724.833	50	1	50	1724.833	50
1400	100	1	100	1400	100
700	1000	0.8	1000	560	1000
350	10000	0.75	10000	262.5	10000
312.8662	100000	0.7	100000	219.0064	100000
312.8662	100001	0.5	100001	156.4331	100001
279.6722	1000000	0.5	1000000	139.8361	1000000
250	10000000	0.4	10000000	100	10000000
		0.3	1E+08		



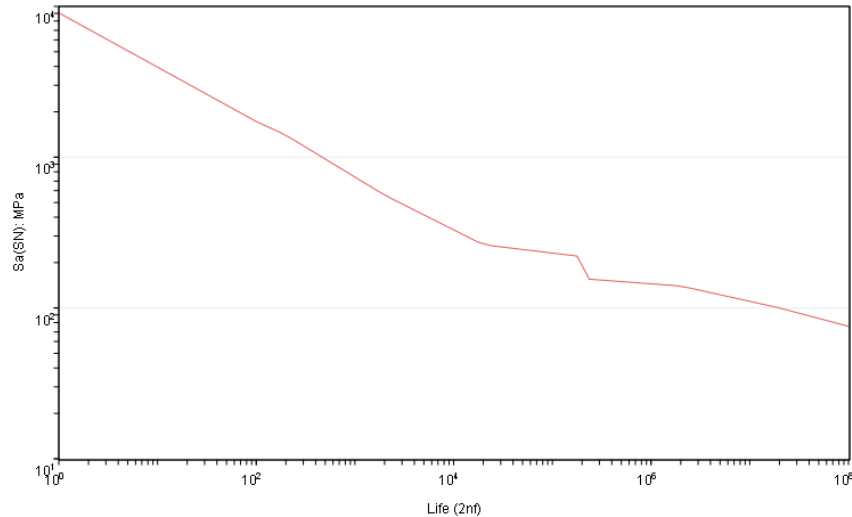


Figure 14.12-1 Original S-N curve, knock-down curve, modified S-N curve

This option applies to stress-based analyses only where the S-N material data is available. The scaling will not be applied if the S-N data is derived from the local strain parameters.

For details of how to define a mean stress correction curve in *fe-safe*, see Appendix E.

14.13 Multiple S-N curves dependant on stress ratio (R-ratio)

Multiple S-N curves may be defined for a material where each S-N curve is associated with a particular stress ratio (R-ratio, see section 8). Where S-N curves are defined for 2 or more R values, *fe-safe* can perform a mean stress correction based on this data. When calculating the fatigue damage for a particular stress cycle in the loading, the mean stress and stress ratio for the cycle are calculated.

For S-N data that is dependent on stress ratio only (constant temperature), the interpolation is performed as follows:

- The stress ratio for the cycle is identified.
- A new S-N curve is created for the stress ratio of interest by linear interpolation between S-N curves for two different stress ratios which bracket the required stress ratio.
- Any value above or below the range of stress ratios defined limits will use the S-N data for the limiting values of R.
- An optional warning is given if data is extrapolated beyond defined stress ratio limits. See section 5.7.4 for details of the optional warning set on the Properties tab of the Analysis Options dialogue

For S-N data that is dependent on both temperature and stress ratio, the interpolation is performed as follows:

- The stress ratio and temperature for the cycle are identified.
- Two new S-N curves are created for two different stress ratios that bracket the required stress ratio, by linear interpolation between S-N curves, for two different temperatures which bracket the required temperature.
- Any value above or below the range of stress ratios defined limits will use the S-N data for the limiting values of R.

- An optional warning is given if data is extrapolated beyond defined temperature limits or defined stress ratio limits. The warning indicates which limit or limits have been exceeded. See section 5.7.4 for details of the optional warning set on the Properties tab of the Analysis Options dialogue.

For finite life calculations the S-N curve for the Stress ratio of the cycle is used. If the Stress ratio falls between two known R-ratios, the S-N data is linearly interpolated between them.

For infinite life calculations the FRF envelope is constructed by looking up the FRF design life on the S-N curves for the appropriate Stress ratio, then adding the corresponding point to the envelope. If the highest mean stress on the envelope is less than the UTS, the envelope is taken horizontally out to the UTS, at which point it drops to 0. If the lowest mean stress on the envelope is greater than the UCS (which if undefined may take its value from the UTS) then the envelope is taken horizontally out to the UCS but does not drop down to zero.

This option can only be used with the following stress-based algorithms: von Mises, normal stress and stress-based Brown Miller.

14.14 Buch calculations

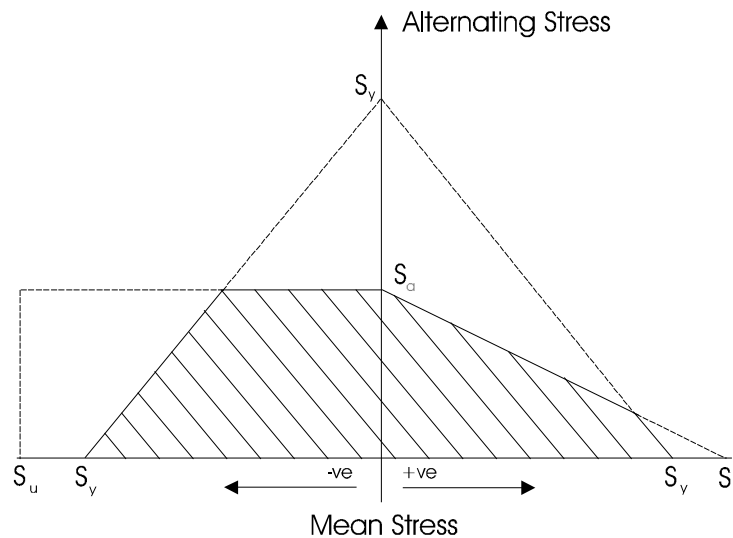


Figure 14.14-1

The Buch calculation is very similar to the fatigue reserve factor (FRF) calculation described in section 17.3, except that the envelope is a function of both the materials UTS (S_u) and the yield stress (S_y). The yield stress is taken to be the 0.2 % proof stress. (Ref : Buch, A 'Fatigue Strength Calculation', Trans Tech Publications, 1988, (6) "Effects of Mean Stress"). This calculation is more conservative than a Goodman calculation for large tensile or large compressive mean stresses. The infinite life envelope is defined as in *Figure 14.8-1*. The diagram indicates that if the stresses are within the shaded area the component will have a calculated infinite life.

The *fe-safe* analysis calculates a Fatigue Reserve Factor value at the node, using the method described in Section 17.

The Buch method has been extended for use in finite life design. As shown in *Figure 14.8.2*, curves for different fatigue endurance values converge to the same curve in the region clipped by the lines joining the yield stresses. It is not possible to determine a fatigue life in this region, and *fe-safe* calculates a pseudo-life in this region. It is assumed that the S-N curve has a constant slope in the high cycle fatigue region, and the slope *b* at an endurance of 10^7 cycles is used as an inverse power on the factor to obtain the fatigue life.

$$FOLBuch = (FRF)^{1/b}; \quad N_{f-Buch} = FOLBuch \times DesignLife$$

This method will provide consistent contour plots for FRF and fatigue life calculations performed with the Buch algorithm. However it should be noted that, for cycles in the ‘clipped’ region, the method will give calculated lives that are a function of the specified design life. In other words, the fatigue life will change with the design life.

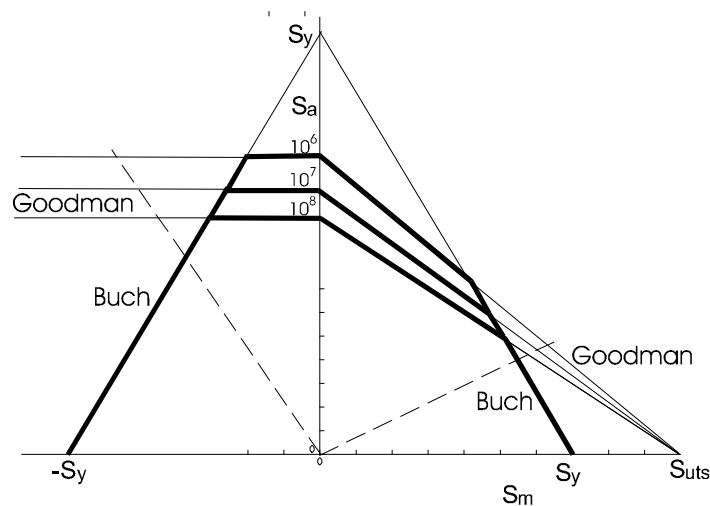


Figure 14.14-2

To allow this algorithm to be selected check on the **Enable non standard fatigue modules** on the **Analysis** tab of the **Analysis Options** dialogue.

14.15 Haigh diagram calculations

The Haigh diagram is a graph of stress amplitude versus mean stress. *fe-safe* calculates the Haigh diagram. At each node, the most damaging cycle is exported. This produces a single diagram showing the most damaging cycle at every node. This can be plotted and listed in *fe-safe* (see section 7), and gives a visual indication of the Fatigue Reserve Factor (FRF).

No fatigue lives are calculated in this analysis, and therefore contour plots of fatigue life cannot be produced.

This module has now been superseded by the diagnostic option for creating Haigh and Smith diagrams for all analysis algorithms.

To allow this algorithm to be selected check on the **Enable non standard fatigue modules** on the **Legacy** tab of the **Analysis Options** dialogue.

14.16 Maximum principal strain analysis (Normal Strain)

This is a critical plane multi-axial fatigue algorithm, using planes perpendicular to the surface. Stress results from an elastic FEA are required. A multi-axial elastic-plastic correction is used to calculate elastic-plastic stress-strains from the elastic FEA stresses

Fatigue lives are calculated on eighteen planes, spaced at 10 degree increments. On each plane

- the principal strains are used to calculate the time history of the strain normal to the plane.
- cycles of normal strain are extracted and corrected for the mean stress
- the fatigue life calculated

The fatigue life is the shortest life calculated for the series of planes

This algorithm uses the strain-life curve defined by the equation:

$$\frac{\Delta \varepsilon}{2} = \frac{\sigma_f'}{E} (2N_f)^b + \varepsilon_f' (2N_f)^c$$

Morrow, Walker, Smith-Watson-Topper, User Defined or no mean stress correction may be selected. See section 14.9 for a definition of the user-defined MSC. For the Morrow mean stress correction the strain-life equation is modified to:

$$\frac{\Delta \varepsilon}{2} = \frac{(\sigma_f' - \sigma_m)}{E} (2N_f)^b + \varepsilon_f' (2N_f)^c$$

For the Walker mean stress correction the strain-life equation is modified to:

$$\frac{\Delta \varepsilon}{2} = \left(\frac{1-R}{2}\right)^{(1-\gamma)} \frac{\sigma_f'}{E} 2N_f^b + \left(\frac{1-R}{2}\right)^{(1-\gamma)} \varepsilon_f' 2N_f^c$$

Rearranging this equation to show the correction applied to the left hand side gives

$$\left(\frac{2}{1-R}\right)^{1-\gamma} \frac{\Delta \varepsilon}{2} = \frac{\sigma_f'}{E} 2N_f^b + \varepsilon_f' 2N_f^c$$

The corrected strain amplitude then forms the damage parameter for the fatigue damage calculations.

Alternatively, an FRF calculation can be used with this algorithm – see section 17.3.

This algorithm can be also be used for fatigue analysis of elastic-plastic FEA results. (See section 15).

Fatigue analysis using principal strains can give very non-conservative results for ductile metals. See the Fatigue Theory Reference Manual, section 7 for the background to this algorithm.

14.17 Maximum shear strain analysis

This is a critical plane multi-axial fatigue algorithm, using planes perpendicular to the surface, and at 45° to the surface. Stress results from an elastic FEA are required. A multi-axial elastic-plastic correction is used to calculate elastic-plastic stress-strains from the elastic FEA stresses

On each of three planes, fatigue lives are calculated on eighteen subsidiary planes, spaced at 10° increments. On each plane:

- the principal strains are used to calculate the time history of shear strain and normal stress
- cycles of shear strain are extracted and corrected for the mean normal stress
- the fatigue life calculated

The fatigue life is the shortest life calculated for the series of planes.

This algorithm uses the strain–life curve defined by the equation:

$$\frac{\Delta\gamma}{2} = 1.3 \frac{\sigma_f'}{E} (2N_f)^b + 1.5 \varepsilon_f' (2N_f)^c$$

Morrow, User Defined or no mean stress correction may be selected. See section 14.9 for a definition of the user–defined MSC. For the Morrow mean stress correction the strain–life equation is modified to:

$$\frac{\Delta\gamma}{2} = 1.3 \frac{(\sigma_f' - \sigma_m)}{E} (2N_f)^b + 1.5 \varepsilon_f' (2N_f)^c$$

This algorithm can be also be used for fatigue analysis of elastic–plastic FEA results. (See section 15).

See the Fatigue Theory Reference Manual, section 7 for the background to this algorithm.

14.18 Brown Miller analysis

This is a critical plane multi–axial fatigue algorithm, using planes perpendicular to the surface, and at 45° to the surface. Stress results from an elastic FEA are required. A multi–axial elastic–plastic correction is used to calculate elastic–plastic stress–strains from the elastic FEA stresses

On each of three planes, fatigue lives are calculated on eighteen subsidiary planes, spaced at 10 degree increments. On each plane

- the principal strains are used to calculate the time history of the shear strain and the strain normal to the plane
- fatigue cycles are extracted and corrected for the effect of the mean normal stress
- the fatigue life is calculated

The fatigue life is the shortest life calculated for the series of planes

This algorithm uses the strain–life curve defined by the equation:

$$\frac{\Delta\gamma}{2} + \frac{\Delta\varepsilon_n}{2} = 1.65 \frac{\sigma_f'}{E} (2N_f)^b + 1.75 \varepsilon_f' (2N_f)^c$$

Morrow, User Defined or no mean stress correction may be selected. See section 14.9 for a definition of the user–defined MSC. For the Morrow mean stress correction the strain–life equation is modified to:

$$\frac{\Delta\gamma}{2} + \frac{\Delta\varepsilon_n}{2} = 1.65 \frac{(\sigma_f' - \sigma_m)}{E} (2N_f)^b + 1.75 \varepsilon_f' (2N_f)^c$$

This algorithm can be also be used for fatigue analysis of elastic-plastic FEA results. (See section 15).

The Brown Miller algorithm is the preferred algorithm for most conventional metals at room temperature and is the default algorithm for most materials in the *fe-safe* materials data base. See the Fatigue Theory Reference Manual, section 7 for the background to this algorithm.

14.19 Dang Van analysis

The Dang Van model is an endurance criterion for analysis of high cycle fatigue (i.e. infinite life design) of components subject to complex multiaxial stresses. The method calculates whether a component has 'infinite life', but does not calculate fatigue lives. It is essentially a 'pass/fail' analysis. See the Fatigue Theory Reference Manual for the background to this algorithm.

Additional materials data is required. The Dang Van failure line is plotted as a straight line using endurance limit stress data for at least two stress ratios, usually $R=0$ (constant amplitude) and $R=-1$. Up to seven points can be defined. Where there are more than two points, *fe-safe* calculates the straight line through these points using a least squares fit.

On the Dang Van diagram the load can be plotted in terms of the deviatoric stress and the hydrostatic stress.

On the first pass through the signal, *fe-safe* considers the elastic shakedown state resulting from the multiaxial load. The hydrostatic stress is subtracted from the direct stress, and the centre of minimum sphere which bounds the full signal is estimated. The minimum sphere that bounds the locus of the signal can be considered as the 'yield domain'.

A second pass through the signal refines the position of the centre of the signal, and calculates the minimum radius of the sphere. The centre of the sphere defines the stable residual stress tensor.

On the third pass through the signal the Tresca stresses are recalculated, where:

direct stress components = direct stress – hydrostatic stress – stable residual direct stress

shear stress components = shear stress – stable residual shear stress

The loading path (time history of loading) is plotted on the Dang Van diagram. The vertical component is the deviatoric Tresca stress and the horizontal component is the hydrostatic stress.

The stress-based factor of strength for any point in the loading is the distance between the loading path and the Dang Van failure line. A safety factor is calculated for each point in the loading as a ratio with respect to the distance from the Dang Van line. The safety factors can be expressed radially (w.r.t. the origin) or vertically (w.r.t. zero shear stress line).

Safety factors less than one imply yielding and a non-infinite life.

14.19.1 Material Definition

Two additional material parameters are required for Dang Van analyses – see section 8.

A sample material database containing Dang Van material parameters is included with *fe-safe*.

14.19.2 Calculated parameters

At each node the Dang Van analysis calculates three parameters, which are written to the results file for contour plotting. The three parameters are

- the FRF using a radial line;
- the FRF using a vertical line;
- a pass/fail (survival) value indicating whether the calculation shows infinite life.

Radial Factor

The radial factor is the ratio a/b , shown in *Figure 14.19-1*.

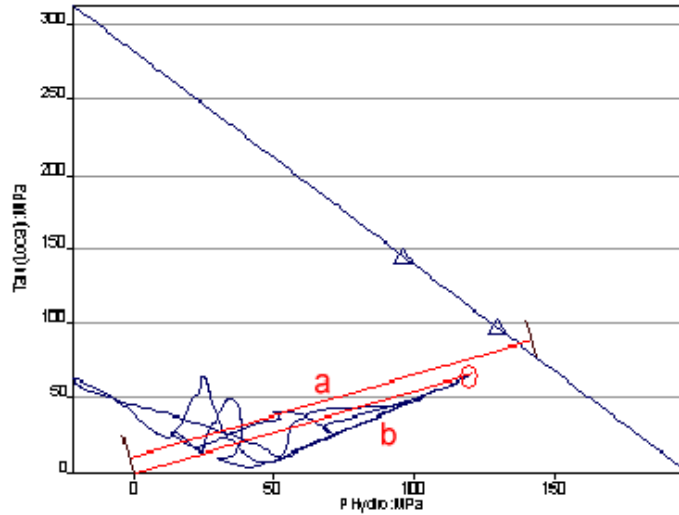


Figure 14.19-1

The loading path is indicated as a vector. The FRF is calculated for the point closest to the Dang Van infinite life line, circled in *Figure 14.19-1*.

Vertical Factor

The FRF is the ratio of b/a , shown in *Figure 14.19-2*.

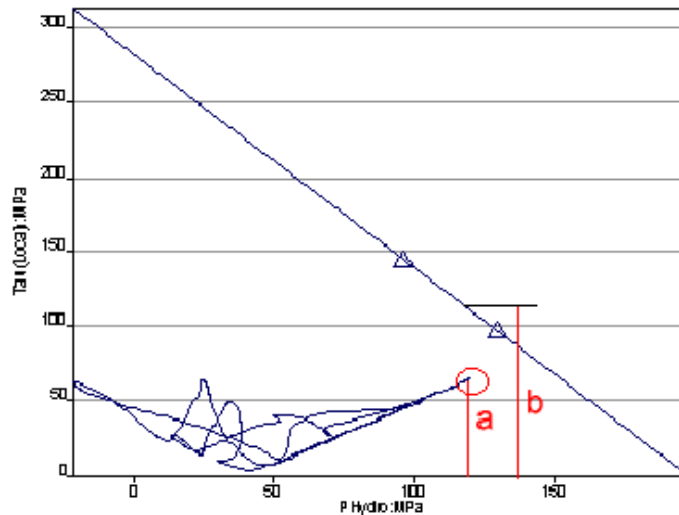


Figure 14.19-2

Prior to version 5.2-05 this calculation was only performed for the sample with the worst radial factor. At 5.2-05 this was modified to perform the calculation for each and every sample. The old behaviour can be enabled by adding the keyword "DANGVAN_VERTALLPTS". With a value of 0 *fe-safe* will do the worst point only calculation (pre 5.2-05 behaviour) and with a value of 1 (the default 5.2-05+ behaviour) *fe-safe* will do the calculation on every point.

Survival

The survival flag set to 1 if the analysis shows infinite life, otherwise it is set to zero.

14.19.3 Diagnostic output

For each analysis a diagnostics log file is created with the same name as the results file and the extension `.log`. This will contain the information displayed in the message log during the analysis.

For a Dang Van analysis, export options include a Dang Van plot and plots of the Hydrostatic pressure and the Local Shear strain. These should be selected in the **Exports and Outputs** dialogue, and the **Export Dang Van Plots** check box should be checked.

The output files will be written to the same location as the results file, with filenames which contain the results filename plus the element and node numbers.

e.g. If the output file is `/data/test1.fil`, then for element 27 node 4 the two created data files will be :

Dang Van Plot	<code>/data/test1_DangVan_27-4.dac</code>
Stress tensors, local shear and hydrostatic stress plots	<code>/data/test1_S-e_27-4.txt</code>

Both data files can be opened in *fe-safe* using **File >> Data Files >>Open Data File** and can then be plotted or listed (see section 7). Example results are shown in *Figure 14.13-3* and *Figure 14.13-4*.

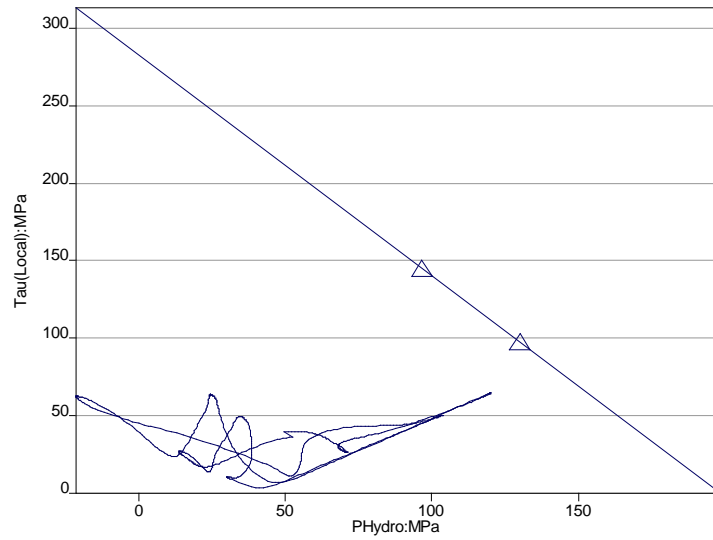


Figure 14.19-3 Dang Van plot.

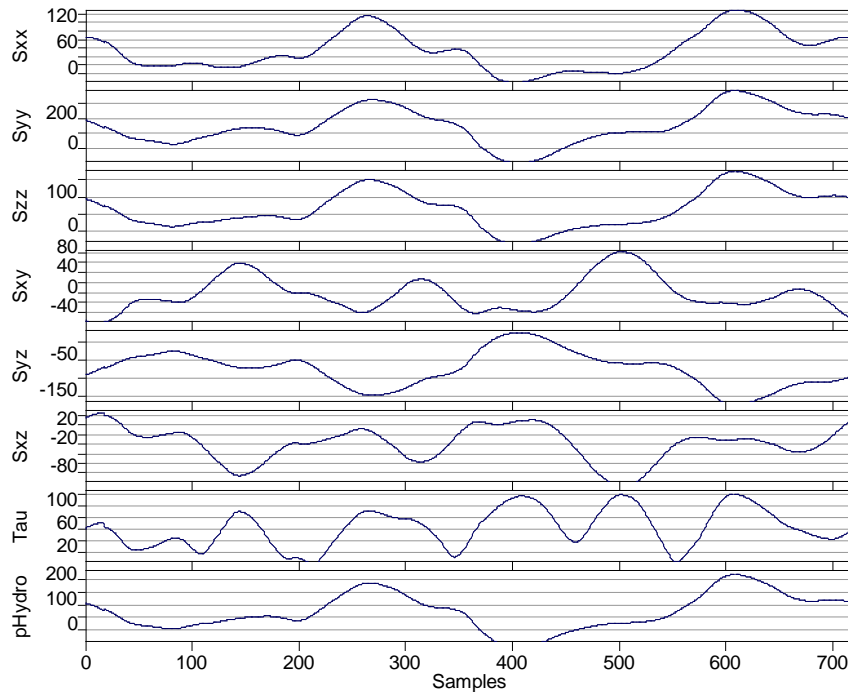
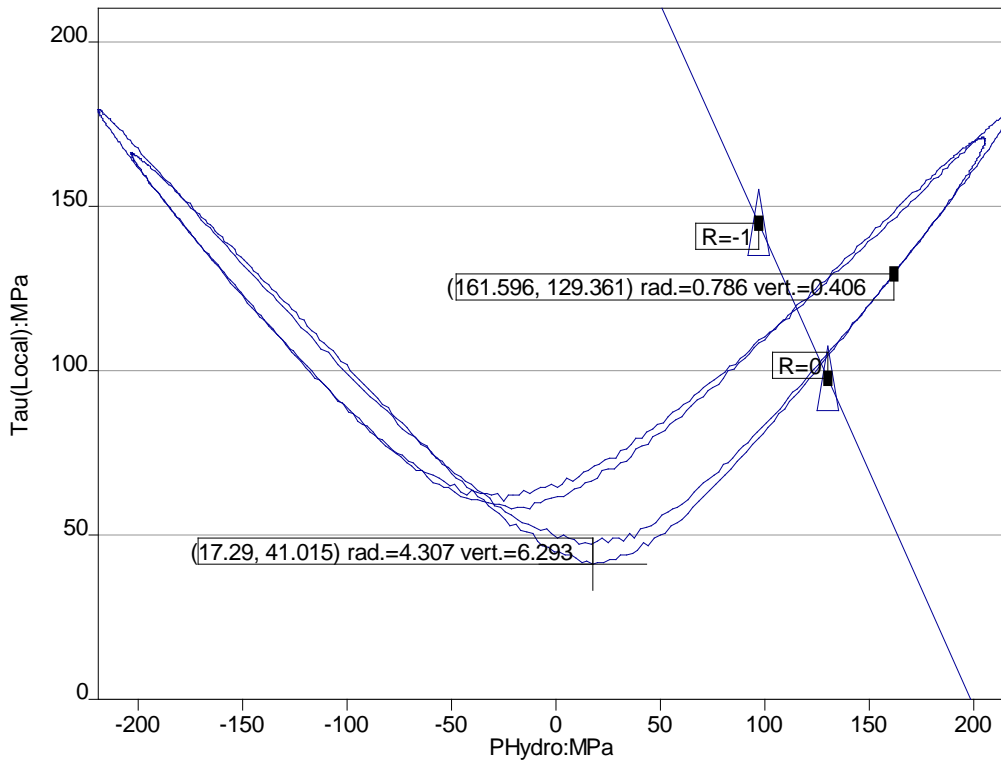


Figure 14.19-4 Plot of tensors, Hydrostatic Pressure and Local Shear Stress.

Using the cursor (Ctrl + T) on the Dang Van plot will show the radial and vertical factors calculated on a point by point basis. The plot below shows an active cursor and a cursor converted to text.



14.20 Prismatic Hull analysis

The Prismatic Hull [1-2] is an infinite life algorithm for complex multiaxial stress histories. It is conceptually similar to the Dang Van method, but addresses non-proportionality better. At each point in the stress tensor history, the stress tensor is decomposed into the hydrostatic stress and the deviatoric stress. Each deviatoric stress tensor can be transformed into a point in an equivalent 5D deviatoric space, by using 5 standard basis tensors [1]. The Prismatic Hull method then evaluates a set of enclosing hypercuboids in this 5D deviatoric space, and thus locates the largest enclosing prismatic hull of the deviatoric stress history. The dimensions of this enclosing hull are combined to produce a multiaxial measure of shear τ_a . This is combined with a weighted contribution from the maximum hydrostatic stress $\sigma_{H,max}$ to produce an overall effective multiaxial shear term for the history, which can be compared with the torsional endurance limit τ_{-1} to provide a safety factor.

The prismatic hull predicts infinite life if $\frac{\tau_a}{\sqrt{2}} + k\sigma_{H,max} < \tau_{-1}$ where τ_{-1} is the torsional endurance limit.

And a safety factor f can be computed as

$$f = \frac{\tau_{-1}}{\frac{\tau_a}{\sqrt{2}} + k\sigma_{H,max}}$$

The weighting factor k depends on the torsional and bending/tension endurance limits:

$$k = 3 \frac{\tau_{-1}}{\sigma_{-1}} - \sqrt{3}$$

and σ_{-1} is the endurance limit under fully reversed loading from the S-N curve.

Note that if residual stresses are defined, then the residual stress will be implicit in the hydrostatic term, but note that the safety factor applies to the total stress (residual + variable).

The accuracy of the multiaxial Prismatic Hull method was assessed for steels and aluminum alloys in [3], considering proportional and non-proportional stress and strain controlled tests reported in the literature. For complex non-proportional histories the Prismatic Hull is generally more conservative than Dang-Van [1,4]. A study of the use of the Prismatic Hull in assessing the fatigue of a Powertrain diesel crankshaft under peak torque conditions is given in [4], which found that the Prismatic Hull correctly predicted the crack location.

14.20.1 Material properties

The shear endurance limit is derived from the T-N curve (torsional form of S-N curve). This may be defined explicitly, similarly to the S-N curve, or derived from the S-N curve using a constant factor (see 8.5.2 and 8.5.4). If no T-N curve is defined, and no such constant factor, then a default constant factor of $1/\sqrt{3}$ is used, which implies $k=0$, so the hydrostatic stress has no effect on the safety factor. This would only be a sensible assumption for ductile materials. Note that the material databases supplied with *fe-safe* do not contain torsional endurance limit data, but the example data in the “Local” database does contain illustrations of the properties. See 8.5.2 and 8.5.4 for further details.

14.20.2 Calculated parameters

The calculated safety factor is output to the FRF-PH contour. In addition the maximum hydrostatic stress and shear amplitude $\frac{\tau_a}{\sqrt{2}}$ will be output to contours called MaxHydro and TauAmp-PH respectively.

If the standard contour for “Worst cycle mean stress and damage parameter” is selected, then the effective shear measure $\frac{\tau_a}{\sqrt{2}} + k\sigma_{H,max}$ will be output as the damage parameter, but the mean stress will be filled with zeroes – this is normally computed on a critical plane and is not really meaningful for this method. In this method the maximum hydrostatic stress is used instead of a mean stress correction.

14.21 High temperature fatigue analysis

Conventional high temperature fatigue analysis is described in section 18.

14.22 Fatigue analysis of cast irons

14.22.1 Technical discussion of fatigue analysis of cast irons

The theoretical background to the analysis of cast irons is described in the Fatigue Theory Reference Manual, section 2.12.

14.22.2 Material parameters for cast iron analyses

The Fatigue Theory Reference Manual, section 3.6 discusses the additional material data required for analysis of cast irons, and the *fe-safe* User Guide, section 8 discusses the implementation of these parameters in *fe-safe*.

The three cast irons sample materials in the database `local.dbase`, show examples of the extra parameters required for

- grey iron;
- compacted iron;
- nodular (SG) iron.

These materials did not require a value of **b2** (see section 8).

14.22.3 Cast iron fatigue damage calculation

A critical plane analysis is used, on planes perpendicular to the surface, using 18 planes at 10 degree intervals, with the normal strain on the plane as the damage parameter – see the Fatigue Theory Reference Manual, section 7.5.

For each plane the fatigue cycles are:

- Extracted;
- corrected for plasticity using a biaxial Neuber's rule;
- corrected for mean-stress.

The following damage equations is used:

$$\Delta D = \frac{(1 - D_i)^{P_i}}{(P_i + 1)N_{fi}}$$

where ΔD is the damage for the cycle, in the current damage increment;

D_i is the damage so far accumulated;

P_i is the damage rate parameter so far;

N_{fi} is the endurance of the cycle.

P_i for a cycle is defined by the relationship:

$$P_i = 2.55(\sigma_{\max} \varepsilon_a)^{-0.8}$$

The constant and exponent in the above equation are configurable in the **Cast Iron** tab of the **Analysis Options** dialogue as shown in *Figure 14.22-1*. The default values are those shown above (constant = 2.55, exponent = -0.8). This equation becomes the conventional linear Miner's summation if P_i is set equal to zero.

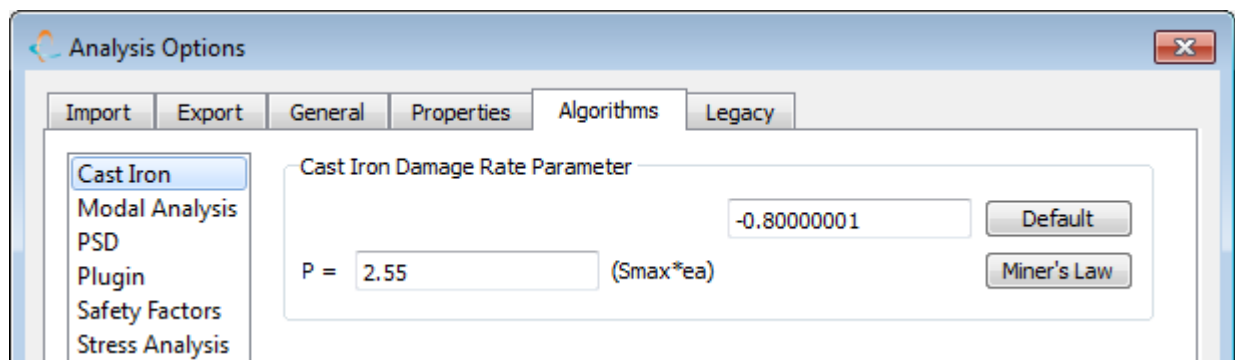


Figure 14.22-1

14.22.4 Mean–stress correction for cast iron analyses

Mean–stress correction can be applied using either Smith Watson Topper or a user–defined mean–stress correction.

Smith Watson Topper (SWT) mean–stress correction

The SWT life curve for cast irons is defined by two parameters in the material database: a slope and an intercept at 1 cycle – (see section 8). These parameters should be determined experimentally.

In the material database, provision is made for a ‘knee’ in the SWT curve, with a second slope **b2** at higher values of endurance. The value of **b2**, and the endurance above which it applies, can be entered if the user finds experimental evidence that a ‘knee’ exists.

Note: This model assumes that totally compressive cycles are non damaging.

14.23 Fatigue analysis of welded joints

Fatigue analysis of welded joints in *fe-safe* is described in section 16.

14.24 Stress–based Brown–Miller analysis

This algorithm allows the Brown–Miller algorithm to be used with stress–life data (S–N curve). See section 14.16 for the equations used with the Brown–Miller algorithm. The life curve is defined as an S–N curve and the code derives the required Brown–Miller parameters from the S–N curve. Morrow, Morrow B, Goodman, Gerber, R ratio SN curves, User Defined or no mean stress correction may be selected.

It is highly recommended to enable the plasticity correction for S–N data in **Analysis Options** dialogue [**FEA Fatigue >> Analysis Options...**], **Stress Analysis** tab (see section 5). If no plasticity correction is performed all nodes with lives beneath 1e6 would probably experience plasticity and hence this algorithm would not be suitable.

14.25 References

- 1 EN Mamiya, JA Araujo, FC Castro (2009). Prismatic Hull – A new measure of shear stress amplitude in multiaxial high cycle fatigue, International Journal of Fatigue 31, 1144–1153
2. FC Castro, JA Araujo, EN Mamiya, N Zouain(2009). Remarks on multiaxial fatigue limit criteria based on prismatic hulls and ellipsoids, International Journal of Fatigue 31, 1875–1881
3. EN Mamiya, FC Castro , JA Araujo (2014) Recent developments on multiaxial fatigue: The contribution of the University of Brasilia, Theoretical and Applied Fracture Mechanics
4. G de Morais Teixeira, J Draper, A Rodrigues et al (2015). Dang–Van, Prismatic Hull and Findley approaches for high cycle fatigue evaluation of powertrain components, NAFEMS conference June 2015.

15 Fatigue analysis of elastic-plastic FEA results

15.1 Performing fatigue analyses from elastic-plastic FEA results in *fe-safe*

By default, *fe-safe* analyses stress datasets that contain elastic stresses. The calculation of elastic-plastic stress-strains, where necessary, is performed in *fe-safe* using an elastic to elastic-plastic correction (using the biaxial “Neuber’s Rule”). This elastic-plastic correction is applied to each node individually, and so it cannot allow for any stress redistribution effects in the FEA model. Where stress redistribution may be significant, it may be necessary to use an elastic-plastic FEA. The plasticity correction (Neuber rule) is then turned off in *fe-safe*.

For an elastic-plastic FEA, *fe-safe* requires that the analysis is a dataset sequence, and that each step in the loading is defined by both a stress and a strain dataset.

Fatigue analysis from elastic-plastic stress-strain pairs is enabled by the use of loading blocks with dataset stress and strain pairs – see section 13.

This type of analysis is supported by the Normal Strain, Brown Miller, Maximum Shear Strain and Cast Iron analysis methods. Factor of Safety and FRF calculations are not supported.

15.2 Importing elastic-plastic stress-strain datasets

To read strain datasets from an FE model, the **Read Strains from FE Model** checkbox must be ticked in the **Analysis Options** dialogue, **Import** tab, or strains must be selected in the **Select Datasets to Read** dialogue, see section 5. Refer to Appendix G for details of how to load strain datasets from specific FEA packages.

For each stress dataset there must be a corresponding strain dataset.

When referencing FE datasets for use in a fatigue analysis care must be exercised when defining dataset numbers to ensure that the defined stress and strain datasets are an elastic-plastic stress-strain pair. For example, a file containing five steps of stress and strain data may be imported. In *fe-safe* the stress data from each step may be listed as datasets 1 to 5, and the strain data from each step may be listed as datasets 6 to 10, so the matching stress-strain pairs would be 1 and 6, 2 and 7, etc..

15.3 Defining elastic-plastic residual stresses

Elastic-plastic residual stresses can be defined as part of the analysis (see section 13).

15.4 FEA hardening models

Kinematic hardening models are used to model cyclic fatigue behaviour. It is important when using elastic-plastic FEA that the FEA analysis is set to the appropriate hardening model. Even when this is done, there seem to be differences between the modelling procedures used in elastic-plastic FEA and those used in fatigue software. This is particularly true for complex stress histories.

15.5 Surface finish effects

A surface finish factor can be applied to a fatigue analysis from elastic-plastic FEA results in the same manner as for results from an elastic analysis.

The theory behind the way in which the software applies the factor to elastic-plastic stress and strain is slightly different to the elastic approach.

For the elastic approach the factor is a straight multiplier on the elastic stresses which are then corrected by the multi-axial Neuber's rule and converted to strains.

For elastic-plastic stresses and strains the strain-life curve is corrected for the given surface finish factor. This degraded strain-life curve is then used in conjunction with any mean-stress correction to evaluate the damage caused by the cycle. For example in *Figure 15.5-1* the degraded strain-life curves for Manten at various values of K_t are shown.

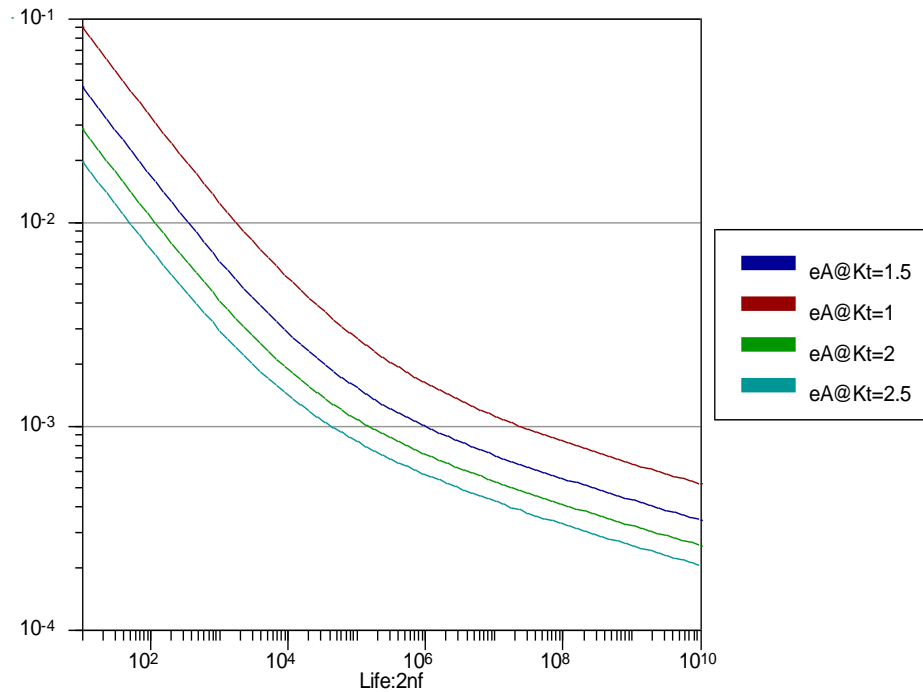


Figure 15.5-1: Strain-life curves degraded by the effect of surface finish factor K_t ,
 $K_t=1$ is the upper curve.

The degraded strain-life curve is calculated at increments on the original strain-life curve (see section 14 for the equation defining the strain-life curve) as follows:

- At a given life (nf) extract the strain amplitude (ea).
- Use the cyclic stress-strain curve to evaluate the associated stress (S) and hence calculate the Neuber's product (np).
- Divide the Neuber's product (np) by the square of the surface finish factor (K_t) to give the effective Neuber's product (np').
- Evaluate the strain amplitude (ea') and the Stress (S') for the applied surface finish factor associated with life (nf) using the cyclic stress-strain curve and the effective Neuber's product (np').

For the Brown-Miller and Maximum Shear Strain algorithms the same ratio of ea/ea' and S/S' are used to correct the algorithms' life curves (see section 14) for the surface finish factor.

Figure 15.5-2 shows an example of the calculated ratios for Manten at a K_t of 1.2. Above lives of $1e10$ there is no plasticity, so K_t is applied as a factor directly to the stresses and strains. As the lives get shorter and the plasticity become more significant, K_t has an increasing effect on the strains and a diminishing effect on the stresses. At lives close to one repeat the effect on the strain has increased to 1.36 and that on the stresses has reduced to 1.06.

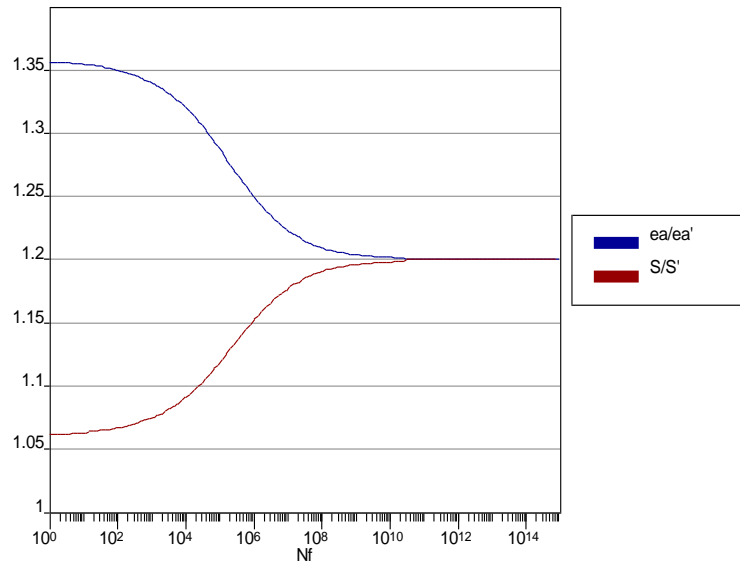


Figure 15.5-2 : Effect of K_t on stress (lower curve) and strain (upper curve).

For a particular analysis diagnostics can be exported displaying the original life curves, modified life curves and the relationship between the two. See section 15.7 for more information.

15.6 Mean-stress effects

The algorithms that support analysis of elastic-plastic FEA results provide three types of mean-stress correction. They are the Morrow, Walker and the User-defined (see section 14) mean-stress corrections.

The Morrow correction has the effect of dragging down the strain-life curve as a function of the mean stress. The elastic-plastic analysis deals with this by building tables of the effect of a unit mean stress on the strain amplitude at each life. This is used to evaluate the life for a given strain amplitude and mean stress. Figure 15.6-1 shows this table as a plot, the y-axis shows the effective reduction in the strain-life curve for each tensile MPa of mean stress.

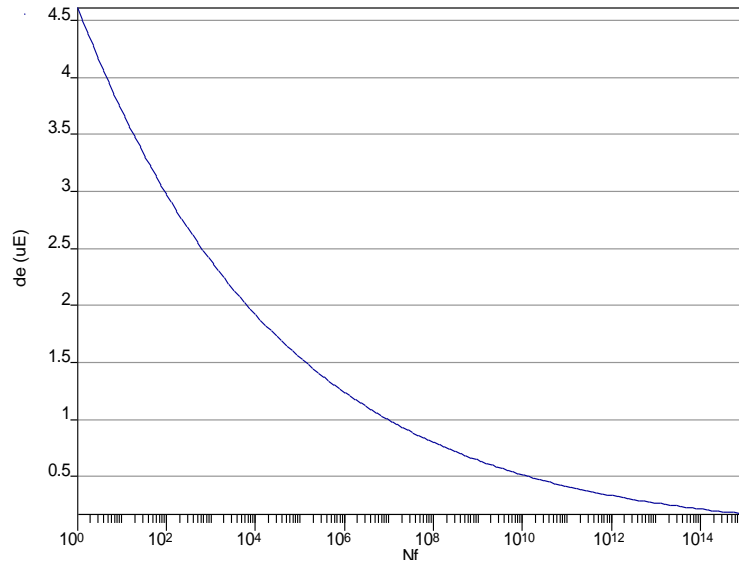


Figure 15.6-1 : Effect of Morrow correction on strain-life curve.

This table can be exported using the diagnostics tools. See section 15.7 for more information.

The User-defined mean-stress correction modifies the strain amplitude by a factor extracted from the User-defined mean-stress curve. This is simulated in the elastic-plastic analysis by iterating until the stress factor for the K_t ; the correction to the strain amplitude for the mean stress and the strain amplitude stabilise for the evaluated life.

15.7 Diagnostics

Two sets of diagnostics specific to elastic-plastic analysis with a surface finish effect are provided. Each is controlled from the **Exports and Outputs** dialogue. This dialogue is obtained by selecting **Exports ...** from the **Fatigue from FEA** dialogue.

Selecting the **Export material diagnostics?** checkbox will turn both sets of diagnostics on, the diagnostics apply to the items (nodes or elements) specified in the **List of Items** text field (see section 22 for a more in depth description of this field).

The first diagnostics are written to the .log file (See section 22.3.2 for more information). For each diagnosed node a table is written as below.

DIAGNOSTICS OF KT MODIFIED LIFE CURVES FOR E/P ANALYSIS

```
Temperature :      0.00
Kt           :      1.20
CAEL amp.    :     141.48
Algorithm    : NormalStrain
```

NOTES: Morrow column show $1e6 * (SNf)^b$ with *bm* or *ms* correction (It may not be used)
 S scaler column indicates how stress @ *Kt=1* and actual *Kt* compare

Nf	ea@Kt=1	ea@Kt	Ratio	Morrow	S scaler
1.00e+000	192000	141549	1.356	4.612	1.062
1.31e+000	169368	124868	1.356	4.495	1.062
1.72e+000	149440	110205	1.356	4.380	1.062
2.26e+000	131894	97291	1.356	4.268	1.062
2.97e+000	116443	85878	1.356	4.159	1.062
3.90e+000	102837	75866	1.356	4.053	1.062
5.11e+000	90854	67045	1.355	3.950	1.063
6.71e+000	80299	59274	1.355	3.849	1.063
8.81e+000	71003	52406	1.355	3.751	1.063
1.16e+001	62813	46380	1.354	3.655	1.063
1.52e+001	55597	41070	1.354	3.562	1.064
1.99e+001	49239	36377	1.354	3.471	1.064
2.61e+001	43636	32246	1.353	3.383	1.064
..... (section removed)					
1.14e+014	198	165	1.200	0.213	1.200
1.49e+014	193	161	1.200	0.208	1.200
1.96e+014	188	157	1.200	0.202	1.200
2.57e+014	183	153	1.200	0.197	1.200
3.37e+014	179	149	1.200	0.192	1.200
4.42e+014	174	145	1.200	0.187	1.200
5.80e+014	170	141	1.200	0.183	1.200
7.62e+014	165	138	1.200	0.178	1.200
1.00e+015	161	134	1.200	0.173	1.200

The columns in the table contain:

Nf	Life in repeats.
ea@Kt=1 (ea)	The strain amplitude for the given life evaluated from the life equations. (See figure 15.6-1)
ea@kt (ea')	The degraded strain amplitude for the specified Kt and life. (See figure 15.6-1)
Ratio	The strain ratio ea/ea'. (See figure 15.6-2)
Morrow	The effective reduction in the strain-life curve (in uE) for each tensile MPa of mean stress at the specified life. (See figure 15.6-3)
S scaler	The stress ratio (S/S') for the given life. (See figure 15.6-2)

The second sets of diagnostics are the plottable files (see section 22 for more information). For each diagnostics node a plot file is created. If the plot file is opened for a particular node after the analysis is completed (using the **File >> Data Files >> Open Data File ...** option) it will contain 3 data channels as shown in *Figure 15.7-1*.

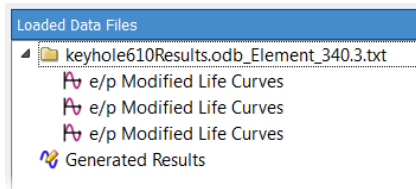


Figure 15.7-1

In this case the diagnostics file was from element 340 node 3. The first channel contains Life information and the second and third channels contain strain amplitude information for the original and degraded strain-life curves. The Life and ea channels can be cross-plotted to create a strain-life curve plot as in *Figure 15.7-2*.

Note: If a Kt of 1 is specified no diagnostics will be displayed for a node.

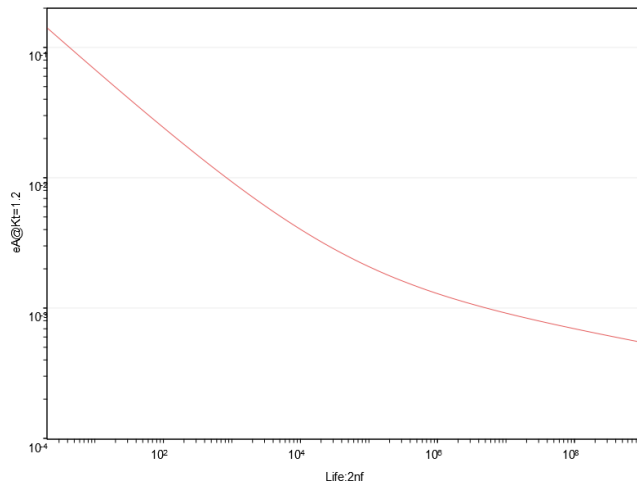


Figure 15.7-2

16 Fatigue analysis of welded steel joints

This chapter outlines the BS5400/BS7608 technique for analysing welds in *fe-safe*. The Verity® technique licensed from the Battelle Memorial Institute can also be used for weldments, it is described in a separate document.

16.1 BS5400/BS7608 analysis for welded joints

This calculation is a critical plane analysis using, at each node, stresses resolved onto planes perpendicular to the surface of the model. The plane with the shortest calculated fatigue life defines the life at the node. For this module the S-N curves are predefined, and are the stress-life relationships defined in BS5400 part10:1980 for welded joints. These curves apply to welds in structural steels.

These relationships are shown below:

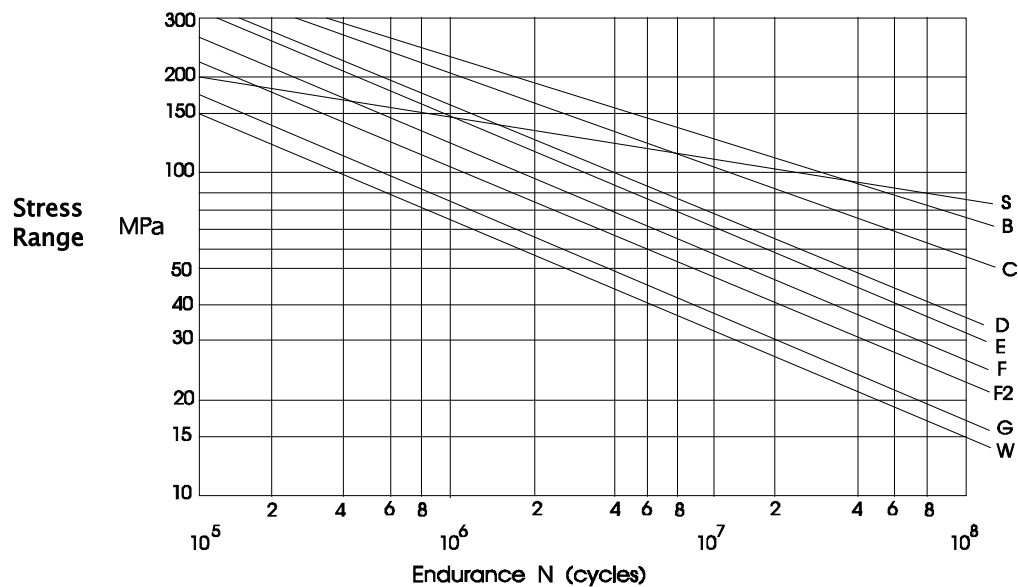


Figure 16.1-1: S-N curves for welded joints

See the Fatigue Theory Reference manual for a discussion of the fatigue analysis of welded joints.

The curves have a constant slope between 10^5 and 10^7 cycles, where the stress-life relationship is defined by the equation (for the mean life):

$$N = \frac{K_0}{S^m}$$

where

N is the endurance in cycles;

S is the nominal stress range;

K_0 is the constant for a particular weld classification;

m is the slope of the S-N curve on log-log axes. For most curves, m has a value of 3, from the Paris crack growth law.

The curve between 10^5 and 10^7 cycles is defined from experimental test data. The curves were extended for longer lives using theoretical calculation. The life to crack initiation for welded joints is a small part of the total life, as most welded joints contain cracks or crack-like defects produced during manufacture. The life is therefore dominated by the propagation of these cracks. Although the defect may initially be small and therefore not affected by small cycles, the larger cycles present in the applied loading may propagate the defect, and as the defect size increases it will be propagated by smaller cycles. The concept of an endurance limit therefore is not appropriate.

The result is that if all the cycles fall below the stress level for 10^7 cycles, the stress history can be considered non-damaging. If larger cycles exist, all cycles must be considered, and the S-N curve is extended indefinitely, with the value of m increased to $(m + 2)$. For very large stress ranges, the curve is extended back at the slope of m until static strength limitations apply.

NOTE: In *fe-safe*, for $N > 10^7$ cycles the value of m is increased to $(m + 2)$ creating a ‘flatter’ curve. For $N < 10^5$ cycles the curves are linearly extrapolated (in log-log terms) back to 1 cycle.

16.2 Operation

The dialogue box is displayed by double-clicking **Algorithm** in the **Fatigue from FEA** dialogue box, and then selecting **BS5400 Weld Life (CP)**.

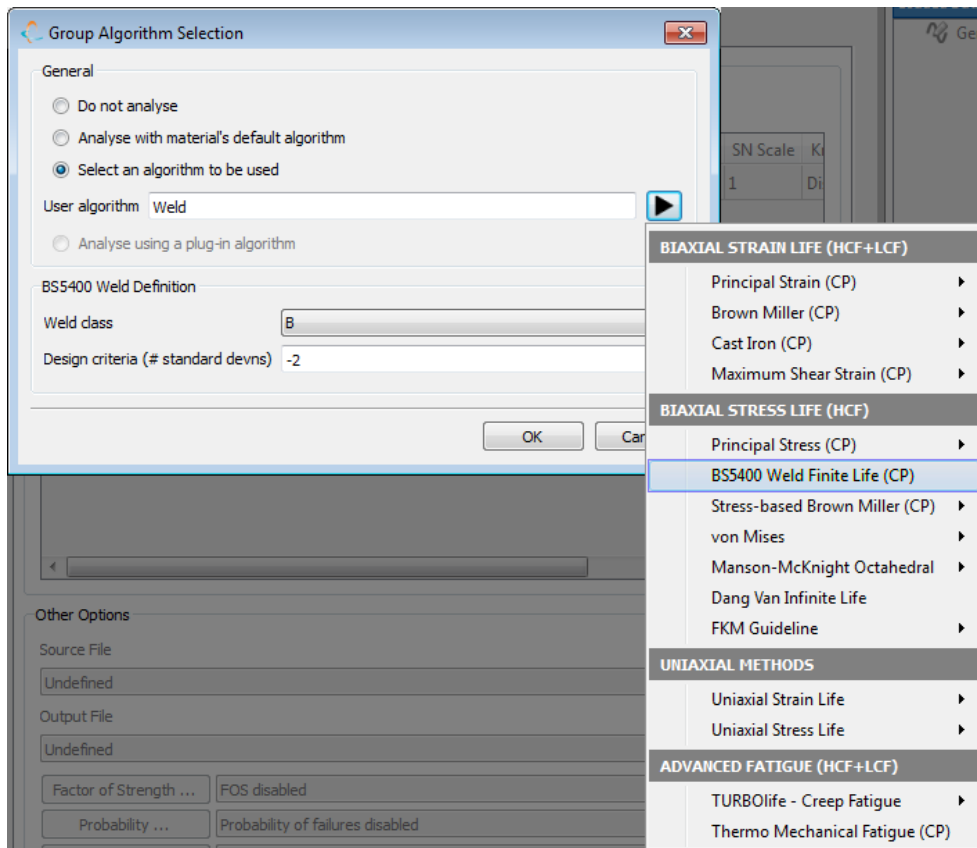


Figure 16.2-1

The user must define the weld class. This defines the S-N curve to be used for the analysis of the model, or the element group. The S-N curves are shown in *Figure 16.1-1*. The user should be familiar with the weld classification selection procedure discussed in BS5400/BS7608 available from BSI.

Note that a different weld class can be defined for each element group.

The user must also select the design criteria. This parameter defines the probability of failure, in terms of the number of standard deviations below the mean life. A value of zero produces a mean life (50% probability) calculation. Example design criteria are:

Design criteria	Probability of failure before the calculated life (%)
0	50
-2	2.3
-3	0.14

16.3 Practical issues relating to the analysis of welded joints

16.3.1 FEA modelling

The S–N curves defined in BS5400/BS7608 are defined in terms of the nominal stress a small distance from the weld or weld toe. The stress concentration effect of the weld geometry is normally included in the S–N curve. This means that

- (i) stresses at the weld toe may give unrealistically low fatigue life results
- (ii) the weld geometry need not normally be modelled in detail

It is usual to define a named group of elements close to the weld toe, and assign the appropriate weld class to this group of elements.

16.3.2 Multi-axial stresses

The treatment of multi-axial stresses as defined in BS5400/BS7608 has not been followed in *fe-safe*. BS5400/BS7608 suggests the stress used in the analysis should be whichever of the two in-plane principal stresses lies within $\pm 45^\circ$ of an axis perpendicular to the weld toe. Instead, *fe-safe* uses a full critical plane analysis.

16.3.3 Welds not covered by the weld classification procedure.

The welded joint analysis in *fe-safe* uses a principal stress calculation, with no mean stress correction. This is selected automatically. For welds not covered by the weld classification procedure, the user may insert an appropriate S–N curve in to the material database, and select a Normal Stress analysis. In this case, mean–stress corrections may be used if required.

17 Factor of strength and probability-based fatigue methods

17.1 Target life

There are three types of target life analyses that can be performed:

- Factor of Strength (FOS) calculation can be performed for any analysis other than the FRF calculations.
- A Fatigue Reserve Factor (FRF) analysis can be performed instead of a fatigue life analysis for certain Biaxial Stress Life or Biaxial Strain Life analyses.
- The Failure Rate for Target Lives calculation can be performed for any analysis other than the FRF calculations.

17.2 Factors of Strength (FOS)

This analysis can be selected when the **Factor of Strength Calculations** dialogue is opened by clicking on the **Factor of Strength...** button in the **Fatigue from FEA** dialogue.

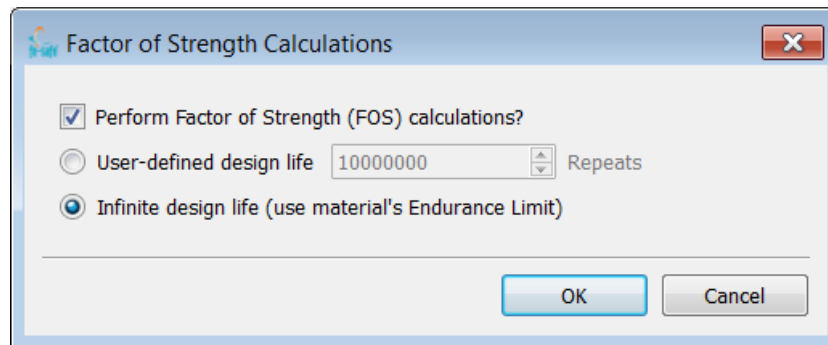


Figure 17.2-1 Factor of Strength Calculations dialogue

To enable FOS calculations check the **Perform Factor of Strength (FOS) Calculations** box. This will enable the target life field to be set. The target life can be a finite life specified in the chosen life units, or 'infinite' life based on the endurance limit for the material

The factor of strength (FOS) is the factor which, when applied to either the loading, or to the elastic stresses in the finite element model, will produce the required target life at the node. The FOS is calculated at each node, and the results written as an additional value to the output file. The FOS values can be plotted as contour plots.

The limits of the FOS values can be configured in the **Band Definitions for FOS Calculations** region of the **Analysis Options** dialogue, **Safety Factors** tab.

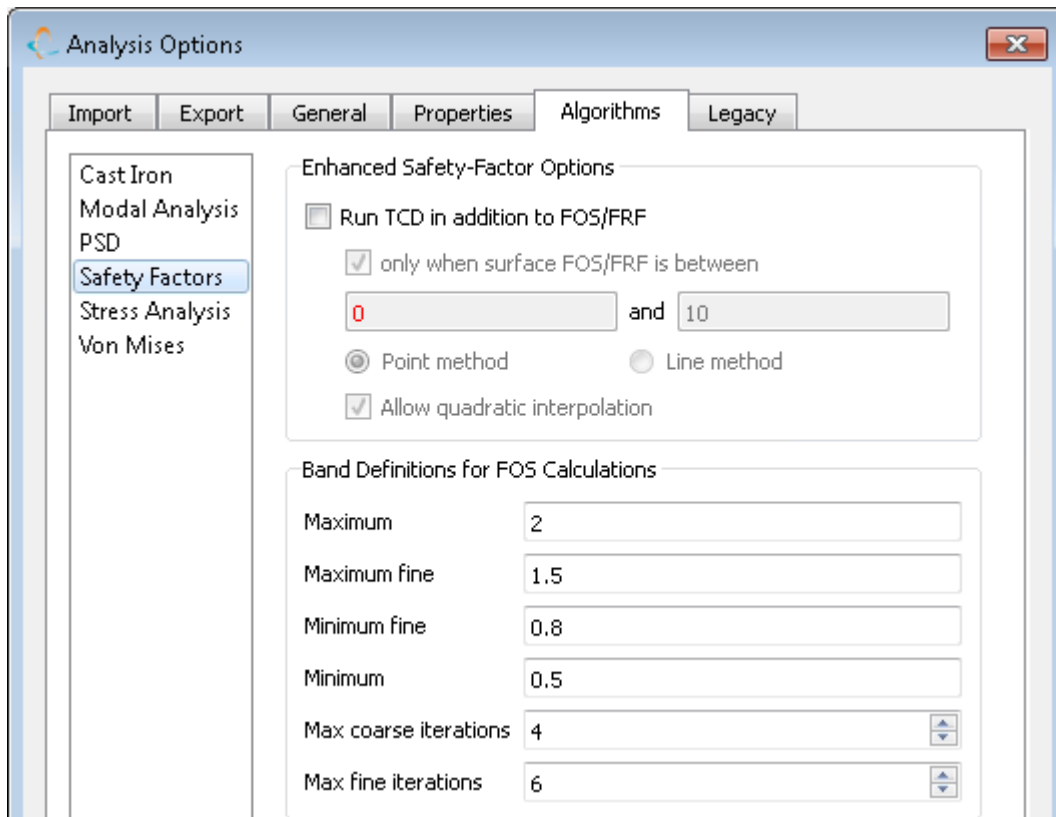


Figure 17.2–2 FOS band definitions

The default limit values are:

Max factor of strength	2.0	all FOS values higher than this will be written as 2.0
Max fine factor of strength	1.5	
Min fine factor of strength	0.8	
Min factor of strength	0.5	all FOS values lower than this will be written as 0.5

The default limits on the number of iterations are:

Max coarse iterations	4
Max fine iterations	6

These both have a minimum value of 1, but no maximum value limit, as in practice the process would find its own natural limit.

The FOS at a node is calculated as follows:

- The calculated life is compared with the target life.
- If the calculated life is lower than the target life, the elastic stresses at the node are scaled by a factor less than 1.0. If the calculated life is greater than the target life, the elastic stresses at the node are scaled by a factor greater than 1.0.
- The elastic stress history is recalculated using the re-scaled nodal stresses.
- For local strain analysis, the cyclic plasticity model is used to recalculate the time history of elastic-plastic stress-strains. The fatigue life is then recalculated.
- For S-N curve analysis, the fatigue life is recalculated from the time history of elastic stresses.

- In the critical plane analysis, the critical plane orientation is re-calculated (see note below).
- The process is repeated with different scale factors until
 - (i) the calculated life is within 5% of the target life or
 - (ii) the FOS exceeds the max factor (default 2.0) or is less than the min factor (default 0.5)
 - (iii) the maximum number of iterations is reached, at which point a message will be displayed with the number of items for which this occurred.

This procedure applies to all analyses, except stress-based analysis using the Buch mean stress correction.

Note: The critical plane is recalculated for each new factor at the node. If a constant critical plane is assumed, the FOS may be unrealistically high. For example, application of the FOS to the mean stress on another plane may cause this stress to exceed the material tensile strength. To avoid this type of problem, the critical plane is constantly recalculated.

17.2.1 Modification of Factor of Strength (FOS) Calculation when using Buch Mean Stress Correction

When a Factor of Strength (FOS) analysis is performed using Buch Mean Stress Correction, the FOS is modified as described below. This analysis is effectively a hybrid of a FOS calculation and an FRF calculation.

FOS values are calculated using both the Goodman and Buch mean stress corrections. The Goodman calculation follows the procedure described above, i.e. the stress history is repeatedly re-scaled and the life recalculated.

The FOS value may also be calculated from the Buch diagram. Referring to *Figure 17.2-3*, the FOS is the ratio $\frac{A}{B}$. For variable amplitude stress histories, the value of the FOS is calculated for the cycle that gives the lowest value of this ratio.

The lowest value of the FOS from the Goodman and Buch calculations is written to the output file.

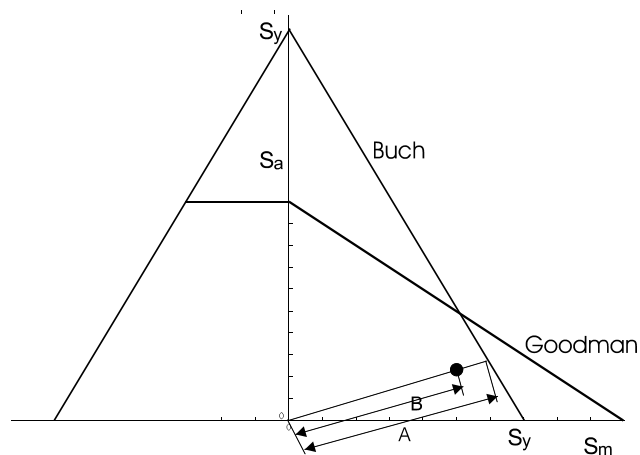


Figure 17.2-3 Buch Factor of Strength.

17.3 Fatigue Reserve Factor (FRF) Analysis

The Fatigue Reserve Factor (sometimes referred to as the Fatigue Reliability Factor) is a linear scale factor obtained from a Goodman-type diagram. This method is inferior to the FOS calculation (see 17.2) because of the technical limitations discussed in section 17.5.

The FRF analysis allows the user to specify an envelope of infinite life for the component as a function of stress/strain cycle amplitude and mean stress (this is similar to a Goodman/Haigh diagram) or as a Smith type diagram. The Smith diagram is internally converted to a Haigh diagram prior to the analysis.

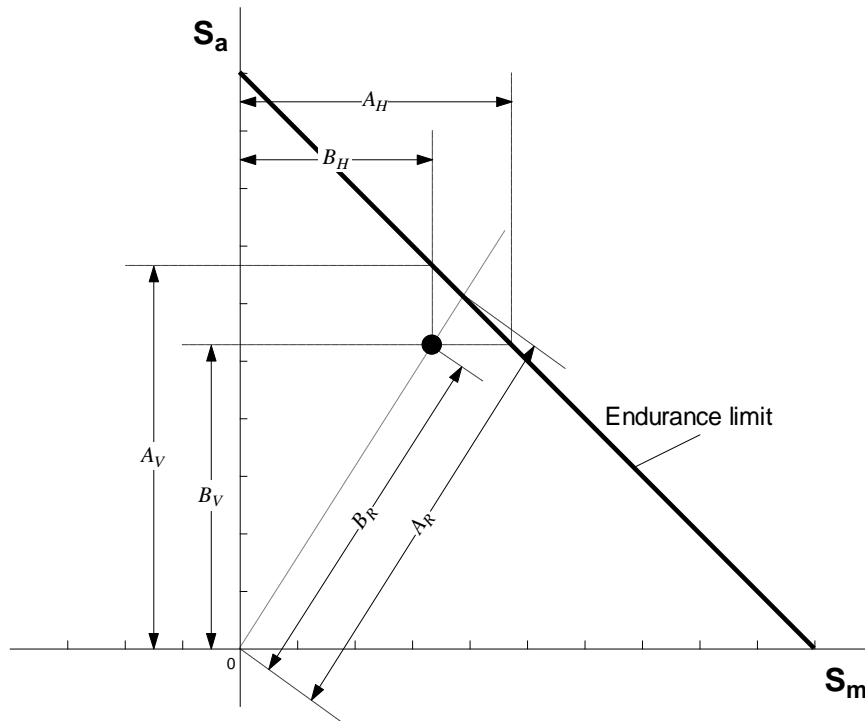


Figure 17.3-1

The ratio of the distance to the infinite life line and the distance to the cycle (Sa, Sm) is calculated for each extracted cycle, to produce four reserve factors, as follows:

Horizontal FRF : $FRF_H = \frac{A_H}{B_H}$

Vertically FRF : $FRF_V = \frac{A_V}{B_V}$

Radial FRF : $FRF_R = \frac{A_R}{B_R}$

Worst FRF : *Worst of above 3 factors.*

Stress or strain amplitudes are used, depending on the analysis selected.

The following rules are followed when calculating Horizontal FRF in fe-safe

1. The Worst Horizontal FRF is taken to be the lowest value from any of the extracted cycles, including negative values.
2. When the mean stress is to the left of the reference origin axis, fe-safe uses the first line segment with
 - a) a point to the left of the origin
 - b) a positive gradient
 - c) amplitudes that bound the cycles amplitude.
3. When the mean stress is to the right of the reference origin axis, fe-safe uses the first line segment with
 - a) a point to the right of the origin
 - b) a negative gradient
 - c) amplitudes that bound the cycles amplitude.

The FRF infinite life curve is defined using the same format rules as the user defined MSC, (see Appendix E). To convert the factors in the envelope to amplitudes, multiply the factors by the amplitude that would cause failure at the target life. The target life is specified in the **Factor of Strength** dialogue when an analysis using the FRF option is selected. The target life is substituted into the life equation for the analysis type to calculate the amplitude that would cause failure at that target life.

At each node, the worst-case reserve factor is calculated, for each of the four FRF types (horizontal, vertical, radial and the worst of the 3). The limitations of this analysis are discussed in section 17.5.

The analysis is selected from the drop-down menu associated with the user-defined algorithm in the **Group Algorithm Selection** dialogue box.

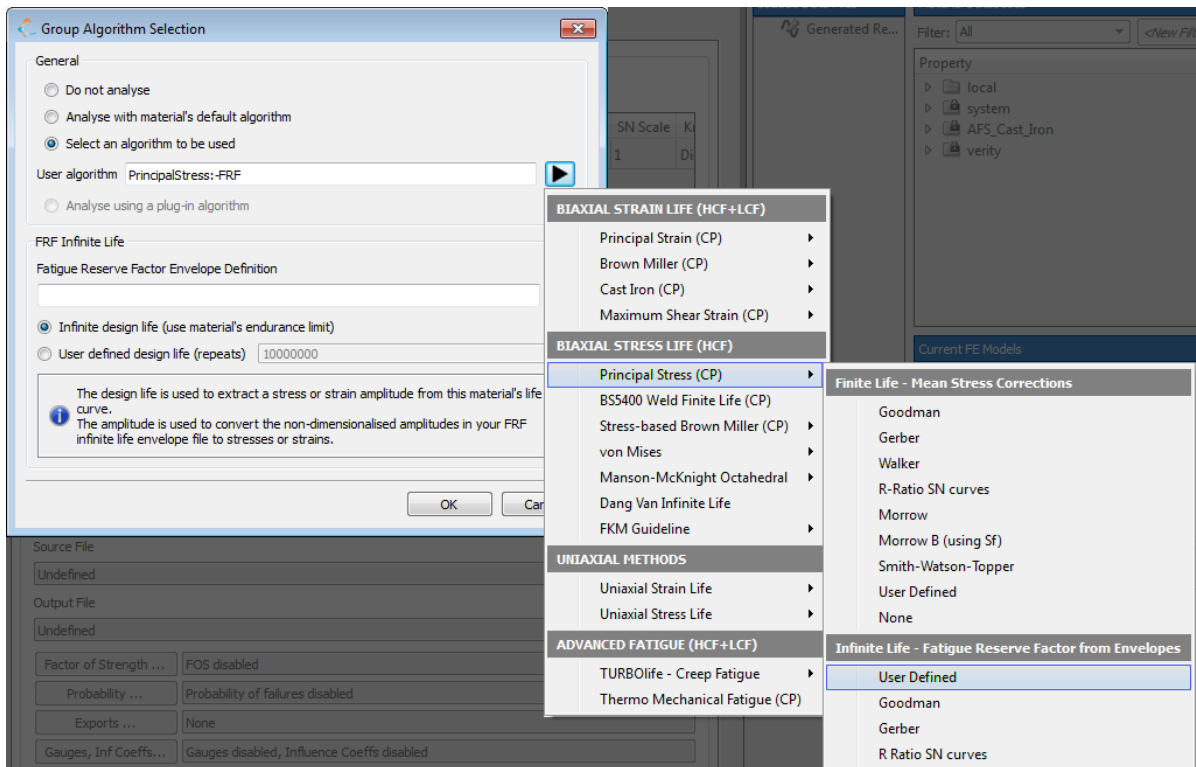


Figure 17.3-2 Algorithm selection dialogue

17.3.1 Residual Stresses

Where an analysis includes a residual stress the point (*residual stress*, 0) rather than the (0,0) is used as the origin for the factor calculations. The use of (0,0) is shown in *Figure 17.3-1*. The residual can either be defined in the **Group Parameters** grid or as an elastic-plastic pair in the load definition file. This residual will have no effect on the vertical factor.

17.4 Failure Rate for Target Lives

This analysis combines variability in the material fatigue strength and variability in the applied loading, to calculate a probability of failure for the life or lives specified. Statistical methods are described in the Fatigue Theory Reference Manual.

The analysis is configured in the **Failure Rate for Target Lives** dialogue, which is opened by clicking on the **Probability...** button in the **Fatigue from FEA** dialogue.

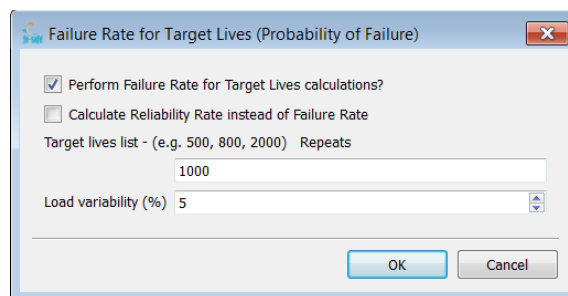


Figure 17.4-1 Probability of failure dialogue

To enable Failure Rate calculations check the box marked **Perform Failure Rate for Target Lives Calculations**.

The failure rate for target lives calculates the % probability of failure at the specified lives (user-defined life units).

For each of the list of target lives a contour plot will be created indicating the % probability of failure at that life. This percentage can either be the % of components that will fail (Failure Rate) or the % that will survive (Reliability Rate) depending upon whether or not the check box **Calculate Reliability Rate instead of Failure Rate** is checked.

The failure rates are calculated as follows:

- (i) The assumption is made that for failure rate analysis to be useful the component must fail in the elastic area of the strain-life curve.
- (ii) A normal or Gaussian distribution is applied to the variation in loading. The % standard deviation of loading is defined, representing the variability of the value of the load amplitude relative to the amplitude defined. For non-constant amplitude loading the code derives an equivalent constant amplitude loading.
- (iii) A Weibull distribution is applied to the material strength. This is defined by three parameters:
 - *The Weibull mean:*
This is the strength at which the life curve exceeds the target life. This value is derived from the material data and the specified target life. The Weibull distribution is centred on this value.
 - *The Weibull slope, Bf:*
This is a shape parameter that varies the probability density.

The value of **Bf** is defined in the material database using the **weibull : Slope BF** parameter, (see section 8).

Examples of the effect of Bf on the shape of the distribution are shown in *Figure 17.4-2*.

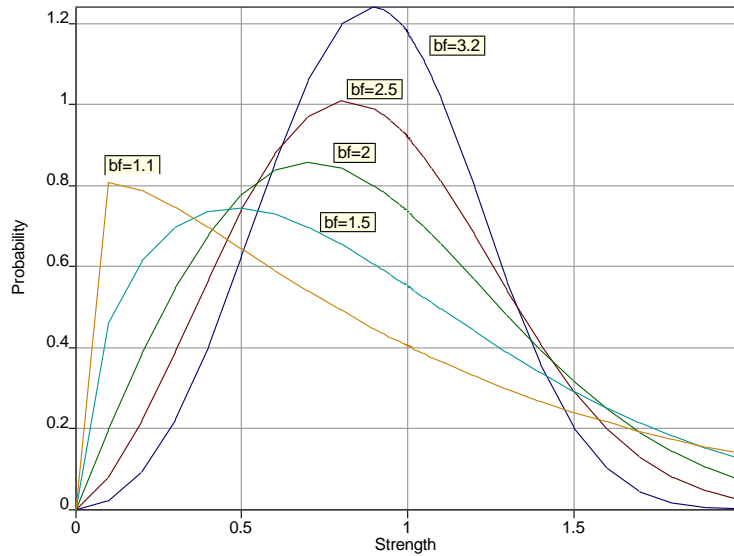


Figure 17.4-2 Variation in probability density for different values of Bf.

○ *The Weibull minimum parameter, Qmuf:*

This minimum parameter parameter determines the width of the distribution:

- as the lower edge of the distribution tends towards zero amplitude, **Qmuf** tends towards zero;
- as the distribution gets narrower, **Qmuf** tends towards one.

For convenience, the minimum parameter is expressed as a ratio of the fatigue strength (i.e. it is normalised by dividing it by the mean strength at the target life).

The value of **Qmuf** is defined in the material database using the **weibull : Min QMUF** parameter, (see section 8).

- (iv) The overlap area of the normal distribution of loading and the Weibull distribution of fatigue strength is calculated for each of the target lives. This represents the probability of failure, as illustrated in *Figure 17.4-3*, below.

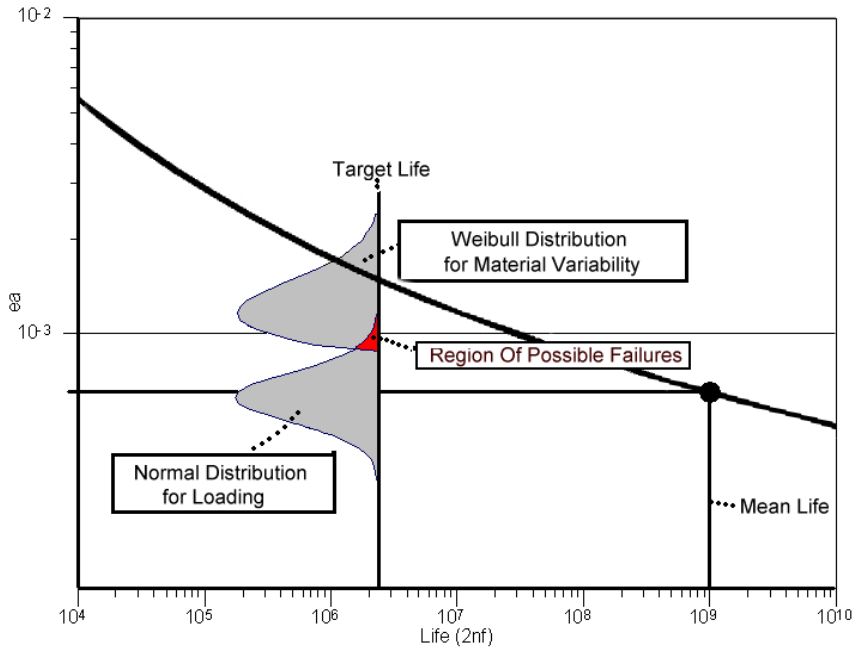


Figure 17.4-3

Note that in *Figure 17.4-3*, for illustrative purposes, the two distributions are plotted on a linear scale, whilst the strain axis is shown plotted on a logarithmic scale.

Figure 17.4-4 illustrates the effect of varying Q_{muf} on the probability of failure (at lives of $1e6$, $1e7$ and $1e8$), for a component with a life of $1e7$.

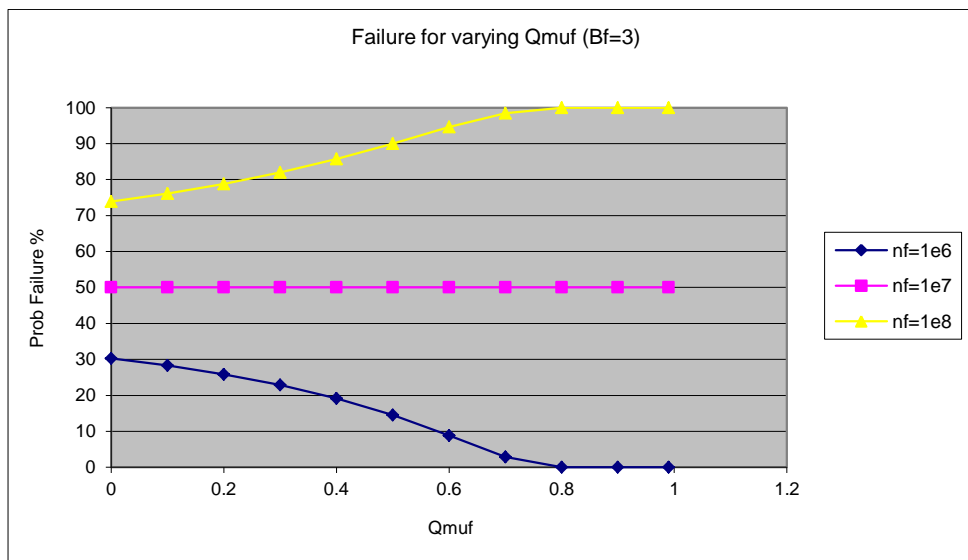


Figure 17.4-4

Using S-N curves.

From version 5.1-01 the use of S-N curve material data can also be used in the probability analysis calculations. The slope of the SN curve at a life of $1e7$ is used in the calculation of the probability of failures. As for local strain data only finite life algorithms can calculate probabilities of failure.

17.5 Technical Discussion of FOS versus FRF

The traditional method of calculating factors (FRFs) is to use an infinite life envelope, usually a Goodman or Gerber diagram. The method, for endurance limit ('infinite life') calculations is as follows.

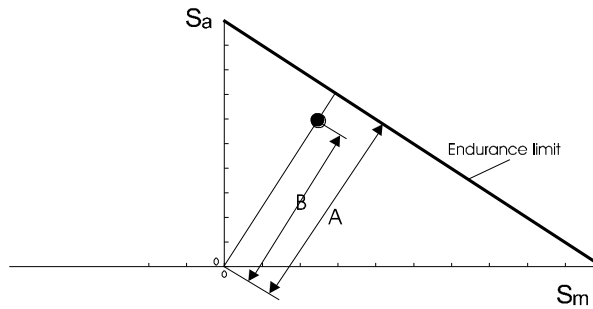


Figure 17.5-1

The most severe cycle, i.e. the one that comes closest to the Goodman line, is plotted on the Goodman diagram. A line is drawn through this point (either vertically, or from the origin). This indicates how much the stress could be increased before it touches the Goodman line. If any cycle crosses the Goodman line the component would not have an infinite life. As all the other cycles in the signal are smaller, they will still be below the endurance limit and contribute no damage. Therefore, the ratio A/B (shown in *Figure 17.5-1*) indicates the factor of strength.

When designing for finite life, the same method cannot be used (except for constant amplitude loading). Consider the case below in *Figure 17.5-2*, where there is 1 occurrence of the largest cycle, and (say) 100 occurrences of the next smallest cycle, shown grey. The target life is (say) 10^5 repeats of the signal.

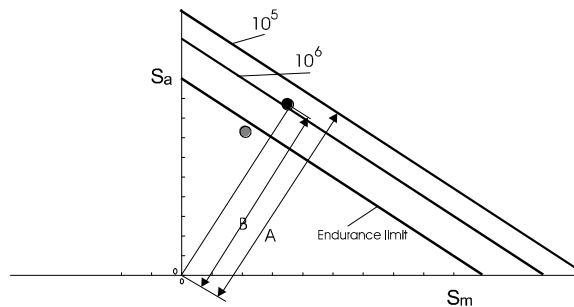


Figure 17.5-2

Under the applied loading, the smaller (grey) cycles would be assumed to be non-damaging. The Goodman analysis would then use the ratio A/B to estimate the factor of strength (FRF). However, scaling the applied loading by this FRF would now make the smaller cycles damaging. As there are many more of these, the FRF would be greatly overestimated, and the analysis would be unsafe.

The same limitations apply to the use of Gerber diagrams to calculate FRF's.

For these reasons, it is strongly recommended that Factors of Strength (FOS) are calculated, instead of FRF's. FOS values are calculated as described in section 17.2, and summarised below:

For a FOS calculation, *fe-safe* calculates the fatigue life. It then applies a scale factor to the elastic stresses in the stress history, and re-calculates the plasticity. The fatigue life is re-calculated. This process is repeated until a scale factor is found which, when applied to the stresses, gives a calculated life equal to the target life. This scale factor is the FOS.

Design life fatigue methods

The FOS analysis is the method recommended in *fe-safe* because it is equally applicable to both complex loading and constant amplitude loading, and to both finite and infinite life design.

Note: The comparison between FRFs calculated using the Goodman technique and the more rigorous *fe-safe* FOS method will only agree for infinite life design, and only for constant amplitude loading. For other cases the results will not agree, for the reasons outlined above. Note also that *fe-safe* reduces the endurance limit when the largest cycle in the stress history becomes damaging.

18 Conventional high temperature fatigue

18.1 High temperature fatigue

There are many instances in engineering design where a component experiences high temperatures which alter the fatigue strength of the material; *fe-safe* can analyse temperature-dependant fatigue.

The high temperature analysis may be used for conventional metallic materials and for cast irons including grey iron.

18.2 Time-dependant phenomena

Conventional high temperature fatigue analysis in *fe-safe* should be used when temperatures are not high enough or times not long enough for time-dependant phenomena or creep fatigue to occur. If creep fatigue needs to be considered, the module *fe-safe/TURBOLife* (creep fatigue) is required – see the separate *fe-safe/TURBOLife* User Manual.

18.3 High temperature fatigue inputs

For a high-temperature fatigue analysis, the following inputs are required:

- an FEA solution for the temperature, or a temperature history, at each node;
- the FEA stresses;
- a description of the materials properties at a number of temperatures.

18.3.1 Temperature datasets

To read temperature datasets into *fe-safe* the FEA solution must contain temperature or temperature-time information at each node. *fe-safe* reads the temperature information from the FEA model. The temperature data need not be in the same model file as the stress solutions.

Before reading in the FEA results file(s), select **FEA Fatigue >> Analysis Options**, General tab, and ensure that the **Disable temperature-based analysis** box is *unchecked*.

If the FEA temperature data is in a separate FEA results file from the stresses, use the **File >> FEA Solutions >> Append Finite Element Model** to append the second and any subsequent FEA results files.

18.3.2 Material data

Fatigue life data (strain-life or S-N) and cyclic stress-strain data may both be adjusted for temperature. Entering fatigue properties for multiple temperatures and interpolation of data for temperature-based analysis is discussed in detail in section 8.

18.3.3 Loading

The loading may consist of

- elastic FEA 'unit loads' stresses with time histories of loading: 'scale and combine'.
- elastic or elastic-plastic FEA stresses as a data set sequence

In both cases the loading is added using the methods outlined in section 13.

The definition of fatigue loading for varying temperature, as discussed in section 13, is not required for conventional high temperature fatigue.

Note: In the conventional high temperature fatigue analysis described here, at each node a single adjustment is made, to the maximum temperature at that node.

18.3.4 Conventional high temperature application in *fe-safe*

At each node, and for each loading block *fe-safe* determines the temperature at the node. Each node may use different material data, appropriate to the temperature at the node. If the material has been defined at multiple temperatures, *fe-safe* interpolates the materials data to apply to this temperature and calculates the fatigue life. Interpolation of data for temperature-based analysis is discussed in section 8.

- When temperature datasets are not opened in *fe-safe* the following is applied:
 - If temperature is not set in the loading block *fe-safe* assumes a default temperature of 0°C.
 - If temperature is set in the loading block *fe-safe* applies the block temperature.
 - For multiple-block loading the transitions block (if enabled) will use the maximum temperature from all blocks in the loading definition.
- When temperature datasets are read from the source model the following is applied:
 - If temperature is not set in the loading block *fe-safe* assumes the worst case scenario, where the maximum temperature from all temperature datasets open in *fe-safe* model is determined and applied.
 - If temperature is set in the loading block *fe-safe* applies the block temperature.
 - For multiple-block loading the transitions block (if enabled) will use the maximum temperature from all blocks in the loading definition.

18.4 Analysis

The analysis proceeds as a normal *fe-safe* analysis.

Conventional high temperature fatigue will not be carried out if the option on the **FEA Fatigue >> Analysis Options** dialogue, **General** tab, entitled **Disable temperature-based analysis** box is *checked*.

19 Fatigue analysis from frequency domain loading

19.1 Introduction

fe-safe can analyse loading defined by a Power Spectral Density diagram (PSD). The PSD is a description of the loading in the frequency domain. See the Signal Processing Reference Manual for the theoretical background to the PSD.

The analysis assumes that although the loading has been defined in the frequency domain, the component is 'rigid', i.e. the stresses in the component are linearly related to the magnitude of the applied load. The analysis applies to a single PSD of loading.

fe-safe transforms the PSD into a Rainflow cycle histogram. The method generates cycle ranges, but does not generate cycle mean values. All cycles are therefore at zero mean. Fatigue analysis from a cycle histogram is faster than the analysis of the load history from which it was obtained, although this difference may only be noticeable on larger FEA models. Because the sequence of events is not retained in the cycle histogram, a strain-life analysis will be less precise (see the Fatigue Theory Reference Manual for a description of strain-life analysis from cycle histograms). Also, the transformation of the PSD into Rainflow cycles generates cycle ranges but does not generate cycle mean values. For this reason the method is most suited to the analysis of welded joints where the effects of mean stress are not significant.

19.2 Background

fe-safe transforms the PSD into a Rainflow cycle histogram. The method generates cycle ranges, but does not generate cycle mean values. All cycles are therefore at zero mean.

The Rainflow cycle histogram is re-formatted as an LDF file. The *fe-safe* analysis then proceeds as for any other LDF file (see section 13 for a description of the LDF file format).

The Fatigue Theory Reference Manual describes the theoretical background to the use of PSD's to define fatigue loading, and gives the method for transforming a PSD into Rainflow cycles. The method was derived for loading which is a Gaussian process, and which is stationary (i.e. its statistical properties do not vary with time). See the Signal Processing Reference Manual for a description of Gaussian processes. The method has been shown to be quite tolerant, in that acceptable fatigue lives can often be obtained for processes which are not strictly Gaussian and not stationary. However, the user should always validate the analysis. The validation method is described in section 19.4.

19.3 Operation

The PSD must be in one of the file formats supported by *fe-safe*, and must consist of values of $(\text{load})^2/\text{Hz}$, at equal intervals of frequency (Hz), with the first value at zero Hz. The interval between frequency values must be defined.

The PSD can be plotted and listed (see section 7).

19.3.1 Transforming the PSD to a Rainflow cycle histogram

The PSD is transformed into a Rainflow cycle histogram, using the **Amplitude >> Rainflow Histogram from PSD** menu option.

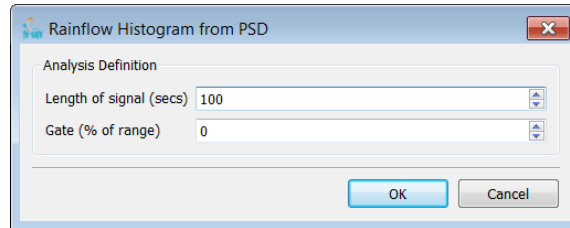


Figure 19.3-1

The dialogue requests that the user define the time (in seconds) to be represented by the cycle histogram. The output file is a range-mean cycle histogram (.cyh).

19.3.2 Converting the Rainflow cycle histogram to a loading definition (LDF) file

The cycle histogram is transformed into an LDF file using the **Amplitude >> Convert Rainflow to LDF for FEA Fatigue** menu option.

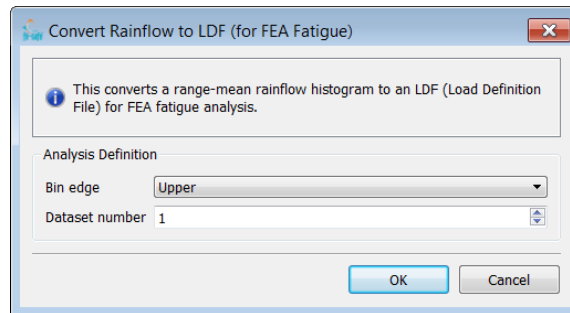


Figure 19.3-2

The user may define whether to take the upper edge of each range bin as the load range, or use centre of each range bin, and must enter the number of the FEA stress data set to be analysed.

The LDF file-name is auto-generated, with extension .ldf. The user may wish to shorten or change the filename.

The file is self-documented, and contains one block for each non-zero bin in the histogram. An example, showing the header and the first three blocks, is given below.

```
# Autogenerated from Range Mean Rainflow Histogram
# Source Histogram : Rainflow_RM_[FButtBP_[sgn_[ ]_created[09.09.35_20-12-02].dac]...
# Bin Edge          : Upper

# Range=12.2773 Mean=-103.215 Bins (1, 25)
BLOCK n=1
ds=1, scale=-109.353
ds=1, scale=-97.0762
END

# Range=12.2773 Mean=-90.9376 Bins (1, 26)
BLOCK n=1
ds=1, scale=-97.0762
ds=1, scale=-84.799
END

# Range=12.2773 Mean=-78.6603 Bins (1, 27)
BLOCK n=2
ds=1, scale=-84.799
ds=1, scale=-72.5217
END
```

The LDF file can then be used as the load definition in *fe-safe* - see section 5. See section 13 for a description of the LDF file format.

19.4 Checking the validity of the PSD analysis.

The user should check the validity of using a PSD to define the fatigue loading. If an original loading history is available, this validity check can be done in *fe-safe*, as follows:

- The PSD of the load history is calculated, using the **Power Spectral Density (PSD)** function – see section 10.
- The PSD is transformed into a Rainflow histogram using the **Rainflow Histogram from PSD** function – see section 10.
- The Rainflow histogram is used as input to one of the fatigue analysis programs, for example the **BS5400 Welded joints from Histograms** function – see section 11.
- The load history is then cycle counted using the **Rainflow (and Cycle Exceedence) from Time Histories** function (see section 10) to produce a range-mean histogram. This is also analysed using the **BS5400 Welded joints from Histograms** function – see section 11.
- The lives from the two analyses are compared. The difference in the lives indicates the potential errors that can occur when using the PSD as the definition of loading.

For the analysis of non-welded components, the user should also check the importance of mean stresses by analysing the load history with and without a mean stress correction. This could be done using (for example) the **S-N Curve Analysis from Time Histories** function (see section 11), with a suitably scaled S-N curve.

20 This section has been superseded and no longer applicable

21 Analysis of axially symmetric models using *fe-safe/Rotate*

21.1 Introduction to *fe-safe/Rotate*

fe-safe/Rotate is the Rotational Symmetry module for use in conjunction with *fe-safe*.

fe-safe/Rotate speeds up the FEA and fatigue analysis of rotating components by taking advantage of their axial symmetry. The module is used to provide a definition of the loading of a rotating component, through one full revolution, from a single static FE analysis. From a single load step, *fe-safe/Rotate* produces a sequence of additional stress results as if the model had been rotated through a sequence of different orientations (or the load had been rotated around the model).

Using *fe-safe/Rotate* on rotating components has several advantages. Because the stress results produced by the FE package need only contain a single loading step, the time taken to produce the stress results can be reduced. Since the results files contain fewer load steps, the results files are smaller, usually by a factor approximately equal to the number of symmetrical segments in the model.

Additional FE solutions can be introduced where the desired rotation increment (i.e. the angle between each fatigue analysis step) is smaller than the angle of symmetry. Each solution takes advantage of the axial symmetry of the component, requiring a single static FE analysis to define the loading for a full revolution.

Stress information from additional load cases can be appended, providing that they contain the same number of solutions, and the same segment properties as the original data.

fe-safe/Rotate automatically generates a definition of the fatigue loading, in the form of an LDF (Loading Definition) file (*.ldf). The LDF file contains the sequences of datasets that describe the rotation, including intermediate load steps if necessary. If additional load cases are appended the loading definition is regenerated as appropriate.

Once the FE model has been imported into *fe-safe* (using *fe-safe/Rotate*), fatigue analysis is performed in the usual way. The fatigue results are produced for the master segment only, but apply equally to all segments. Some FE packages can expand the data so that it can be viewed for the whole model.

fe-safe/Rotate is particularly suitable where the complete model exhibits axially symmetry, for example: wheels, bearings, etc.. However, the module can also be used where only a part of the model exhibits axial symmetry, for example to analyse the hub of a cam.

The *fe-safe/Rotate* module is included as standard in *fe-safe*, and currently supports ANSYS RST results files (*.rst) and Abaqus FIL (*.fil) files (binary and ASCII) containing element-nodal data.

21.2 Terminology

This section defines some of the terms used in the *fe-safe/Rotate* module.

- Angle of symmetry, S :
the angle of each symmetrical segment.
- Master segment :
the segment being rotated.
- Master segment angle, M :
the angle of the *master segment*, (equal to the *angle of symmetry*, S).

- Solution :
a set of static stress results produced (for the whole model) by the FE package, for a particular orientation of the model.
- Rotated solution :
a solution produced as if the model had been rotated by an angle equal to the *rotated solution angle, R*.

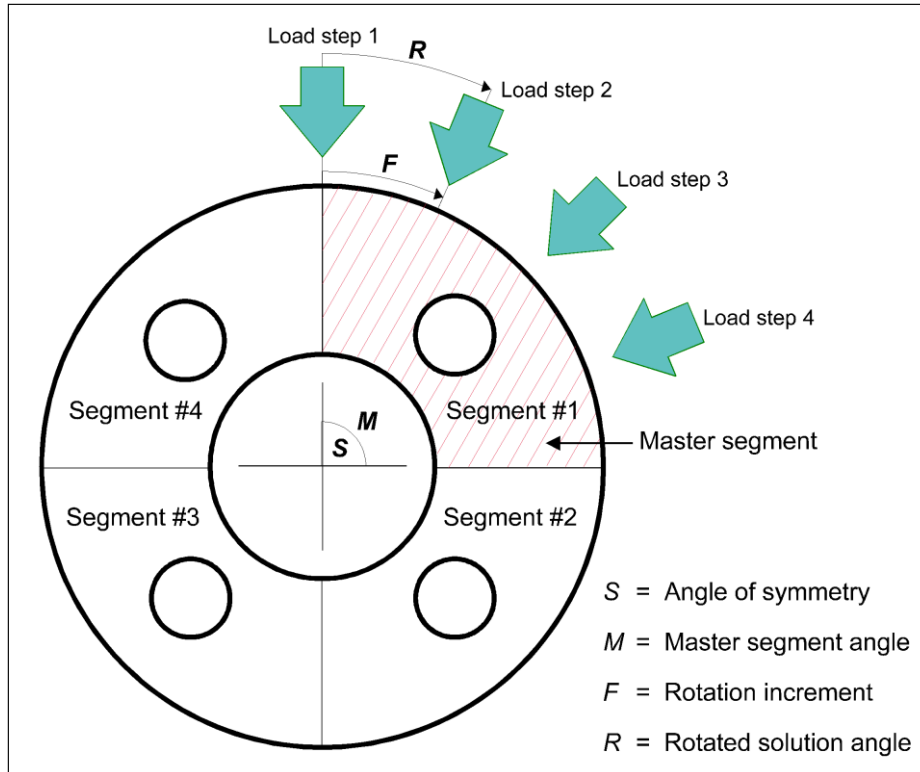


Figure 21.2-1

- Rotated solution angle, R :
the assumed angle through which the model is rotated to produce a rotated solution. The *master segment angle, M*, must be an integral multiple of the *rotated solution angle, R*, i.e.

$$R \times i = M,$$
 where i is an integer.
- Rotation increment, F :
the angle between fatigue data sets, (equal to the rotated solution angle, R).
- Fatigue data set :
a data set, derived from the FE model, loaded into *fe-safe* and written to the FED file. *fe-safe* performs fatigue analysis on a loaded fatigue data set or sequence of loaded data sets.

- Load Definition (LDF) file:
a file defining the loading to be applied for a fatigue analysis, (see 13.9). For this application, the LDF file contains a loading block or series of loading blocks, with each loading block describing a *data set sequence*.
- Data set sequence :
a list of stress data sets defining the variation in load over a sequence of events - in this instance, a sequence of angles - defined in the LDF file, (see 13).
- FED file :
The loaded FE model file - (see Appendix E).
- Element groups :
See section 21.4.3.

21.3 Method

Consider a component that exhibits axial symmetry. The component can be divided into a number of axially symmetrical segments. By definition, these segments are of equal shape and size, but differ in their orientation about an axis. To take advantage of the axial symmetry of the segments, the elements and nodes in each segment must be identical - see section 21.4.

Consider a simple two-dimensional model as shown in *Figure 21.3-1*, below:

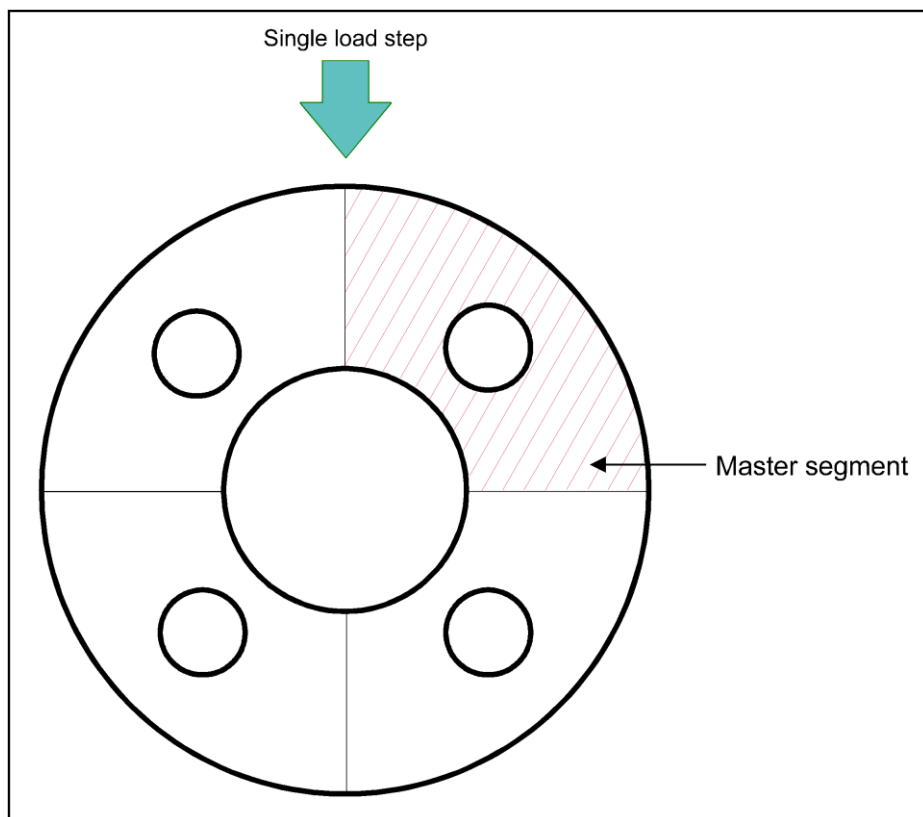


Figure 21.3-1

The model has four modes of axial symmetry - i.e. the model has four segments of equal shape and size. Assume that the model has been prepared with identical elements and nodes in each segment.

One of the segments is defined as the master segment, (see the guide to terminology in section 21.2). To distinguish the master segment from the rest of the model it must be allocated a unique named element group, or groups, in the FE solution – see sections 21.4.3 and 21.4.4, below.

If any elements in the model do not form part of the axially symmetric region, then these must be excluded during the *fe-safe/Rotate* read process by defining one or more element groups that contain the elements to be excluded – see section 21.4.5, below.

The model is now loaded and constrained for a particular axial orientation. An FE solution of the static stresses under these conditions is produced, and written to an FE results file.

In *fe-safe*, *fe-safe/Rotate* is used to import the FE stress results for the model. *fe-safe/Rotate* produces a sequence of additional stress results as if the model had been rotated through a sequence of angles.

The first fatigue data set uses stress data from the elements in the master segment.

To produce the additional fatigue data sets, *fe-safe/Rotate* first has to determine associated elements from each of the other segments for every element in the master segment. In this example, where there are four axially symmetrical segments, *fe-safe/Rotate* finds three elements associated with each element in the master segment. The first associated element (the element from segment #2) is the equivalent element that lies 90° (360°/4) clockwise from the element in the master segment. Similarly, the second and third associated elements, from segments #3 and #4, are the equivalent elements that lie at 180° and 270° from the element in the master segment.

When *fe-safe/Rotate* searches for an associated element it accepts as the closest match the element whose centroid is nearest to the target location. If the centroid of the matched element is further away from the target location than a specified tolerance, then *fe-safe/Rotate* displays a warning, for example:

```
WARNING! There are n matched elements that are out  
of tolerance (i.e. > x away from target location).
```

where *n* is the number of matched elements that are out of tolerance, and *x* is the specified tolerance.

By default, the tolerance, *T*, is calculated by *fe-safe/Rotate* as a function of the number of elements in the master segment, *N_E*, and the number of segments, *N_S*, where:

$$T = 10 / (N_E \times N_S)$$

Alternatively, a user-defined tolerance can be specified.

The tolerance is used only to generate a warning. *fe-safe/Rotate* will use the element with the closest match, even if the warning tolerance has been exceeded.

The first fatigue data set is produced by reading the stress tensors for the elements in the master segment, and writing them to the *fe-safe* FED file.

For the second fatigue data set, *fe-safe/Rotate* reads the stress tensors from all of the associated elements in the second segment. The tensors are rotated through the segment angle (in this case 90°) and then written to the equivalent element in the master segment.

The remaining fatigue data sets are produced in the same way. Each data set now contains a set of stress tensors pertaining to the elements in the master segment, with each data set corresponding to a different rotation angle.

In this example, four data sets are produced, with the tensors from segment #2, #3 and #4 (rotated 90°, 180° and 270° respectively), mapped onto the master segment.

A summary of each fatigue data set automatically appears in *fe-safe*, in the **Current FE Models** window.

The name of the data set describes it in an abbreviated form. For example, the following data set name:

```
3 - FIL = 1; LC = 1; SOL = 1; SEG = 3; <<name>>
```

can be interpreted as follows:

```
3           :      this is data set 3;
FIL = 1     :      the stress data is from the first FE file loaded;
LC = 1      :      this fatigue data set pertains to the first load case;
SOL = 1     :      this fatigue data set pertains to the first solution;
SEG = 3     :      this fatigue data set pertains to the third segment;
<<name>>    :      the stress data is from the loadcase called <<name>>.
```

fe-safe/Rotate automatically produces a load definition (LDF) file that is used by *fe-safe* when performing the fatigue analysis. The LDF file comprises a loading block containing a data set sequence. The data set sequence lists the stress data sets that define the variation in load over a sequence of angles.

For the example in *Figure 21.3-1*, the data set sequence would list four fatigue data sets, DS1 to DS4, describing a complete rotation in four steps: 0° >> 90° >> 180° >> 270°.

The LDF file can be modified as necessary, for example to incorporate scaling information. However, it is important that the order of data sets and blocks is preserved.

If the desired rotation increment (i.e. the angle between fatigue data sets - see 21.2) is smaller than the angle of symmetry, then *fe-safe/Rotate* can be instructed to consider more than one solution.

Each solution takes advantage of the axial symmetry of the component, requiring a single static FE analysis to define the loading for a full revolution. The FE results for the first solution are prepared by considering the model in its original orientation. The next solution is prepared as if the model has been rotated through the rotated solution angle, *R*.

The master segment angle, M , must be an integral multiple of the rotated solution angle, R , i.e.

$$R \times i = M, \text{ where } i \text{ is an integer.}$$

Consider a case similar to the model in *Figure 21.3-1* but this time with the addition of three rotated solutions, as in *Figure 21.3-2*, below:

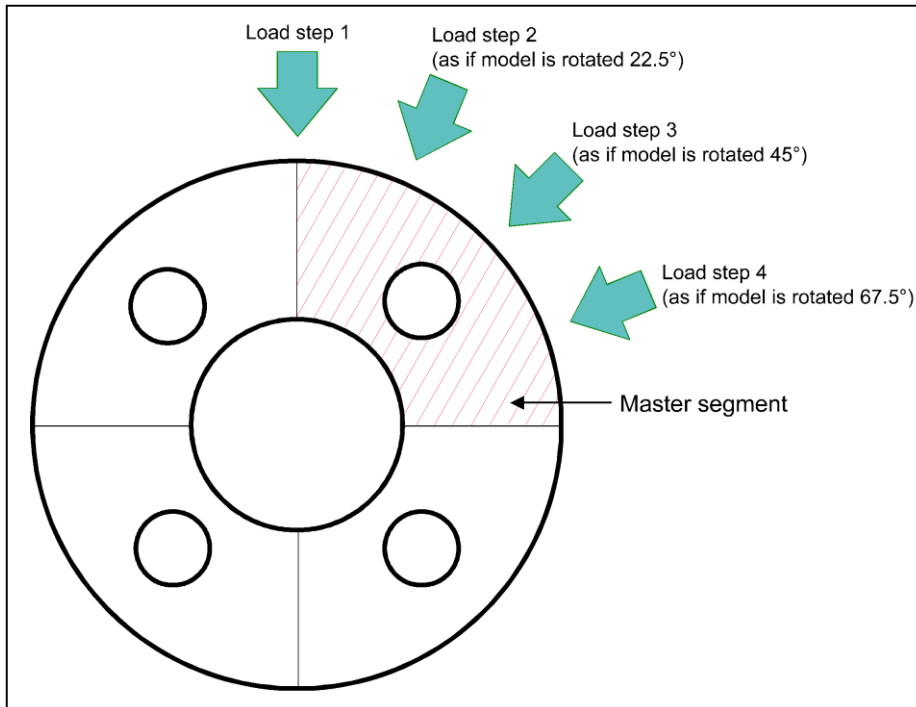


Figure 21.3-2

Again the model has four modes of axial symmetry, but we now need to consider four separate FE stress solutions. Performing a stress analysis with the model in its original orientation produces the first solution. The second solution is produced by loading and constraining the model as if it had been rotated through 22.5° ($90^\circ / 4$). Similarly the third and fourth solutions are produced as if the model had been rotated through 45° and 67.5° , respectively.

In this example, the fatigue data sets are derived from the FE stress solutions and written to fatigue data sets in the following order:

<i>For fatigue data set...</i>	<i>...stresses are read from FE stress solution...</i>	<i>...which was prepared as if the model had been rotated through...</i>	<i>Tensors are read from elements in segment...</i>	<i>...and are rotated through...</i>	<i>...then written to their associated elements in segment...</i>	<i>The equivalent model rotation angle is...</i>
1	1	0°	1	0°	1	0.0°
2	1	0°	2	90°	1	90.0°
3	1	0°	3	180°	1	180.0°
4	1	0°	4	270°	1	270.0°
5	2	22.5°	1	0°	1	22.5°
6	2	22.5°	2	90°	1	112.5°
7	2	22.5°	3	180°	1	202.5°
8	2	22.5°	4	270°	1	292.5°
9	3	45°	1	0°	1	45.0°
10	3	45°	2	90°	1	135.0°
11	3	45°	3	180°	1	225.0°
12	3	45°	4	270°	1	315.0°
13	4	67.5°	1	0°	1	67.5°
14	4	67.5°	2	90°	1	157.5°
15	4	67.5°	3	180°	1	247.5°
16	4	67.5°	4	270°	1	337.5°

The LDF file, automatically generated by *fe-safe/Rotate*, reconstructs the fatigue data sets in the correct sequence to simulate rotation of the model. The data sets constitute a single loading block, with the following sequence:

```

BLOCK n = 1
ds = 1
ds = 5
ds = 9
ds = 13
ds = 2
ds = 6
ds = 10
ds = 14
ds = 3
ds = 7
ds = 11
ds = 15
ds = 4
ds = 8
ds = 12
ds = 16
END

```

The advantages of using *fe-safe/Rotate* can be clearly seen in this example, where sixteen fatigue data sets have been created in the FED file, at equivalent rotational intervals of 22.5°, from just four sets of FE stress data.

21.4 Preparation of the FE model for use with *fe-safe/Rotate*

By definition, axial symmetry implies that it must be possible to rotate the elements in one segment onto the elements of another segment, rather than (or as well as) mirroring them.

To take advantage of axial symmetry *fe-safe/Rotate* must be able to find equivalent elements in each segment that correspond with the elements in the master segment, i.e. elements must coincide when rotated about the axis by the angle of symmetry. Therefore, the FE model should be prepared so that the elements and nodes in each section are identical. *fe-safe/Rotate* works with full models and half-models. There are additional implications from using half-models that must be considered when the model is prepared – see section 21.4.2, below.

The FE model must be axially symmetric about one of the global Cartesian axes, i.e. the rotational axis of the FE model must coincide with one of the Cartesian axes. Models whose axes are parallel to, but not coincident with, one of the global Cartesian axes are not supported.

If the mesh in each segment is not identical, *fe-safe/Rotate* will match elements whose centroids are nearest to their ideal target locations. This can lead to unexpected results, since the elements found in the rotated segments may not be good representations of the equivalent element in the master segment – they could be a different size and shape, or even a different number of nodes. The stresses in such elements are unlikely to be representative.

Note: It is not recommended to use non-identical segments with *fe-safe/Rotate*.

The *fe-safe/Rotate* module currently supports Ansys RST results files (*.rst) and Abaqus FIL (*.fil) files (binary and ASCII) containing element-nodal data.

Note: Currently shell elements are not supported in *fe-safe/Rotate*.

21.4.1 Working with full models

For full models:

- the model must be axisymmetric about one of the global Cartesian axes;
- the boundaries of each segment must not overlap.

To create a full model with identical segments:

- create the geometry for a single segment and mesh it;
- duplicate this segment to create a full axisymmetric model.

21.4.2 Working with half-models

For half models:

- the model must be axisymmetric about one of the global Cartesian axes;
- the half-model must be symmetric in one of the resulting planes, i.e.:

Line of axial symmetry	Plane of half-model symmetry
Global Cartesian X-axis	X-Y or X-Z
Global Cartesian Y-axis	X-Y or Y-Z
Global Cartesian Z-axis	X-Z or Y-Z

- since the radial symmetry of each segment and the half-model mirror symmetry are not exclusive, each half-model segment must be symmetrical about its own radial centre-line;
- the radial boundaries of half-model segments must not overlap;
- for each matched element, *fe-safe/Rotate* also performs a node match, since the node order is likely to change because of the mirroring process.

To create a half model with identical segments (see the example in *Figure 21.4.2-1*):

- create the geometry for a half-segment and mesh it (the light grey area in *Figure 21.4.2-1*);
- create a mirror copy of the meshed half-segment (the dark grey area in *Figure 21.4.2-1*);
- the two mirror-image half-segments constitute one full segment;
- if the model has an even number of segments, duplicate the full segment to create the remainder of the half-model;
- if the model has an odd number of segments, duplicate the half-segments (mirrored and unmirrored, as appropriate) to create the remainder of the half-model.

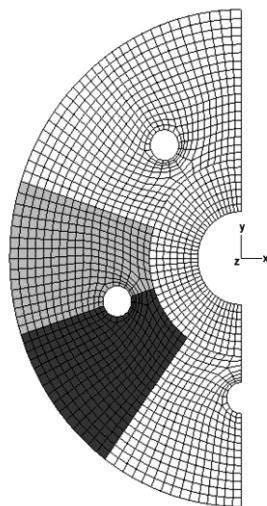


Figure 21.4.2-1

21.4.3 Defining element groups

The *fe-safe/Rotate* interface uses element groups to define:

- the elements that make up the master segment;
 - any elements in the model that do form part of the axially symmetric region or master segment.
- or
- any elements in the model that do **not** form part of the axially symmetric region.

fe-safe uses the term “group” to describe either a list of element numbers (i.e. an ‘element group’) or a list of node numbers (i.e. a ‘node group’).

fe-safe/Rotate supports only element-nodal data. Therefore, in this context, we are concerned only with element groups.

The semantics used to describe element groups differ in different FE packages – this is discussed in Appendix G.

Ansys

Ansys does not export element and node groups directly to the RST file. Therefore, groups are supported in Ansys by the use of the material number.

Abaqus

In Abaqus, element groups are referred to as “Element Sets”.

21.4.4 Defining the ‘Master’ segment

To distinguish the master segment from the rest of the model, the elements contained in the segment must be allocated to a unique group, or to a number of unique groups. All elements in the group(s) must lie within the master segment.

The groups that make up the master segment are defined in the **Open Finite Element Model Using Rotational Symmetry** dialogue box, as follows:

Ansys RST files:

- A list of element group numbers (corresponding to material numbers – see 21.4.3) is entered in the **List of group names defining master segment** box, separated by commas. All elements from the listed groups will be included in the master segment.

Abaqus FIL files:

- A list of element group names is entered in the **List of group names defining master segment** box, separated by commas. All elements from the listed groups will be included in the master segment.
- If the **Automatically add groups starting with ‘M_’** option is selected, then all elements from groups with names that begin with the two characters “M_” (M, underscore) are included in the master segment.

21.4.5 Excluding or including groups of elements in the axially symmetric region

When a model is being read by *fe-safe/Rotate*, it is sometimes necessary to exclude some parts from the axially symmetric region. This allows parts like spigots, bolts, flanges, or any other non-symmetric region to be excluded

from the matching process. Conversely elements can be excluded by default and only those in specified groups will be included.

The exclusion from the axially symmetric region is defined in the **Open Finite Element Model using Rotational Symmetry** dialogue box, as follows:

- The **excluded** and **included** radio buttons determines whether listed groups are used to exclude or include element groups respectively.

Ansys RST files:

- A list of element group numbers (corresponding to material numbers – see 21.4.3) is entered in the **List of groups names to be:** edit box, separated by commas. All elements from the listed groups will be included/excluded from the axially symmetric region.
- If the **Automatically include/exclude groups from rotation, starting with 'X_' or with *.rst files, material numbers 100 or more** option is selected, then all elements from group numbers (corresponding to material numbers – see 21.4.3) greater than or equal to 100 will automatically be include/excluded.

Abaqus FIL files:

- A list of element group names is entered in **List of groups names to be:** edit box, separated by commas. All elements from the listed groups will be included/excluded from the axially symmetric region.
- If the **Automatically include/exclude groups from rotation, starting with 'X_' or with *.rst files, material numbers 100 or more** option is selected, then all elements from groups with names that begin with the two characters "X_" (X, underscore) are included/excluded from the axially symmetric region.

ASCII files:

- A list of element group names is entered in **List of groups names to be:** edit box, separated by commas. All elements from the listed groups will be included/excluded from the axially symmetric region. Spaces in the group name should be replaced with underscores '_'. Note that pre-element matching export of .rst models prefixes 'Material ' to the group names e.g. group (i.e. material) 6 will be called 'Material 6'.
- If the **Automatically include/exclude groups from rotation, starting with 'X_' or with *.rst files, material numbers 100 or more** option is selected, then all elements from groups with names that begin with the two characters "X_" (X, underscore) are included/excluded from the axially symmetric region.

21.4.6 Exporting coordinates

Abaqus FIL files:

fe-safe/Rotate requires the FIL file to contain Cartesian coordinates. This is achieved using the following instruction in the input deck:

```
*NODE FILE
COORD
```

This instruction must be part of the first increment.

21.5 Using *fe-safe/Rotate*

The *fe-safe/Rotate* dialogue is opened by selecting **File >> FEA Solutions >> Open Finite Element Model Using Rotational Symmetry...** from the main *fe-safe* menu.

The following dialogue appears:

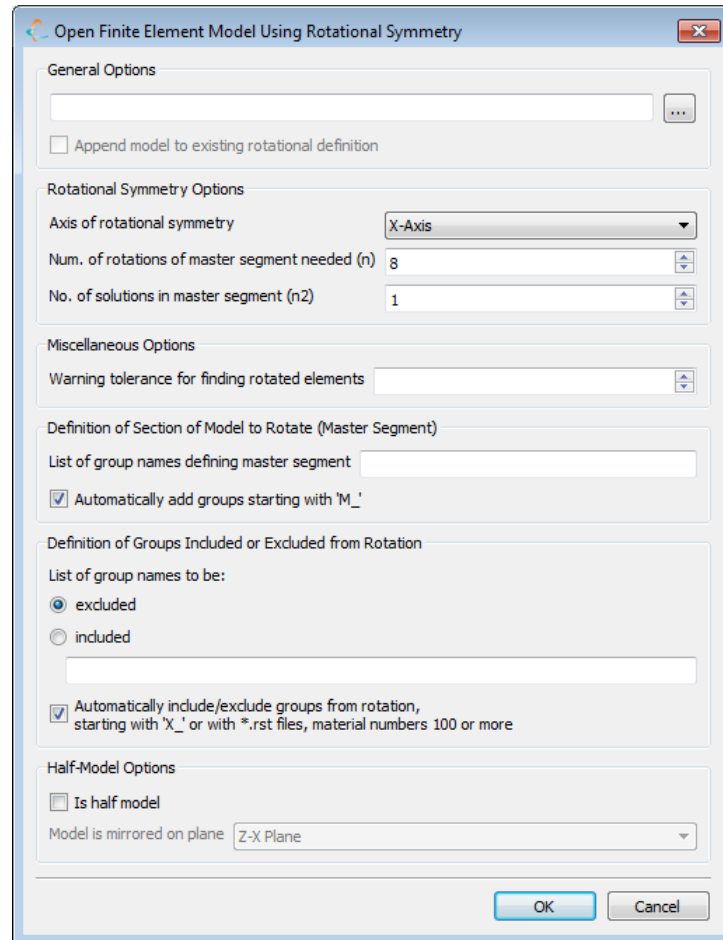


Figure 21.5-1

The name of the FE results file is entered at the top of the dialogue. Clicking on the button labelled '...' allows the user to browse for a file.

The *fe-safe/Rotate* module currently supports Ansys RST results files (*.rst) and Abaqus FIL (*.fil) files (binary and ASCII) containing element-nodal data.

The axis of rotational symmetry should be entered. The FE model must be axially symmetric about one of the global Cartesian axes, i.e. the rotational axis of the FE model must coincide with one of the Cartesian axes. Models whose axes are parallel to, but not coincident with, one of the global Cartesian axes are not supported.

The number of segments and the number of solutions in each segment should be entered. There must be at least one set of FE stress results in the FE results file for each solution.

If there are more sets of stresses in the FE results file than the number of solutions entered, then *fe-safe/Rotate* assumes that the additional sets apply to an additional load case. Therefore, the number of result sets must be an

integral multiple of the number of solutions. If not, then *fe-safe/Rotate* returns an error when it attempts to read the model.

A user-defined warning tolerance can be entered. If the warning tolerance is left blank then *fe-safe/Rotate* calculates a tolerance criterion automatically - see 21.3.

The master segment must be defined, as described in section 21.4.4.

Groups that should be excluded from the rotational region should be defined as described in section 21.4.5.

To append a load case to an existing model (loaded using *fe-safe/Rotate*), select the **Append model to existing rotational definition** option. The appended model must have the same master segment definitions and axis of rotation as the original model. Therefore, if the append option is selected, the file name control is enabled, but all other controls in the dialogue are disabled.

The model is loaded by clicking on the **OK** button. Here there is the option to pre-scan the file in case not all datasets are required. As *fe-safe/Rotate* loads the model, information about the file and the data that it contains is written to the file:

```
<ProjectDir>\Model\reader.log.
```

This information is also displayed in the **Message Log** window.

When the model has finished loading, a summary of the open model appears in the **Current FE Models** window, showing the loaded datasets and element group information.

fe-safe/Rotate also produces a load definition (LDF) file that is used by *fe-safe* when performing the fatigue analysis. The loading details are automatically reconfigured to use the LDF file.

21.6 *fe-safe/Rotate* keywords

For a full description of keywords see Appendix E.

21.6.1 Functional keywords

The functional keywords controlled using the *fe-safe/Rotate* dialogue are included in the overall table of keywords in Appendix E.

21.6.2 Diagnostics keywords

The keywords in *Table 21.6.2-1* provide additional functionality for diagnostic purposes. These keywords should be reset to their default values after use.

Note: The diagnostics facilities below should be used with caution, as they can increase considerably the time taken to import an FE file, and can lead to a very large `reader.log` file.

Keyword	System Keyword	Group Keyword	Function and valid values
DUMP_FEROTATE			0 (default) = do not dump imported model to an ASCII FE tensor file; 1 = dump the imported model to an ASCII FE tensor file (see E.4.1).
ROTATIONAL_SKIPMATCHEDDELS			1 (default) = skip matched elements; 0 = don't skip matched elements.
ROTATIONAL_FORCESROTATE			1 (default) = forces rotate; 0 = model rotates.
ROTATIONAL_FEDDIAGLEVEL			0 (default) = do not write diagnostics to FED file; 1, 2, 5 or 6 = write diagnostics to FED file - see <i>Table 21.6.2-2</i> .
ROTATIONAL_DIAGLEVEL			0 (default) = do not write diagnostics to <code>reader.log</code> file; > 0 = write diagnostics to <code>reader.log</code> file - see <i>Table 21.6.2-3</i> .
ROTATIONAL_NTENSDIAGLINES			>= 0 : number of lines of tensor diagnostics to be written to the tables of rotated tensors; -1 (default) = all elements are written to table of rotated tensors. Note : this value only has an effect if option 14 of ROTATIONAL_DIAGLEVEL is selected – see <i>Table 29.6.2-3</i> .
ROTATIONAL_ECHO_AUTO_LDF			1 (default) = echo LDF file to the message log window and to the <code>reader.log</code> file; 2 = echo LDF file to <code>reader.log</code> file only; 0 = do not echo LDF file.

Table 21.6.2-1

Skip Matched Elements [ROTATIONAL_SKIPMATCHEDDELS] - this option is used to improve the time taken to match elements in the master segment to elements in the other segments. The default option is to skip matched elements - in other words to not attempt to match elements if they have already been matched. This can considerably reduce the number of matching operations, depending on the geometry, the number of segments, and so on.

Forces Rotate [ROTATIONAL_FORCESROTATE] - sets the method by which rotated solutions are applied. The default method (forces rotate) assumes that the rotated solutions are prepared as if the model has rotated through a specified angle.

The alternative method (model rotates) is not available in this release.

FED Diagnostics Level [ROTATIONAL_FEDDIAGLEVEL] - this facility enables diagnostic values to be exported and viewed in an FE viewer. The following options are available:

Function	ROTATIONAL_FEDDIAGLEVEL						
	0	1	2	3	4	5	6
Do not write any diagnostics to the FED file.	✓						
Write values of Cartesian (x, y, z) and cylindrical (r, theta, z) coordinates for each element / node to the FED file, in place of the stress tensors, where: X, Y and Z Cartesian coordinates replace SXX, SYY and SZZ, respectively, and R, Theta and Z cylindrical coordinates replace SXY, SYZ and SXZ, respectively.		✓				✓	
Write unrotated tensors to the FED file, in place of rotated tensors, where unrotated tensors SXX, SYY, SZZ, SXY, SYZ, SXZ replace rotated tensors SXX, SYY, SZZ, SXY, SYZ, SXZ, respectively.			✓				✓
Write the rotational segment number to the FED file in place of the temperature.		✓	✓				
Write the associated (master segment) element number to the FED file in place of the temperature.						✓	✓

Table 21.6.2-2

Rotational Diagnostics Level [ROTATIONAL_DIAGLEVEL] - this facility enables diagnostic values to be exported to the `reader.log` file. The following options are available:

Option	Switch value	Function
00	1	echo node array details
01	2	list individual node record details
02	4	list node reference number, true node number and node coordinates
03	8	list nodes in each element
04	16	echo node coordinate values
05	32	list coordinates of element centroids
06	64	evaluate elements - echo inputs
07	128	evaluate elements - list coordinates, etc.
08	256	evaluate elements - echo segment constraints
09	512	evaluate elements - echo best match for element
10	1024	evaluate elements - echo out of tolerance warnings
11	2048	evaluate elements - write table of element information (all elements)
12	4096	evaluate elements - echo direction cosine information
13	8192	echo element associations when getting data sets
14	16384	write tables of rotated tensors for each element (per data set)

Table 21.6.2-3

The `ROTATIONAL_DIAGLEVEL` keyword can be used to set any combination of the above options by adding the switch values for the required options. For example, to select options 10, 11 and 14, set `ROTATIONAL_DIAGLEVEL` to 19456 (= 1024 + 2048 + 16384).

Number of lines of tensor diagnostics [ROTATIONAL_NTENSDIAGLINES] - used in conjunction with option 14 of Rotational Diagnostics (see above), this value specifies the number of lines of tensor diagnostics per table of rotated tensors. There is one table for each segment.

If option 14 of rotational diagnostics is not set, then this keyword is ignored.

If `ROTATIONAL_NTENSDIAGLINES` is not specified, but option 14 of Rotational Diagnostics is set, then all elements in each table are written to the `reader.log` file.

22 Diagnostic techniques including additional outputs

This section describes techniques for exporting additional contours, tables and load history style plots from Finite Element model based fatigue analyses. This includes tables of individual cycle lives, nodal information, critical-plane orientation, traffic life contours, Smith diagrams, Haigh diagrams, strain gauge type outputs, and influence coefficients.

22.1 Exports and Outputs

The dialogue is obtained by selecting **Exports...** from the **Fatigue from FEA** dialogue. This displays a multi tab dialogue. Each of the tabs are outlined in the sections below:

22.1.1 Contours

A contour is numerical output per analysis item (node or element). This tab allows the configuration of which contour variables are exported to the Output File, which is typically a copy of the FEA results file, for plotting in your FEA viewer.

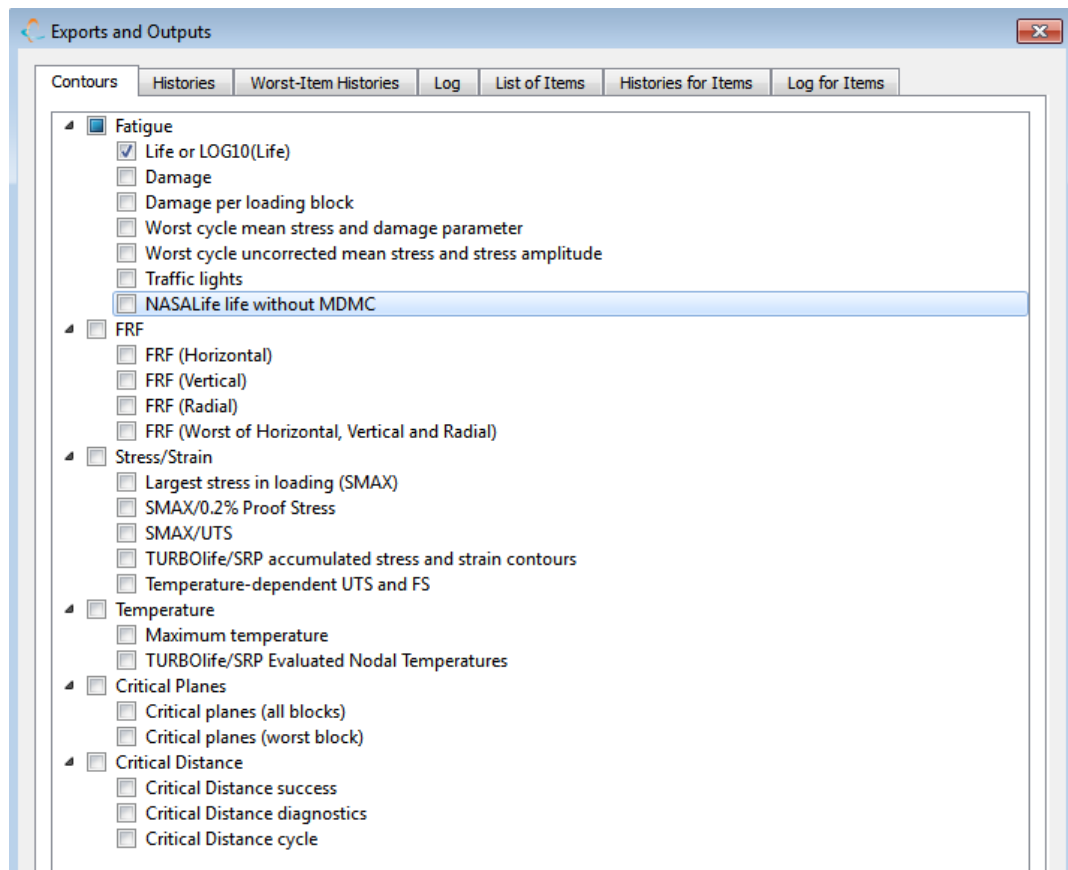


Figure 22.1.1-1

fe-safe currently allows a maximum of 64 scalars to be exported, but note that some options correspond to multiple scalars, e.g. vectors and per-block contours. If the resulting number of scalars exceeds this limit, then the selected contours will be truncated.

Life or LOG10(Life)

This contour indicates the number of repeats of the loading definition which will cause a fatigue failure. However, when editing the loading, it is possible to assign an interval, e.g. in hours or miles, which corresponds to one repeat, so that life is then reported in hours or miles. This is achieved by double-clicking on **Loading is equivalent to 1 Repeats** under the **Settings** node of the loading definition. A dialogue appears in which a numerical scale and a description of the units may be set.

In the event of an item experiencing zero damage, a particular value indicating infinite life will be reported, which is configured using setting [job.infinite life value]. Reserved value -1 indicates that the material's value of the Constant Amplitude Endurance Limit should be reported. If none is defined, a hard-coded value of 1e15 is used.

By default the contour of fatigue lives is in Log base 10, for the best post-processing. Linear versus logarithmic contour output is controlled by selecting **Analysis Options** from the **FEA Fatigue** menu and toggling the option **Export logarithmic lives to results file** on the **Export** tab (see section 5 for more details).

Note that logarithms are not used in the progress table and analysis summary which appear in the analysis log and the Message Log window.

Damage

This contour indicates the fatigue damage that arises from a single repeat of the loading. Damage is defined such that values exceeding unity indicate a fatigue failure. Since most fatigue algorithms accumulate damage according to Miner's rule, damage is then the reciprocal of the fatigue life (in repeats). Damage is calculated by multiplying the damage for each block by its number of repeats (decremented when transitions are used) and summing them, i.e. Miner's rule is implicit. Thus the use of multiple loading blocks or multiple repeats of a block is not appropriate to algorithms which do not use Miners' rule.

Damage per loading block

This contour indicates the fatigue damage that arises from each loading block, taking into account the specified number of repeats of the block (not decremented when transitions are used). Currently, the damage from the transition block (if used) is not reported.

Worst-cycle mean stress and damage parameter

For analysis algorithms that base the fatigue calculation on extracting cycles, the most damaging cycle seen at a node can be exported. Algorithms such as Dang Van that base their calculations at least partially on non-cycle extraction techniques do not export this variable. The Prismatic Hull algorithm will output the effective shear amplitude (including hydrostatic term) for the damage parameter, but the mean stress will default to zero as there is no mean stress correction, and the evaluation is not based on cycle counting.

The amplitude will be the damage parameter for the selected algorithm. A few examples are given in the table 22.1.1-1

Algorithm	Damage Parameter
Normal Stress	Normal stress amplitude (MPa or ksi)
Brown Miller/ BM Combined Direct and Shear Stress	Brown Miller parameter (uE)
Normal Strain	Normal direct strain amplitude (uE)
Von Mises	Von Mises stress (MPa or ksi)
Verity	Equivalent structural stress amplitude (MPa or ksi)
Prismatic Hull	Effective shear stress (MPa), including max hydrostatic term.

Table 22.1.1-1

Worst-cycle uncorrected mean stress and stress amplitude

For analysis algorithms that base the fatigue calculation on extracting cycles, the most damaging cycle seen at a node can have its mean stress and stress amplitude exported. This is similar to the previous item but the stress mean and amplitude here are prior to any plasticity correction on the cycle. Also, unlike the damage parameter of the previous item, it is always stress amplitude that is exported, even though the cycle may be identified by other damage parameters.

Traffic Lights

This is a traffic-light style contour plot of the fatigue lives.

Upper and lower design life thresholds (in user-selected units) can be entered.

The values exported to this contour are:

- 0 (zero) – for a node or element that fails to achieve the design life;
- 0.5 – for a node or element that may or may not achieve the design life (further analysis is necessary);
- 1 – for a node or element that clearly exceeds the design life.

NASALife life without MDMC

This contour is only applicable to the Manson-McKnight Octahedral algorithm. See *Section 14.7*.

FRF Contours

These four parameters are selected by default. However these contours will only be exported in the case that an infinite life Fatigue Reserve Factor (FRF) analysis is selected in configuration options (see sections 5 and 17 for more details).

Prismatic Hull FRF-PH and related contours

The FRF-PH contour will be output if the Prismatic Hull algorithm is selected. It contains the value of the Prismatic Hull safety factor. Like other FRF methods, a value above 1 indicates that infinite life is expected. In addition the Prismatic Hull estimate of the deviatoric shear amplitude and maximum hydrostatic stress are output to contours TauAmp-PH and MaxHydro respectively. See section 14.20. These contours are always output when the Prismatic Hull method is selected. The weighted combination of the TauAmp-PH and MaxHydro stress values is the damage parameter used for the Prismatic Hull (as in Table 22.1.1-1 above).

Largest stress in loading (SMAX)

Selecting this item will export the numerically largest principal stress experienced by a node during the fatigue loading. If a plasticity correction is performed then the stress will include the plasticity effect. The correction performed is based upon the assumption that the cyclic stress-strain curve is transitioned from zero to SMAX.

SMAX/UTS, SMAX/0.2% Proof Stress

Non-dimensional versions of SMAX can be exported. Either the UTS or the proof stress is used to make SMAX non-dimensional.

Note: The UTS is used for both positive and negative stresses, the UCS is not used.

TURBOlife/SRP accumulated stress and strain contours

See the fe-safe/TURBOlife User Manual.

Temperature-dependent UTS and FS

The UTS and Fatigue Strength based on the maximum temperature of each item can be output. When temperature-dependent material properties have been supplied, these will generally be interpolated between the supplied material data points. The Fatigue Strength is calculated as the (zero mean) stress amplitude which will give a life of either 1E4 or 1E7 repeats (so there are actually two Fatigue Strength contours). The Fatigue Strength will be calculated from the required lives as is most appropriate to the analysis algorithm (S-N curve for stress algorithms, strain-life for strain algorithms, Smith-Watson-Topper grey iron curve for Cast Iron algorithm).

Maximum temperature

The maximum temperature used for each analysis item (and used in the calculation of the temperature-dependent UTS and FS contours above) can also be output. The item temperature for each block is taken from the non-varying temperature assigned in the loading, if one is set. Otherwise, FE temperature datasets are used. In the latter case, the datasets used depend on the algorithm: for FKM and plug-in algorithms, the largest temperature in datasets attached to the loading block is used, but for other algorithms, all temperature datasets loaded from the FE solution are used and none need be attached to the block. The block temperature is then the largest of the datasets used; the temperature reported is the maximum over all blocks.

TURBOlife/SRP Evaluated Nodal Temperatures

See the fe-safe/TURBOlife User Manual.

Critical planes

For plane-based algorithms that calculate the fatigue life it is possible to export a vector that is the normal of the critical plane, scaled by $(1/N_f)$.

For a single loading block configuration the two export options are identical and only one will be exported.

Depending on the format of the results file and position of the data, the results will be exported as:

- A vector field (available only in Abaqus `.odb` and I-DEAS `.unv`)
- A tensor field with the results on the diagonal of the tensor and the other components zeroed.
- Three scalar fields.

For result files that support both vector and tensor fields an option is given for choosing a tensor field rather than the vector.

Note: Vector plotting is disabled by default. Please contact your local support office for enabling instructions.

Critical distance success

This option creates a contour on surface nodes for which critical-distance calculations were performed, containing an integer indicating success or failure. See Section 26.3 for details.

Critical distance diagnostics

This option creates a contour on surface nodes for which critical-distance calculations were performed, containing diagnostic codes (integers) which indicate the outcome. See Section 26.5.1 for details.

Critical distance cycle

This option controls two contours: **CritDist-StressAmplitude** and **CritDist-MeanStress**. They contain the amplitude and mean of the largest stress cycle at the sub-surface critical point, or their averages along the critical line.

22.1.2 Histories

This tab allows the export of history plot files for the analysis to be defined. These plot files relate to the whole analysis and generally have one sample per node. The created plot files can be plotted using the **Loaded Data Files** window and **Plot** menu options. If any of the check boxes are selected then a whole analysis plot file is created. Its name is created by appending 'histories.txt' to the specified output file name. For example, for the output file `\data\results.odb` the history file `\data\results.odb-histories.txt` will be created.

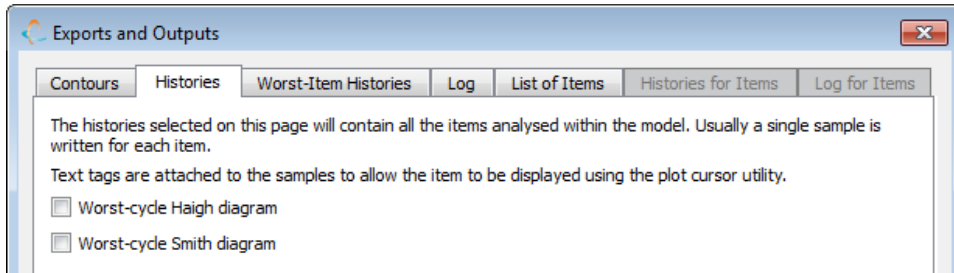


Figure 22.1.2-1

Haigh diagram

For each node the worst cycle's mean stress and damage parameter amplitude are cross-plotted – this is named '**Haigh-all items**'. The damage parameter amplitude varies from algorithm to algorithm, see table 22.1.1-1. If multiple algorithms are used within a single analysis then the amplitude for each node will be a different parameter, i.e. stresses for some and strains for others.

Under certain conditions an infinite-life envelope is added to the history file – this will be named '**Infinite life Haigh diagram for ...**'. The conditions are that:

- the specified analysis has a user-defined MSC or an infinite-life FRF envelope;
- the default MSC is defined for the materials used in the analysis. (If more than one material is used in the analysis there will be an infinite life envelope for each material).

The infinite-life envelope will use the damage parameter amplitude at the FRF design life or the constant-amplitude endurance limit life to scale the non-dimensional MSC or FRF.

See figure 22.1.2-3.

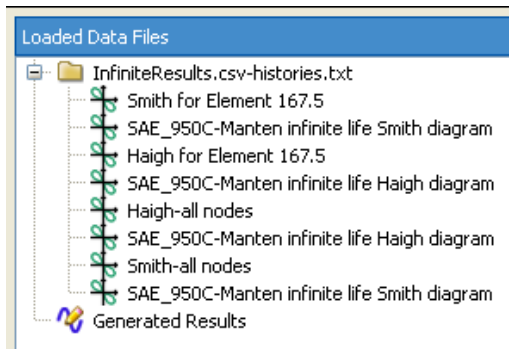


Figure 22.1.2-3

Overlaying the infinite life envelope and the Haigh diagram is shown in figure 22.1.2-4. Each cross represents the most damaging cycle for a node. There are about 80000 nodes on this plot.

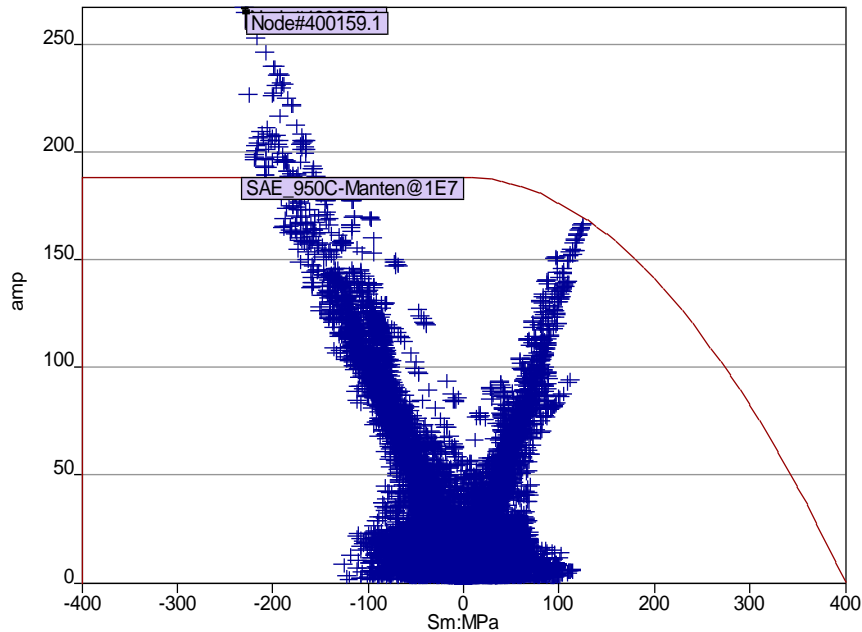


Figure 22.1.2-4

The worst 2 nodes from the whole analysis are marked on the plots. Using the cursor facility allows the node ID for any cycle to be viewed.

How is the worst cycle evaluated?

For nodes with multiple cycles (or those that require a critical plane algorithm) the most damaging cycle is evaluated as follows:

For finite-life algorithms the damage due to each cycle is tracked. If there is no damage, the ratio of the damage parameter to the damage parameter at the constant amplitude endurance limit is used. If two cycles have identical damages then the first encountered is used.

For infinite-life factor algorithms (FRF) the shortest radial factor is tracked and this is used to identify the worst cycle. As with finite-life algorithms, if two cycles have the same factor then the first encountered is used.

Smith diagram

For each node the worst cycle's mean stress and the stresses at each of the turning points are cross-plotted – this is named '**Smith-all items**'. Under the same conditions as those defined in the previous section, an infinite-life envelope is also added to the history file – this is named '**Infinite life Smith diagram for ...**'. See figure 22.1.2-3.

Smith diagrams can only be created for analyses that use a stress parameter as the damage parameter; cross plotting the turning points in strain would be meaningless.

Overlaying the infinite-life envelope and the Smith diagram is shown in figure 22.1.2-5.

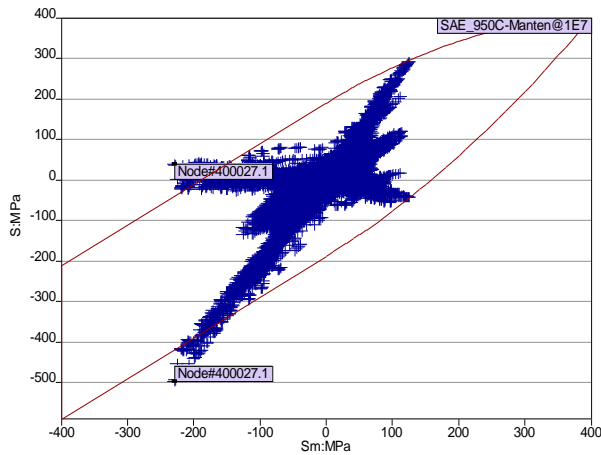


Figure 22.1.2-5

The worst node in the analysis is marked on the plots. Using the cursor facility allows the node ID for any cycle to be viewed. The two turning points for the most damaged node (400027.1) are marked as 'important' tags. All the other turning points are marked with normal tags that can be seen using the cursor facility.

The same criterion is used to evaluate the worst cycle for the Smith diagram as for the Haigh diagram.

22.1.3 Worst-Item Histories

This tab allows the export of history plot files for the most damaging item in the analysis. If a finite-life calculation is being performed and there is no damage then the plots will not be created. The plot files can be plotted using the **Loaded Data Files** window and **Plot** menu options. If any of the check-boxes are selected then a whole-analysis plot file is created. Its name is created by appending 'histories.txt' to the specified output file name. For example, for output file `\data\results.odg` the history file `\data\results.odg-histories.txt` will be created.

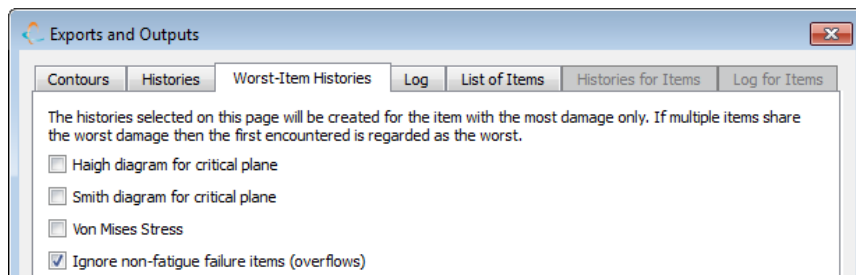


Figure 22.1.3-1

Figure 22.1.2-3 shows a history plot file that contains both the worst-item histories and the whole-analysis histories. In this example the channels named '****for Element 1.3' are the worst-item history plots.

The definition of the most damaging item neglects any non-fatigue failure items that occur when **Ignore non-fatigue failure items (overflows)** is checked. If two items have the same life/FRF values then the first encountered is deemed to be the worst.

Haigh diagram

The Haigh diagram contains all the damaging cycles on the critical plane for the most damaging item in the analysis. Tags indicating the sample numbers for the turning points in the loading are stored with each item. Zero is the first sample in the loading. For infinite-life calculations (FRF), if a residual stress is included in the analysis then the mean value imparted by this residual is also shown on the Haigh diagram, as shown in figure 22.1.3-2.

A sample Haigh diagram is shown below with several of the tags converted to text using the context menu item **Convert Cursor Values to Text**.

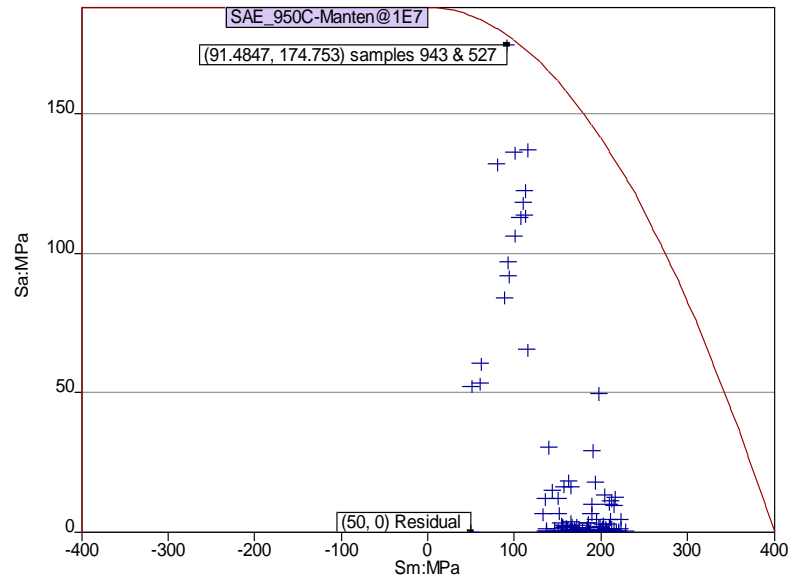


Figure 22.1.3-2

Smith diagram

The Smith diagram also contains all the damaging cycles on the critical plane for the most damaging item in the analysis. Each cycle has a sample for each turning point in stress. Tags indicating the sample numbers for the turning point in the loading are stored with each item. The Smith diagram for the same analysis as in 22.1.3-2 is shown in figure 22.1.3-3.

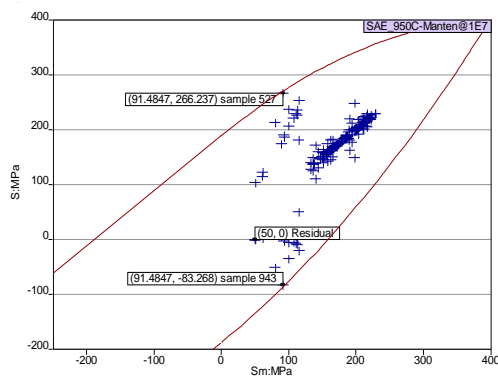


Figure 22.1.3-3

Von Mises

The von Mises stress for the worst item can also be exported. The way in which the sign of the von Mises stress is assigned is controlled from the **von Mises** tab in the **Analysis Options** dialogue. If this is using the Hydrostatic stress then the label of this plot will be **SvM-Hy:MPa** and if this is the Largest Principal stress then the label will be **SvM-LP:MPa**. As with all 'representative' stress variables that have their sign defined by some criteria, there is the possibility of sign oscillation. For the von Mises stress this occurs when the Hydrostatic stress is close to zero (i.e. the major two principal stresses are similar in magnitude and opposite). This is why using such 'representative' stress values for fatigue analysis can cause spurious hot-spots. In areas where this could occur, the von Mises stress plot will mark the sample with a black filled circle as shown in figure 22.1.3-4. A threshold criterion is used to identify samples where the sign is questionable. This criterion is when the hydrostatic stress is less than 2.5% of the von Mises stress.

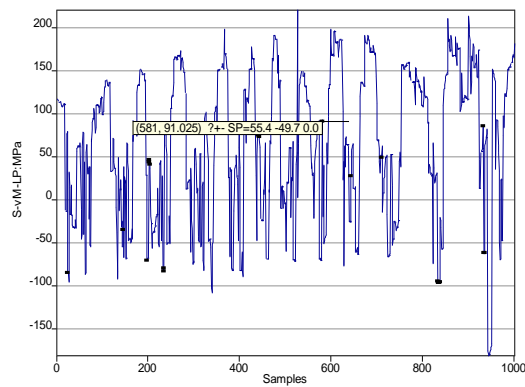


Figure 22.1.3-4

Displaying the cursor values at one of these black circles indicates the principal stresses at the sample.

22.1.4 Log

The **Log** tab allows text-based diagnostics relating to the whole analysis to be written to the .log text file. The name of the log file is derived from the output file name.

For example, if the output file name is:

```
c:\data\testResults_01.fil
```

Then the text-based diagnostics are written to the file:

```
c:\data\testResults_01.log
```

The log file can be viewed in a text-editor.

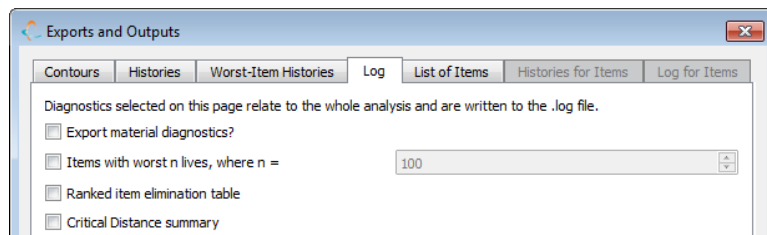


Figure 22.1.4-1

Material Diagnostics

This allows the detailed material parameters to be dumped to the analysis log.

Items with worst n lives

A table of the worst n items can be created for the analysis. A sample table is shown below:

```
ITEM LIFE TABLE
Element           Life           % est.
                  Repeats      Amp/End. Amp
e677.10           8.01e+06      584
e738.5            8.06e+06      584
e735.10           8.87e+06      579
e740.1            1.15e+07      560
e738.9            1.36e+07      551
e738.10           1.48e+07      545
e735.8            1.53e+07      544
e735.7            1.57e+07      542
e735.3            1.65e+07      538
e735.2            1.71e+07      536
```

```
6849 items had non-zero damage
```

```
Item list for use with export dialogue:
```

```
677.10 738.5 735.10 740.1 738.9 738.10 735.8 735.7 735.3 735.2
```

The *%est. Amp/End. Amp* column indicates a nodal elimination estimate that was made for the particular nodes (See the next section).

The list of items can be used in conjunction with the **List of Items** tab to just re-analyse the worst n nodes when trying what-if scenarios.

Ranked item elimination table

The item elimination table can be exported when 'nodal' elimination is turned on (See **Analysis Options** dialogue, **General** tab) and when the loading is a number of scale-and-combines. The code attempts to estimate the worst possible stress/strain ranges using the tensor principals and the load history maxima and minima. This estimate is used to decide if an item can be skipped and not analysed. This information for all items is shown in this table. The extract below shows 3 sections of the table for a 2000 element analysis.

ITEM ELIMINATION TABLE
 For each item, the amplitude of the estimated worst cycle is compared with that of the mean-stress-corrected constant-amplitude endurance limit (CAEL).
 This is used to eliminate items from the fatigue analysis.
 In this table, items are sorted by this ratio, which is listed as a percentage.

Item	% cycle/ CAEL amp	Eliminated ?	Estimated cycle		CAEL amp
			MPa	MPa	MPa
1092.7	1	1	3	-3	172
1209.2	1	1	3	-3	172
79.6	2	1	4	-4	172
79.7	2	1	4	-4	172
.....					
1326.1	79	1	136	-136	172
1332.5	79	1	137	-137	172
86.2	80	0	138	-138	172
87.6	80	0	139	-139	172
.....					
740.1	560	0	966	-966	172
735.10	579	0	998	-998	172
677.10	584	0	1007	-1007	172
738.5	584	0	1007	-1007	172

Critical Distance summary

At the end of the analysis, a summary is produced of the number of items for which each diagnostic code was produced. During the analysis, a line is written to the log for each such node. See *Section 26.3*.

22.1.5 List of Items

For the diagnostics selected in the **Log for Items** and **Histories for Items** tabs, this tab defines the applicable element or nodes. This tab can also be used to limit the analysis to just the listed set of element or node IDs using the **Only analyse listed items** check-box.

If the analysis is to be limited to a few element or nodes, it may be worth writing the results to either the *fe-safe* results file format (.fer) or to an ASCII text file (.csv) to avoid the overhead associated with exporting results to an FE format.

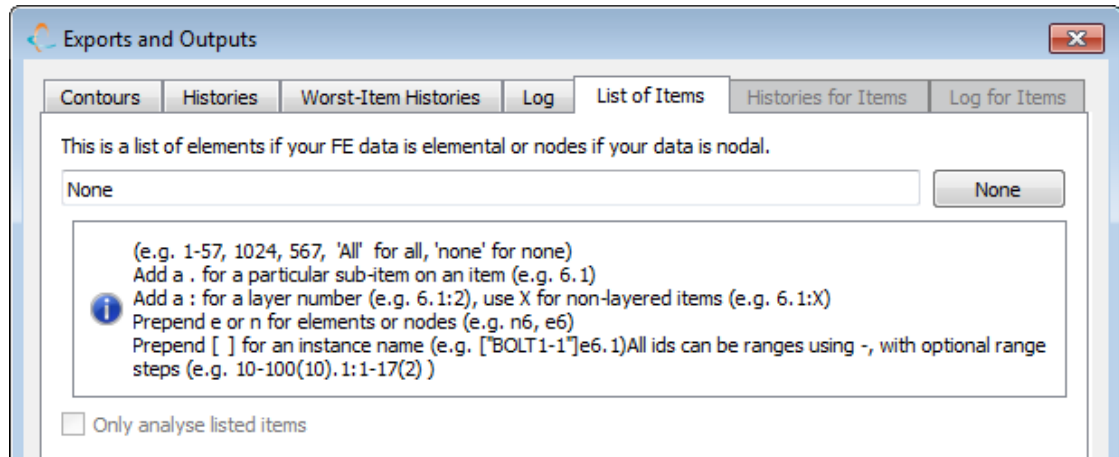


Figure 22.1.5-1

An item is either a node or an element, depending on the context, i.e.:

- elements if stresses are elemental;
- nodes if stresses are nodal.

See Appendix G for a description of how these terms relate to individual FEA suites.

The following syntax rules apply to the item list:

- each item in the list must be separated from the next by a comma;
- to specify a range of items a '-' (minus) character can be used;
- to specify all items, enter 'All';
- to specify no items, enter 'None';
- add a '.' (period) character to specify a node on an element, e.g.: 6.1 (element 6, node 1);
- add a ':' (colon) character to specify a shell layer number, e.g.: 6.1:2 (element 6, node 1, shell layer 2).

22.1.6 Histories for Items

This tab allows a plot file to be created for each node listed in the **List of Items** tab window. The plot files can then be viewed using the **Loaded Data Files** window and **View** menu options.

A separate plottable history file will be created for each item listed in the **List of Items** box. The number of channels of data in the history file will depend on the options selected. If no options are selected, the plot files will not be created.

The name of each plottable file is derived from the output file name and the item.

For example, if the output file name is:

```
/data/testResults_01.fil
```

Then the plottable histories for element 8, node 3 will be written to the file:

```
/data/testResults_01.fil_Element_8.3.txt
```

Plottable history files can be opened using **File >> Data Files >> Open Data File...**, and plotted in the same way as any other loading history or data files, as described in section 7.5.

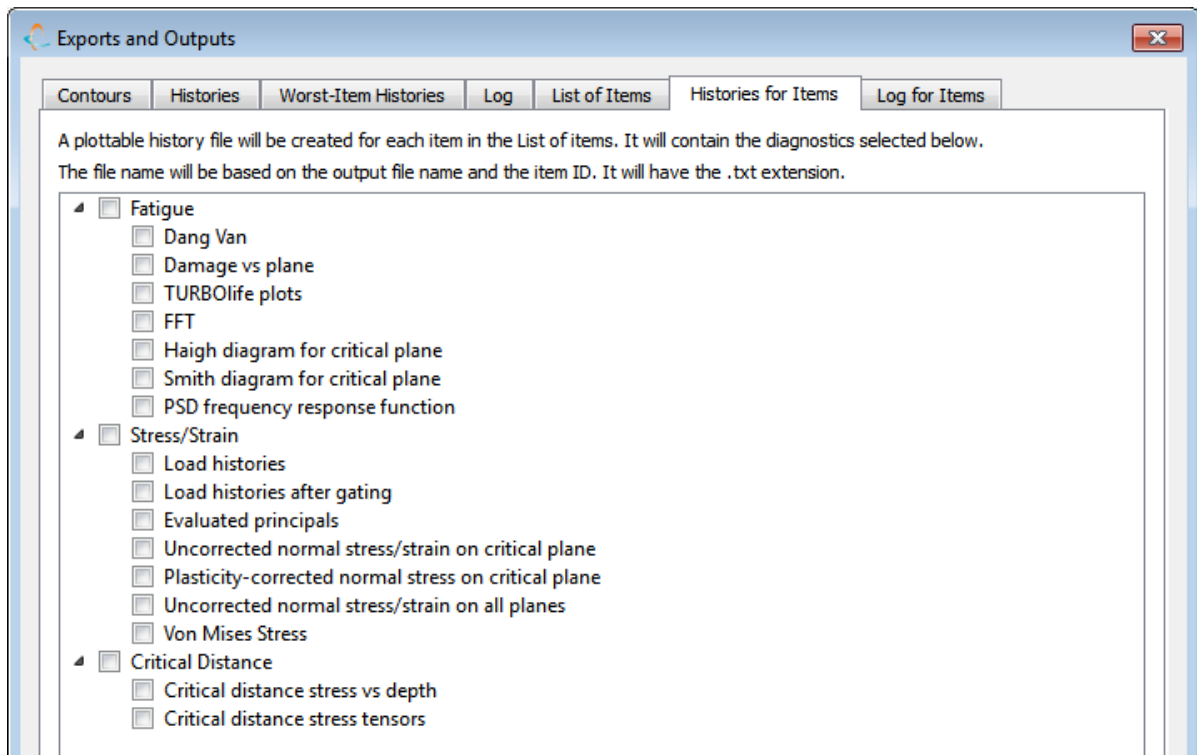


Figure 22.1.6-1

Dang-Van plots

See section 14.12.3.

Damage vs plane plots

The damage-vs-plane plots indicate the damage calculated on each of the planes during the critical-plane calculation. There will be three of these for the Brown-Miller and Maximum Shear algorithms, as the damage is calculated on three different shear-planes.

These are repeated for each block.

If the code performs a triaxial stress calculation within a block then these will be repeated for each triaxial stress plane.

The angle is measured from the orientation of the principals at the reference sample. The reference sample is the stress tensor used to evaluate the orientation of the surface. This is marked in the .log file if the principal stress and strain diagnostics are exported.

If the surface is in the XY plane then the name of the damage channel will include the angle between the X-axis and the critical plane. I.e. **Damage (NOTE: X -> C/P 70 degs)** indicates that the critical plane is 70 degrees clockwise from the X-axis.

A sample set of three plots from a Brown-Miller calculation is shown in figure 21.6.1-2.

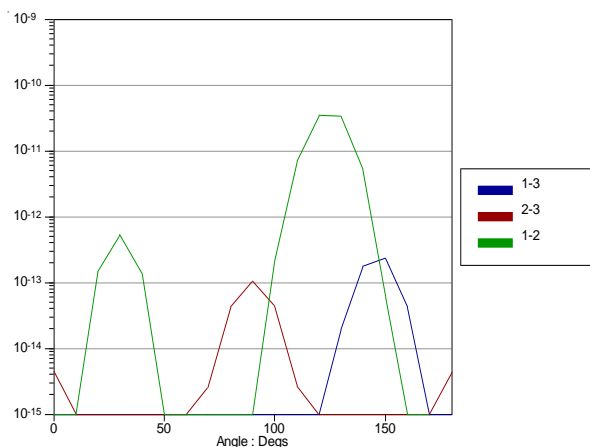


Figure 21.6.1-2

TURBOlife plots

See the *fe-safe/TURBOlife* User Manual.

FFT plots

See section 25.8.

Haigh diagram for critical plane

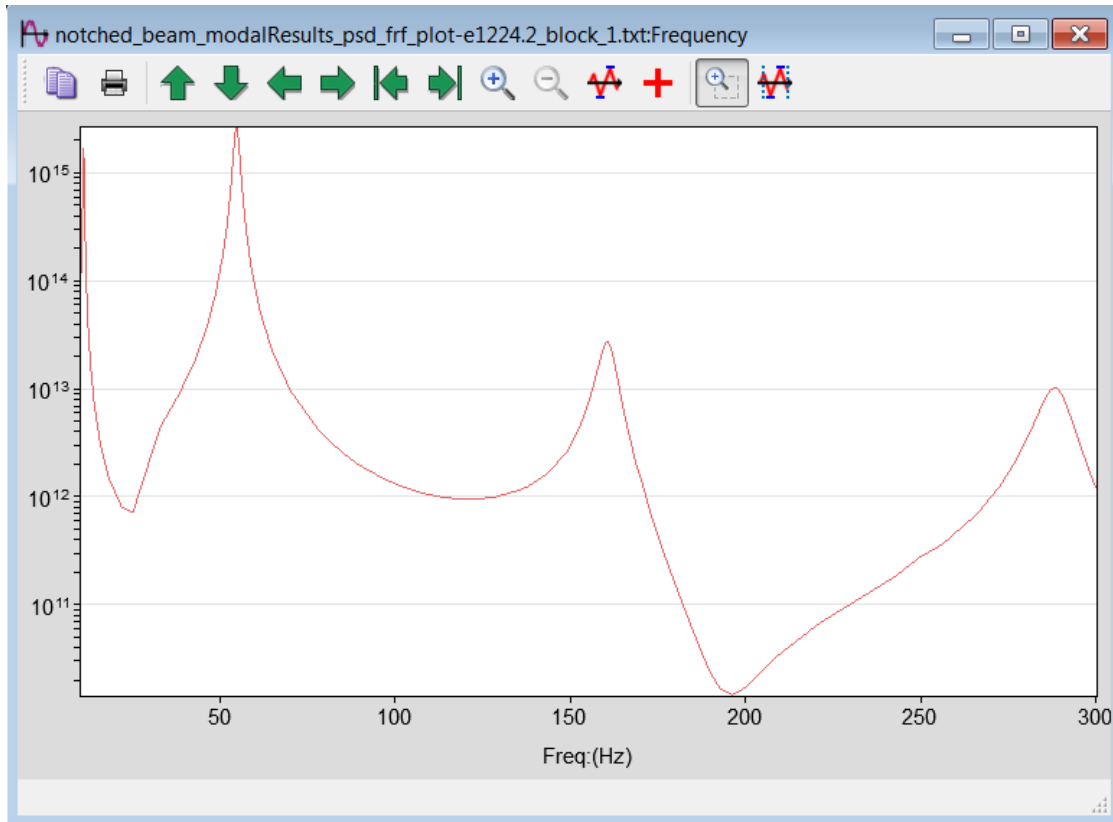
This is the same as the **Worst Node Histories** Haigh diagram except for the specified ID. See section 22.1.3.

Smith diagram for critical plane

This is the same as the **Worst Node Histories** Smith diagram except for the specified ID. See section 22.1.3.

PSD Frequency response function

This shows the response PSD against frequency of the selected item. This is derived by combing the input PSDs with the Generalized Displacements (called Modal Participation Factors in ANSYS) and modal stresses and summing across multiple channels (including channel cross-correlation terms if defined). Note that this cannot usually be directly compared with the input PSD, because it involves a convolution with the modal responses. An example plot is shown below.



Load histories

This exports the full fatigue loading stress tensors. For fatigue analysis from elastic-plastic stress-strain pairs the strains are also exported. These are the stresses (and strains) prior to the code applying a plasticity correction. If there are multiple blocks in the analysis then there will be 6 stress tensor channels per block.

Load histories after gating

If gating is enabled and the analysis algorithm allows gating then these are the tensors after gating.

Evaluated principals

These are the evaluated principal stresses and strains *SP1*, *SP2*, *SP3*, *eP1*, *eP2* and *eP3*. No plasticity correction is applied to these outputs i.e. for fatigue analyses from elastic FEA these are elastic values. The angle between *SP1* at any point in the loading and the reference principal stress is indicated by the channel *Theta1*. Positive values are clockwise. The reference sample is the one used by the code to evaluate the orientation of the surface. *SP1* and *SP2* are in the surface being analysed and *SP3* is out of it.

If the fatigue loading contains multiple blocks then there will be one set of principals for each block.

If the stresses are triaxial then the principals will be repeated for each of the triaxial surfaces that the code analyses. See Technical Note 3 (TN-003) for some examples of triaxial stress diagnostics.

Uncorrected normal stress/strain on critical plane

These are the normal stresses and strains on the critical plane.

Plasticity-corrected normal stress on critical plane

These are the plasticity-corrected normal stresses on the critical plane at the start and end of each cycle. Note that because the plasticity correction is applied to the range of a whole cycle, not individual points within the cycle, this history will typically correspond to a subset of the uncorrected values of the previous export. For more meaningful comparison with uncorrected values, this subset of uncorrected normal stresses is also output whenever this export is selected.

Uncorrected normal stress/strain on all planes

These are the normal stresses and strains on each plane analysed.

Von Mises stress

This is the same as the **Worst Node Histories** von Mises stress except for the specified ID. *See section 22.1.3.*

22.1.7 Log for Items

The **Log for Items** tab allows text-based diagnostics relating to each of the nodes listed in the **List of Items** tab to be written to the .log text file. The name of the log file is derived from the output file name.

For example, if the output file name is:

```
c:\data\testResults_01.fil
```

then the text-based diagnostics are written to the file:

```
c:\data\testResults_01.log
```

The log file can be viewed in a text-editor.

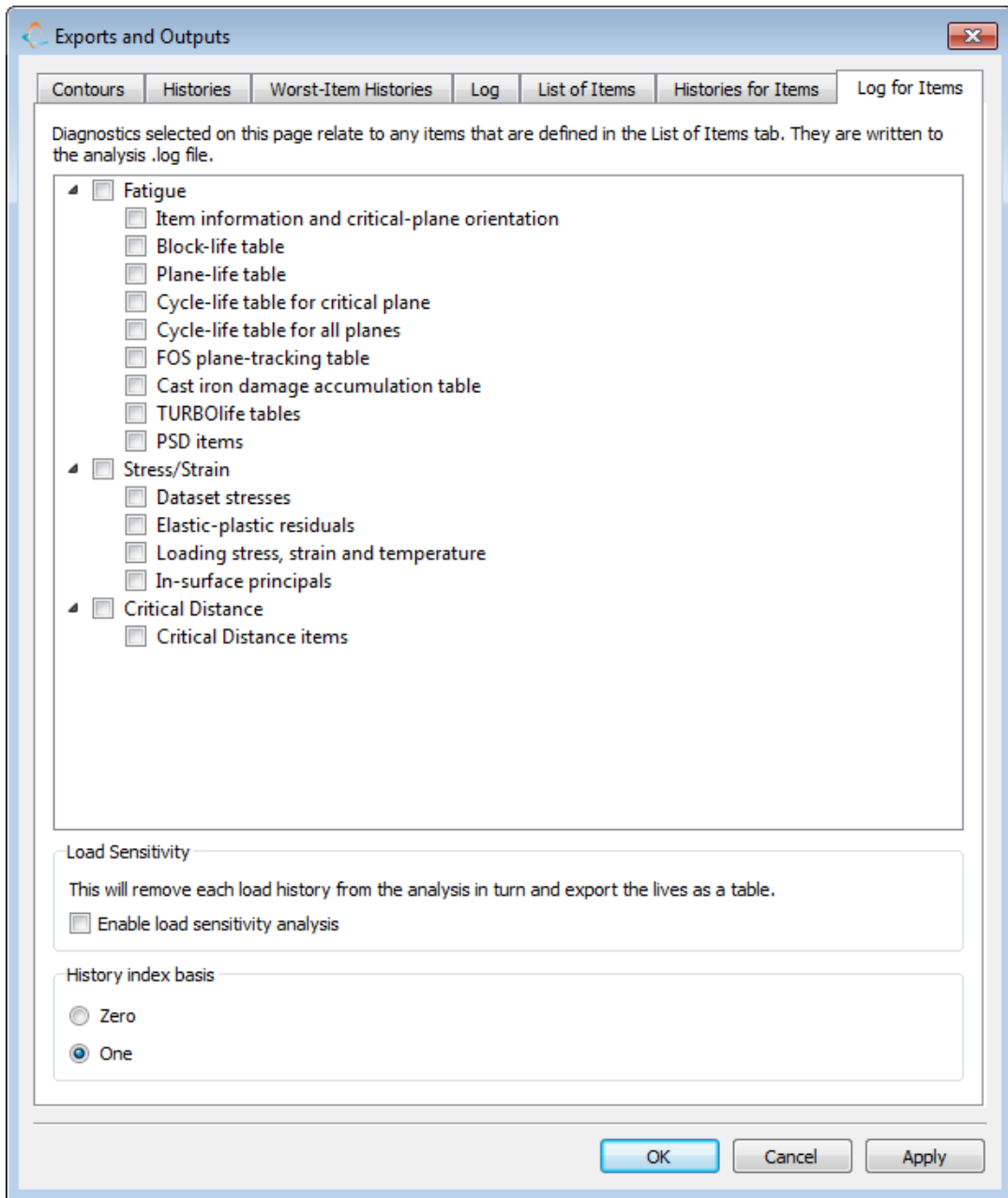


Figure 22.1.7-1

Item Information and Critical Plane Orientation

This can be used to determine the type of stress state that *fe-safe* has evaluated for a node or element and the critical plane determined to have the most damage.

After the analysis the .log file will contain the following table:

Element	Block	Group	T (degC)	S_type	CP / X->Ang (deg)	CP Directional cosines	Tensor Size
						X->CP Y->CP Z->CP	Pre Post
70114.1	1	Remainder	205	Triaxial	171 2-3 177	-0.998 0.031 -0.047	2 2
70114.2	1	Remainder	205	Triaxial	171 2-3 178	-0.999 -0.031 -0.027	2 2
70114.3	1	Remainder	205	Triaxial	171 2-3 ...	0.897 0.437 -0.067	2 2
70114.4	1	Remainder	205	Triaxial	171 1-2 ...	0.194 -0.320 0.927	2 2

Column 5 indicates the stress type. In this example it is triaxial. Column 4 indicates the temperature at the node; columns 6 to 8 indicate the orientation of the critical plane. See Technical Note 3 (TN-003) for more information on diagnostic options for triaxial stresses.

Block-life table

This table lists the damage caused by each loading block, expressed as a fatigue life (N_f) and taking into account its number of repetitions.

BLOCK-BY-BLOCK LIFE TABLE for Element [0]7273.1

Block	n	Nf reps (for n repeats)
Transitions	1	4.974e+06
1	0	Infinite
2	99	2.223e+04
3	9	5.528e+05
Total		2.128e+04

2.128e+04 Repeats

In this example, a transition block was used. This means that the effective number of repetitions n of each of the other blocks is one less than the numbers defined in the loading definition (1, 100 and 10 respectively). Therefore no damage is reported for block 1, since it is taken care of by the transition block.

On the total line, the life may be expressed in a custom unit such as hours or miles, as well as in repeats.

Plane-life table

Damage-versus-critical-plane information can be output to the diagnostics log as a table. In the **Log for Items** tab, check **Plane-life table**, then press **OK**.

For the Brown-Miller and Maximum Shear Strain algorithms there will be 1-2, 2-3 and 1-3 shear-planes for each of the triaxial planes analysed. For the other critical plane algorithms there will be just one.

Diagnostic techniques including additional outputs

The .log file after the analysis will contain a damage-vs-angle table for each node:

PLANE LIFE TABLE for Element 70114.1 (These figures do not include repeats of the blocks for LDF loading, i.e 'n' is not considered)

Block.Tri	Shr Plane	C/P Ang	nf (Repeats)
1.1	1-3	140	5.50e+012
1.1	1-3	150	4.01e+012
1.1	2-3	90	7.64e+012
1.1	1-2	20	7.41e+012
1.1	1-2	30	1.97e+012
1.1	1-2	40	7.68e+012
1.1	1-2	100	4.97e+012
1.1	1-2	110	1.35e+011
1.1	1-2	120	2.56e+010
1.1	1-2	130	2.52e+010
1.1	1-2	140	1.51e+011
1.2	2-3	0	6.01e+010
1.2	2-3	10	9.05e+010
1.2	2-3	20	3.23e+011
1.2	2-3	30	3.15e+012
1.2	2-3	40	9.77e+013
1.2	2-3	150	3.40e+012
1.2	2-3	160	3.39e+011
1.2	2-3	170	9.27e+010
1.2	2-3	180	6.01e+010
1.2	1-2	20	1.33e+012
1.2	1-2	30	3.38e+011
1.2	1-2	40	3.39e+011
1.2	1-2	50	1.50e+012
1.3	2-3	0	1.03e+009
1.3	2-3	10	1.24e+009
1.3	2-3	20	2.04e+009
1.3	2-3	30	4.47e+009
1.3	2-3	40	1.23e+010
1.3	2-3	50	3.87e+010
1.3	2-3	60	1.22e+011
1.3	2-3	70	3.22e+011
1.3	2-3	80	6.22e+011
1.3	2-3	90	7.57e+011
1.3	2-3	120	9.58e+010
1.3	2-3	130	3.00e+010
1.3	2-3	140	9.75e+009
1.3	2-3	150	3.71e+009
1.3	2-3	160	1.79e+009
1.3	2-3	170	1.16e+009
1.3	2-3	180	1.03e+009

Column 1 indicates the block and triaxial plane number i.e. 1.3 is block 1, triaxial plane 3. Column 2 is the shear plane; column 3 is the angle on the critical plane and column 4 the life on the plane. Only planes with some damage are added to this table.

Cycle-life table for critical plane

Plane-based algorithms that are based on rainflow cycles can output a list of their most damaging cycles to the diagnostic log. The example below is for a strain-based shear algorithm which used a plasticity-correction with elastic FE data.

WORST PLANE CYCLE LIFE TABLE for Element [0]e7273.3 Block 3 Triaxial Plane 1 of 3 1-2 120 degs (Maximum of 100 most damaging cycles shown, 1 is the first point)

nf (Repeats)	Cycle		Pt 1	Pt 2	S 1	S 2	elasE1	elasE2	elasS1	elasS2
	uE	uE								
4.98e+02	19182.210	-24133.477	423	1918	187.333	-196.918	8944	-10265	404	-463
8.57e+02	17787.716	-17211.810	1635	1150	183.938	-181.675	8532	-8309	385	-375
8.71e+02	-17965.025	16810.640	2264	1995	-182.676	182.381	-8485	8289	-383	374
1.01e+03	15887.079	-17011.647	2189	2098	179.985	-180.266	8010	-8199	361	-370
1.16e+03	16663.937	-14482.335	1107	624	180.616	-174.910	8172	-7497	369	-338
1.19e+03	12265.298	-18662.061	1827	1814	169.838	-185.079	6870	-8731	310	-394
1.30e+03	-13843.711	15981.418	738	649	-173.086	178.706	-7295	7957	-329	359
1.38e+03	-13487.283	15614.968	400	2267	-170.333	179.349	-7090	7930	-320	358
1.62e+03	-13416.292	14034.792	1038	993	-171.760	172.906	-7154	7328	-323	331
1.84e+03	14818.355	-11350.020	1691	1678	174.715	-165.849	7555	-6500	341	-293

The header line lists the following information:

- the model instance (in square brackets; applies to ODB models only);
- the item and sub-item (here, the element and element-node);
- the loading block;
- the search axis, shear plane and angle (relative to a reference Cartesian axis) of the critical plane, in the local coordinate-system assigned by fe-safe to the sub-item;
- the basis (zero or one) used when reporting points in the load history.

The columns shown refer to:

- The damage caused by a single repeat of each rainflow cycle, expressed as a fatigue life (for finite-life algorithms) or the FRF of the rainflow cycle (for infinite-life algorithms)
- The values of the algorithm-dependent damage parameter at the cycle's end-points;
- The indices of the cycle's end-points in the block's loading history (after gating, if used);
- The stresses at the cycle's end-points, normally after plasticity correction. These define the mean stress used in the mean-stress correction. They may not be displayed if no mean-stress correction is used, nor if the damage parameter is the stress;

Diagnostic techniques including additional outputs

- Elastic strain at the cycle's end-points. This column will not appear if a stress-based method is used;
- Elastic stress at the cycle's end-points. This column will not appear if a stress-based method is used.

Cycle-life table for all planes

This table presents the same information as the **Cycle-life table for critical plane** (above), except that all the analysed planes are tabulated, rather than just the one that experienced the greatest damage. Therefore, the details of the analysis plane appear in each line of the tabulation, rather than in its header. Very large volumes of data may be output.

In the following sample, lines have been truncated for ease of presentation. The omitted columns are identical to those of the **Cycle-life table for critical plane**. In addition, numerous lines have been omitted from the sample. The first line refers to loading block 1, plane-search axis 1, shear mode 1-2, plane angle 120 degrees.

CYCLE LIFE TABLE for Element [0]e7273.3 (Maximum of 100 most damaging cycles shown per plane shown, 1 is the first point)

Block.	Shr Plane	C/P Ang	nf (Repeats) ...
Triax Pl.		Deg.	
1.1	1-2	120	6.52e+06 ...
1.1	1-2	120	7.37e+06 ...
1.2	1-2	120	1.50e+07 ...
1.2	1-2	120	3.32e+07 ...
1.3	2-3	0	9.85e+07 ...
1.3	2-3	0	1.14e+08 ...
2.1	1-2	30	1.89e+06 ...
2.1	1-2	30	2.26e+06 ...
3.1	1-2	120	4.98e+02 ...
3.1	1-2	120	8.57e+02 ...
3.2	2-3	90	1.11e+03 ...
3.2	2-3	90	1.13e+03 ...
3.3	2-3	0	1.06e+03 ...
3.3	2-3	0	1.89e+03 ...

FOS plane-tracking table

Tabulates the FOS on each analysed plane and, for the critical plane, the fatigue life for each iteration of the FOS scale. See *Section 17.2*.

Cast iron damage accumulation table

Tabulates the accumulation of damage as described in *Section 14.21.3*.

TURBOLife tables

Dumps intermediate results of creep/fatigue analyses performed using the TURBOLife Ductility Exhaustion and Strain-Range Partitioning algorithms. See the *fe-safe/TURBOLife User Manual*.

PSD items

Lists, for each block in a PSD analysis:

- The 0th, 1st, 2nd and 4th moments
- The number of peaks per second
- Upward mean crossings per second
- The irregularity factor
- The central frequency
- RMS stress

See Section 19 for more details of frequency-domain fatigue analyses.

Dataset stresses

Tabulates all stress, strain and temperature datasets loaded from the FE model, both in the model's physical units and in the units chosen for exports.

Elastic-plastic residuals

For each loading block, tabulates the residual stress and strain normal to each analysis plane.

Loading stress, strain and temperature

Tabulates the stress, strain and temperature histories resulting from the loading definition. The six components of stress and strain are exported for each time step (sample). In the table **S**** denotes a stress and **E**** a strain. See Technical Note 3 (TN-003) for treatment of triaxial stresses.

LOADING for Element 70114.1 Block 1. Sample indices are one-based.

Sample	Sxx	Syy	Sxy	Szz	Syz	Sxz	Exx	Eyy	Exy	Ezz	Eyz	Exz
	MPa	MPa	MPa	MPa	MPa	MPa	uE	uE	uE	uE	uE	uE
1	-13	-119	-3	-37	-4	-1	520	-1434	-127	78	-133	-46
2	-11	-1	-1	-1	-0	-1	-98	-18	-41	53	-16	-22

In-surface principals

Tabulates idealised principal stresses at each surface node. The principals are idealised in the sense that a surface normal is first surmised from the stress history. The history is then rotated to form a local coordinate system in which the z-axis denotes the normal and the yz- and zx- shear components are set to zero. The principals reported are then the normals in the z-axis (principal 3) and two orthogonal directions in the x-y plane (principals 1 and 2).

The resulting table is shown below. The second and third sets of in-surface principals indicate that *fe-safe* has treated this node as triaxial. In the log file output *SP** denotes a stress and *eP** a strain. *theta* denotes the angle between SP1 (first principal stress) at a sample and the reference plane in the surface. A positive value of theta is clockwise. See Technical Note 3 (TN-003) for treatment of triaxial stresses.

IN-SURFACE PRINCIPALS for Element 70114.1 Block 1 Triaxial Plane 1 of 3 Sample indices are one-based. ** indicates reference sample.

Pt	SP1	SP2	SP3	eP1	eP2	eP3	theta	
	MPa	MPa	MPa	uE	uE	uE	deg	
1	-13	-120	-37	523	-1439	80	-0	**DP
2	-11	-1	-1	-97	-18	52	9	D

Non-Proportional

IN-SURFACE PRINCIPALS for Element 70114.1 Block 1 Triaxial Plane 2 of 3 Sample indices are one-based. ** indicates reference sample.

Pt	SP1	SP2	SP3	eP1	eP2	eP3	theta	
	MPa	MPa	MPa	uE	uE	uE	deg	
1	-120	-37	-13	-1439	80	523	0	**DP
2	-1	-1	-11	-20	52	-96	-3	D

Non-Proportional (constant-direction principals)

IN-SURFACE PRINCIPALS for Element 70114.1 Block 1 Triaxial Plane 3 of 3 Sample indices are one-based. ** indicates reference sample.

Pt	SP1	SP2	SP3	eP1	eP2	eP3	theta	
	MPa	MPa	MPa	uE	uE	uE	deg	
1	-13	-37	-120	523	80	-1439	-0	**DP
2	-11	-1	-1	-96	53	-20	-3	D

Non-Proportional (constant-direction principals)

Critical Distance items

For surface nodes, details of the Critical Distance calculation (if performed) are added to the analysis log. See *Section 26.3*.

Load sensitivity analysis

This function repeats the analysis, removing each load history in turn, to determine which load history causes the most damage. It also provides useful information on which loads are helpful in prolonging the life of your component. For example, in the table below, removing the load associated with dataset 2 reduces the life of the component by more than a factor of 10.

Lives for each repeat of the analysis are reported in a table in the diagnostics log file (see 22.1.3). A sample output table for an analysis containing 22 scale-and-combines would be:

SENSITIVITY ANALYSIS for Element 1.1 (The life is for 1 repeat of the block (i.e n=1), it does not consider the n Value if this is an LDF analysis)

Life (Reps)	%	Omission
4923.79	100	None
3759.04	76	DS#1 * 330c_2d_meas_comp_stick_a.asc:#2
465.69	9	DS#2 * 330c_2d_meas_comp_stick_a.asc:#3
9055.95	184	DS#3 * 330c_2d_meas_comp_stick_a.asc:#4
6318.56	128	DS#4 * 330c_2d_meas_comp_stick_a.asc:#5
7011.65	142	DS#5 * 330c_2d_meas_comp_stick_a.asc:#6
1240.22	25	DS#6 * 330c_2d_meas_comp_stick_a.asc:#7
7165.32	146	DS#7 * 330c_2d_meas_comp_stick_a.asc:#8
6313.75	128	DS#8 * 330c_2d_meas_comp_stick_a.asc:#9
7667.34	156	DS#9 * 330c_2d_meas_comp_stick_a.asc:#10
7239.29	147	DS#10 * 330c_2d_meas_comp_stick_a.asc:#11
8020.3	163	DS#11 * 330c_2d_meas_comp_stick_a.asc:#12
7409.04	150	DS#12 * 330c_2d_meas_comp_stick_a.asc:#13
7531.32	153	DS#13 * 330c_2d_meas_comp_stick_a.asc:#14
6659.44	135	DS#14 * 330c_2d_meas_comp_stick_a.asc:#15
6920.51	141	DS#15 * 330c_2d_meas_comp_stick_a.asc:#16
8327.25	169	DS#16 * 330c_2d_meas_comp_stick_a.asc:#17
8654.95	176	DS#17 * 330c_2d_meas_comp_stick_a.asc:#18
782.96	16	DS#18 * 330c_2d_meas_comp_stick_a.asc:#19
17256.65	350	DS#19 * 330c_2d_meas_comp_stick_a.asc:#20
1719.62	35	DS#20 * 330c_2d_meas_comp_stick_a.asc:#21
7565.82	154	DS#21 * 330c_2d_meas_comp_stick_a.asc:#22
7854.54	160	DS#22 * 330c_2d_meas_comp_stick_a.asc:#23

It should be noted that performing a load sensitivity analysis on a large number of the items in your model could increase the overall analysis time substantially.

22.2 Influence coefficients

For a node, the influence coefficients are the individual contribution of a unit load case dataset on the strains or stresses in a particular direction.

A number of locations (strain gauges) and a number of unit load cases can be defined to create an influence coefficient matrix. This is written to the `.inf` results file, the `.log` file, and the histogram plot files.

This feature is designed for use with centroidal data and elastic stresses. If stress data is used with multiple values per element then the calculation will be performed on each node.

Evaluation of influence coefficients is disabled when analysing stresses and strains from an elastic-plastic FEA analysis.

Only shells, membranes and other two-dimensional elements with coordinate systems defined in the surface of the element should be analysed, i.e. the surface of the component is the XY surface of the element.

To use models containing 3D elements, the model should be skinned with membrane or shell elements.

This 2D limitation is so that the gauge orientation can be specified in straightforward manner.

The influence coefficients consider scale factors and conversion factors for the stresses but do not consider surface finish effects or residual stress effects.

22.2.1 Defining influence coefficients

The **Influence Coefficients and Gauges** dialogue shown in *Figure 22.2.1-1* is displayed by clicking the **Gauges,Inf Coeffs ...** button on the **Fatigue from FEA** dialogue.

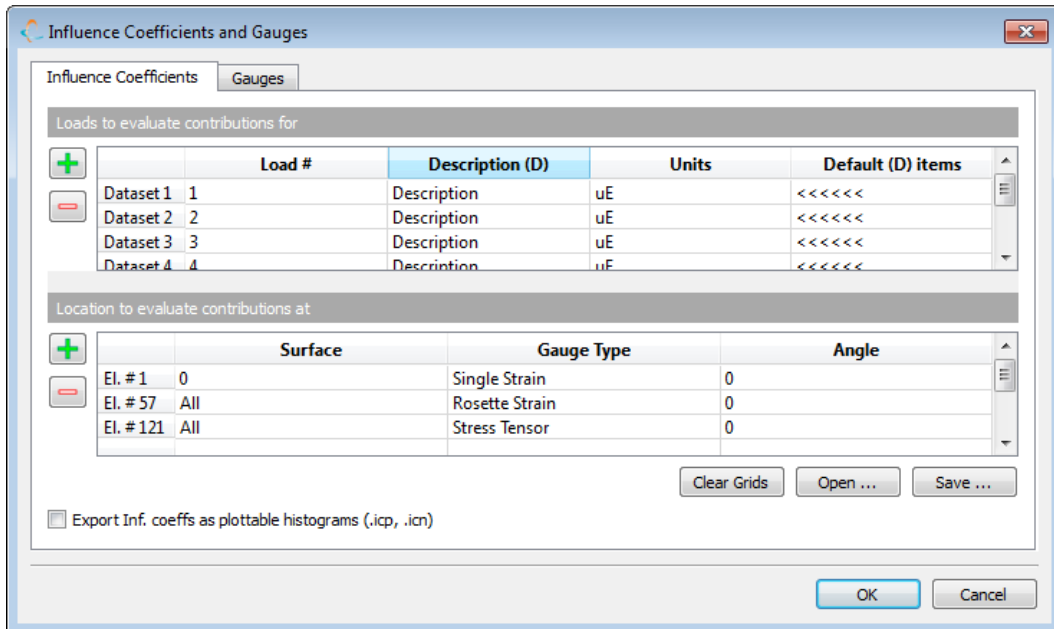


Figure 22.2.1-1

The left tab is used to define the influence coefficients. The top grid defines the loads (or datasets) for which one would like to know their contribution - these are the dataset numbers in the **Current FE Models** window.

To add new loads press the + button at the top of the dialogue. This will display the **Add IC Load** dialogue, as shown in *Figure 22.2.1-2*.

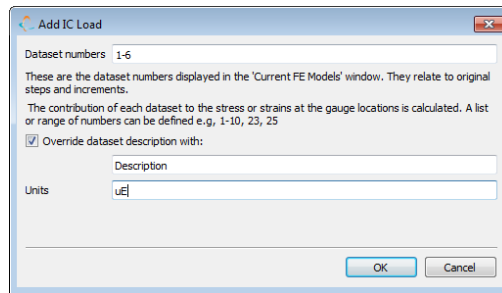


Figure 22.2.1-2

The **Description** and **Units** are text strings that will be displayed in the output matrices. Multiple loads (datasets) can be added by specifying a range or list of datasets. Dataset ranges are specified using a '-' (minus) character, e.g.:

1-17, 24, 27.

Pressing the **OK** button adds the new loads to the loads grid. The **Load #**, **Description** and **Units** columns are editable.

In-cell editing is supported.

Pressing the <<<<<< button on the grid for a load definition will set the description to that associated with the dataset in the **Current FE Models** window.

The **Load #** defines a unique identification number for the load (this can be just the dataset number).

To edit multiple loads simultaneously select the required rows by selecting them with the left mouse button. To highlight additional loads after the first one has been highlighted, hold down the **CTRL** key on the keyboard and click the on the additional rows using the left mouse button. When the required loads (rows) are highlighted, click on the header of the appropriate column to edit that parameter. The relevant dialogue will be displayed. Only columns marked with a * can be edited in this manner - see *Figure 22.2.1-3*.

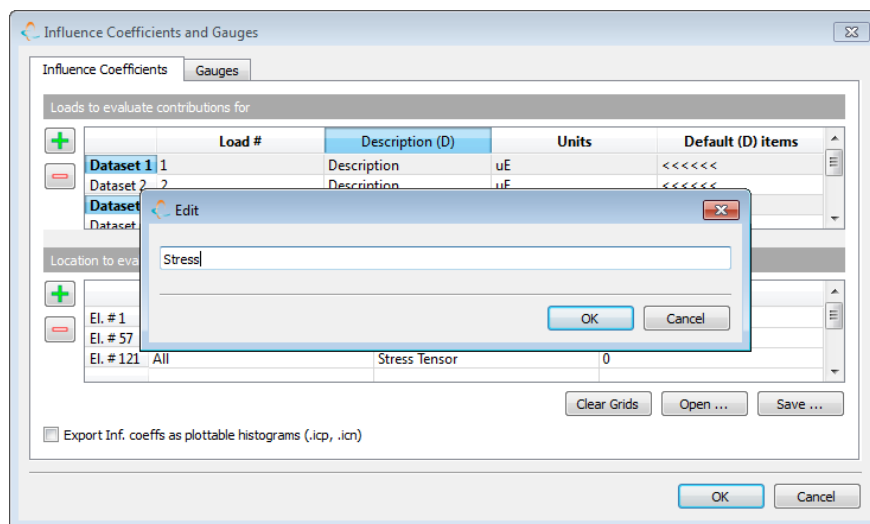


Figure 22.2.1-3

The – button allows multiple loads to be removed.

The **Clear Grids** button removes all loads and gauges from the influence coefficient definition grids.

The second grid (**Location to evaluate contributions at**) is used to define the gauge locations. At each location the individual contribution of the load is evaluated as a stress or strain value. Each location is defined by:

- An element or node number. If a particular node on an element is required then the syntax *el.node* is used.
- A surface. The valid values are shown in the table below:

Value	Meaning
All	All layers in the shell.
Not a shell	The element is not a shell
A number	Defines the surface of interest

- Gauge type. The valid values are shown in the table below :

Value	Meaning
Single Stress	A single or one-armed stress gauge.
Single Strain	A single or one-armed strain gauge.
Rosette Strain	A rosette strain gauge – three gauges are created at 0°, 45° and 90° to the specified orientation.
Stress Tensor	Gauges are simulated as the three stress tensors Sxx, Syy and Sxy. The orientation is ignored.

- Angle. This is the angle from the x-axis to the first arm of the gauge. 0° is along the x-axis and 90° is along the y-axis. This is ignored for the Stress Tensor gauge type.

To add new gauges click the + button above the locations grid of the dialogue. This will display the **Add Gauges** dialogue, shown in *Figure 22.2.1-4*.

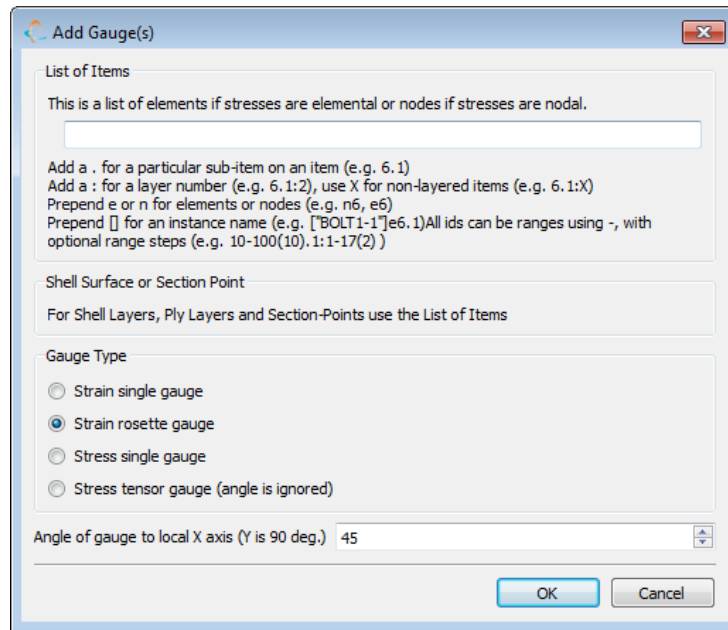


Figure 22.2.1-4

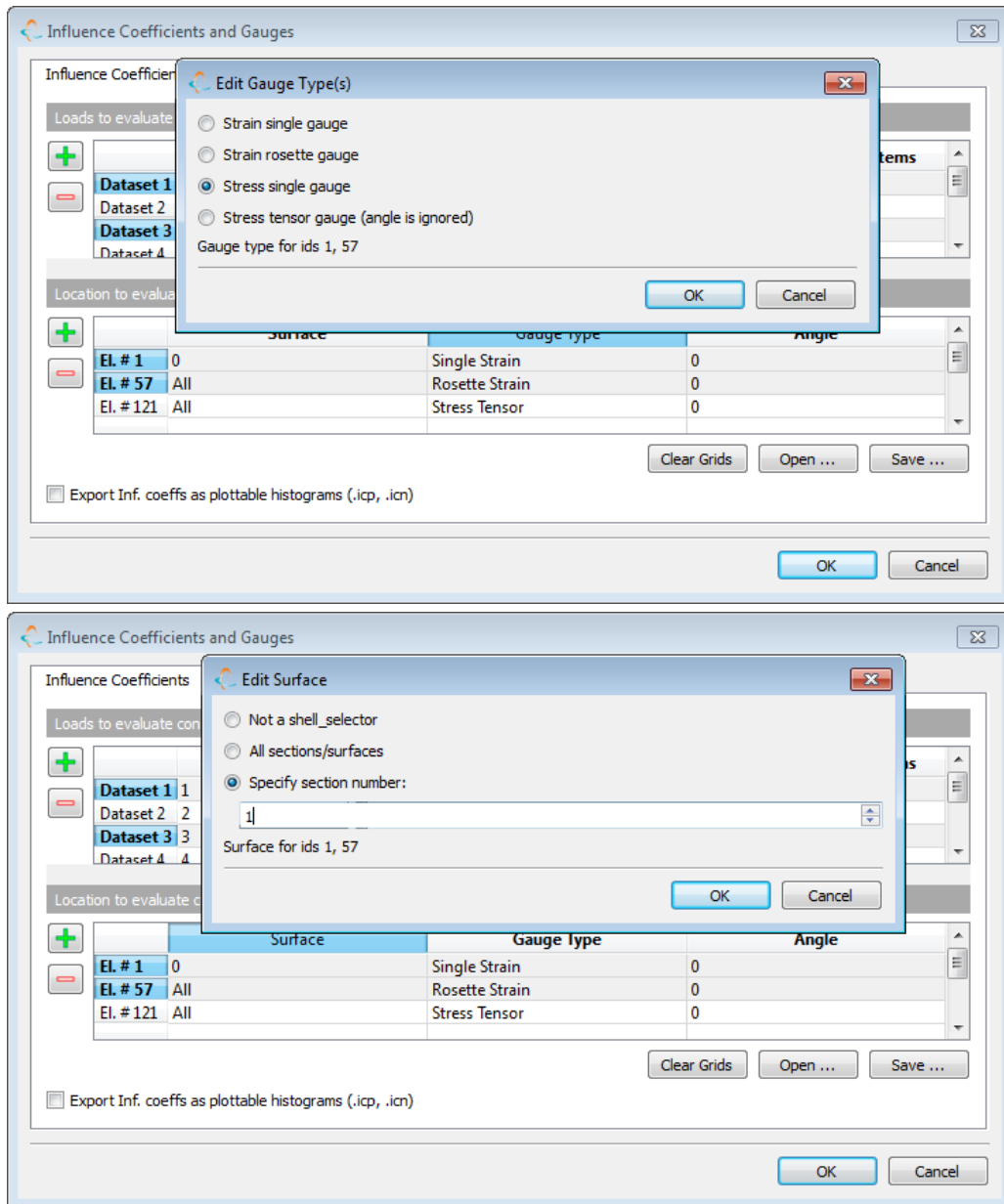
Element ranges and lists are specified using a '-' (minus) character, e.g.:

1-17, 24, 27.

Pressing the **OK** button adds the new gauges to the grid. The **Surface**, **Gauge Type** and **Angle** columns are editable.

Diagnostic techniques including additional outputs

In-cell and multi-gauge editing are supported in the same way as for the load definitions (described above) – see *Figures 22.2.1-5 and 22.2.1-6*.



Figures 22.2.1-5 and 22.2.1-6

The **Open ...** and **Save ...** buttons allow the influence gauge definitions to be saved to a file and reloaded at a later date. See section 22.4 for the format of these files.

22.2.2 Technical discussion and limitations of influence coefficients

The influence coefficients are calculated by resolving the elastic stresses or strains from each load (dataset) into the direction of each of the arms of the gauge. The elastic stresses or strains are calculated from X and Y tensors within the element only, no plasticity correction is performed. Therefore, this should only be used with elements that have their surface and the local XY plane coincidental, i.e. shells and membranes. The reason for this limitation is so that the gauge orientation can be specified in a straightforward manner.

Where out-of-surface direct stresses or shear stresses are found a note is appended to the influence coefficients matrix in the .log file. The gauges where this occurs are marked with a '#' and '!' token in the matrix, e.g.:

```
INFLUENCE COEFFICIENTS

Load ID          Influence Coefficients for gauges
          Descr          SG,670.1  SG,672.1  SG,677.9
                                0          0          0
                                Units    uE          uE          uE
1          FORWARD BEND          MPa    -0.567!#    0.562!#    -0.815!#
2          REVERSE BEND - 1.1X FWD MPa    -0.624!#    -0.619!#    -0.896!#
3          MEAN                    MPa    -0.0284!#    -0.0281!#    -0.0407!#
4          AMP                      MPa     0.596!#     0.59!#     0.856!#

# indicates that at a gauge location there are out of plane direct stresses (Szz != 0)

! indicates that at a gauge location there are out of plane shear stresses (Syz != 0
or Szx != 0)
```

Figure 22.2.2-1

22.2.3 Influence coefficient matrix

For each specified gauge type there are a number of responses for each load case. The table below indicates how many responses and their names.

Gauge type	No. Responses	Names	Notes
Single Stress	1	ST	Stresses are in MPa.
Single Strain	1	SG	Strains are in uE.
Rosette Strain	3	RT, RF, RL	0, 45 and 90 degrees. Strains are in uE.
Stress Tensor	3	SX, SY, SXY	Orientation is ignored. Stresses are in MPa.

The influence coefficient matrix can be exported in three formats as outlined in the following sections.

22.2.4 Influence coefficient matrix in .log format

The format of the matrix written to the analysis log (.log) file for 4 loads and 3 responses is:

Load ID	Descr	Units	Influence G#1 Name, Location G#1 angle G#1 units	Coefficients G#2 Name, Location G#2 angle G#2 units	for gauges G#3 Name, Location G#3 angle G#3 units
L#1 ID	L#1 description	L#1 units	IC for G#1 L#1	IC for G#2 L#1	IC for G#3 L#1
L#2 ID	L#2 description	L#2 units	IC for G#1 L#2	IC for G#2 L#2	IC for G#3 L#2
L#3 ID	L#3 description	L#3 units	IC for G#1 L#3	IC for G#2 L#3	IC for G#3 L#3
L#4 ID	L#4 description	L#4 units	IC for G#1 L#4	IC for G#2 L#4	IC for G#3 L#4

Where G# indicates a parameters associated with a numbered gauge response.

L# indicates a parameters associated with a numbered load.

When out-of-surface direct stresses occur at a gauge location a '#' character is added to the IC for the gauge and when out-of-surface shear stresses occur a '!' character is added to the IC for the gauge. An example of the output written to the analysis log file is shown in *Figure 22.2.2-1*. This includes the out-of-surface markers.

In addition to the matrix, a summary of the influence coefficients will be added to the .log file.

```

Inf. coeffs to      P:\data\fullmodeltests\curResults\500-11-ic7.inf
... plottable      P:\data\fullmodeltests\curResults\500-11-ic7.icp
Influence coeffs.  Dataset   Load ID  Descr.                               Units
                   1           1        ->STRESS 1                          MPa
                   2           2        ->STRESS 2                          MPa
                   3           3        ->STRESS 3                          MPa
                   4           4        ->STRESS 4                          MPa

                   Gauge ID   Surface   Type           Angle
                   1           Not a shell Rosette Strain
    
```

If there were gauges defined in the influence coefficients definition that were not included in the analysis then a message is appended to the .log file similar to the one below:

```

The following Elements were not part of your analysis:
57,89.Z2,104,1234567
    
```

22.2.5 Influence coefficient matrix in .inf format

The .inf file stores the influence coefficients in a format that can be read by FORTRAN programs. This file will be created if the influence coefficients are enabled. It will have the same stem as the output filename.

Where a gauge has a surface set to 'Not a shell', a Z1 and Z2 output gauge will be added to the matrix in the .inf file. The values of Z1 and Z2 will be identical.

The matrix section is formatted with a (I10, 2X, I10, 2X, 1PG15.7) statement. The SLOPE parameter is the influence coefficient. In the example below there are 4 gauges creating 12 responses and 4 loads.

```

INFLUENCE COEFFICIENT MATRIX
VERSION 5.00-09
  Thu Jul 10 15:44:07 2003
  fe-safe LOAD TABLE: 0
CHANNEL SUMMARY:
  NUMBER OF SINGLE GAGES      - 0
  NUMBER OF ROSETTE GAGES     - 2   (6 CHANNELS)
  NUMBER OF STRESS GAGES      - 2   (6 CHANNELS)
  NUMBER OF LOAD CHANNELS     - 4
RESPONSE PARAMETERS:
  NO.  NAME                                UNITS
-----
  1  SX,1,Z1,0.00                          MPA
  2  SX,1,Z2,0.00                          MPA
  3  SY,1,Z1,0.00                          MPA
  4  SY,1,Z2,0.00                          MPA
  5  SXY,1,Z1,0.00                         MPA
  6  SXY,1,Z2,0.00                         MPA
  7  RT,1,Z1,0.00                          MSTRN
  8  RT,1,Z2,0.00                          MSTRN
  9  RF,1,Z1,0.00                          MSTRN
 10  RF,1,Z2,0.00                          MSTRN
 11  RL,1.1,Z1,0.00                        MSTRN
 12  RL,1.1,Z2,0.00                        MSTRN
LOAD PARAMETERS:
  NO.  NAME                                UNITS
-----
  1  Dataset # 1                          MPa
  2  Dataset # 2                          MPa
  3  Dataset # 3                          MPa
  4  Dataset # 4                          MPa
MATRIX:
  RESP. NO      LOAD      SLOPE
-----
  1             1      -3.9937299E-02
  1             2      -0.1823680
  1             3      -5.2025900E-03
  1             4       2.3488799E-02
  2             1      -3.9937299E-02
  2             2      -0.1823680
  2             3      -5.2025900E-03
  2             4       2.3488799E-02
  3             1      -1.6808899E-02
  3             2      -8.8843599E-02

```

Diagnostic techniques including additional outputs

3	3	-9.5912404E-03
3	4	5.1717898E-03
4	1	-1.6808899E-02
4	2	-8.8843599E-02
4	3	-9.5912404E-03
4	4	5.1717898E-03
5	1	4.2707700E-02
5	2	0.2475940
5	3	-2.3491900E-02
5	4	-3.5227101E-02
6	1	4.2707700E-02
6	2	0.2475940
6	3	-2.3491900E-02
6	4	-3.5227101E-02
7	1	-0.1698290
7	2	-0.7558005
7	3	-1.0061632E-02
7	4	0.1075660
8	1	-0.1698290
8	2	-0.7558005
8	3	-1.0061632E-02
8	4	0.1075660
9	1	-0.3743764
9	2	-2.074844
9	3	0.1298187
9	4	0.2787820
10	1	-0.3743764
10	2	-2.074844
10	3	0.1298187
10	4	0.2787820
11	1	-1.7923901E-02
11	2	-0.1415415
11	3	-3.8885854E-02
11	4	-1.2738341E-02
12	1	-1.7923901E-02
12	2	-0.1415415
12	3	-3.8885854E-02
12	4	-1.2738341E-02

22.2.6 Influence coefficients as plottable histograms

To export influence coefficients as plottable histograms ensure the **Export Inf. Coeffs as plottable histograms (.icp, .icn)** checkbox on the **Influence Coefficients and Gauges** dialogue is checked (see *Figure 22.2.1-1*). This will create two histogram matrices, one with the positive influence coefficients (.icp), and one with the negative influence coefficients (.icn). To plot the matrices use the **File >> Data Files >> Open Data File** menu option to open the .icn and .icp files in the **Loaded Data Files** window. Plotting can then be performed using the **View** menu options. The correlation between *response#* and the gauge orientation is shown in the .log and .inf files.

The **Response#** and **Load ID** are associated with the centre of the bins in the histograms.

A sample output for 22 loads and 24 responses is shown below as a plot tilted at 90°.

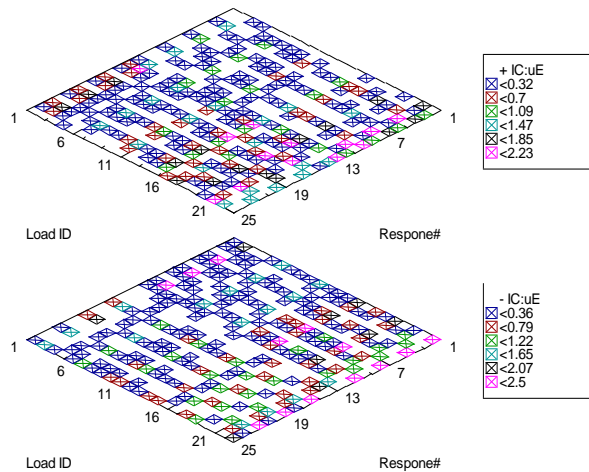


Figure 22.2.6-1

This can also be displayed in tabular format as below, where the columns are loads, and the rows are responses.

Numerical listing of 500-11-ic11.icn										
Load ID vs Response#										
- IC:uE (Mid)										
	1.00	2.00	3.00	4.00	5.00	6.00	7.00	8.00	9.00	10.00
1.00	--	--	0.17876622	0.18395431	--	--	0.060572363	--	--	--
2.00	0.017923901	0.14154153	0.038885854	0.012738341	--	--	--	--	--	0.1832207
3.00	0.37437639	2.0748439	--	--	0.23815262	1.4509829	0.0036597915	0.33694214	0.22395937	1.4190544
4.00	--	--	0.15424596	0.17500004	--	--	0.032835823	--	--	--
5.00	0.016883725	0.19756506	0.027063183	0.0060905144	--	0.027755644	--	--	--	0.22442397
6.00	0.40422505	2.1683264	--	--	0.25650939	1.4995984	0.016861767	0.33627969	0.22949719	1.4963421
7.00	--	--	0.15523857	0.20056874	--	--	0.039692186	--	--	--
8.00	0.046349149	0.24089815	0.047556102	--	--	0.060904928	0.013298384	--	--	0.25316337
9.00	0.42454532	2.3635421	--	--	0.24694295	1.582154	0.0015783908	0.31718063	0.18002151	1.6718802
10.00	--	--	0.17977495	0.20994137	--	--	0.067540757	--	--	--
11.00	0.047870882	0.18557985	0.059713963	0.0061322073	0.003809321	0.02497012	0.023327414	--	--	0.21242654
12.00	0.3950288	2.2732477	--	--	0.22842978	1.5348876	--	0.31753129	0.17367551	1.5974571
13.00	0.46940356	1.6555268	0.022696763	--	0.3889384	1.249684	0.17084818	0.35640156	0.44421172	1.0599253
14.00	--	--	--	0.044942264	--	--	--	0.017487943	0.073952861	--
15.00	--	--	--	0.25834545	--	--	--	--	--	--
16.00	0.45984474	1.7738811	--	--	0.3645986	1.3092554	0.14311208	0.3530764	0.40133813	1.160436
17.00	--	--	--	0.03829439	--	--	--	0.023729106	0.065581568	--
18.00	--	--	--	0.25072461	--	--	--	--	--	--
19.00	0.41739294	1.4575753	--	--	0.36008668	1.144437	0.14996806	0.35651591	0.4422375	0.89628708

Figure 22.2.6-2

22.3 Gauges

At a node the strains or stresses in a particular direction can be exported using the gauges facility. If a plasticity correction is performed within the fatigue analysis this will be included in the calculation of the gauge value. Any surface finish factor will be ignored. Residual stresses will be included. For analysis with multiple blocks the gauge output will be a concatenation of a single repeat of each block.

Only shells, membranes and other two-dimensional elements with coordinate systems defined in the surface of the element should be analysed, i.e. the surface of the component is the XY surface of the element.

Models containing 3D elements should be skinned with membrane or shell elements.

This 2D limitation is so that the gauge orientation can be specified in a straightforward manner.

This module will enable the comparison of measured strains and those evaluated in the fatigue analysis software.

22.3.1 Defining gauges

The **Influence Coefficients and Gauges** dialogue shown in *Figure 22.3.1-1* is displayed by pressing the **Gauges, Inf Coeffs ...** button on the **Fatigue from FEA** dialogue.

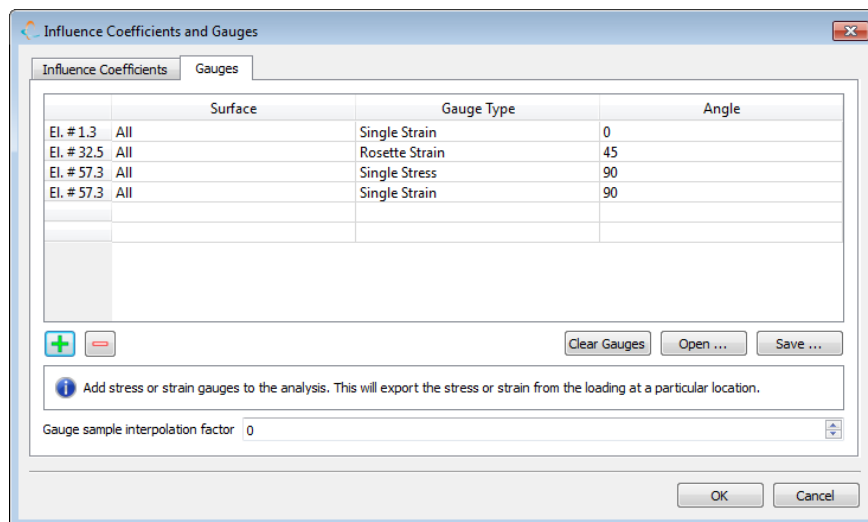


Figure 22.3.1-1

The right tab of the Influence Coefficients and Gauges dialog is used to define the gauges. The grid displays the gauge locations. Each location is defined by:

- An element or node number. If a particular node on an element is required then the syntax *el.node* is used.
- A surface. The valid values are shown in the table below:

Value	Meaning
All	All layers in the shell.
Not a shell	The element is not a shell
A number	Defines the surface of interest

- Gauge type. The valid values are shown in the table below :

Value	Meaning
Single Stress	A single or one-armed stress gauge.
Single Strain	A single or one-armed strain gauge.
Rosette Strain	A rosette strain gauge – three gauges are created at 0°, 45° and 90° to the specified orientation.

- Angle. This is the angle from the x-axis to the first arm of the gauge. 0° is along the x-axis and 90° is along the y-axis.

To add new gauges press the + button beneath the gauges grid. This will display the dialogue shown in *Figure 22.3.1-2*.

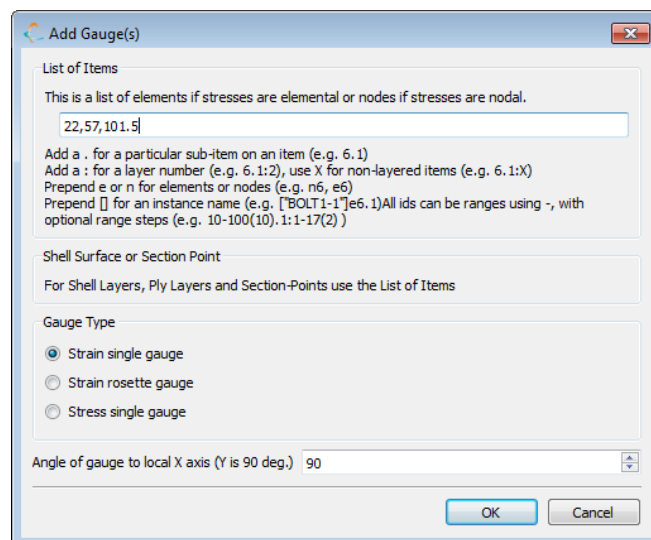


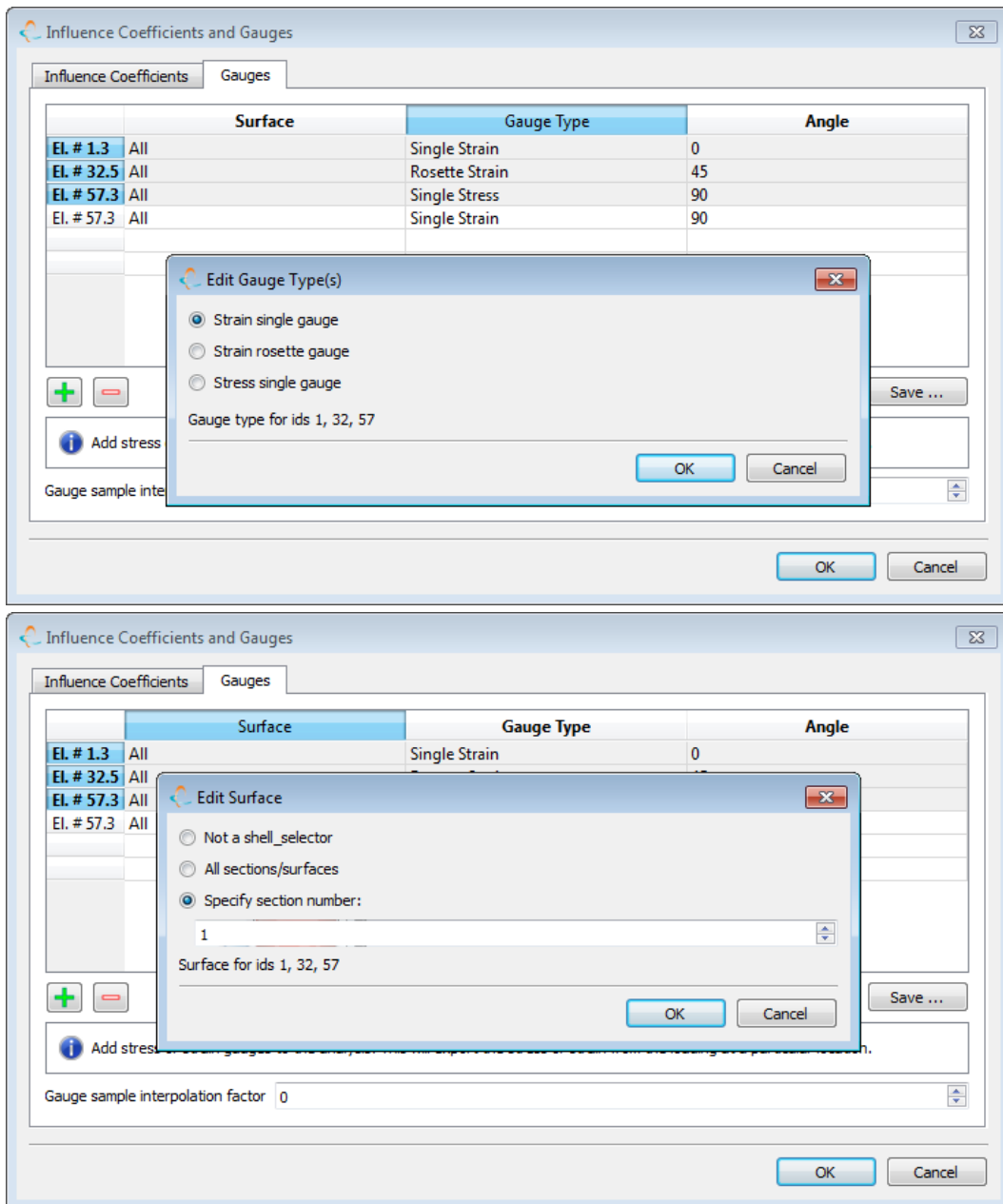
Figure 22.3.1-2

Element ranges and lists are specified using a '-' (minus) character, e.g.:

1-17, 24, 27.

Pressing the **OK** button adds the new gauges to the grid. The **Surface**, **Gauge Type** and **Angle** columns are editable.

In-cell and multi-gauge editing are supported in the same way as for the load definitions (see 22.2.1, above). Only columns marked with a * can be edited in this manner. See *Figures 22.3.1-3 and 22.3.1-4*.



Figures 22.3.1-3 and 22.3.1-4

The – button allows multiple gauges to be removed.

The **Clear Gauges** button will remove all gauges from the gauges definition grids.

The **Open ...** and **Save ...** buttons allow the gauge definitions to be saved to a file and reloaded at a later date. See section 22.4 for the format of these files.

The **Gauge sample interpolation factor** allows samples to be inserted into the tensors built for a node. This allows smoother hysteresis loops to be plotted where a small number of samples define the loading cycle. This is discussed more in the section 22.3.3.

22.3.2 Gauge outputs

For each specified gauge type there are a number of outputs as shown in the table below.

Gauge type	No. of outputs	Notes
Single Stress	1	Stresses are in MPa.
Single Strain	1	Strains are in uE.
Rosette Strain	3	0°, 45° and 90°. Strains are in uE.

The gauge outputs will be written to the plot file for a node. The plot file names are derived as described in section 22.1.2. One plottable output will be created for each arm of the gauge. These will be named as follows:

Name	Description
<i>EPS_gauge_ang</i>	Elastic-plastic strains
<i>SIG_gauge_ang</i>	Elastic-plastic stresses
<i>E_gauge_ang</i>	Elastic strains
<i>S_gauge_ang</i>	Elastic stresses

Diagnostic techniques including additional outputs

Where an elastic-plastic correction is performed in the fatigue software, or the input stresses and strains are interpreted as elastic-plastic, then the elastic-plastic versions of stress and strain will be written. The table below shows this in more detail. An × denotes an algorithm does not support a particular analysis.

Analysis type	Outputs from analysis of elastic data	Outputs from analysis of elastic-plastic data
Normal Strain	EPS and SIG	EPS and SIG
Brown Miller	EPS and SIG	EPS and SIG
Max Shear	EPS and SIG	EPS and SIG
Cast Iron	EPS and SIG	EPS and SIG
Normal Stress Life from SN data	E and S	×
Normal Stress Life from strain life data	EPS and SIG	×
Weld Life	E and S (for strains, Young's Modulus is 203000 Mpa)	×
Dang Van	E and S	×
Von Mises life	E and S	×
Uniaxial Strain Life	EPS and SIG	×
Uniaxial Stress Life from SN data	E and S	×
Uniaxial Stress Life from strain life data	EPS and SIG	×

Figure 22.3.2-1

Where a plasticity correction is performed the strain and stress gauge outputs will vary from the “Normals” described in section 22.1. In section 22.1 elastic stresses and strains are exported when a plasticity correction is performed.

After the analysis is complete, a nodes plot file for a specified gauge can be opened in the **Loaded Data Files** window, using **File >> Data Files >> Open Data File ...**:

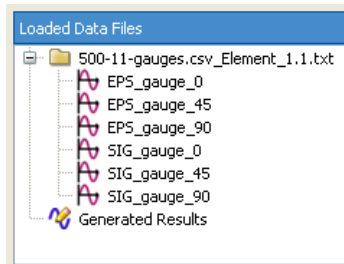


Figure 22.3.2-2

The example in *Figure 22.3.2-2* was created with a rosette strain gauge and 3 single stress gauges. The gauges are plotted in *Figure 22.3.2-3* below.

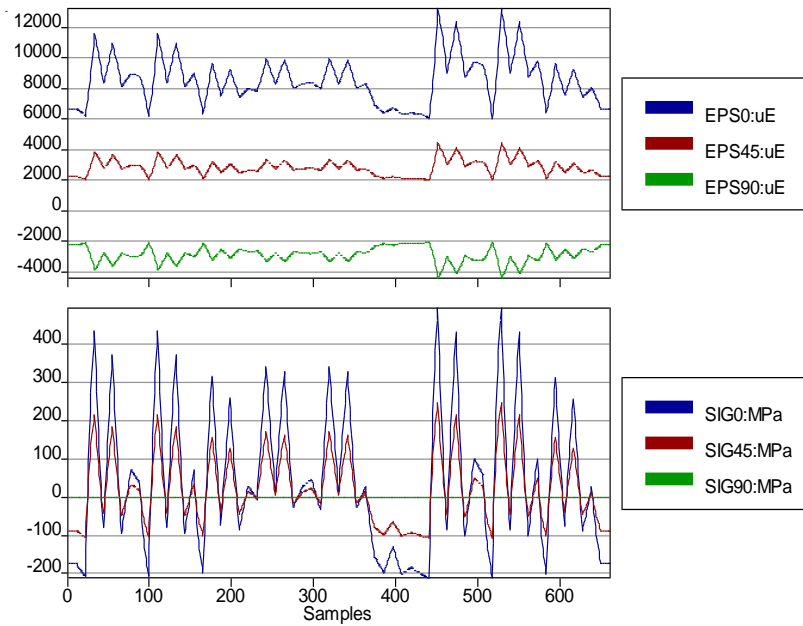


Figure 22.3.2-3

In addition to the plottable outputs the `.log` file will contain a summary of the gauges defined for an analysis.

```

Gauges
      ID      Surface      Type      Angle
      1      Not a shell  Rosette Strain  0
      1      Not a shell  Single Stress  0
      1      Not a shell  Single Stress  45
      1      Not a shell  Single Stress  90
      7      Not a shell  Single Strain  45
      7      Not a shell  Single Strain  90
      7      Not a shell  Single Strain  45
      9      Z2          Single Strain  90
Gauge Interpolation  0
  
```

and will display a list of gauge outputs created.

```

Gauge 'EPS_gauge_0' added to plot file P:\500-11-gauges12a.csv_Element_1.1.txt
Gauge 'EPS_gauge_45' added to plot file P:\500-11-gauges12a.csv_Element_1.1.txt
Gauge 'EPS_gauge_90' added to plot file P:\500-11-gauges12a.csv_Element_1.1.txt
Gauge 'SIG_gauge_0' added to plot file P:\500-11-gauges12a.csv_Element_1.1.txt
  
```

If there were gauges that were not a part of the analysis then a message similar to the one shown below will be added to the `.log`.

```

WARNING:The following ids defined in your Gauges were not part of your analysis :
67
  
```

When out-of-surface direct stresses occur at a gauge location a '#' character is added to the gauge name and when out-of-surface shear stresses occur a '!' character is added to the gauge name. An example of the output of surface markers is shown in *Figure 22.3.2-4*.

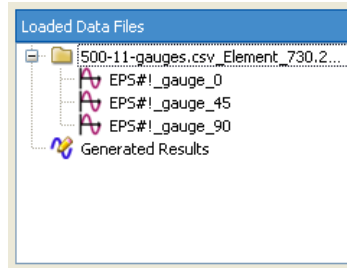


Figure 22.3.2-4

22.3.3 Technical discussion and limitations

Surface finish factors are not considered in this module for practical reasons. To apply a strain gauge to a component a smooth or ground surface is required. The surface perturbations that cause the increase in local stress and strains are unlikely to be picked up by such gauges. As a consequence the surface finish is ignored in the gauge calculations. Only elements that have the surface in the XY plane should be analysed.

To plot hysteresis loops at a particular orientation add a stress and a strain gauge. Then cross plot the *EPS* output channel and the *SIG* output channel. It should be noted that only those analysis methods that perform a plasticity correction (see *Figure 22.3.2-1*) will create hysteresis loops that are anything other than straight lines.

If the input loading is defined by just the cycle turning points the hysteresis plots will similarly be just straight lines. This can be seen in *Figure 22.3.3-1*.

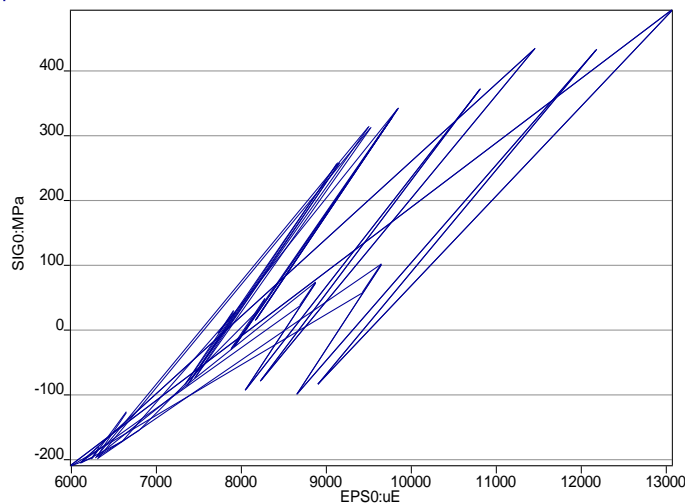


Figure 22.3.3-1

The **Gauge sample interpolation factor** (see *Figure 22.3.1-1*) can be used to insert extra samples between each of the samples in the loading to provide better hysteresis loop definition. *Figure 22.3.3-2* shows the same loading as *Figure 22.3.3-1* with an interpolation factor of 10.

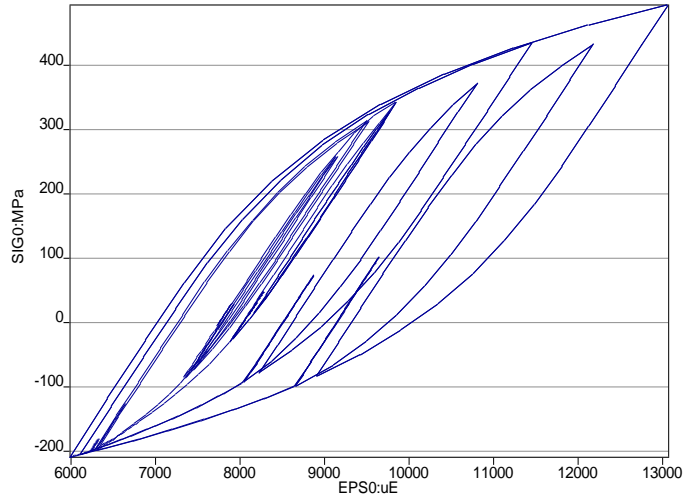


Figure 22.3.3.2

It should be noted that for fatigue analysis from elastic-plastic FEA results the interpolation factor does not improve the hysteresis loop shapes as a plasticity correction is not applied in the fatigue software. Superimposing the interpolated and non-interpolated outputs shows the areas between the peaks and valleys forming the shapes of the hysteresis loops - see *Figure 22.3.3-3*.

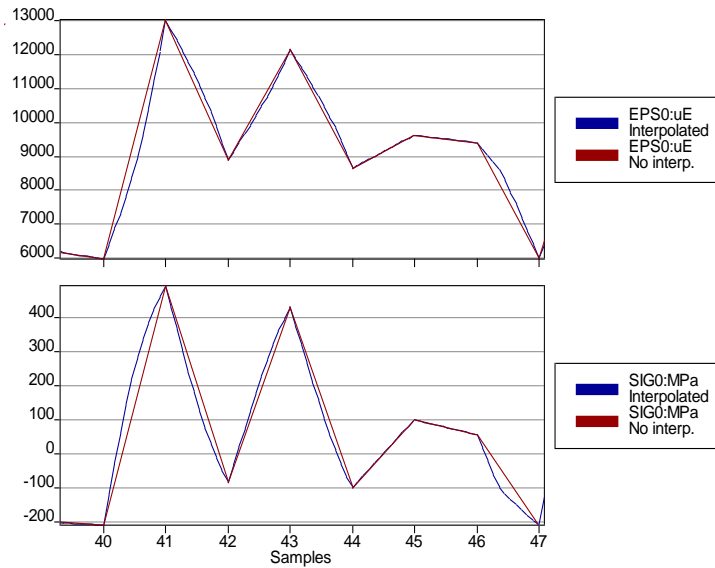


Figure 22.3.3-3

For non-proportional constant amplitude loading sample hysteresis loops look like *Figure 22.3.3-4*. These are similar to those shown in Socie and Marquis (Ref. 22.1).

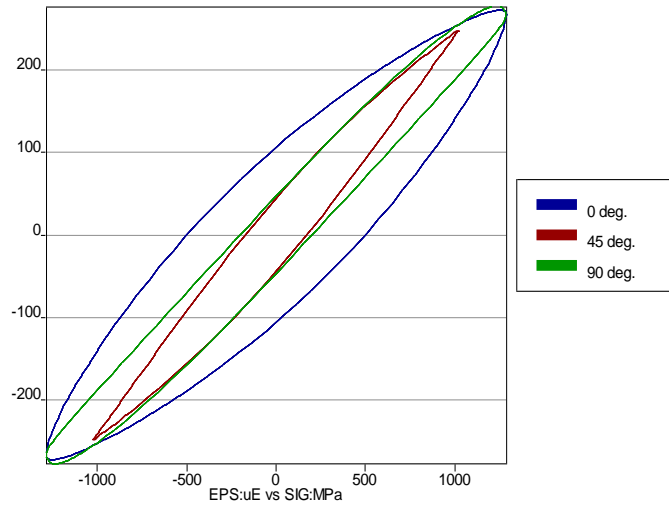
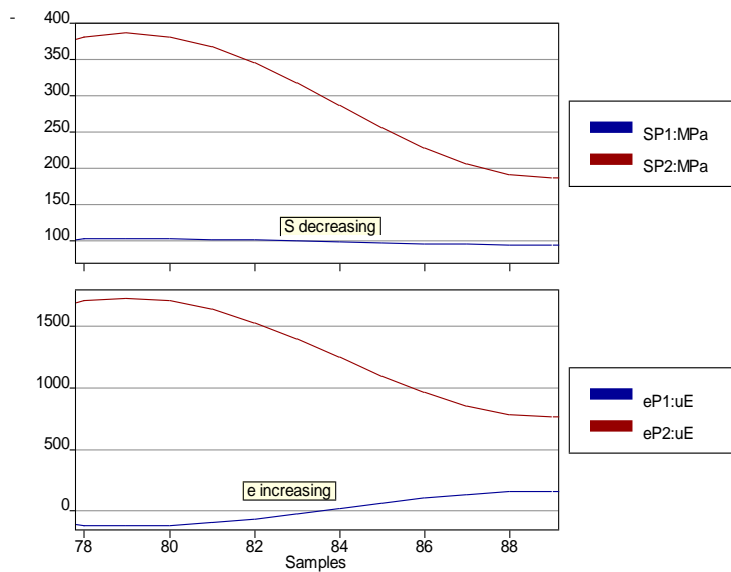


Figure 22.3.3-3

For more complex loading, effects such as backward hysteresis loops can be seen. This occurs where the strain increases on a plane as the stress reduces (or vice versa). An example of a section of loading where this occurs is shown below in *Figure 22.3.3-4*, (eP1 and SP1 are the lower plots in each segment). The plots are of elastic principals which, in this example, are not changing direction. The first principal (ep1 and sP1) exhibit backward hysteresis behaviour due to the overriding effect of the Poisson's strains. SP2 is much bigger than SP1 in the displayed area.



NOTE:
eP1 and SP1 are the lower plots in each segment.

Figure 22.3.3-4

Cross plotting the elastic stress and strains and the elastic-plastic stress and strains in the direction of eP1 and SP1 displays the backward hysteresis loops - see *Figure 22.3.3-5*.

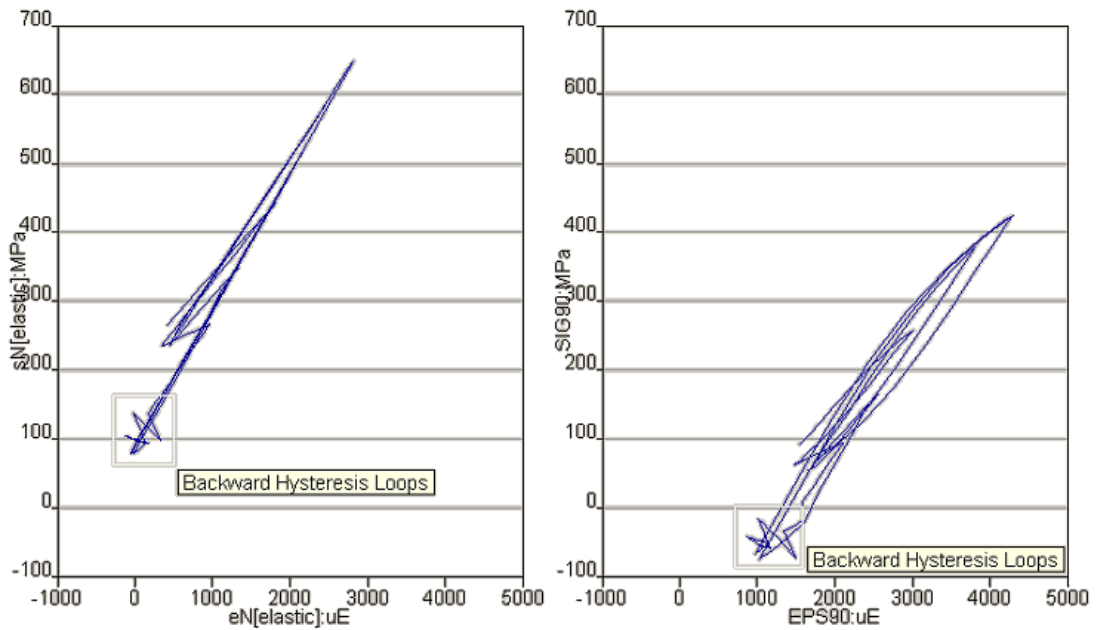


Figure 22.3.3-5

22.4 Gauge and IC definition format.

Definition files allow analysis configurations for influence coefficients and gauges to be saved to disk and reloaded at a later time. The files are text-based and consist of a list of single line commands.

22.4.1 LOAD

This defines a load to be considered in the influence coefficients matrix. This is in the format:

```
LOAD, load_case_id, load_number, descr, units
```

where:

load_case_id	Specifies the input FEA load case. The <i>fe-safe</i> dataset number.
load_number	This is a unique ID for the load to be used for the output <i>.inf</i> file.
descr	Textual description of load.
Units	Textual description of units.

e.g.: `LOAD 1, 1, Vertical Reaction, MPa`

Note: If the definition is for gauge outputs rather than an influence coefficient matrix then LOAD commands are ignored.

22.4.2 SINGLE

This defines a single strain gauge output. This is defined in the format:

```
SINGLE el_num, surface, angle
```

where:

el_num Is the element number for the gauge.

Surface Defines the surface. This is either:

Z1 top surface (1st surface).

Z2 bottom surface (2nd surface).

BOTH top and bottom.

ONE There is only one surface but Z1 and Z2 are written as the same.

a number defining the surface (for Abaqus).

angle Angle to x-axis, y is 90°. If omitted assume to be 0°.

e.g.: SINGLE 1007,BOTH, 25

22.4.3 ROSETTE

This defines a 45° strain gauge rosette. This is defined in the format:

```
ROSETTE el_num, surface, angle
```

where the parameters are the same as those for the SINGLE command.

e.g.: ROSETTE 2006, Z1

22.4.4 STRESS

This defines a stress tensor output. This is defined in the format:

```
STRESS el_num, surface
```

where the parameters are the same as those for the SINGLE command.

e.g.: STRESS 2006, Z1

22.4.5 SINGLES

This defines a single stress output. This is defined in the format:

```
SINGLES el_num, surface, angle
```

where the parameters are the same as those for the SINGLE command.

e.g.: SINGLES 2006, z1, 67

22.5 Hotspot Detection

Hotspots are areas in the model that have relatively high or low results, depending on a user's selection, in relation to the surrounding elements. An alternative to picking these areas by hand using a post-processor, is to use a required formula in conjunction with software which can read in mesh topology data, such as *fe-safe*.

In *fe-safe* a hotspot is defined as a group of locally connected (i.e. adjacent) nodes or elements that satisfy a defined criterion. If the selected variable is less than or greater than the specified criterion, then those results are stored and sorted into locally connected hotspots. These hotspots are then arranged in a "from worst" order meaning furthest from the defined criterion, and can be used for additional analysis, diagnostics and/or reporting.

This capability can be used with solid and shell models and can identify hotspots both on the surface and inside a component. In the current version of *fe-safe* it is limited to nodal-averaged and element-nodal data.

22.5.1 Finding Hotspots

Following an analysis in *fe-safe*, right-click in the **Current FE Models** window and select **Find Hotspots...** to open the **Find Hotspots** dialogue:

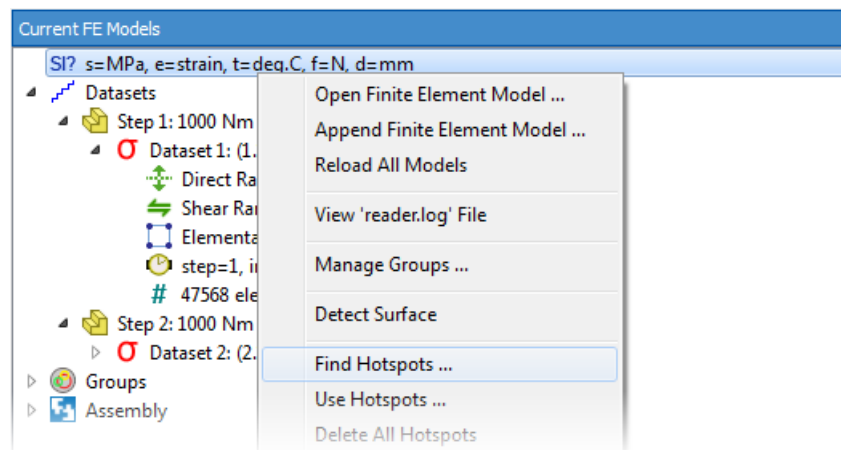


Figure 22.5.1-1

The **Find Hotspots** dialogue can then be used to define the criterion for detecting hotspots:

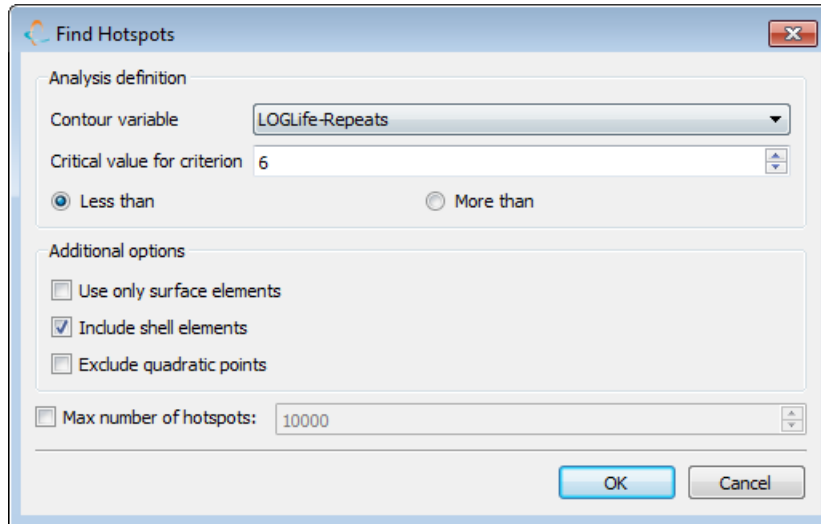


Figure 22.5.1-2 Find Hotspots dialogue

Contour variable

The required variable can be selected from the drop-down list. This list will include all contour variables that were requested for exporting, see Section 22.1.1.

Critical value for criterion

Numerical value for the criterion used to determine hotspots. This will be referencing data in the selected contour variable.

Less than / More than

Specifies hotspots as groups of connected elements with values less than or more than the Critical value for criterion.

Additional Options can be used to further control the hotspot detection - to **Use only surface elements**, **Include shell elements** and **Exclude quadratic points** for higher-order elements.

Max number of hotspots

Specifies maximum number of hotspot areas which will be found, in the "from worst" order. The default value 100, the maximum number is currently limited to 10 000.

As hotspots are detected, a new item **Hotspots** is added to the Current FE Models window:

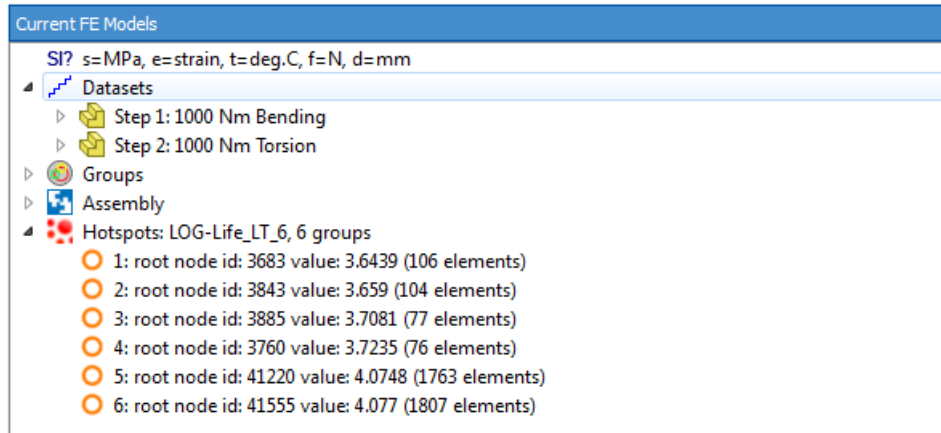


Figure 22.5.1-3

The window will be updated after all hotspots have been detected. Names of detected hotspots are based on the name of the contour variable used, `_LT_` or `_GR_` for less than and greater than and the value of the criterion used. Short information about each hotspot group is revealed by clicking the `[+]` symbol, which includes:

- id of the hotspot group,
- global id of the worst node in the group,
- value at the worst node and number of all nodes in the group

22.5.2 Using Hotspots

Following Hotspot detection, right-click in the **Current FE Models** window and select **Use Hotspots...** to open the **Use Hotspots** dialogue:

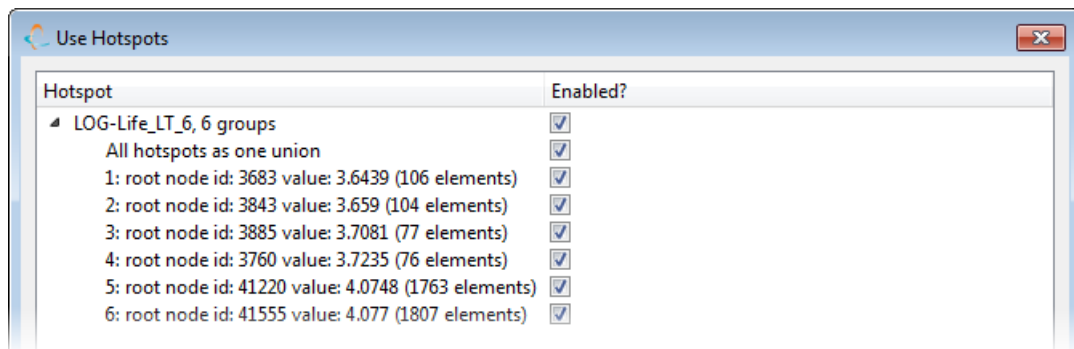


Figure 22.5.2-1 Use Hotspots dialogue

The **Use Hotspots** dialogue can then be used to select hotspots to be converted to element groups, by default all detected hotspots will be selected. A Union group containing all selected hotspots can also be created. Clicking **OK** will create element groups from selected hotspots, which can then be used for a subsequent fatigue analysis configuration:

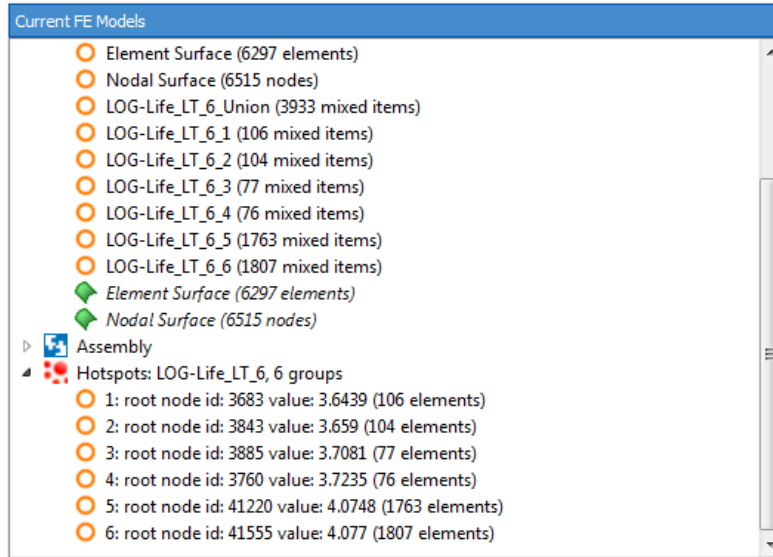


Figure 22.5.2-2

22.5.3 Deleting Hotspots

Detected hotspots are saved in a file FESAFE.hspots in the user results directory. To delete all information about detected hotspots, and remove the FESAFE.hspots file, right-click in the **Current FE Models** window and select **Delete All Hotspots**. A confirmation message will be displayed before deleting all hotspots information.

22.6 Saving FE model datasets to the FER file

Individual datasets from the original model can be saved (unanalysed) to the *fe-safe* intermediate file format, *.fer, (see Appendix E, section 205.8.5) as follows:

- highlight a dataset in the **Current FE Models** window;
- select **File >> FEA Solutions >> Save Loaded FE Models...**;
- change the file type to **Intermediate FE Results format, *.fer**;
- enter a filename to save the file to;
- click **Save**.

The .fer file can now be saved to another output format. For example to save to an OP2 file:

- select **File >> Save FE Fatigue Results as...**;
- set the results file to the .fer file just created as described above;
- set the output file to have the desired extension - e.g. myResults.op2.
- click **Save**.

22.7 Saving FE model datasets to an ASCII tensor file

All datasets from the original model can be saved (unanalysed) to an ASCII tensor file (see Appendix G, section 207.10) as follows:

- select **File >> FEA Solutions >> Save Loaded FE Models...**;
- change the file type to **Text, *.txt**;
- enter a filename to save the file to;
- click **Save**.

The ASCII tensor file can subsequently be re-imported as an FE model:

- select **File >> FEA Solutions >> Open FE Models**;
- select the `.txt` file created as described above.

22.8 References

22.1 Socie D F and Marquis G B

Multiaxial Fatigue

SAE International, 2000, pp 286.

23 Macros and batch mode operation

23.1 Macros in *fe-safe*

23.1.1 The auto-generated macro script

During an *fe-safe* session, analysis and manipulation functions are automatically recorded in a default macro script, called `current.macro`. This script can be viewed by selecting **View Macro from Current Session** from the **Tools** menu.

The auto-generated macro script contains a macro command line for each analysis or manipulation function performed during an *fe-safe* session. A comment line showing the format of the macro command line precedes each macro command line.

23.1.2 Macro command format

Each command in the macro file is of the format:

```
<token> <arg_1>, <arg_2>, ....., <arg_n>
```

where

<token> is the token for the required function,

for example:

the token `RAINFL` performs the **Rainflow and Cycle Exceedence** function;

the token `RF2LDF` performs the **Convert Rainflow to LDF** function.

and

<arg_1>, <arg_2>,, <arg_n> is a comma-separated list of arguments required by the function.

23.1.3 Creating a user-defined macro

A user-defined macro can be created using a text editor. Macro files normally have a `.macro` extension.

23.1.4 Checking the syntax of a macro

When running a macro, *fe-safe* will normally stop processing if a syntax error is encountered. To check for syntax errors before running a potentially long macro, select **Check Macro Syntax...** from the **Tools** menu. This displays a file selection dialogue. Select the required macro file and click **Open** to check the macro for syntax errors.

23.1.5 Running a macro

To run a macro, select **Play Macro ...** from the **Tools** menu. This displays a file selection dialogue. Select the required macro file and click **Open** to run the macro.

23.1.6 Executing analysis and manipulation operations using macros

To obtain the correct token and arguments for a function, run a similar analysis using the GUI, then view the macro command line and associated comment line in the auto-generated macro script.

Example 1:

The following macro script was saved when the **Rainflow and Cycle Exceedance** function was used on the file `whitelon.dac` to produce a rainflow histogram, then the **Convert Rainflow to LDF** function was used to produce an LDF file from the resulting rainflow:

```

Line 1  # Auto generated log of session
Line 2
Line 3  # RAINFL <input_file>, <input_chan>, <start_time=start>, <end_time=end>,
      <numBins=64>, <gate=0>, <min_limit=yMin>, <max_limit=yMax>, <Range-Mean=Y>,
      <Range-Only=Y>, <Range-Exceedance=Y>, <Cycle-Density=Y>, <output_file>
Line 4  RAINFL C:\my_data\WHITELON.DAC, 1, , ,64, 0.1, , ,Y, Y, N, N,
      c:\ResultsArchive\Rainflow_RM_[WHITELON.DAC]_created[07.59.37_22-12-02].cyh
Line 5
Line 6  # RF2LDF <input_file>, <input_chan>, <dataset=1>, <bintype=upper>, <output_file>
Line 7  RF2LDF c:\ResultsArchive\Rainflow_RM_[WHITELON.DAC]_created[07.59.37_22-12-
      02].cyh, 1, 1, Upper,
      c:\ResultsArchive\FromRainflow_[Rainflow_RM_[WHITELON.DAC]_created[07.59.37_22-
      12-02].cyh]_created[07.59.47_22-12-02].ldf
    
```

Note that lines beginning with a # character are comment lines. Automatically generated comment lines indicate the required syntax for that particular macro function.

The easiest way to run functions from a macro is to edit a recorded script, and save it to a new file.

Example 2:

The script in Example 1 was saved and edited so that the same functions would be processed, but on different input files. This time the **Rainflow and Cycle Exceedance** function was used on the file `sinlong.dac` to produce a rainflow histogram, `sinlon_rainflow_01.cyh`. This file was then converted to an LDF file called `ldf_from_sinlon_rainflow_01.ldf` using the **Convert Rainflow to LDF** function.

```

Line 1  # Modified macro file
Line 2
Line 3  # RAINFL <input_file>, <input_chan>, <start_time=start>, <end_time=end>,
      <numBins=64>, <gate=0>, <min_limit=yMin>, <max_limit=yMax>, <Range-Mean=Y>,
      <Range-Only=Y>, <Range-Exceedance=Y>, <Cycle-Density=Y>, <output_file>
Line 4  RAINFL C:\my_data\SINLONG.DAC, 1, , ,64, 0.1, , ,Y, Y, N, N,
      c:\my_data\sinlon_rainflow_01.cyh
Line 5
Line 6  # RF2LDF <input_file>, <input_chan>, <dataset=1>, <bintype=upper>, <output_file>
Line 7  RF2LDF c:\my_data\sinlon_rainflow_01.cyh, 1, 1, Upper,
      c:\my_data\ldf_from_sinlon_rainflow_01.ldf
    
```

Support for macros in the command line is also available using the `macro=` command line option – see section 23.2 below

23.1.7 Executing FEA fatigue analyses from within a macro

To execute an FEA fatigue analysis from within a macro use the token:

```
fe-safe
```

Following the token, comma-delimited arguments are entered as described in section 23.2, below. The parameters supported within macros are: `j=`, `v=`, `b=`, `o=`, `log=`, `<kwd>=`, `material=` and `mode=`. If values of arguments contain any spaces they should be surrounded by double quotes e.g. `macro="c:\My Documents\test2.fil"`. File references should include a full path, on Windows the path should include the drive letter. See **Running *fe-safe* from the command line** below for examples.

23.1.8 Executing Pre-scanning of a FEA Model from within a macro

To pre-scan an FEA Model from within a macro use the token:

```
pre-scan
```

Following the token, commands and corresponding arguments and values are entered as described in section 23.4, below. The commands supported within macros are: `files`, `position`, `select`, `deselect`, `open`, `append`, and `delete`. A `pre-scan` token cannot be used in the same line with any other token.

Pre-scan commands and arguments can be entered in separate lines, in form of token followed by command, or can be entered all in one line beginning with the token and followed by a comma-separated list of commands (their arguments separated by spaces).

Pre-scanning in a macro represents a method to extract datasets from the source FE model which is described in section 5.

23.1.9 Defining element or node groups from within a macro

To open a user defined group from within a macro use the token:

```
groups
```

Following the token, commands and corresponding arguments and values are entered as described in section 23.5, below. The commands supported within macros are: `load`, `save`, and `list`. A `group` token cannot be used in the same line with any other token.

Group commands and arguments can be entered in separate lines, in form of token followed by command, or can be entered all in one line beginning with the token and followed by a comma-separated list of commands (their arguments separated by spaces).

Defining element or node groups in a macro represents a method of Managing groups used for FEA fatigue analysis which is described in section 5.

23.1.10 Combining pre-scanning, user defined groups, and FEA fatigue analysis in a macro

An example of combining pre-scanning, user defined groups, and FEA fatigue analysis in a macro is shown below:

```
pre-scan files "c:\files\keyhole_01.fil" "c:\My Files\keyhole.op2"
pre-scan position centroidal
pre-scan deselect detect-surface
pre-scan select all stress, deselect step 1 inc 1
pre-scan open selected
groups load "c:\groups\group.csv"
groups list select "c:\groups\GROUP*"
groups list deselect "c:\groups\GROUP3"
fe-safe b=c:\files\analysis.stlx
```

Such macro can then be run from a command line as follows:

```
fe-safe_cl macro="c:/My Documents/combining.macros"
```

23.1.11 Modifying settings within a macro

Any setting can be changed in a macro, the syntax for this is:

[path.to.setting] = new_value

The path to a setting is the hierarchy used in the settings files. To aim this, where settings can be changed within the UI, tools tips display the settings path to use in a macro e.g. for the settings to read strain datasets while performing a full read is *[project.model.extract strains]* as shown in figure 23.1.11.1

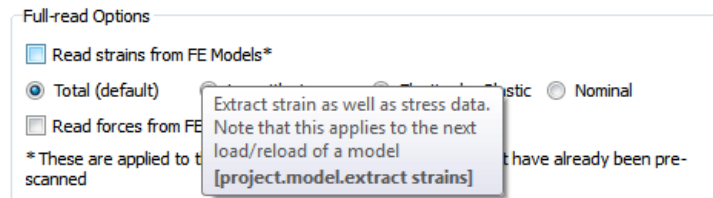


Figure 23.1.11.1

Where a setting path is unambiguous part or the entire path prefix can be ignored e.g. as 'extract strains' is unique *[model.extract strains]* or *[extract strains]* can be used in place of *[project.model.extract strains]*.

A secondary way to reduce duplication in a macro and increase readability is to use the setting *with* syntax where a setting name prefixed with *.* will be relative to the last defined *with* settings path e.g. the following:

```
[job.exports.plots.Haigh] = true
[job.exports.plots.Smith] = true
[job.exports.plots.principals] = false
[job.exports.plots.damage] = true
```

Can be replaced with:

```
with [job.exports.plots]
.[Haigh] = true
.[Smith] = true
.[principals] = false
.[damage] = true
```

The setting type determines what values are allowed, in all cases and white-space after the equals symbol (=) and at the end of the line is ignored. Where the value includes spaces double quotes can be used to clarify the value.

Type	Expected Values	Example
Boolean	1, true, yes or 0, false, no	[add surface groups] = true
Integer	32-bit integer, digits only, no separators	[nastran.data position] = 256
Real	IEEE 754-2008 binary32	[stress units.scale] = 1e6
String	Any characters	[csv.separator character] = tab
File	A valid file path (it need not exist)	[source file] = c:\myfile.odb
Directory	A valid directory (it need not exist)	[generated results directory] = c:\tmp
Enumerator	The name of the enumerator value or its numeric index	[history indexing] = one

For string, file and directory setting types the value can contain dynamically resolved tokens:

- Environment variables using \${NAME} syntax e.g. [source file] = \${TEMP}/myfile.odb
- The current macro directory can be referenced with <%macro_dir> e.g. [source file] = <%macro_dir>/myfile.odb
- Paths are auto resolved to the correct / or \, any drive is stripped on Linux and on Windows a missing drive is replaced with the setting [UNIX drive] (which defaults to c:)

For arrays of settings such as for groups and materials, the settings can be accessed using (number) e.g. the first groups algorithm can be set using [groups(1).algorithm] = "WeldLife"

23.2 Running *fe-safe* from the command line

Most *fe-safe* analysis capability is available from the command line. Command-line parameters are available to define an analysis. Command-line options are available to control particular behaviour of the program. Command-line instructions are entered in a terminal or console window in the conventional way. Any messages generated are written to the console window.

Note: the examples below assume that the *fe-safe* application is run by typing `fe-safe_cl` on the command line.

On Windows platforms this is found in the *fe-safe* installation exe sub-directory.

On Linux platforms this there is a script in the base *fe-safe* installation directory called `fe-safe_cl` – see section 3.

On Windows platforms, if the main *fe-safe* executable `fe-safe.exe` is used for macro or batch processing instead of `fe-safe_cl.exe`, a dialogue pops up showing the command being executed. Any messages generated are displayed in the pop-up console window. This is a legacy option and does not support all available features.

23.2.1 Command-line parameters

IMPORTANT NOTE – Supporting legacy FEA Fatigue definition file formats.

From version 6.0-00 onwards

The *fe-safe* Project Definition (*.stlx) file has replaced the previous FEA Fatigue definition file (*.kwd) format – see Appendix E. Henceforth the KWD file format will be supported as a legacy format.

New users should always use the STLX file in command-line analyses. Existing users can still use KWD files produced in previous versions of *fe-safe* or generated by scripts in command line analyses.

Existing keywords from the Table of Keyword by type in Appendix E will continue to be useful in combination with STLX files as optional command line parameters. See supported optional parameters below.

Each command-line parameter that has a value is of the format `parameter=value`. If `value` contains any spaces it should be surrounded by double quotes e.g. `macro="c:\My Documents\My Macro.macro"`.

File references should include the full path. On Windows the path should include the drive letter, e.g.:

```
C:\data\test_models\model_01.fil or "C:\My Documents\test_models\model_01.fil"
```

Note: While **macros run by executables *fe-safe* and *fe-safe_cl* require commas between parameters (see Section 23.1.2)**, the command line does not need them (though it is unaffected by them).

Command-line parameters fall into 2 categories: process commands and optional parameters. The supported process commands are:

Process Command Parameter	Command
<code>j=<modelfile></code>	Open the FE model <model>, multiple <code>j=<modelfile></code> parameters cause appending of one FE model to another. <code>j= refresh</code> means the FE models referenced in the Project Definition will be re-loaded
<code>v=<deffile></code>	Perform a Verity (TM) analysis using the welds in definition file <deffile>.
<code>b=<projdeffile></code>	Perform a fatigue analysis defined by a Project Definition file <projdeffile>. Alternatively <code>b=run</code> may be used to run the project in its current state. Legacy support allows for referencing legacy keyword (*.kwd) files with this command parameter
<code>macro=<macrofile></code>	Run the macro file <macrofile>.

If a macro (`macro=`), load FE model (`j=`), a Verity analysis (`v=`) or fatigue analysis (`b=`) are specified, the command(s) will be processed. A macro command cannot be run with any other command line parameters; all other parameters will be ignored except `-project`, `-h` and `-v` (see below).

The other three commands may be specified in any order, but will always be executed in the order of: loading the FE model, performing a Verity analysis then performing a fatigue analysis.

If no processes are specified, *fe-safe* will display the help screen.

Referencing a project definition file (*.stlx) using the fatigue analysis parameter (`b=`) will cause the loaded settings to overwrite the current project and job settings. As the file is opened, any paths defined in the file are interpreted assuming the following path hierarchy:

- Absolute path (as defined in the `.stlx` file)

- Location of the `.stlx` file
- Current project path

Any paths defined in the referenced `.ldf` file will also be interpreted in a similar way and the loading definition will then be saved as the new `current.ldf` (for the current job).

Legacy Keyword format and Stripped Keyword (`*.kwd` and `*.xkwd`) files can also be used as the value of the fatigue analysis parameter (`b=`) from analyses completed in an earlier version of *fe-safe*.

The supported optional parameters are:

Optional Parameter	Description
[<setting>]=<value>	Overrides setting <setting> with <value>
<kwd>=<value>	Overrides the setting having legacy keyword <kwd> with <value>
-h	Displays usage information for fe-safe_cl.
-import_project <ProjectArc>	Imports the project archive into the current project directory, it will overwrite any existing files. <ProjectArc> can be relative to the current working directory.
-l <location>	Looks for licence at <location>.
log=<logfile>	Overrides analysis output log to <logfile>
-macro_check=<checktype>	Checks that the macro can be successfully run, rather than executing it
-macro_exit=<condition>	Sets the condition for stopping execution (or checking) of a macro
material=<mode>	Forces material data to 'refresh' from database, use 'cached' from .stlx file or 'auto' decide (default)
mode=<mode>	Extra model file processing of 'rotate', 'geometry', 'surface' or 'psd'
o=<outputfile>	Overrides analysis output file to <outputfile>
-overwrite_project	When importing a project archive with -project option, an existing files will be overwritten.
-project <ProjectArc>	Overrides project directory to <ProjectArc> stripped of its file suffix, and imports the project archive into the new project directory. The import will abort if there are existing files. <ProjectArc> can be relative to the current working directory.
-project <ProjectDir>	Overrides project directory to <ProjectDir>
-s	Silent, minimal output will be displayed
-timeout <minutes>	Overrides the default licence timeout of 15 minutes to <minutes>
-v [all]	Displays the application version
-w	Wait for user input before exiting

-project <ProjectDir>

The current location of the project directory is overridden, see section 5 for more details.

```
[<setting>]=<value>
```

Changing a setting value can be done via the [`<setting>`]= command, however there are a number of restrictions compared to changing a setting from within a macro:

- Setting names containing spaces must be enclosed in double quotes
- Accessing setting arrays (e.g. the groups) can only be done via index and not via a name
- Spaces prior or following the = are not allowed
- Values with spaces in must be enclosed in double quotes
- Using a comma, double quotes or any platform special characters in a value is not possible

```
<kwd>=<value>
```

Changing a keyword value can be done via the `<kwd>=` command, group keywords are set using the suffix `.n` for group `n` e.g. `MyKeyword.3=MyValue` will set keyword 'MyKeyword' in group 3 to 'MyValue'. If a keyword file is loaded, any keyword set on the command line takes precedence.

```
-l <location>
```

This can be used to redirect the licence server location for the session. A hostname (or IP address) should be passed through, with an optional port number. (e.g. `MYHOSTNAME@7171`).

```
log=<logfile>
```

During an analysis with the `b=` command, the log file can be redirected with the `log=` command rather than its name being paired with that of the output file.

```
-macro_check=<checktype>
```

This option can be used to check the macro for several types of errors. Running a check will not change project settings or create any files. The following checks are supported:

- Check for syntax errors using `-macro_check=syntax`. These errors include unknown commands and formatting errors. By default a syntax error will cause the macro check to stop; see `-macro_exit`.
- Check semantic errors using `-macro_check=semantics`. This includes checking that command arguments don't conflict, for the existence of input files and that output file names are viable. Note that for complex commands some of these types of errors will only be detected when executing the command. Checking for semantic errors will also check for syntax errors.
- Check licensing errors using `-macro_check=licence`. This checks for basic licensing requirements. These do not include add-ons used in a fatigue analysis, as the settings are not changed and so cannot be used to determine the state when all commands would have been run. Checking for licensing errors will also check for semantic and syntax errors.

```
-macro_exit=<exitcondition>
```

This option can be used to change the condition under which a macro run (or check) is stopped:

- To continue to the end of a macro regardless of any errors, use `-macro_exit=macro_end`.
- To stop running a macro when a syntax error is encountered, use `-macro_exit=syntax_error`. This is the default.
- To stop running a macro when a semantic error is encountered, use `-macro_exit=semantic_error`. This will also stop if a syntax error is encountered.
- To stop running a macro when a licensing error is encountered, use `-macro_exit=licence_error`. This will also stop if a semantic or syntax error is encountered.
- To stop running a macro when macro command fails, use `-macro_exit=execute_error`. This will also stop if a licensing, semantic or syntax error is encountered.

```
material=<mode>
```

Forces the material data:

- to be reloaded from the relevant database (`material=refresh`) – any material keywords set on the command line will be ignored;
- to use data from the `.stlx` file (`material=cached`) – required properties can be modified from the command line;
- to be reloaded from the relevant database unless material keywords are set on the command line, in which case data from the `.stlx` file will be used (`material=auto`). This is the default option.

`mode=<mode>`

Specifies that the model(s) being opened with the `j=` command should be loaded:

- via the Rotate module (`mode=rotate`);
- with the geometry information of the first model (`mode=geometry`);
- with geometry and surface-detection (`mode=surface`);
- as frequency-domain data (`mode=psd`).

`o=<outputfile>`

When performing the analysis with the `b=` command the output file name specified in the keyword file is overridden.

This will also override the analysis log file name unless the `log=` command is used.

`-s`

Stops most messages from being echoed to the console, though some errors will still get displayed.

`-v`

Displays the version of *fe-safe* and exits.

`-w`

Causes *fe-safe* to wait for the <Enter> key to be pressed before exiting.

`-timeout <minutes>`

The default command-line *fe-safe* licence time-out is 15 minutes; this option can be used to change the time-out to <minutes> minutes. This changes the timeout for all future runs.

23.2.2 Command line examples

Example 1:

The macro script in Example 2, above, was run from a windows command prompt, using:

```
fe-safe_cl macro=c:\my_macros\macro_02.macro
```

The following is displayed in the console window:

```
Playing Macro file c:\my_macros\macro_02.macro

Processing Line 3: RAINFL C:\my_data\SINLONG.DAC, 1, , ,64, 0.1, , ,Y, Y, N, N,
c:\my_data\sinlon_rainflow_01.cyh
Processing Line 6: RF2LDF c:\my_data\sinlon_rainflow_01.cyh, 1, 1, Upper,
c:\my_data\ldf_from_sinlon_rainflow_01.ldf
Play Macro completed
```

Example 2:

The following command is entered in a Linux console window at the shell prompt:

```
fe-safe_cl j=/data/test1.fil j=/data/test2.fil b=/data/test.stlx o=/data/res.csv
```

This loads the project definition file `/data/test.stlx`, then loads FE analysis results from the two FIL model files, `test1.fil` and `test2.fil`. Fatigue analysis results are written to the file `/data/res.csv`. The program exits when the analysis is complete.

Example 3:

The following command is entered in a Linux console window at the shell prompt:

```
fe-safe_cl j=refresh j=/data/test2.fil b=/data/test.stlx o=/data/res.csv ELASMOD.2=200000
```

This reloads the FE analysis results referenced in the project definition file `/data/test.stlx`, applies the settings in the project definition except that the elastic modulus for element group 2 is modified to 200000 using the ELASMOD keyword.

23.3 Batch Operation

A batch sequence of operations can be configured using either a macro script or a conventional batch script.

23.3.1 Using a macro (All platforms)

A macro can be used to perform a sequence of operations, as described in section 23.1, above. The macro can be launched either from the GUI, (by selecting **Play Macro...** from the **Tools** menu - see 23.1.4), or by running the macro from the command line, as in Example 1, above.

The advantage of using a macro over using a conventional batch script is that *fe-safe* does not need to be shut down after each process and then re-launched.

23.3.2 Using a conventional batch script (Linux)

A conventional batch script can be created containing a sequence of command line operations.

For example a file called `my_batch_linux.sh` may contain the following lines:

```
fe-safe_cl j=/data/test1.fil j=/data/test2.fil b=/data/test.stlx o=/data/res_a.csv
fe-safe_cl j=/data/test3.fil j=/data/test4.fil b=/data/test.stlx o=/data/res_b.csv
fe-safe_cl j=/data/test5.fil j=/data/test6.fil b=/data/test.stlx o=/data/res_c.csv
fe-safe_cl j=/data/test7.fil j=/data/test8.fil b=/data/test.stlx o=/data/res_d.csv
```

where `fe-safe_cl` is the `fe-safe_cl.exe` executable, or an alias to the script `fe-safe_cl` (see 23.2, above).

The script `my_batch_linux.sh` can be run from the command line by typing:

```
./my_batch_linux.sh
```

As the script is executed, each command line launches an instance of *fe-safe*, executes the analysis then shuts down *fe-safe* before the next line executes. So, in this example *fe-safe* would be launched and shut down four times.

23.3.3 Using a conventional batch script (Windows)

The technique described in 23.3.2 above does not work in Windows, because the second command line executes without waiting for the first one to finish. This could cause data to become corrupted.

However, including:

```
start /wait
```

at the beginning of each line in the batch file ensures that the current command completes before executing the command in the next line.

So, for example, a file called `my_batch_win.bat` may contain the following lines:

```
start /wait fe-safe_cl j=/data/test1.fil j=/data/test2.fil b=/data/test.stlx o=/data/res_a.csv
start /wait fe-safe_cl j=/data/test3.fil j=/data/test4.fil b=/data/test.stlx o=/data/res_b.csv
start /wait fe-safe_cl j=/data/test5.fil j=/data/test6.fil b=/data/test.stlx o=/data/res_c.csv
start /wait fe-safe_cl j=/data/test7.fil j=/data/test8.fil b=/data/test.stlx o=/data/res_d.csv
```

where `fe-safe_cl` is the `fe-safe_cl.exe` executable.

The script `my_batch_win.bat` can be run from the command line by typing:

```
my_batch_win.bat
```

As the script is executed, each command line launches an instance of *fe-safe*, executes the analysis then shuts down. So, in this example *fe-safe* would be launched four times.

23.4 Macro Commands for Pre-scanning

A macro file can contain `pre-scan` tokens indicating which FEA model files to pre-scan and what data to open in order to use the FEA results later in *fe-safe* analysis.

To see the correct syntax of commands and arguments for a pre-scan, open a similar FEA Model and use the pre-scanning function in the GUI, then view the auto-generated macro script (`current_macro`). See section 5 for importing datasets from FE models in the *fe-safe* GUI.

Each command parameter that has a value is of the format `parameter value` with a space in between. File references should include either a full path or a project-relative path, in either case the path must always be enclosed in double quotes. On Windows the path should include the drive letter, e.g.:

```
"\data\test_models\model_01.fil" or "C:\data\test_models\model_01.fil"
```

OR

```
"c:\My Documents\projects\project_1\model_01.fil"
```

OR if `c:\My Documents\projects\project_1` is the current project

```
".\model_01.fil".
```

23.4.1 Pre-scanning Commands

Macro commands for pre-scanning can be organized into three required steps and a fourth optional step. They are:

Step	Command Type	Command
Files Selection Command		
1	Files	Specifies files to be pre-scanned
1	More	Specifies files to be pre-scanned in 2 nd and subsequent lines
Datasets Selection Commands		
	Selection Commands	
2	Position	Specifies position to read the results from
2	Select	Select datasets to read from the files
2	Deselect	Deselect datasets not to be read (optional)
Read Commands		
3	open selected	Open files and read the selected datasets into <i>fe-safe</i>
3	append selected	Append selected datasets into <i>fe-safe</i> (optional)
Delete Command		
4	Delete	Delete pre-scan files created in the macro (optional)

The pre-scanning commands should be set in order, as follows:

Step 1: Selecting files to pre-scan

All `pre-scan files` commands are read in order, regardless of whether they are set in the same or separate lines. For example:

```
pre-scan files "c:\files\keyhole_01.fil" "c:\My Files\keyhole.op2"
```

OR

```
pre-scan files "c:\files\keyhole_01.fil"
pre-scan more "c:\my files\keyhole.op2"
```

The files will be pre-scanned in the order of `keyhole_01.fil` followed by `keyhole.op2`. If geometry import or surface detection options are requested in the fatigue analysis (using the `mode=` parameter) the required data would be loaded from the first model, if available.

Those commands do not load any data into *fe-safe* immediately – appropriate datasets must next be selected and then opened.

Step 2: Selecting datasets to load

To select all datasets in a file a command `select all` can be used, for example:

```
pre-scan files "c:\my_files\keyhole_01.fil"
pre-scan select all
```

OR

```
pre-scan files "c:\my_files\keyhole_01.fil", select all
```

Appropriate datasets can then be deselected, as required, for example:

```
pre-scan files "c:\my_files\keyhole_01.fil"
pre-scan select all
pre-scan deselect step last
```

OR

```
pre-scan files "c:\my_files\keyhole_01.fil", select all, deselect step last
```

The selected datasets are all of the steps in the pre-scanned file, less the last step. The parameter `step` and the value `last` are part of a list of parameters and values that can be used with the `select` or `deselect` commands:

Parameter	Value
<code>step</code>	Step number <code>n</code> , step name, <code>first</code> , <code>last</code> or <code>all</code>
<code>inc</code>	Increment number <code>n</code> , <code>first</code> , <code>last</code> or <code>all</code>
<code>time</code>	Time <code>t</code> , <code>first</code> , <code>last</code> or <code>all</code>
<code>ds</code>	Dataset number <code>n</code> or dataset name, <code>first</code> , <code>last</code> or <code>all</code>
<code>source</code>	A file name <code>filename</code> of a file pre-scanned using the pre-scan file command including the full path, e.g: "c:\my_files*.fil", if more than one. Alternatively use <code>first</code> , <code>last</code> or <code>all</code>
<code>type</code>	Result type: <code>all</code> , <code>stress</code> , <code>strain</code> , <code>force</code> , <code>temperature</code> , <code>history</code> , <code>misc</code> and/or <code>custom(CustomName)</code>

N

Note: Number `n` can be an integer `n`, or a range `n-m`, e.g.: 2-25 or 1-6(2).

Names `name` or `filename` are case sensitive and are set as a text strings within double quotes with optional `*` wildcards, e.g.: `"*heat"`.

Time `t` can be a real and must include the decimal point, even for 0, e.g.: 0.0

Custom variable `CustomName` refers to the data type name used in CMF algorithms.

For example:

```
pre-scan select inc last
```

Selects the last increment of every step

```
pre-scan select all, deselect step last strain temperature
```

Selects all datasets, then from the last step deselect all strain and temperature datasets.

```
pre-scan select ds 1-100 stress force
```

Selects any stress or force dataset in the first 100 datasets.

```
pre-scan select step "*Static*" step 1-10 stress
```

Selects any stress dataset in the first 10 steps that also have a step name containing *Static*.

Optionally, `select` and `deselect` commands can be used with 2 special qualifiers, `geometry` and `detect-surface`. These can be used to control geometry-reading and surface-detection in the same way as in the pre-scan dialogue, see section 5.7.2. For example:

```
pre-scan select geometry
```

Reads geometry from the first file

```
pre-scan select detect-surface
```

Detects surfaces when opening datasets

`Position` command is used to control the position the data is read from FEA result files. Available arguments are: `elemental`, `nodal`, `integration`, `centroidal` or `element-and-centroidal`.

For example:

```
pre-scan files "c:\files\keyhole_01.fil" "c:\My Files\keyhole.op2"
```

```
pre-scan position integration
```

```
pre-scan deselect detect-surface
```

```
pre-scan select all stress, deselect step 1 inc 1
```

Selects all datasets except step 1, increment 1, from each of the two files, does not detect surfaces when opening datasets, reads stress results at integration points.

The above commands do not load any data into *fe-safe* immediately – appropriate datasets must next be opened.

Step 3: Reading selected datasets

Open command is used to load the selected datasets from specified files, for example:

```
pre-scan open selected
```

Optional `append` command can be used to append additional datasets to the datasets already opened, for example:

```
pre-scan append selected
```

A combined usage of `files`, `select`, and `open selected` commands:

```
pre-scan files "c:\files\keyhole_01.fil" "c:\My Files\keyhole.op2"  
pre-scan position centroidal  
pre-scan select detect-surface  
pre-scan select all stress strain, deselect step 1 inc 1  
pre-scan open selected
```

Step 4: Deleting pre-scanned information

The `delete` command can be used to delete some or all pre-scan data and accepts wild cards. For example:

```
pre-scan delete "c:\file\a_file.rst" "c:\*.odb"
```

23.5 Macro Commands for Managing Groups

A macro file can contain `groups` tokens which can be used to manage elemental or nodal groups to be used in *fe-safe* analysis.

To see the correct syntax of commands and arguments for a group specification, open a similar group file using the GUI, then view the auto-generated macro script (`current.macro`). See section 5.5.4 for using groups in the *fe-safe* GUI.

Each command line parameter that has a value is of the format `parameter value` with a space in between. File references should include a full path, and are always enclosed in double quotes. On Windows the path should include the drive letter, e.g:

```
"\data\test_models\model_01.fil" or "C:\data\test_models\model_01.fil"
```

OR

```
"c:\My Documents\model_01.fil".
```

23.5.1 Group Commands

The commands supported for managing groups within macros are: `load`, `save`, and `list`.

To load an existing *fe-safe* ASCII (*.csv, *.txt, *.asc) or binary (*.grp) group file, a `groups` token should be used, followed by the `load` command, appropriate `filename` and optional parameter `defaulttype`, to identify whether the group contains nodes or elements. If the group type is not set it will default to elemental. For example:

```
groups load "c:\groups\a_group.csv", defaulttype=elemental
OR
groups load "c:\My Documents\Copy_of_FESAFE.grp", defaulttype=nodal
```

Groups definitions from FEA model files are automatically extracted when such files are loaded into *fe-safe*. To load an FEA model file the following command can be used:

```
fe-safe j=/data/test1.fil
```

To save existing groups, the `groups` token should be used, followed by the `save` command, optional parameter `binary`, to control whether group information is to be saved to a binary (*.grp) file, and the target `filename`. For example:

```
groups save binary "c:\New_Binary_Groups.grp"
OR
groups save "c:\New_Binary_Groups.csv"
```

To select, deselect, and remove groups from the group parameters list, the `groups` token should be used, followed by the `list` command, `select`, `deselect` or `remove` parameters and a `group name`. For more information on managing groups see section 5. Group names are case sensitive and are set as a text strings within double quotes with optional '*' wildcards, e.g.: "GROUP*". An `all` operator can be used instead of a `group name` to manage all existing groups. For example:

```
fe-safe j=/data/test1.fil
groups list select all
```

Selects all loaded groups for the fatigue analysis.

```
fe-safe j=/data/test1.fil
groups list select "GROUP2"
groups list select "GROUP*"
```

Selects a group named GROUP2, followed by all other groups with names starting in GROUP for the fatigue analysis.

Note: Selection order dictates positions of the selected groups in their parameters list. For more information see section 5.

```
fe-safe j=/data/test1.fil
groups list deselect "GROUP3"
```

A group named GROUP3 will not be used to set the fatigue analysis options.

```
groups load "c:\groups\a_group.csv"
groups list remove "*node*"
```

Loads groups from a file `a_group.csv` and then removes all groups containing 'node' in their names.

```
groups load "a_group.csv"
groups list select "GROUP*"
groups list deselect "GROUP3"
```

Loads groups from a file `a_group.csv` and selects all groups with names starting in GROUP, apart from GROUP3, for the fatigue analysis.

To create new groups the `groups` token should be used, followed by the `create` command, the name of the new group and then the `equation` representing the new groups' contents (this is identical to the contents when creating an advanced group, see section 5). Optionally the `create` command can be followed by `,` `type=elemental` or `,` `type=nodal` – this sets the default group type and required when specific items are used to identify the type. For example:

```
groups create "NewGroup" "GroupA AND GroupB"
```

This will create a group called `NewGroup` containing items common to both group `GroupA` and group `GroupB`.

```
groups create "NewGroup2" "1-100, n50", type=elemental
```

This will create a group called `NewGroup2` containing elements 1 to 100 and node 50.

There are a number of special identifiers that can be used to specify mesh based groups:

Identifier name (case-insensitive)	Definition
<code>from_mesh(all)</code>	All elements if default group type is elements, otherwise all nodes
<code>from_mesh(elements)</code>	All elements
<code>from_mesh(nodes)</code>	All nodes
<code>from_mesh(surface)</code>	All surface elements if default group type is elements, otherwise all surface nodes
<code>from_mesh(solids)</code>	All solid elements
<code>from_mesh(shells)</code>	All shell elements
<code>from_mesh(brick)</code>	All brick elements
<code>from_mesh(wedge)</code>	All wedge elements
<code>from_mesh(octahedral)</code>	All octahedral elements
<code>from_mesh(pyramid)</code>	All pyramid elements
<code>from_mesh(tetrahedron)</code>	All tet elements
<code>from_mesh(quadilateral shell)</code>	All quad shell elements
<code>from_mesh(triangular shell)</code>	All triangular shell elements
<code>from_mesh(quadilateral)</code>	All quad elements
<code>from_mesh(triangular)</code>	All triangular elements
<code>from_mesh(beam)</code>	All beam elements
<code>from_mesh(conn)</code>	All connector elements
<code>from_mesh(unsupported)</code>	All unsupported/unclassified elements

23.6 Macro Commands for Managing Projects

There are commands related to changing, exporting and importing projects (or project archives). These commands accept absolute file paths or file paths relative to the macro location.

23.6.1 Changing the current project

```
SwitchToProject <Project Directory>
```

This command changes the current project to **<Project Directory>**. If this is not an existing project, a new project will be created. If for any reason the specified project directory is invalid, e.g. permissions restrictions, the project will not be changed.

23.6.2 Importing a project archive into a new project

```
SwitchToProject <Project Archive> [,<Optional Project Directory>]
```

This command creates a new project; the directory can either be specified using the optional parameter **<Optional Project Directory>** or based on the **<Project Archive>** file path, stripped of all extensions. The archive is then extracted to the new project, which then becomes the current project.

If the new project directory is invalid or would cause any files to be overwritten, the operation is aborted and no change will occur.

23.6.3 Importing project settings into the current project

```
Import <Project Settings.stlx>
```

This command imports the project settings file and replaces settings values for all settings listed in the settings file. If all other settings should be at their defaults, call CLEARKWD first.

23.6.4 Importing a project archive into the current project

```
Import <Project Archive>
```

This command imports the archive into the current project; any existing files will be overwritten.

23.6.5 Exporting project settings for the current project

```
Export [Project] <Project Settings.stlx>
```

This command exports the project to a stlx file. The optional Project can be replaced with User for the user settings.

23.6.6 Exporting the current project to a directory or project archive

```
Export Archive <Project Location>
```

This command exports the current project to **<Project Location>**. The project location is treated as a directory if it is an existing directory or the file path is not an existing file and it does not end in 7z. If this is the case, the export will be treated as a project copy to the directory, otherwise the project location is treated as the file name of a project archive to be created. In either case, if there are files that exist that would be overwritten, the export is aborted – this can be prevented by calling macro command `rm` or `rmdir` to remove any existing file or directory.

There are several categories of project file. By default, all except any external FE models are exported. If there is missing project model data (e.g. no FESAFE.FED), then any external FE models will be selected instead.

Files external to the project that are selected for export will be copied to a location relative to the exported project, e.g. exporting to `c:\Archive\project_01` will cause external files to be copied to `c:\Archive\project_01\external_files` (or one of its subdirectories). The exported project settings will reflect the new relative locations which the external files are now in.

Optionally the categories of project files selected to be exported can be changed using the token names:

Macro token name	Affected project files
Project	The project settings and all miscellaneous project files
Fe_model	The source FE models
Datasets	The loaded datasets
Mesh	The loaded mesh
Groups	The loaded groups
Job	The job settings and all miscellaneous job files
Loading	Any used ldf or hldf file in the project directory
Histories	Any history files used that are in the project directory
External>Loading	Any used ldf or hldf file outside the project directory
External>Histories	Any history files used that are outside the project directory
Results	The results stored in the intermediate results file
Exported>Results	The results exported to the target output file
Export>Prereq	The file required to create the exported results file
Result>Diag	The diagnostic results files created during the analysis

Categories are separated from the **Export** command and each other by commas, e.g.

```
Export Archive c:\temp\project1, Project, Job, Groups
```

23.6.7 Reverting a projects settings to the defaults

While using `SwitchToProject` can be used to create a new project (see 23.6.1) based on the default setting and without any existing data, it is possible to reset just the settings to the defaults by using the command:

```
CLEARKWD
```


24 Nodal mapping of material properties

24.1 Spatially varying mechanical and fatigue properties

In some components the material properties can be different at different locations in the component. Examples are:

- Variations in elastic properties and tensile strength throughout a casting
- Variations in yield strength throughout a forging
- Variations in fatigue properties as a result of surface treatments or heat treatment, e.g. shafts, gears, etc.

These local variations in properties may change the fatigue behaviour of the material at each location.

Using *fe-safe*, these variations can be accounted for through the capabilities of nodal property mapping:

- Material properties can be defined independently for each node on the model using property mapping.
- The property map can include material properties for all or just part of the model, e.g.: a heat treated region of a shaft. If properties for a node are not specifically included in the property map, then the properties of the material that are set in the Group Parameters region (see section 5) will be used based on the group the node is part of.
- A property map does not have to include all material properties – just those that vary spatially. For example, it is possible that only a mechanical property such as UTS is affected. Alternately a fatigue property such as the tabular stress–life endurance curve may be affected. All other properties for the node will come from the material set in the Group Parameters table (see section 5 for details) based on the group the node is part of.
- Any material parameter defined in *fe-safe* can be used in a property map (see section 8 for details). The effect of the mapped property on fatigue results will depend on how each property is used in *fe-safe*, for instance UTS is frequently used to determine surface finish factor. See sections 14 and 15 for fatigue analysis of Elastic and Elastic–Plastic FEA results respectively.
- Temperature–dependent variation with property mapping is comprehensive and powerful:
 - Not all nodes have to use temperature–dependent properties, and those that do can have a different number of temperatures listed. Properties will be interpolated as described in section 8.
 - The nodal properties can be temperature–dependent, even if the main properties for the material are not temperature–dependent. For example, the nodal property map may contain temperature–dependent UTS and nothing else, whereas the properties of the material defined for the group that the node belongs to could be defined only at one temperature (for instance at room temperature). In this example, temperature–dependent UTS will be used, even though the other properties are isothermal.
 - Such variation makes use of existing conventional high temperature fatigue in *fe-safe* (see section 18 for details).

24.2 Nodal property mapping

Nodal property mapping in *fe-safe* allows for a fatigue analysis to take into consideration the spatial variation of physical properties throughout a component. A nodal property definition file for a component is used by *fe-safe* to account for different material properties at each node in a FEA model. Using existing material property definitions in *fe-safe*, relevant material properties (e.g. elastic

coefficients, tensile strength, cyclical stress strain properties, strain–life curves, stress life curves, ..) can be specified at each node in a FEA model. These properties can also be temperature dependent.

If the model contains elemental data, *fe-safe* reads the geometry/mesh information and generates an element/node table to cross-reference the nodal properties. If the model contains nodal data, reading the geometry/mesh information is not necessary. See section 2 for complete analysis process examples including loading application. A fatigue analysis in *fe-safe* using nodal property mapping uses the following logic,

- *fe-safe* checks if nodal properties have been defined for a given node
- if they are defined, nodal property data will take precedence over any corresponding material data
- If no nodal properties have been defined for a node, the properties defined in the **Group Parameters** configuration table are used
- All existing features of *fe-safe* remain unaltered – all loading definition options, including residual stresses, are still available

24.3 Using nodal property mapping in *fe-safe*

To enable nodal property mapping in *fe-safe* select **FEA Fatigue >> Analysis Options...**, **General** tab, and select the **Enable nodal property mapping** option (default).

Once the option is enabled the nodal property definition NPD file (*.npd) described below in section 24.4 can be opened in *fe-safe* using the context-sensitive menu (accessible by using the right-mouse-button) in the **Current FE Models** window. Select **Open Nodal Properties...** from the pop-up menu and use the **Open Nodal Properties** dialogue to navigate to the directory containing the NPD file, select **Open**.

The **Nodal properties** will appear in the tree view in the **Current FE Models** window. Beneath the **Nodal Properties** heading the path and name of the file opened are shown as well as the first node defined with the properties list for that node. Opening an NPD file enables nodal property mapping through a keyword (NODALPROPS=) referencing the fully qualified path and file name of the *.npd file. This can be used to reference nodal property definitions during command line or macro analyses, for more information see section 23.

By selecting **Close Nodal Properties** from the pop-up menu this information is removed from the tree and the analysis keyword is cleared. Note that when a new FE model is opened the nodal properties are automatically cleared.

24.3.2 Using Nodal property mapping with Elemental data types

If the stress data is element–nodal then additional information is required to map from the elemental node number to the global node number. This information is acquired after a pre–scan of the stress data file on the Select **Datasets to Read** dialogue by and selecting **Read geometry from first model** as shown in *Figure 24.3–1*.

If the stress data is nodal then merely opening the nodal properties file is sufficient.

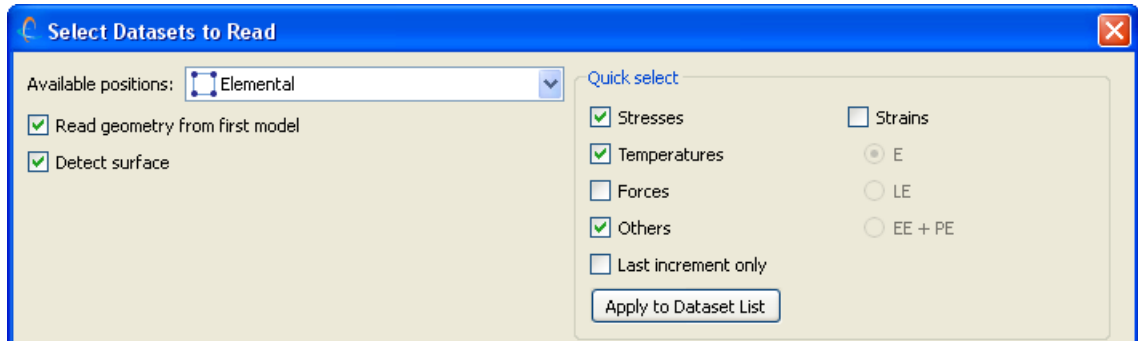


Figure 24.3–1

Once the geometry has been read a summary will appear in the “Open FE Models” model tree as shown in *Figure 24.3–2*.

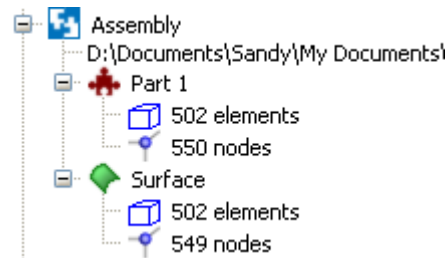


Figure 24.3–2

24.4 Defining nodal properties

Material properties can be defined on a node–by–node basis for all or part of a model, and any material property can be defined, including the fatigue algorithm (see section 8 for material properties).

Nodal material properties are imported from an ASCII “nodal property definition” (*.npd) file; the NPD file is based on the existing syntax of the *fe-safe* database.

24.4.1 Overview of the nodal property definition NPD file (*.npd)

The NPD file is an ASCII database file used to specify nodal properties. Its format is based on the existing syntax of the *fe-safe* material database (*.dbase) and (*.template) files, which are described in section 8 and Appendix E. The supported material properties are defined in section 8.5.

The syntax of an NPD file consists of the following three items:

- Comment lines starting with a hash, #, character.
- Keywords.
- **Tab** delimited metadata and data.

The metadata section should contain entries describing all properties to be modified using the nodal property file and the syntax should be identical to the corresponding entries of the material properties of interest in the *.template file.

The NPD file includes sections for:

- Temperature metadata (optional)
- List of temperatures (optional)
- Nodal property metadata
- List of nodal properties (varying by temperature if specified)

The recognised keywords, in the order in which they should appear in an NPD file, are as follows:

For the optional temperature section:

TEMPERATURE_METADATA_START	Start of the meta-data section describing the table of temperatures.
TEMPERATURE_METADATA_END	End of previous section.
TEMPERATURE_LIST_START	Start of the data section containing the table of temperatures.
TEMPERATURE_LIST_END	End of previous section.

For the required nodal section:

NODAL_METADATA_START	Start of the meta-data section describing the table of nodal properties.
NODAL_METADATA_END	End of previous section.
NODAL_LIST_START	Start of the data section containing the table of nodal properties.
NODAL_LIST_END	End of previous section.

Each metadata section can contain a metadata definition of one or more material parameters. For the Temperature metadata section, this is limited to the Temperature_List variable only. For the Nodal metadata section, any variable (including Temperature_List) that can be included in a material database in *fe-safe* can be referenced. This is done by accessing the metadata section from an existing material database file (*.dbase).

For example, many commonly used material properties for fatigue analysis in *fe-safe* are included in the local database, accessible in the Local Directory as an ASCII file `<LocalDir>\local.dbase`. A copy of the local database can be made and accessed to find examples of metadata lines corresponding to material properties of interest for property mapping. Find each variable on its own line, and copy the lines of interest to build the metadata sections of a nodal property definition (NPD) file.

The first column in the table of nodal properties contains each node number to define nodal properties for, subsequent columns (**tab**-delimited) should contain the relevant data in the same order as defined in the metadata section. If a temperature list is specified this means that multiple values for each variable, corresponding to the temperatures should be listed (in **space**-delimited form).

Below are a few examples to show the use of metadata and the corresponding values listed at a short subset of nodes in an FE model. The example files are available from the directory `<DataDir>\NPD` and can be opened using the right-mouse button in the **Current FE Models** window and selecting **Open Nodal Properties....**

24.4.2 Example 1 – Basic nodal property definition NPD file (*.npd)

An example nodal property definition without defined temperature metadata or temperature list is shown below, note that some lines have been truncated to fit the page:

```
#<document link title> <keyword> <unused> <display text> <units> <size> <Extra_info>
NODAL_METADATA_START
BSName STANDARD_&_GRADE      UNUSED BSName None      72
E      E      UNUSED ~~gen::~~E      MPa    32000 "Edit=Table2d,..."
UTS    UTS    UNUSED ~~gen::~~UTS    MPa    32000 "Edit=Table2d,..."
NODAL_METADATA_END

#<NODE_ID> <E> <UTS>
NODAL_LIST_START
550    203000 400
163    203001 401
48     203002 402
NODAL_LIST_END
```

Note: some lines above have been truncated to fit the page. A sample file can be found in the directory `<DataDir>\NPD` to examine the full metadata definition for each parameter, and consider the impact of **tab** and **space** delimiting.

The lines in the nodal metadata section came from a material template file and reference the two variables: Young's Modulus (E) and Ultimate Tensile Strength (UTS) in the nodal list.

The first column in the table of nodal properties contains labels that are the node numbers, subsequent columns (**tab**-delimited) should contain the relevant data in the same order as defined in the metadata section. For example, for node 550 the Young's modulus was set to 203000 MPa and the Ultimate tensile strength was set to 400 MPa. These columns were separated from each other by **tabs**.

Once opened in the GUI the metadata above, and values for the first node defined in the Nodal List are shown in the Current FE Models window as follows in *Figure 24.4-1*:

Nodal mapping of material properties

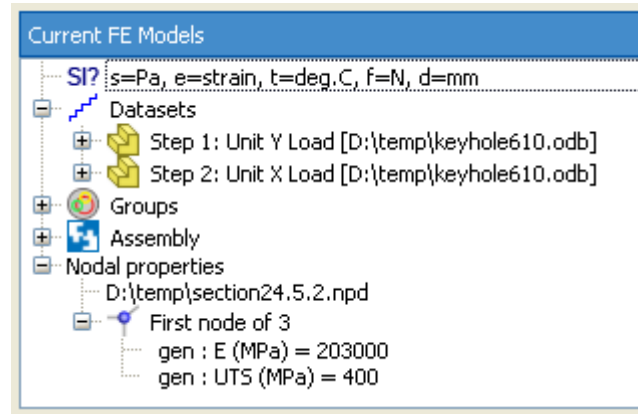


Figure 24.4-1

24.4.3 Example 2 – NPD file with temperature list defined for all nodes

For Nodal List data, when a temperature list has been defined, additional values for each variable are **tab** delimited. This means that for the example above, at node 550, Young's Modulus (E) is defined at the three temperatures (20, 200, and 250) as (203001, 190820, and 168490 respectively). Once opened in the GUI the metadata above, and values for the first node defined in the Nodal List (corresponding to the temperature list) are shown in the Current FE Models window as follows in:

An example nodal property definition with a defined temperature metadata and temperature list is shown below, note that some lines have been truncated to fit the page:

```
#<document link title> <keyword> <unused> <display text> <units> <size> <Extra_info>
TEMPERATURE_METADATA_START
BSName      STANDARD_&_GRADE      UNUSED BSName None    72
Temperature_List TempList UNUSED  ~~gen::~Temperature~List deg.C 200 "Edit=...
TEMPERATURE_METADATA_END
```

```
#<List of temperatures>
TEMPERATURE_LIST_START
Temperatures      20 200 350
TEMPERATURE_LIST_END
```

```
#<document link title> <keyword> <unused> <display text> <units> <size> <Extra_info>
NODAL_METADATA_START
BSName      STANDARD_&_GRADE      UNUSED BSName None    72
E   E        UNUSED  ~~gen::~E      MPa    32000 "Edit=Table2d,...
UTS UTS      UNUSED  ~~gen::~UTS   MPa    32000 "Edit=Table2d,...
SN_Curve_S_Values SN_Curve_S_Values UNUSED  ~sn~curve::~S~Values MPa 32000...
SN_Curve_N_Values SN_Curve_N_Values UNUSED  ~sn~curve::~N~Values nf 32000...
NODAL_METADATA_END
```

```
#<NODE_ID>      <E>      <UTS> <SN-S> <SN-N>
NODAL_LIST_START
550 203000 190820 168490 400 254 100 (400 200) (200 100) (100 50) 1e4 1e7
163 203001, 190820, 168490 401, 254, 100 (401, 201) (201, 101) (101, 51) 1e4, 1e7
48 203002, 190820, 168490 402 254 100 (402 202) (202 102) (102 52) 1e4, 1e7
NODAL_LIST_END
```

Note: some lines above have been truncated to fit the page. A sample file can be found in the directory <DataDir>\NPD to examine the full metadata definition for each parameter, and consider the impact of **tab** and **space** delimiting and **parenthesis**.

The lines in the temperature and nodal metadata sections came from a material template file and reference the variables in the temperature and nodal lists. In this example a temperature list of 20, 200, 350 degrees C was defined (note the list is **space** delimited).

All nodal properties should contain **space** delimited lists of data corresponding to the temperatures in the temperature list. Multi-dimensional properties (e.g. S-N curve datapoints) should be grouped by temperature in **parentheses**, and each group should be **tab** delimited. Node 550 for example, Young's Modulus (E) was set to 203000 MPa at 20 degrees C, 190820 MPa at 200 degrees C, and 168490 MPa at 350 degrees C. The values in this list were separated from each other by spaces, while the list of Moduli was separated from the column indicating the node number and the list of

Ultimate Tensile Strengths by tabs. Tabular stress–life data was defined for Node 550 for example as being 400 MPa at 1e4 cycles and 400 MPa at 1e7 cycles, for 20 degrees C.

Once opened in the GUI the metadata above, and values for the first node defined in the Nodal List are shown in the Current FE Models window as follows in *Figure 24.4–2*:

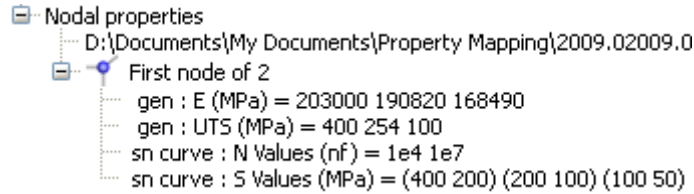


Figure 24.4–2

24.4.4 Example 3 – NPD file with temperature lists varying at each node

An alternative approach to defining temperature dependent material properties is by omitting the separate temperature list and specifying different temperature lists for each node as follows, note that some lines have been truncated to fit the page:

```
#<document link title> <keyword> <unused> <display text> <units> <size> <Extra_info>
NODAL_METADATA_START
BSName STANDARD_&_GRADE UNUSED BSName None 72
Temperature_List TempList UNUSED ~~gen::~Temperature~List deg.C 200...
E E UNUSED ~~gen::~E MPa 32000 "Edit=Table2d,...
UTS UTS UNUSED ~~gen::~UTS MPa 32000 "Edit=Table2d,...
NODAL_METADATA_END

#<NODE_ID> <List of temperatures> <E> <UTS>
NODAL_LIST_START
550 20 200 350 203000 190820 168490 400 254 100
163 40 250 400 203001 190820 168490 401 254 100
48 60 300 450 203002 190820 168490 402 254 100
NODAL_LIST_END
```

Note: some lines above have been truncated to fit the page. A sample file can be found in the directory <DataDir>\NPD to examine the full metadata definition for each parameter, and consider the impact of **tab** and **space** delimiting.

The lines in the nodal metadata section (including a temperature list variable) came from a material template file and reference the variables in the nodal list. In this example a temperature list of 20, 200, 350 degrees C is defined for node 550 only, and a different temperature list is specified at each node in the nodal list.

All nodal properties should contain **space** delimited lists of data and the columns following the temperature list should correspond to the temperatures in the temperature list column respectively. Multi-dimensional properties (e.g. S–N curve datapoints) should be grouped by temperature in parentheses, and each group should be **tab** delimited. Node 550 for example, Young’s Modulus (E) was set to 203000 MPa at 20 degrees C, 190820 MPa at 200 degrees C, and 168490 MPa at 350 degrees C. The values in this list were separated from each other by spaces, while the list of Moduli was separated from the column indicating the node number and the list of Ultimate Tensile

Strengths by tabs. To show the flexibility of varying temperature lists, node 163 had Moduli defined at 40, 250, and 400 degrees C instead.

Note that in Example 3, each node includes a temperature list of three temperatures. In fact, each node can have a different number of temperatures in the list. In such a case, the data in each column would vary accordingly. A sample file can be found in the directory <DataDir>\NPD to examine an example wherein the temperature lists are different length for each node.

Once opened in the GUI the metadata above, and values for the first node defined in the Nodal List are shown in the Current FE Models window as follows in *Figure 24.4-3*:

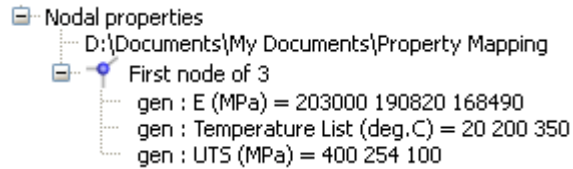


Figure 24.4-3

25 Fatigue analysis of linear dynamics FEA results

25.1 Performing fatigue analysis from linear dynamics FEA results.

Fatigue analysis from linear dynamics FEA results can be performed for steady-state dynamics and modal dynamics results.

For steady-state dynamics the FEA package calculates the real and imaginary FFTs of stresses for the specified exciting frequencies. This section outlines how *fe-safe* analyses this type of data.

For modal dynamic results the FEA calculates the response of the system in the time domain. This can be treated as a dataset sequence as outlined in sections 13 and 14.

Analysis of random response FEA results is not supported in the current release of the software.

Combining both modal dynamics and steady-state dynamics results within one analysis is supported.

25.2 Importing steady-state dynamics datasets

Abaqus .odb, Abaqus .fil, NASTRAN .op2 and ASCII steady-state dynamics data can be read into *fe-safe* with a normal open of the finite element model using the **File** menu item **Open Finite Element Model**.

Note: ANSYS .rst steady-state dynamic analysis results are also supported, but the RST file does not contain information to tell *fe-safe* what frequency each dataset corresponds to, or which datasets contain the real data and which contain the imaginary data. However, this limitation is overcome by manually defining the frequency and the datasets in the loading definition.

. As the model is being read, the contents of the dynamics results are reported to the message log. The real and imaginary stress tensors are read from two separate datasets for each exciting frequency.

Example:

```
From Step      : 7
Description    : S : 7: FREQUENCY RESPONSE: STEADY-STATE DYNAMICS, - (incr=1, t=80)
Direct Min/Max : -385798 385185
Shear Min/Max  : -178440 150127
No. Elements   : 2400
```

Frequency	: 80 Hz
Type	: real data

```
From Step      : 7
Description    : S : 7: FREQUENCY RESPONSE: STEADY-STATE DYNAMICS, - (incr=1, t=80)
Direct Min/Max : -12090.5 11097.3
Shear Min/Max  : -4803.59 5619.19
No. Elements   : 2400
```

Frequency	: 80 Hz
Type	: imaginary data

```
From Step      : 7
Description    : S : 7: FREQUENCY RESPONSE: STEADY-STATE DYNAMICS, - (incr=2, t=81.2593)
Direct Min/Max : -708147 681072
Shear Min/Max  : -324905 273456
No. Elements   : 2400
```

Fatigue analysis of linear dynamics FEA results

```
Frequency      : 81.2593 Hz
Type           : real data

From Step      : 7
Description    : S : 7: FREQUENCY RESPONSE: STEADY-STATE DYNAMICS, - (incr=2, t=81.2593)
Direct Min/Max : -48599.6 46264.6
Shear Min/Max  : -19579.7 22807.2
No. Elements   : 2400
Frequency      : 81.2593 Hz
Type           : imaginary data
```

The frequency value for each dataset will be stored for use within the analysis. Once the whole model has been read, the **Current FE Models** window will display a summary of the model as shown in *Figure 25.2-1*.

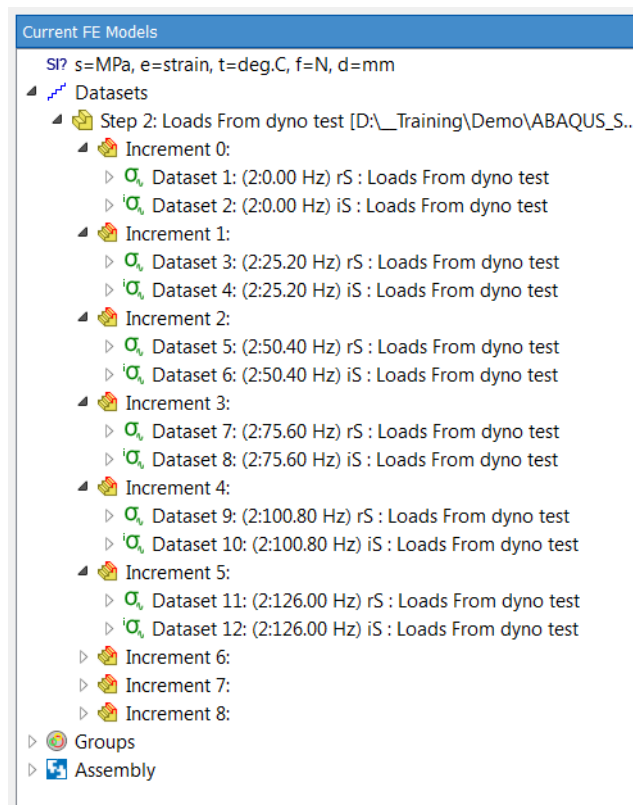


Figure 25.2-1

The icon associated with each dataset indicates whether it is a real or an imaginary dataset. The original step and the frequency information are also displayed for each dataset.

25.3 Analysing steady-state dynamics datasets

For *fe-safe* to analyse a set of steady-state dynamics FEA data, the load definition file must be used, see section 13. To tell *fe-safe* that the loading block within a .ldf file contains steady-state dynamics data, the keyword pair `modal=steady` must be used in the BLOCK statement. The simplest load definition uses the auto-detect feature to build the loading from all the real and imaginary datasets in the **Current FE Models** window.

e.g.

```
# LDF file containing 100 seconds of data
# let fe-safe work out how many repeats and which datasets

BLOCK modal=steady, dt=100
END
```

A time `dt` must be specified on the block statement for steady-state dynamics loading. When the simplest form is used, *fe-safe* will report which datasets it has paired together in the analysis .log file.

e.g.

```
Reading LDF file /data/fullmodeltests/501-00-modalres01.ldf
  Line 4 - Start of block processed - Repeats=1 Scale=1.00 dTime=100 Temp=-300
    (BLOCK modal=steady, dt=100)
  Line 6 - Modal block with no defined frequency datasets found
    ... Reading FED to auto-define block
      rds=3, ids=4, freq=4
      rds=5, ids=6, freq=8
      rds=7, ids=8, freq=12
      rds=9, ids=10, freq=16
      rds=11, ids=12, freq=20
      rds=13, ids=14, freq=24
      rds=15, ids=16, freq=28
      rds=17, ids=18, freq=32
      rds=19, ids=20, freq=36
  For modal block auto-calculated 'n' as 80. This used the lowest
    modal frequency 4 Hz and a factor of 5 to evaluate the minimum
    time for 1 repeat of the block as 1.25 seconds
  Line 6 - End of block processed
    (END)
End of read LDF file /data/fullmodeltests/501-00-modalres01.ldf
```

The other form of loading allows the user to specify which datasets to pair together in the analysis. The `freq` parameter is optional, as shown for the line `rds=5, ids=6`. In this case *fe-safe* will extract the frequency from the loaded FE models.

e.g.

```
# LDF file containing 100 seconds of data
# let fe-safe work out how many repeats and which datasets
BLOCK modal=steady, dt=100
rds=3, ids=4, freq=4
rds=5, ids=6
rds=7, ids=8, freq=12
rds=9, ids=10, freq=16
rds=11, ids=12, freq=20
rds=13, ids=14, freq=24
rds=15, ids=16, freq=28
rds=17, ids=18, freq=32
rds=19, ids=20, freq=36
END
```

Where the frequency is omitted *fe-safe* will report the frequency it found in the .log file, e.g.:

```
Extracted frequency 8 Hz for real dataset 5
and imaginary dataset 6
```

The loading is built up in the time domain to match the amplitude and phase relationships of the frequency domain stresses, see section 25. However, it is not recommended to superimpose multiple exciting frequencies, as it produces over-conservative results.

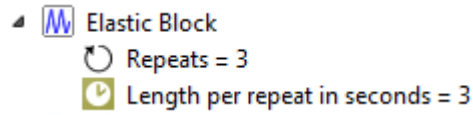
The length of a single repeat of the block may be defined in an .ldf file in one of two ways:

- Allow *fe-safe* to evaluate it using the lowest exciting frequency. A factor of 5 is applied to the period of the lowest frequency to ensure that there will be 5 cycles per repeat. This is then used to divide `dt`, the total block time for `n` repeats, to determine the unsupplied number of repeats `n`. This technique should be used when multiple frequencies are being used, which however is not recommended.
- Specify it using the `n=` parameter in the block definition. In this case the length of a single repeat is `dt/n`. This may be used to reduce the length of a repeat and improve performance if a single frequency is selected.

Conversely, the sample rate of the time-domain data is the product of the highest exciting frequency used and an integer samples-per-cycle setting which defaults to 10 (see Section 25.6).

Careful definition of the time information is important, since the speed of an analysis can be adversely affected if large amounts of data are generated and analysed for each and every node.

The time `dt` in the `.ldf` file will show the total time for all of a block's repeats, but in the Loading Settings tab in *fe-safe's* Fatigue from FEA panel, the time shown is for 1 repeat of the block, e.g.



will be written as `BLOCK n=3, dt=9`

A residual pair of stresses and strains can be used as an offset for the generated data, see section 13 for more details.

25.4 Mixing steady-state dynamic and time domain blocks of loading

The fatigue loading can consist of any number of time domain and frequency domain blocks of loading. Each block is analysed individually and the damages are summed using Miner's rule. The time domain blocks can be from elastic FEA results of elastic-plastic FEA results.

25.5 Discussion of methodology

For a frequency-domain block of loading, *fe-safe* will extract the real and imaginary parts of the FFTs. For each frequency specified in the load definition, the amplitude and phase relationship between each of the tensor components is evaluated and a sine wave of the required amplitude and phase is created. These time-domain signals are summed to provide the time-domain stress tensor history. This technique maintains the correct phase relationship between tensors in the loading and hence between the principals within the loading. However, it is not recommended to superimpose multiple frequencies. Consideration should be given to whether it is realistic to excite them simultaneously.

The utility to construct a history from the real and imaginary part of a FFT buffer uses an identical technique. This can be accessed from the **Generate** menu option **Generate Time History from FFT Buffer**. The PSD utility in *fe-safe* allows the FFT buffers to be exported. If this utility is used then care should be taken not to use the cosine tapering in the PSD module.

There are limitations with performing a PSD analysis on the diagnostic stress histories generated using the modal analysis technique. The PSD divides the frequency domain up into equal increments that may not lie exactly on the modes. This causes the frequency content of a mode to be split between adjacent FFT buffer coefficients rather than being concentrated on a single coefficient.

25.6 Configuration Options

The configuration options for analysis from steady-state dynamics are located on the **Analysis Options** dialogue, **Modal Analysis** selection in the **Algorithms** tab (**FEA Fatigue** menu). See *Figure 25.6-1*.

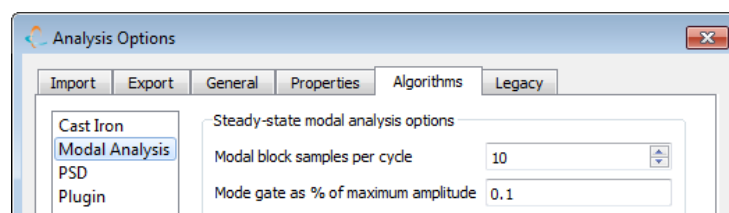


Figure 25.6-1

Modal block samples per cycle

This setting controls the sample rate of the generated stress tensors. It is a multiplier on the highest frequency found in the model, e.g. if the steady-state dynamics FEA results contain exciting frequencies of 12, 19 and 23Hz and this parameter is equal to 5 then the stress tensors will be generated with a sample rate of 115Hz (23Hz × 5).

Mode gate as % of maximum amplitude

This value allows speeding up of the stress tensor generation. If an exciting frequency has an amplitude less than this percentage of the maximum amplitude found in any frequency, then the frequency’s contribution is gated out.

25.7 Sensitivity Analysis

Sensitivity analysis for a frequency-domain block can be performed to see the effect of each frequency.

For the diagnostic nodes this repeats the analysis, omitting one frequency at a time. The effect of omitting each frequency is displayed in a table in the analysis .log file. This is enabled from the **Exports and Outputs** dialogue, **Log for Items** tab, (**Exports ...** button on **Fatigue from FEA** dialogue). A sample excerpt from a .log file is shown below:

```
SENSITIVITY ANALYSIS for Element 1.1 (The life is for 1 repeat of the
block (i.e n=1), it does not consider the n Value if this is an LDF analysis)
```

Life (Reps)	%	Ommision
3942360.41	100	None
3415936.57	87	4 Hz, rDS#13 iDS#14
Infinite	...	8 Hz, rDS#15 iDS#16
9986882.73	253	12 Hz, rDS#17 iDS#18
3476011.1	88	16 Hz, rDS#19 iDS#20
3875728.62	98	20 Hz, rDS#21 iDS#22
4222260.35	107	24 Hz, rDS#23 iDS#24
3754882.94	95	28 Hz, rDS#25 iDS#26
3143896.35	80	32 Hz, rDS#27 iDS#28
4692368.27	119	36 Hz, rDS#29 iDS#30
Infinite	...	40 Hz, rDS#31 iDS#32
4355837.63	110	44 Hz, rDS#33 iDS#34
3300559.67	84	48 Hz, rDS#35 iDS#36
3917764.79	99	52 Hz, rDS#37 iDS#38
4023822.01	102	56 Hz, rDS#39 iDS#40

25.8 Diagnostics

A set of diagnostics specific to steady-state dynamics analysis is provided. This is controlled from the **Exports and Outputs** dialogue. The dialogue is obtained by selecting the **Exports ...** button on the **Fatigue from FEA** dialogue. Select the **FFT** checkbox located in the **Histories for Items** tab. Diagnostic nodes can be defined on the **List of Items** tab.

For each diagnostic node a plot file is created. If the plot file is opened for a particular node after the analysis is completed (using the **File >> Data Files >> Open Data File** option) it will contain 13 channels, the real and imaginary FFT for each of the tensors and the frequency values.

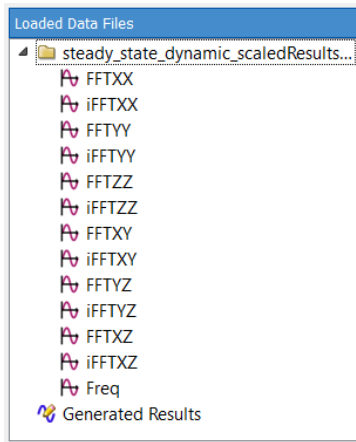


Figure 25.8-1

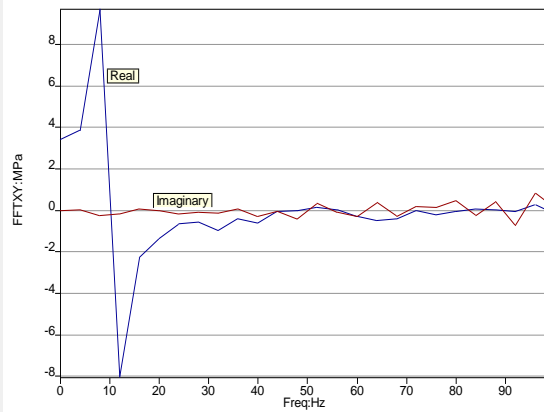


Figure 25.8-2

Cross-plotting the frequency channel and the real and imaginary channels creates a plot of the FFT buffers. An example showing the XY component of stress is shown in *Figure 25.8-2*.

The generated tensors in the time domain can also be exported and plotted in the same way as for other time-domain analyses in *fe-safe*. When the tensors are exported for a steady-state dynamics block, the title will indicate which exciting frequencies were used to build the tensor (and which were gated out). This can be seen by selecting the properties of the tensor diagnostics channel as shown in *Figure 25.8-3* (right-click on the desired channel).

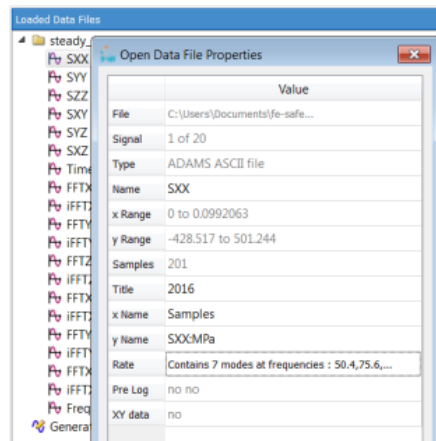


Figure 25.8-3

Plotting the tensor channels shows the generation technique based upon a series of sine waves. See *Figure 25.8-4*.

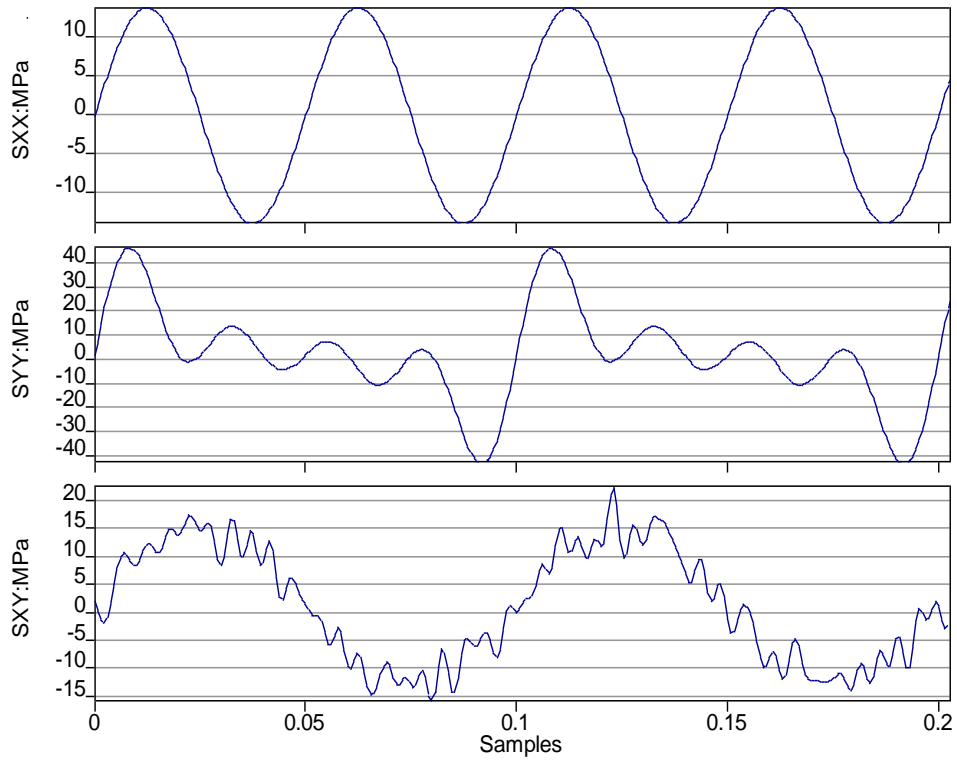


Figure 25.8-4

26 Theory of Critical Distance methods (TCD)

26.1 Introduction to the Critical Distance methods (TCD)

These methods allow a stress gradient to be used to assess notch sensitivity on a FE model and are particularly useful for analysing high-cycle fatigue behaviour of components with stress concentrations coming from holes, grooves, corners etc.

Stress gradient has little effect on fatigue lives to crack initiation. Almost all steel and aluminium materials can be treated as fully notch-sensitive so stress gradient effects are not required for accurate life prediction. However, for cast irons, and particularly grey cast iron, this approach may be excessively conservative because of the presence of crack-like graphite.

For such materials it is more appropriate to calculate FOS/FRF using Theory of Critical Distances (TCD) point method or line method. For more information on FOS or FRF methods see section 17.

The internal stress cycle is evaluated at a certain distance inside the material in the case of the Point Method (PM), or averaged along a line in the case of the Line Method (LM), as determined by the critical distance parameter for the material. See section 26.6 below for details.

26.2 Calculating the FOS/FRF using TCD

The critical distance FOS/FRF calculations can be enabled using options on the **Safety Factors** section of the **Algorithms** tab of the **FEA Fatigue >> Analysis Options** dialogue window:

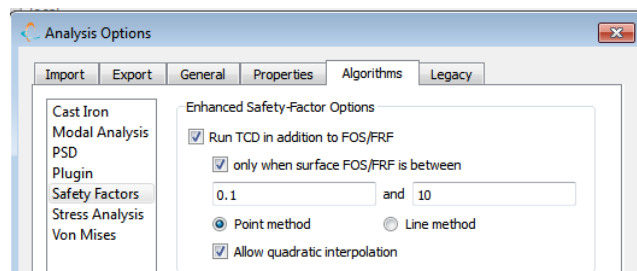


Figure 26.2-1

The Critical Distance method options can be found in the **Enhanced Safety-Factor Options** region. Selecting the **Run TCD in addition to FOS/FRF** checkbox enables the calculation.

Choice of the required method can be made by selecting either **using critical-distance point method** or **using critical-distance line method** option as appropriate.

A limit to apply enhanced safety factor calculations using TCD **only when surface FOS/FRF is between** specified values is possible. Nodes outside those values will be omitted from the calculation. By default the limits are applied within the thresholds of 0 and 10 for FRF and those shown in Figure 26.2-1 for FOS.

Note: Even when an FRF or FOS is within the specified limits, it may be at the limit of meaningful values, e.g. 10, denoting no damage, for an FRF, or the maximum/minimum band for FOS. In this case, no TCD calculation is performed and the surface value of the factor is reproduced in the Critical Distance output. Similarly, if a TCD calculation takes a factor outside its defined range, the TCD value output is limited to that range.

The Critical Distance methods (TCD)

If a TCD method has been selected then an additional contour is written containing the Critical Distance FRF or FOS value, called FRF-R@CritDist or FOS-R@CritDist respectively. If for some reason the critical distance calculation cannot be performed, then the value contour will contain the surface radial FRF (or FOS). Also, as the crack propagation threshold cannot be worse than the crack initiation threshold, if the TCD factor is lower than the conventional surface factor, then the TCD factor is replaced with the surface radial FRF (or FOS). See section 26.3 below for details on contours and diagnostics included in the TCD outputs

26.2.1 Material Studies of stress-concentrations at notches has led to the definition of a stress-intensity factor K for a given notch of radius a and nominal stress σ :

$$K = \sigma\sqrt{\pi a}$$

This parameter can be used to predict crack-growth due to fatigue, which will only occur when the range ΔK of stress intensity exceeds a threshold ΔK_{th} , which is a material property that is constant for a given stress ratio $R = \sigma_{min} / \sigma_{max} = K_{min} / K_{max}$. This property is defined in the *fe-safe* material database for the case of $R = -1$ corresponding to zero mean stress and is denoted `taylor : Kthreshold@R:-1`, in units MPa m^{1/2}. *fe-safe* can use this to calculate the critical distance parameter L (see section 26.6 below), or alternatively the critical distance can be directly specified as the material property `taylor : L (mm)`.

26.2.2 Mean stress correction

The Critical Distance calculation corrects stress-cycles for mean-stress effects using the same mean stress correction (MSC) that was selected for the FRF (or FOS) analysis, although there are certain restrictions on available MSC algorithms, in addition to any usual restrictions on which MSC algorithm is available for the selected analysis algorithm. The following are available with Critical Distance:

- Goodman
- Gerber
- Smith Watson Topper
- Walker
- User defined mean stress correction (via user supplied .msc or .frf file)
- R Ratio SN Curves

If an analysis algorithm is selected with an MSC which is not available in Critical Distance (e.g. FOS analysis with Morrow MSC), then the analysis will proceed, but the Walker MSC will be used instead in Critical Distance, and a warning will be issued. If the Walker exponent parameters have not been set then 0.5 will be assumed (i.e. similar to Smith Watson Topper).

For more information on the mean stress corrections and required material properties see sections 14 and 8 respectively.

26.3 Outputs

Critical Distance Radial FRF or FOS values are exported as contours when their surface FOS/FRF factor counterparts are calculated by an analysis and selected for export as contours. The worst Critical Distance factors are also reported in the analysis summary:

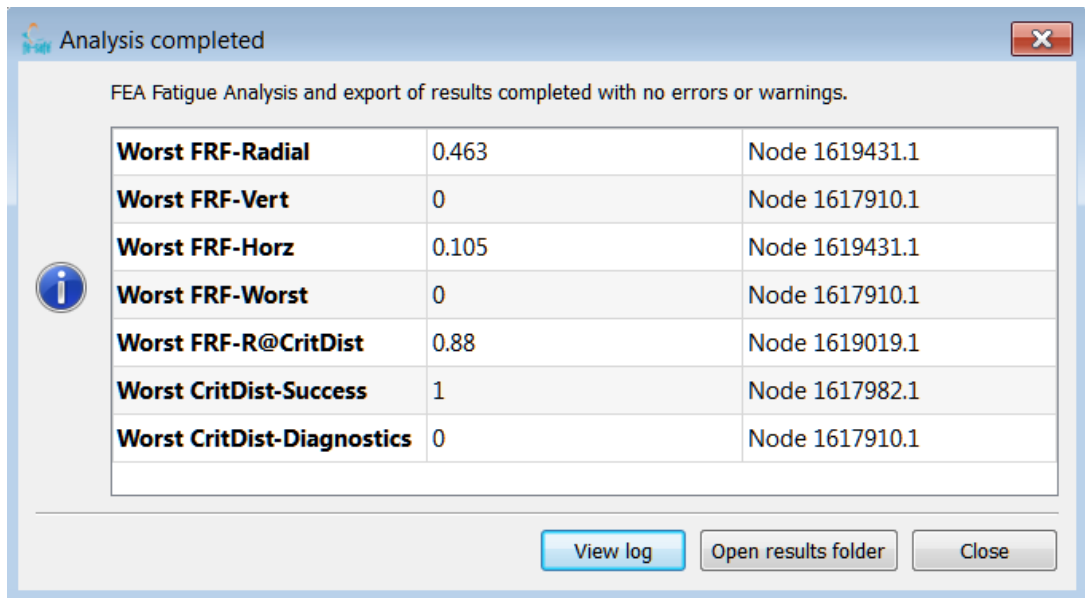


Figure 26.3-1

In some cases the FRF or FOS may not differ from those calculated using TCD methods. In such cases the R@CritDist contour will contain the surface FRF or FOS value. *fe-safe* optionally outputs two additional contours called CritDist-Success and CritDist-Diagnostics so that any problem nodes can be identified. These can be selected via the **Contours** tab of the **Exports and Outputs** dialog, opened via the **Exports...** button of the **Analysis Settings** tab of the **Fatigue from FEA** dialog. The difference between the success and diagnostics contours is that the former gives a simple summary of success or failure, whereas the latter gives detailed reasons for the failures. The coding of the success contour is

- 0 = Failure
- 1 = Warning (Calculation succeeded but with a warning)
- 2 = Success

A complete description of all the diagnostic codes is given in section 26.5 below. In brief, negative codes are used if the calculation did not proceed at all (e.g. the node was out of the defined FRF band, or required material data was unavailable), zero if there was no failures or warnings, and positive for a warning or error encountered during the calculation.

There are two possible warnings issued if

- The mesh is considered coarse which may cause potential interpolation inaccuracies.
- The critical distance FRF (or FOS) is worse than the corresponding surface value.

A complete list of errors and an example of viewing the diagnostic contour is given in section 26.5 below. A common cause of error is when the stress gradient path leaves the model before reaching the specified critical distance (e.g. when analysing thin structures).

If the **Export Critical Distance summary** checkbox has been selected on the **Log** tab of the **Exports and Outputs** dialog, then a warning summary will appear at the end of the analysis giving the total number of problem nodes under each category, and the corresponding diagnostic code. The diagnostic code may be useful when viewing the diagnostic contour (details in section 26.5 below). If the **Export Critical Distance summary** checkbox has been

The Critical Distance methods (TCD)

selected, then further details on each node with a Critical Distance warning or error are written to the *fe-safe* log file. Each such node has a line in the log giving node ID, numeric diagnostic code and short text explanation. The number of nodes in any failure category in this file is limited to 10,000.

An example comparing the conventional surface FRF with the Critical Distance FRF contour is shown below for an open-source crank throw model (*Figure 26.3-2*). It can be seen that the worst-case FRF region is improved on the Critical Distance contour (the red hotspots disappear).

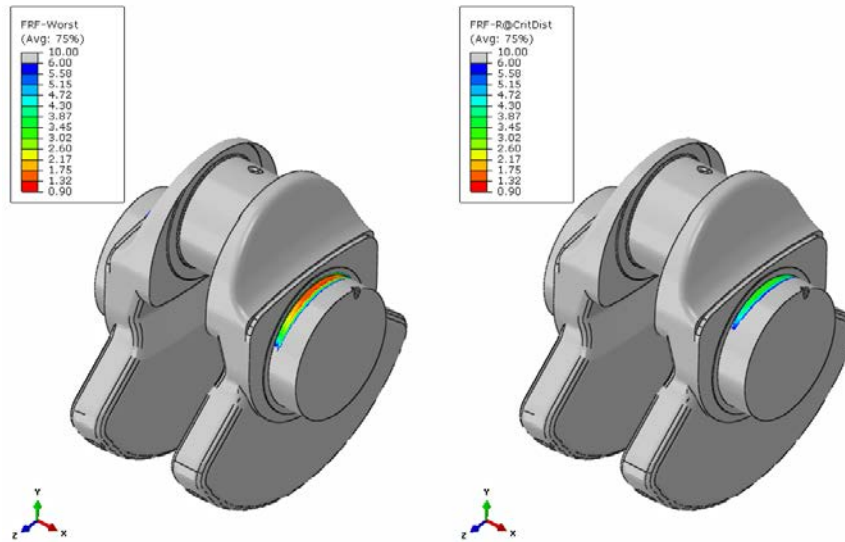


Figure 26.3-2

It is also possible to produce more detailed information showing details of the calculation and the stress tensors interpolated along the stress gradient path. Values are output at element boundaries. These additional outputs can be selected for specific items by specifying the required item IDs in the **List of Items** tab in the Exports dialog box.

if the **Critical Distance items** checkbox is selected on the **Log for Items** tab, then further details will be written to the log file detailing:

- the surface node's coordinates and surface-normal;
- the critical plane search;
- the plane, block and sample numbers of the most damaging cycle;
- the elements intersected by the critical path, with topology information;
- the minimum and maximum normal stress upon entering and leaving each element;
- the stress mean, amplitude and mean-stress-corrected range at the critical point (for the Point Method), or integrated along the critical path (for the Line Method);
- the material's fatigue strength at the node's temperature;
- the resultant FRF.

If the **Critical Distance stress-vs-depth** checkbox is selected on the **Histories for Items** tab, then each node in the list of items has a plottable text (.txt) file created, listing the following as a function of depth: critical plane min/max normal stress, the associated cycle mean and amplitude, and $\Delta\sigma_o$ (see section 26.6 below). These plottable text files appear in the results directory, and are appended with the node ID (e.g.

crankResults_CritDist_Line1_Depth_n60035.txt for input model file named crank). Note that there may be two depth text files when using the line method because there can be different critical planes determined at the surface and the point method depth; if these differ then results for both are output (1=surface, 2=point method depth).

If the **Critical Distance stress tensors** checkbox is selected on the **Histories for Items** tab, then each node in the list of items has a second plottable text file created, listing the six tensor components of the interpolated stress as a function of depth. These plottable text files likewise appear in the results directory and are suffixed with the node ID (e.g. crankResults_CritDist_Point_TensorHistory_n60035.txt for an input model file named crank).

26.4 Notes on The Critical Distance methods

The following should be noted:

- The methods require that geometry is imported from the FE model and its surface is detected. (See section 5.4.2).
- It is assumed that the material parameters on the surface and inside the model are identical. Critical Distance methods may not be applicable otherwise
- A residual stress dataset may be defined for the Transition Block set in the **Settings** section of the **Loading Settings** dialogue. In-plane residual stresses specified in the group parameters area of the **Fatigue from FEA** dialogue are not supported, as the internal residual stresses may differ than those thus defined at the surface.
- Materials SN curves are always used when present, even if this is deselected in the **FEA Fatigue >> Analysis Options** dialogue by selecting *Use stress-life curve defined using s_f and b* . Only if no SN tabular data is defined in the material, will the s_f and b parameters will be used.
- The methods use a critical plane search around an axis defined by the geometric surface normal, which is calculated using a weighted mean of the surface normals evaluated over all elements containing the surface node. The weighting factor is the angle subtended by each face at the node.
- Shell elements are not supported.
- Second-order solid elements are supported and the stress interpolation function used is then quadratic by default. Linear interpolation in second-order elements may be selected in the **FEA Fatigue >> Analysis Options** dialogue (see Figure 26.2-1 above). Second-order elements from Ansys RST models do not include mid-node stresses, so they are always interpolated linearly. A least-squares fit is used to all the nodes of the element, but this may lead to residual errors if linear interpolation is used on quadratic or otherwise non-linear stress functions. These errors could become significant if a coarse mesh (on the scale of the critical distance for the material) is combined with second-order elements where the second derivatives of the stress function are large. However, such errors will tend to be smoothed out in the Line Method integration.
- For best results, meshing should be configured so that element lengths do not exceed $L/4$ for the Point Method or $L/2$ for the Line method, where L is the critical distance (see 26.6 below).
- Symmetry boundary conditions of the FEA model are currently not recognised, so planes of symmetry will be treated as a free surface of the model.
- The methods rely on interpolation within finite elements based on nodal values. Therefore, it is necessary to use nodal averaged or element-nodal data; integration-point or centroidal data would give poor results

and are not currently supported. No internal averaging of element-nodal data is done, so the user should ensure that discontinuities in stress between adjacent elements near the critical point or line are small.

- The current *fe-safe* Critical Distance implementation is only valid for infinite life, and when performing a FOS analysis the Critical Distance calculation will always calculate $\Delta\sigma_o$ (see section 26.6 below for details) from the constant amplitude endurance limit, even if a finite life has been specified for the FOS analysis. A warning message box is displayed prior to running the analysis in this circumstance.
- The Critical Distance methods only use stresses. The software does allow analyses to be run which combine strain-based algorithms with Critical Distance, but the stress-based Critical Distance results may not be directly comparable with the conventional FRF/FOS. A warning message box is displayed prior to running the analysis in such cases.
- Because of the computational overhead in interpolating along the critical path, complex signals (scale-and-combine) are heavily gated prior to the Critical Distance algorithm, essentially preserving only the overall minimum and maximum. This does not apply to loadings specified as dataset sequences.
- The internal ray-tracing and interpolation used in the Critical Distance calculations mean that the method is computationally expensive. When running on large models where computational time may become an issue, it may be worth restricting the FRF/FOS band more tightly so that only the nodes of most concern are analysed using TCD and FRF/FOS. Alternatively a conventional analysis can be run first and then hotspot detection (see section 22.5 for details) used to limit analysis to elements on which Critical Distance calculations can be profitably performed.

26.5 Diagnostic Codes and viewing the diagnostic contour

26.5.1 Diagnostic Codes

The following tables list all Critical Distance warning and error codes appearing in the diagnostic contour and their specific meaning. Note that some errors mean that the analysis cannot proceed at all (negative codes), whereas others are errors during the analysis.

The analysis never proceeds in the following cases:

Error Code	Value	Meaning
Rejected_Not_On_Surface	-5	The node is not on the surface.
Rejected_Outside_FRF_Band	-1	The surface FRF value is outside the specified analysis range
Rejected_Outside_FOS_Band	-9	The surface FOS value is outside the specified analysis range

The analysis is immediately aborted in the following cases:

Error Code	Value	Meaning
Error_No_Node_Coordinates	-14	The node is lacking position data
Error_No_Normal	-13	The surface normal cannot be computed

Error_No_Material_Data	-12	The material does not contain required properties for TCD analysis
Error_Unsupported_Data_Position	-11	The dataset type is not supported
Error_Not_FRF_or_FOS	-10	Neither an FRF nor FOS analysis is being conducted
Error_Missing_Node	-8	The analysis node does not belong to the element specified for analysis
Error_No_Element	-7	An element specified for analysis does not exist in the model
Error_Unsupported_Element_Type	-6	An element type specified for analysis is not supported

Otherwise the analysis proceeds but may fail with one of the following errors.

Error Code	Value	Meaning
Error_Path_Left_Model	10	The critical path appropriate to the method (PM or LM) normal to the node left the model (or entered a hole).
Error_Missing_Stress	11	An element along the path had no associated stress data (or not all its corner nodes had stress data).
Error_Bad_Mesh	12	There were mesh inconsistencies during ray tracing, or the surface normal never entered the model.
Error_Singular_Matrix	13	Geometry of an element led to non-invertible matrix during stress interpolation, probably due to a collapsed element with several coinciding nodes
Error_Msc_Failed	14	Mean stress correction failed
Error_Internal_Problem	20	Any other undiagnosed problems

The following codes indicate either success or completion with warning:

Code	Value	Meaning
Success	0	The calculation was successful and no warnings were issued.
Warning_Poor_Mesh	1	The analysis was successful but may be inaccurate because the mesh may be too coarse (too few elements encountered during critical distance ray tracing, indicating possible

		interpolation errors).
Warning_Worse_Than_Surface	2	The analysis was successful, but the Critical Distance result was worse than that at the surface and so was ignored (replaced with surface value).

Certain FE analysis packages may use quadratic elements but only export stresses at corner nodes. This still allows a successful analysis, but there may be some errors in the log file indicating nodes with missing stress values.

26.5.2 Visualising Problem Node Cases

As stated above, nodes where the Critical Distance calculation could not be performed will receive a diagnostic code in the CritDist-Diagnostic contour (if this contour was selected for output on the **Contours** tab). The warning summary at the end of the analysis will give the corresponding diagnostic code for each error type (if the **Export Critical Distance summary** checkbox has been selected on the **Log** tab). If it is desired to investigate where these nodes are in the model then a suitable post-processor may be used to view the contour. It will normally be necessary to adjust the contour legend in the post-processor to highlight the error code of interest. For example when analysing a thread model, *fe-safe* reported that there were several hundred nodes where the critical path left the model (diagnostic code 10). The results file was loaded into a suitable post-processing tool, where the diagnostic contour could be visualised. The colour coding was adjusted to highlight the diagnostic code 10 value, and creating a plane cut through the model shows that these problem cases lie along the edge of the thread where the critical distance is greater than the thread thickness.

Note: Some post-processors may produce spurious local interpolation effects when displaying integer diagnostic codes as floating point values. Averaging should be switched off if possible to negate these effects.

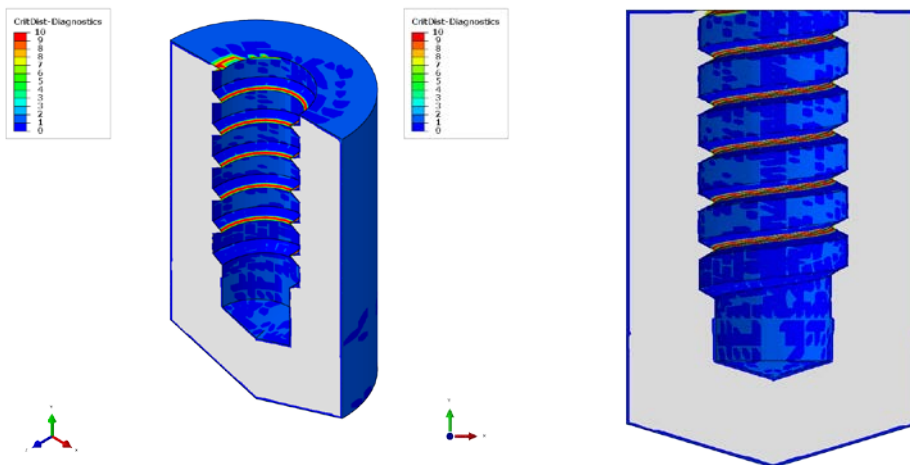


Figure 26.5-1 Diagnostic contour showing location of “path left model” nodes on a thread

26.6 Technical Background

In general the fatigue life to crack initiation is determined by the sequence of stress/strain on the surface of a component. This is the fatigue life that *fe-safe* calculates using the various strain-life methods.

At stress concentrations there may be a stress gradient, with sub-surface stresses significantly lower than those at the surface. Whether or not this crack will propagate depends on the stresses at a certain distance below the surface, see *Figure 26.6-1* below. This distance is a material property, the difference between the two stresses is an indication of the material 'notch sensitivity' – the larger the critical distance the lower the 'notch sensitivity'. In *Figure 26.6-1*, $r_c = L/2$, where L is the 'critical distance' for the material.

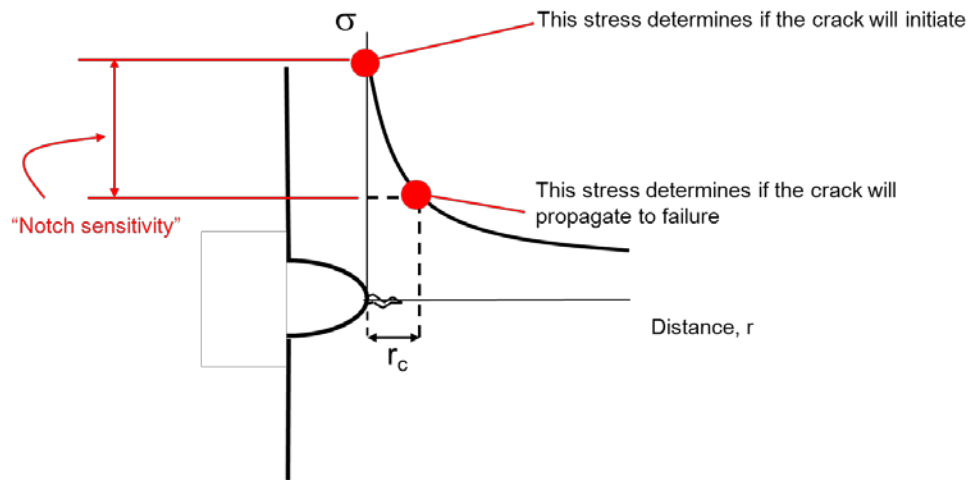


Figure 26.6-1 Stress gradient at a 'notch'

Critical distances can vary from less than 0.1mm for high strength steels, to 4mm for some grey irons.

For sharper notches (i.e. at higher values of K_t) there will be a bigger difference between the stresses at the surface and the stresses at the critical distance. Hence there is more chance that the crack will not propagate. This difference will be greater for lower strength materials because the critical distance is greater. Critical distance methods are therefore most applicable to relatively sharp notches in cast irons, but may have an effect on other materials as well depending on K_t .

The benefit of using critical distances is that higher stresses may be used. If the crack will not propagate it may be possible to increase the stresses to a value where the crack will just not propagate. However, the designer is then moving from a 'crack initiation' design criterion to one in which cracks are allowed.

The Critical Distance methods (TCD)

Critical Distance methods are described in detail in Ref. 26.1. Critical distance parameters for many materials are given in Ref. 26.2. If no critical distance (L) material property is specified in the material database (see section 8 of the *fe-safe* User Guide), then the critical distance is calculated using the threshold value of the crack growth parameter ΔK_{th} . The relationship is

$$L = \frac{1}{\pi} \left(\frac{\Delta K_{th}}{\Delta \sigma_o} \right)^2$$

Where:

L is the critical distance for the material and

$\Delta \sigma_o$ is the stress amplitude at the constant amplitude endurance limit (CAEL) from a conventional uniaxial stress S-N curve at zero mean stress. Note that even if L is instead specified as a material property, $\Delta \sigma_o$ is still calculated from the CAEL, as it is also needed for the FRF (or FOS) factor calculation.

A review of Critical Distance applications is given in Ref. 26.3.

Critical Distance factor values are derived from $\Delta \sigma_o$, which is determined by the conventional constant amplitude endurance limit for the material. The use of Critical Distance methods should therefore be applied only for 'infinite life' calculations. In particular if a FOS analysis is being conducted and a target life lower than the CAEL has been specified, the Critical Distance calculations will still use the CAEL. The use of Critical Distance methods in finite life analyses is the subject of current research (see Ref. 26.4 and 26.5). However, finite life Critical Distance analyses are not supported in the current release of *fe-safe*.

26.6.1 Cast Irons

There is an exception made in the calculation of $\Delta \sigma_o$ when the Cast Iron algorithm is used. The Smith-Watson-Topper (SWT) life curve is used instead of the S-N curve in such cases to convert the CAEL to $\Delta \sigma_o$.

The CAEL (n say) is converted to an equivalent Grey Iron SWT value thus:

$$G_{SWT} = An^b,$$

where A and b are `SWTLifeCurveCoeff` and `SWTLifeCurveExponent` material properties (see section 8 of the *fe-safe* User Guide) (e.g. $b=-0.25$ for `Downing : GreyIron`).

Then assuming elasticity in the SWT stress-strain product, *fe-safe* sets $\Delta \sigma_o$ using Young's modulus E as follows:

$$\Delta \sigma_o = 2\sqrt{EG_{SWT}}$$

This gives for the Cast Iron algorithm:

$$L = \frac{\Delta K^2}{4\pi E A n^b}$$

It is recommended that iron materials have the material database property for L explicitly specified whenever possible.

26.6.2 Critical Distance Methods

The Point Method (PM) postulates that the condition for fatigue failure is that the stress-range at a distance $L/2$ (critical distance) from the crack tip exceeds the fatigue strength $\Delta\sigma_o$, the stress-range that corresponds to infinite life according to the material SN curve, see *Figure 26.6-2* below.

Thus, the stress-range at a distance $L/2$ from the surface may be compared with $\Delta\sigma_o$ to compute Fatigue Reserve Factors (FRF) or Factors of Strength (FOS).

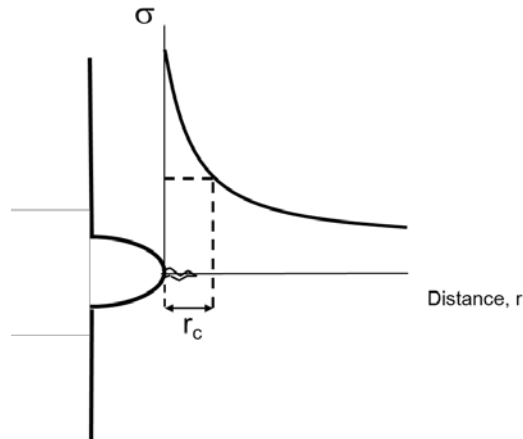


Figure 26.6-2 The Point Method

Similarly, the Line Method (LM) uses the mean of the stress-range integrated over a path of length $2L$ along the normal to the surface, see *Figure 26.6-3* below.

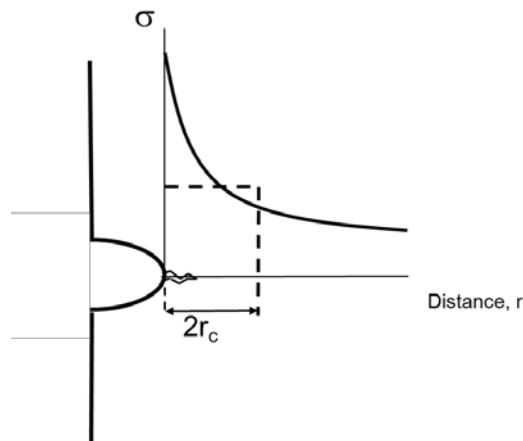


Figure 26.6-3 The Line Method

When performing the integral to calculate the (spatial) mean stress it is necessary to determine a critical plane and a minimum and maximum point in the stress cycle. Strictly speaking these could vary with depth, leaving the line integral somewhat ill-defined. *fe-safe* first takes the critical plane and worst block stress range at the surface, and the line integral is evaluated for this critical plane/block. If the point method gives a different critical plane or block at the point method depth, then the integral is also performed for this plane/block, and the higher integrated stress is used.

26.6.3 References

26.1. Taylor D. The Theory of Critical Distances. A New Perspective in Fracture Mechanics.

Elsevier, 2007

26.2. Susmel L. Multiaxial notch fatigue.

Woodhead, 2009

26.3. Susmel, L. (2008). The theory of critical distances: a review of its applications in fatigue. *Engineering Fracture Mechanics*, 75(7), 1706-1724.

26.4. Susmel, L., & Taylor, D. (2010). An elasto-plastic reformulation of the theory of critical distances to estimate lifetime of notched components failing in the low/medium-cycle fatigue regime. *Journal of engineering materials and technology*, 132(2).

26.5. Susmel, L., & Taylor, D. (2012). A critical distance/plane method to estimate finite life of notched components under variable amplitude uniaxial/multiaxial fatigue loading. *International Journal of Fatigue*, 38, 7-24.

27 Fatigue Analysis Using PSD data

27.1 Introduction

27.1.1 PSD capability

This capability allows the user to define the fatigue loading in terms of power spectral density (PSD). It is used in conjunction with a frequency-response analysis in the FE software. In *fe-safe* multiple PSDs of loading can be applied together with cross spectral densities (CSDs), if they are available. The following capabilities are supported:

- Multiple PSDs of loading.
- Multi-block loading.

There is a choice of analysis algorithms to calculate expected life once a suitable PSD response has been calculated:

- The Dirlik algorithm [1]
- The Tovo-Benasciutti algorithm [2-4] with fixed (per node) mean stress defined by residual stress.
- The Tovo-Benasciutti algorithm [2-4] with randomly distributed mean stress centred on the specified residual stress.

The Dirlik algorithm only considers cycle amplitudes, so if residual stresses are present one of the Tovo-Benasciutti methods should be used. Note that if no residual stress is defined then the Tovo-Benasciutti algorithm will use zero overall mean, but even the fixed mean option may still give slightly different results to Dirlik because the amplitude distribution is slightly different.

Four methods are available for computing response PSDs:

- Von Mises stress for ductile metals.
- A normal-stress critical plane algorithm.
- A shear-stress critical plane algorithm.
- A combined shear and normal stress critical plane algorithm.

The normal-stress critical plane algorithm searches a full hemisphere, but to obtain a reasonable computation time, the shear algorithm searches a more restricted set of critical planes, which are planes at 90 degrees or 45 degrees to the surface normal. Since this implies that the surface normal at each node is defined, the shear stress PSD algorithm can only be run on the surface group. The combined shear and normal stress algorithm is a kind of modified shear algorithm. The set of evaluated critical planes is still exactly as for the shear case, but a contribution of normal stress (projected onto each plane) is added with configurable weighting k (default 0.25).

The PSD approach may also be used in FOS calculations on expected life.

27.12 Outline of approach

Performing a frequency sweep (such as Harmonic or Steady State Dynamic) analysis in the FE analysis software will produce the following data which can be utilised by *fe-safe*:

- Modal stress solutions and Generalized Displacements (GDs) (also called modal participation factors in ANSYS). Such data characterises a structure’s harmonic response when subjected to a defined load over a pre-defined frequency range. From this data the component’s frequency response functions per mode, per node can be calculated.

For the *fe-safe* analysis the user must provide:

- Sets of PSDs to characterise the applied loads. Loading may be acceleration, force, displacement, etc...
- Optionally, CSDs may be input. For information on CSDs see section 10.

The *fe-safe* analysis procedure is shown in Figure 27.1-1. In summary, *fe-safe* processes the FEA results and the user-supplied PSDs (and CSDs, if available) by calculating the response PSDs at each node. This response data is then used by the fatigue damage algorithm.

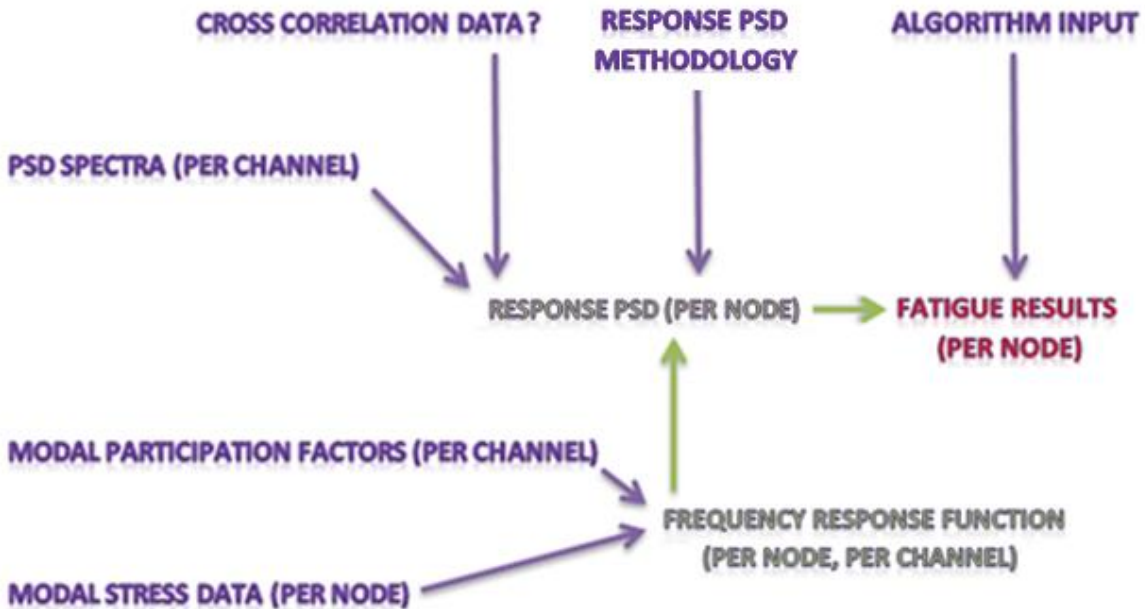


Figure 27.1-1 Outline of data interaction during the *fe-safe* PSD calculation procedure (for a typical multi-channel, single loading block example). Note that the purple text indicates input data, the green text indicates calculated data and the red text indicates output.

27.2 Inputting of data for a PSD Analysis

To perform a PSD analysis *fe-safe* can currently use data generated by Abaqus, ANSYS and NASTRAN FEA software. Table 27.2-1 illustrates how some of the terminology used in this document relates to that used in FEA software.

Terminology	FEA Software Description
Channel	Step, load step.
Stress data	Modal stresses, or stress variables from an eigenfrequency analysis, extracted at a discrete number of natural frequencies.
Stress amplitude	Stress magnitude from an analysis that contains complex-valued results.

Table 27.2-1 The terminology used in this document and the equivalent FEA software-related descriptions.

27.2.1 Files required: Abaqus

The necessary files are specified in Table 27.2-2. Note that either a single ODB may be used, containing both modal stress data and generalized displacements steps for the GDs, or these may be separated into two .odb files. To create such files the user must follow a two-step procedure. Firstly, the modal stresses need to be calculated in a Natural Frequency Extraction Analysis. An excerpt from a sample input deck is displayed in Figure 27.2-1(a). Note that Abaqus-specific modal participation factors will be created by the Natural Frequency Extraction Analysis [5]; however, these are not to be confused with the Generalized Displacements required for an *fe-safe* PSD analysis. Once the above-mentioned analysis is complete the user must run a Steady State Dynamic Analysis to generate a set of displacement-related results. Such displacement data represent the GDs required for an *fe-safe* analysis. Since calculations outlined in Figure 27.1-1 are carried out on a per-channel basis, it is necessary for the GD data for each node to be matched to the associated channel number. This is addressed by making the assumption that the channel order (from 1 to n) follows the step sequence detailed in the associated input file, i.e. the first step is taken to be the first channel, the second step is taken to be the second channel, etc... It is assumed that the frequency interval for which output is requested for each step is identical. An excerpt from a sample input deck for the Steady State Dynamic Analysis is displayed in Figure 27.2-1(b). Note that displacement variable (U) from the Natural Frequency Extraction step is not required for *fe-safe*, but is useful in Abaqus/CAE to visualise the eigenmodes.

Data	File extension	Data Description
Stress Datasets	.odb	Stress data per node, per mode.
<i>fe-safe</i> Generalized Displacements	.odb	Frequency (Hz) vs. Generalized Displacement data (per mode) in rectangular or polar form. It is assumed that all the GD data, i.e. every channel-related set of GD data, is provided by one file.

Table 27.2-2 Necessary Abaqus generated files for an *fe-safe* PSD analysis.

```

** FIELD OUTPUT: F-Output-1
**
*Output, field
** -----
*Node Output
U,
*Element Output, directions=YES, position=nodes
S,
** -----
    
```

(a)

```

** HISTORY OUTPUT: H-Output-1
**
*Output, history
** -----
** specifically request generalized displacement output
** -----
*Modal Output
GU, GPU
** -----
    
```

(b)

Figure 27.2-1 Abaqus input file extracts to request the necessary output in including (a) modal stress data and (b) *fe-safe* Generalized Displacement data for a PSD analysis.

27.22 Files required: ANSYS

The necessary files are specified in Table 27.2-3. It is assumed that the modal stress data is created in a Modal Analysis, whilst the associated Generalized Displacement data (also called Modal Participation Factors in ANSYS) is generated in a Harmonic Analysis. The stress data in the .rst file and the GDs in the .mcf files can then be combined (by *fe-safe*) to calculate the frequency response functions. There is a strict format associated with these files [6]. Furthermore, it is assumed that for an analysis of *n* channels the frequency range of each Harmonic Analysis (leading to an applied .mcf file) are identical.

The content of each .mcf file does not explicitly state the associated channel (loading location and direction). Such information is necessary for a multi-channel PSD analysis since the calculations outlined in Figure 27.1-1 are carried out on a per-channel basis. To overcome this problem the following naming convention must be obeyed. For *n* channels there will be *n* .mcf files. It is expected that each .mcf file has a unique channel-specific number at the end of its name, located between a '_' and the file extension. It is assumed that such channel identifiers are numbered in a continuous manner (from 1 to *n*), e.g. say a .rst file has two associated .mcf files then these files should be called x_1.mcf and y_2.mcf (where x and y denotes a valid file name).

Data	File extension	Data Description
Stress Datasets	.rst	Stress data per node, per mode.
Generalized Displacements	.mcf	Frequency (Hz) vs. Generalized Displacement data (per mode) (also called Modal Participation Factors in ANSYS) in rectangular or polar form. It is assumed that there will be one .mcf (GD) file per channel.

Table 27.2-3 Necessary ANSYS generated files for an *fe-safe* PSD analysis.

27.2.3 Files required: NASTRAN

The necessary files are specified in Table 27.2-34. It is assumed that the modal stress data is created in a Modal Analysis (SOL 103) whilst the associated Generalized Displacement data is generated in a Frequency Response Analysis (SOL 111) and written to the punch file (.pch). The stress data in the .op2 file and the GDs in the .pch files can then be combined (by *fe-safe*) to calculate the frequency response functions. Furthermore, it is assumed that for an analysis of n channels the frequency range of each Harmonic Analysis (leading to an applied .pch file) are identical.

The content of each .pch file does not explicitly state the associated channel (loading location and direction). Such information is necessary for a multi-channel PSD analysis since the calculations outlined in Figure 27.1-1 are carried out on a per-channel basis. To overcome this problem the following naming convention must be obeyed. For n channels there will be n .pch files. It is expected that each .pch file has a unique channel-specific number at the end of its name, located between a '_' and the file extension. It is assumed that such channel identifiers are numbered in a continuous manner (from 1 to n), e.g. say an .op2 file has two associated .pch files then these files should be called x_1.pch and y_2.pch (where x and y denotes a valid file name).

Data	File extension	Data Description
Stress Datasets	.op2	Stress data per node, per mode.
Generalized Displacements	.pch	Frequency (Hz) vs. Generalized Displacement data (per mode) in rectangular or polar form. It is assumed that there will be one punch file per channel.

Table 27.2-4 Necessary NASTRAN generated files for an *fe-safe* PSD analysis.

27.24 Opening Finite Element models for PSD Analysis in *fe-safe*

To Open an FE model for use in a PSD analysis select **Open Finite Element Model For PSD Analysis...** from the **FEA Solutions** section of the **File** menu. The dialogue box shown in

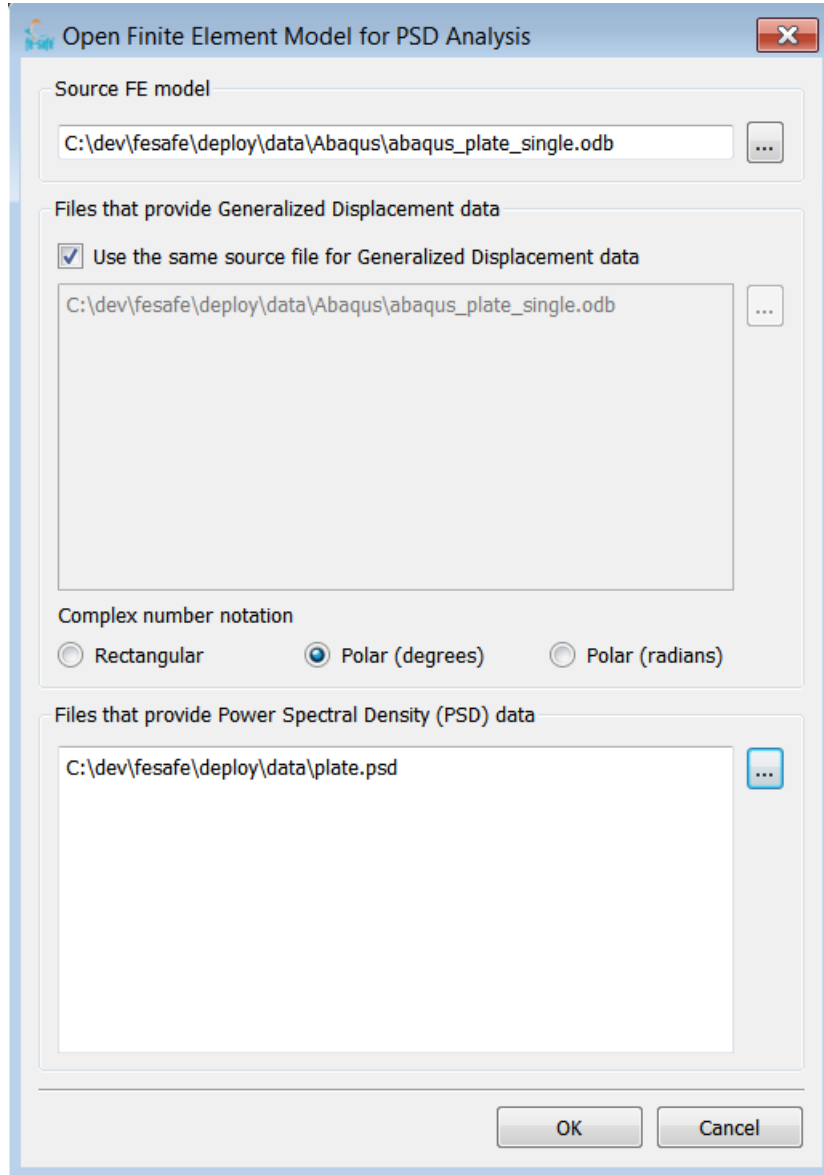


Figure 27.2-2 will then be displayed. Under the **Source FE model** section select the Abaqus .odb, ANSYS .rst, or NASTRAN .op2 file which contains the modal stresses. In the **Files that provide Generalized Displacement data** section select either the .odb containing the Generalized Displacement data steps in the case of an Abaqus model, or all the .mcf or .pch files containing this data for an ANSYS or NASTRAN model respectively. If the source model is an ODB then by default the *Use the same source file for Generalized Displacement data* checkbox is set, and it is only necessary to select the one source model; alternatively deselect the checkbox and select the second .odb file containing the Generalized Displacement data if this is stored in a different file.

The complex Generalized Displacements being imported into *fe-safe* can be expressed in either polar or rectangular form (this data will be converted to rectangular form for use in the PSD loading process in *fe-safe*). By default, an Abaqus Steady State Dynamics Analysis exports such data in polar form, i.e. with modulus and

argument components (where the angles are expressed in degrees). Meanwhile, the default settings for an ANSYS Harmonic Analysis or NASTRAN Frequency Response Analysis results in complex-valued data that is exported in rectangular form, i.e. with real and imaginary components. With the above in mind, it is imperative that the appropriate **Complex number notation** radio button is selected by the user.

Finally, in the **Files that provide Power Spectral density (PSD) data** section select the files containing PSD data. Click **OK**, then the option to pre-scan the file will be displayed and the procedure for Selecting datasets to read will proceed as with other pre-scanning operations (see section 5).

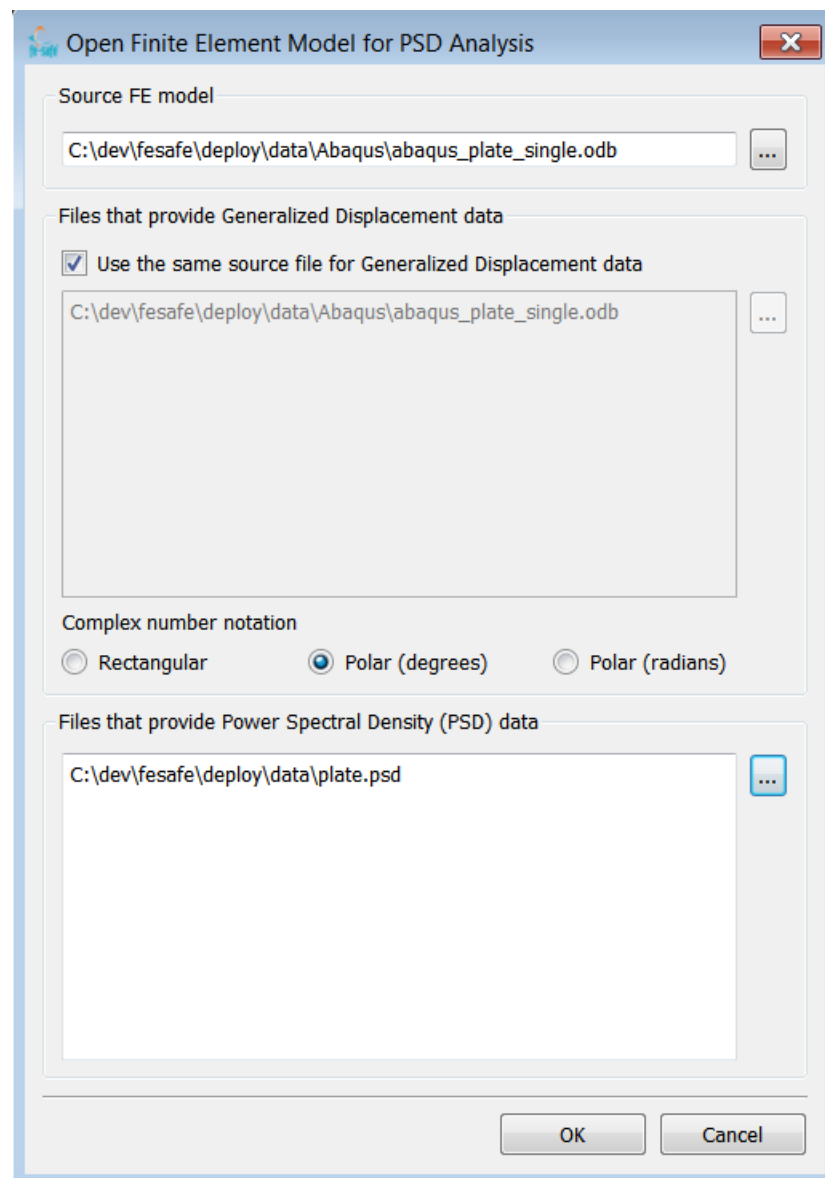


Figure 27.2-2 Open Finite Element Model for PSD Analysis dialog box.

27.3 Format of PSD data

27.3.1 Background

To understand the file format it is useful to understand how the PSD spectra are used in part of the calculation procedure. The simplest case is that which neglects the contribution of CSDs. Here, the user has to supply the real components of the PSDs over m discrete frequencies for n channels (assuming that each PSD has been measured with respect to the channels defined by the FEA software - if the PSD data is gathered before the FE analysis the channels will be numbered with respect to the experimental setup instead). At run-time a set of m matrices will be formed, i.e.

$$\begin{pmatrix} PSD_{11}(f_j) & 0 & \dots & 0 \\ 0 & PSD_{22}(f_j) & \dots & \dots \\ \dots & \dots & \dots & 0 \\ 0 & \dots & 0 & PSD_{nn}(f_j) \end{pmatrix} \quad (1)$$

where $j = 1, \dots, m$ and the $PSD(f)$ entries represent the PSD terms. Such data can be viewed as a single *fe-safe* loading block (see section 13) and can be used in combination with the modal stresses and generalized displacements in order to calculate the response PSD (per node).

Matrix (1) is a special case of

$$\begin{pmatrix} PSD_{11}(f_j) & Cross_{12}(f_j) & \dots & Cross_{1n}(f_j) \\ Cross_{12}^*(f_j) & PSD_{22}(f_j) & \dots & \dots \\ \dots & \dots & \dots & Cross_{n-1n}(f_j) \\ Cross_{1n}^*(f_j) & \dots & Cross_{n-1n}^*(f_j) & PSD_{nn}(f_j) \end{pmatrix} \quad (2)$$

where the $Cross(f)$ entries represent the complex CSDs in rectangular form, i.e. such data is assumed to have real and imaginary components. If the user possesses CSD data then further cases, or loading blocks, may be constructed by creating case-specific combinations of matrix (2). Note that the matrix is Hermitian [7]. So, given n sets of PSD data, i.e. one set per channel, calculations can be implemented for *any* unique combination of cross correlation components above the matrix diagonal, over m discrete frequencies.

To clarify the above, consider a three channel example where PSD spectra are provided over, say, 100 discrete frequencies. Here, a loading block that neglects the contribution of the cross correlation terms will make use of the diagonal terms only, i.e. the following matrix will be formed (at run-time)

$$\begin{pmatrix} PSD_{11}(f_j) & 0 & 0 \\ 0 & PSD_{22}(f_j) & 0 \\ 0 & 0 & PSD_{33}(f_j) \end{pmatrix} \quad (3)$$

where $j = 1, \dots, 100$.

If CSD data is available (over the entire frequency range) then seven further loading blocks can be created by considering any unique combination of the three components above the matrix diagonal, e.g.

$$\begin{pmatrix} PSD_{11}(f_j) & Cross_{12}(f_j) & 0 \\ Cross_{12}^*(f_j) & PSD_{22}(f_j) & 0 \\ 0 & 0 & PSD_{33}(f_j) \end{pmatrix} \quad (4)$$

would characterise one suitable loading block and

$$\begin{pmatrix} PSD_{11}(f_j) & Cross_{12}(f_j) & 0 \\ Cross_{12}^*(f_j) & PSD_{22}(f_j) & Cross_{23}(f_j) \\ 0 & Cross_{23}^*(f_j) & PSD_{33}(f_j) \end{pmatrix} \quad (5)$$

would characterise another.

Given suitable modal stress data, the FEA Fatigue analysis process can then be implemented (per loading block).

Note: Given matrix (2), it is possible to use the *coherence function* [8] to provide a quantitative estimate of causality between two sets of PSD spectra (per loading block); i.e. at frequency j the cross correlation term at row p , column q should satisfy

$$0 \leq |Cross_{pq}(f_j)|^2 / (PSD_{pp}(f_j)PSD_{qq}(f_j)) \leq 1. \quad (6)$$

Failure to satisfy this inequality will indicate that unsuitable PSD data has been provided by the user.

27.3.2 File format

A description of the expected data is outlined in Table 27.3-1. All PSDs must be expressed in units squared per Hz and must be consistent with the FEA software generated data, e.g. if an excitation *force* is used to generate the modal stresses then all PSD spectra should be expressed in N²/Hz. Given that this is user-supplied data a suitable file format has been designed.

Data	Data Description
PSD	Frequency (Hz) vs. real-valued PSD data per channel.
CSD	Frequency (Hz) vs. real and imaginary-valued cross correlation data per channel combination.

Table 27.3-1 User-supplied PSD data required for fatigue calculations.

To provide the necessary input data to *fe-safe* the user must create a file for every loading block under consideration, e.g. the cases characterised by matrices (3) to (5) would require three files (see section 27.3.1). Each file should be an ASCII file using ANSI encoding with a .psd extension and should contain PSDs and CSDs (if available) over the frequency range of interest, which in turn, will indicate the matrix configuration (per loading block). With the above in mind a typical PSD file needs to be formatted as follows:

- Related PSD and cross correlation data (for n channels) should be written in one ASCII file.
- Any comment line should start with a '#'.
- The first non-empty, non-commented line is assumed to be a header specifying the number of channels. The line must contain the text "Number of channels =" and must end with the user-specified value for n .
- A second optional header line may be used to specify the associated signal time length in the form Exposure time = <t>, where <t> is the signal length in seconds.
- After the initial header line it is assumed that n sets of 2-columned PSD data will be provided, i.e. columns of frequency and real-valued data. Each set must be separated by either an empty line or a comment line.
- After the sets of PSD data further 3-columned sets of CSD data, i.e. frequency, real and imaginary-valued data, may be defined. If there is no CSD data, i.e. a loading block characterised by matrix (1) (see section 27.3.1), then the space after the last PSD data set should be empty (or contain comments). Alternatively, if the user wants to supply CSD data, i.e. any unique loading block configuration characterised by matrix (2) (see section 27.3.1), then $n(n-1)/2$ sets of 3-columned data must exist in ascending column (then ascending row) order. As mentioned earlier the associated complex conjugate entries, i.e. those below the diagonal in matrix (2), are not required.
- A single row of zeroes (per matrix entry) is sufficient to represent zero-valued CSD data over the entire frequency range.

An example of a typical file is displayed in

Figure 27.3-1. Note that this file contains data for 1001 matrices based on the template represented by matrix (5) (see section 27.3.1).

```

Number of channels = 3

#PSD11_frq      PSD11_real
0.0             0.0014311
0.1             0.0001870
...
100.0          0.0000001

#PSD22_frq      PSD22_real
0.0             0.0014895
0.1             0.0001932
...
100.0          0.0000021

#PSD33_frq      PSD33_real
0.0             0.0014821
0.1             0.0003933
...
100.0          0.0000003

#CPSD12_frq     CPSD12_real     CPSD12_imag
0.0             0.0014821     0.0000121
0.1             0.0003933     0.0000212
...
100.0          0.0000024     -0.0000023

#CPSD13_frq     CPSD13_real     CPSD13_imag - no cross correlation data
0.0             0.0000000     0.0000000

#CPSD23_frq     CPSD23_real     CPSD23_imag
0.0             -0.0013822     0.0000721
0.1             -0.0003233     0.0000270
...
100.0          0.0000014     -0.0000021

```

Figure 27.3-1 An example of a suitably formatted three-channel .psd file with data provided over a frequency range of 0 to 100Hz (at increments of 0.1Hz).

CSD (or single PSD) format files can be derived from time signals by using the Cross-Spectral Density Matrix File option in the Frequency menu of fe-safe. This uses a 10% buffer overlap with cosine tapering and defaults to 1024 bins in the FFT buffer (this may be reduced for short signals with fewer than 1024 datapoints). The exposure time is also written to the PSD file header. The output .psd file is created in the project results directory (<project>/results), and users may wish to rename it. Note that for single channel signals there is also an option on the Frequency menu called Power Spectrum Density (PSD), which generates a single channel PSD file in .dac format. This is less convenient for PSD fatigue analysis, as it would be necessary to manually convert the file format, by re-saving in some Ascii format and then providing the necessary headers and frequency column. It is therefore recommended

that the general Cross-Spectral Density Matrix File option be used to generate PSD files from time signals, even for the simple case of a simple PSD for a single signal. The frequency domain signal processing options are described further in Chapter 10.

When multiple PSD files are being used, i.e. when there is more than one loading block per analysis, it is assumed that:

- The frequency values per .psd file are identical.
- The frequency values per .psd file must match those specified in other .psd files.
- The frequencies in a .psd file may differ from those specified in the generalized displacement data.

The response PSD frequency set is restricted to the larger lower bound of the .psd and generalized displacement data and the smaller upper bound (so no extrapolation is performed), and is set to the union of the input PSD and generalized displacement data frequency sets lying within these joint bounds. The response PSD for each analysis node at each such frequency is computed by combining the input PSD channels, the generalized displacements, and the modal stress tensors, with interpolation as required, and appropriate projection in the critical plane approaches. Algorithm details are given in [9,10].

27.4 PSD analysis options

Analysis options for a PSD analysis can be accessed by selecting **Analysis Options** from the **FEA Fatigue** menu and navigating to the PSD tab.

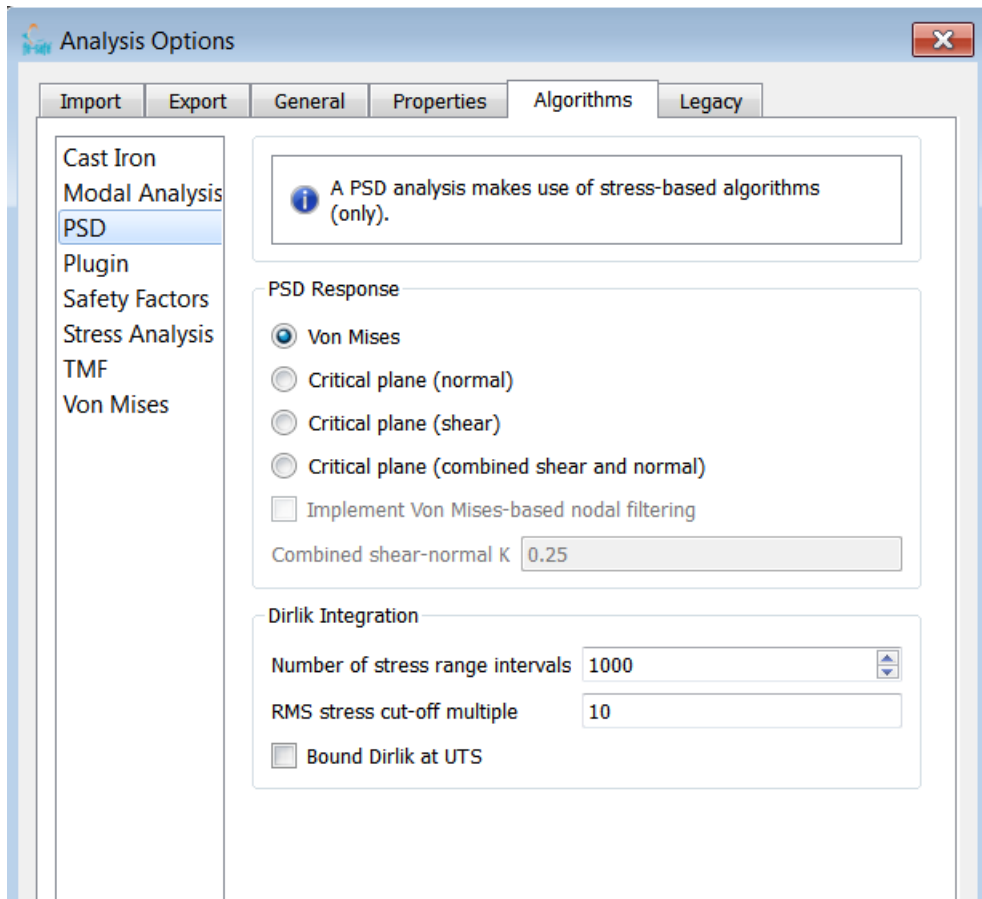


Figure 27.4-1 shows the options available.

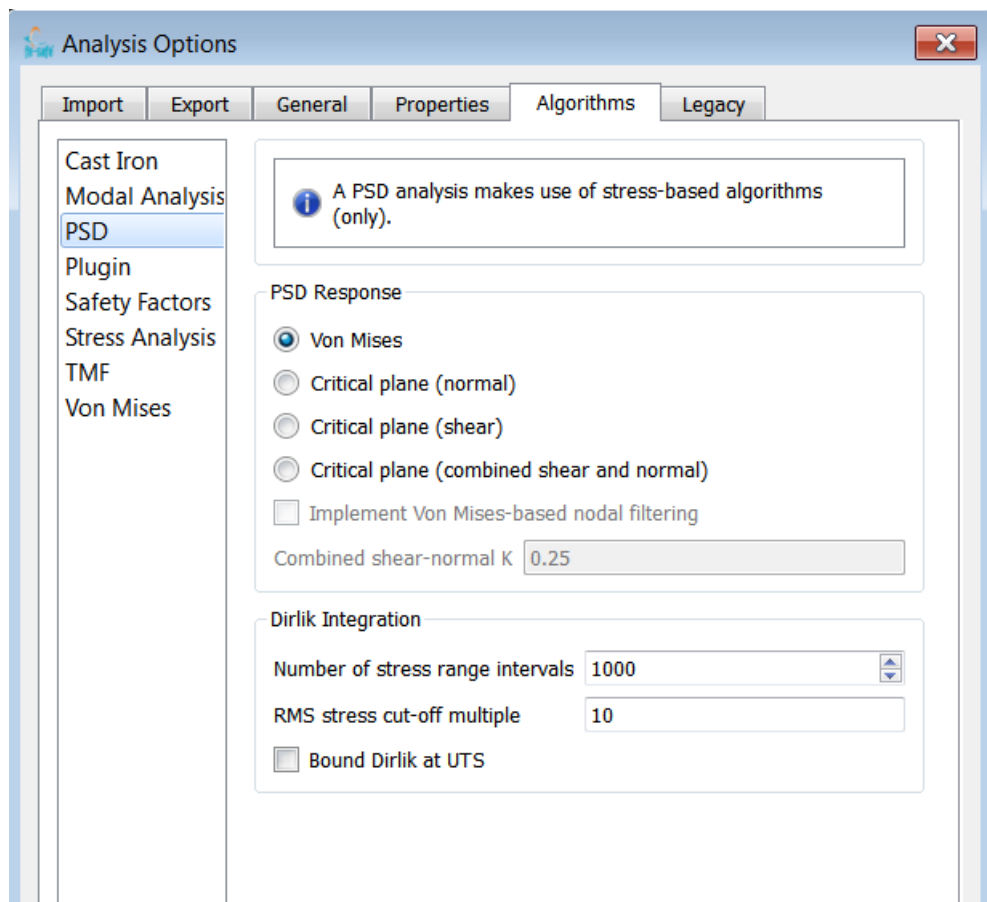


Figure 27.4-1 PSD analysis options

At present *fe-safe* offers four methods to calculate the PSD of the damage parameter:

- Von Mises stress.
- Critical plane search – normal stress evaluation.
- Critical plane search – shear stress evaluation.
- Critical plane search – combined shear and normal stress evaluation.

A description of each method is beyond the scope of this document (see refs. [2,9,10,11] for further details). However, note that in *fe-safe* the granularity of the critical plane search can be varied by selecting the **Critical plane search count** field in **FEA Fatigue->Analysis Options->General tab** (see section 5). For most cases the default value of 18 (which leads to a search increment of 10 degrees) should suffice. The combined normal and shear algorithm is taken from Macha & Nieslony [11], and uses the same set of critical planes as the shear algorithm. So this can be viewed as a kind of modified shear method, where some contribution of the normal stress σ_n to the damage is included. The normal contribution is controlled by a configurable parameter k (in $[0,1]$) which can be set in the above dialog (default 0.25). For ductile materials a similar parameter in the Findley (time-domain) algorithm is in the range $[0.2,0.3]$. The damage parameter is in effect:

$$\frac{2\tau_s + k\sigma_n}{1 + k}$$

The *Implement Von Mises-based nodal filtering* check box is a potential speed-up option which is available when a critical plane option is selected. If checked, the box indicates that *fe-safe* will implement 'nodal filtering'. Response PSD moments will be initially calculated (for all nodes) by using the Von Mises stress, and only nodes with significant stress (i.e. finite life below constant amplitude endurance limit (CAEL)) will be further processed using a critical plane search. In models where most of the lives are infinite, this allows faster processing of the majority of nodes which undergo no (or low) damage. More precisely, nodes with very low stress (RMS below 15% of CAEL fatigue strength) are immediately filtered out, whereas nodes with obviously significant stress (RMS exceeding 40% of CAEL fatigue strength) are immediately passed on for critical plane processing. Nodes with RMS values in between these thresholds have an approximate life calculated using a conservative narrowband approximation of Bendat [12], for which an analytical solution is available for expected life, with a 20% error margin applied to the Von Mises RMS. If this conservative life is below the CAEL then the critical plane processing is invoked.

The damage integral for the Dirlik algorithm is affected by the setting of the *RMS stress cut-off multiple*. It is recommended that the default value be normally retained. Also note that these settings (cut-off and *Number of stress range intervals*) are only applied to the Dirlik algorithm. The damage is upper bounded at the value implied by the limit, and the remaining tail of the stress PDF is integrated using this damage upper bound (or 1 if the damage would be more than 1). Also note that Dirlik's algorithm is defined in terms of stress ranges (not amplitudes), and so the limit in the case of Dirlik is applied to the stress range (not amplitude). Hence the default setting of 10 can be thought of as covering 5 standard deviations of the amplitude distribution. The Tovo-Benasciutti method has a more complicated way of handling the integral, and limits are affected by the mean stress under consideration. Therefore for Tovo-Benasciutti the limits are always the lower of the SN curve intercept point or 5 RMS values, subsequently modified by the current mean. Finally the number of stress range intervals is also only applied to Dirlik, since with Tovo-Benasciutti there is a closed form for the integral for single-segment SN curves, and otherwise a lower number of 100 intervals is used when also doing a double integral over the randomly varying mean. If running Dirlik on a large model a small speed-up can be obtained by reducing the *Number of stress range intervals*. It can typically be dropped to 100 without materially affecting accuracy, but values under 50 are not recommended.

There is a further option, selectable by checkbox, to apply a further bound to the Dirlik damage integral at the Ultimate Tensile Strength (UTS) of the configured material. If the UTS is lower than the SN curve intercept point, then the effect is to use the UTS in place of the SN curve intercept as an additional bound on the upper limit of the integral, after which the tail is treated as having damage of 1. Use of this option is usually over-conservative at low life, as for most materials the UTS is lower than the SN curve intercept, but is provided for backwards compatibility with earlier versions of *fe-safe* (6.5-02, 6.5-03, 2016), or for when a material's SN curve is not regarded as valid beyond the UTS. For the medium to high life region, use of this option will have little or no effect, as the stress range limit would already be below the UTS.

fe-safe calculates fatigue results using either the Dirlik method [1] or the Tovo-Benasciutti method [2-4]. Both provide a closed form solution to estimate the Probability Density Function (PDF) $p(S)$ of stress range S from the spectral moments of the response PSD, and hence calculate a histogram of Rainflow cycle ranges. Expected fatigue damage can be calculated from this cycle histogram by integration of $D(S)p(S)$, where $D(S)$ is the damage incurred by a cycle of range S . Earlier versions of *fe-safe* (up to *fe-safe* 2016) provided only Dirlik's algorithm for

converting the response PSD spectral moments to a PDF. Dirlik's PDF is a semi-empirical mixture model of three distributions which suffers from two issues:

- a) It only assesses cycle amplitudes, and there is no adjustment for cycle means, neither random variation in the mean, nor a non-zero global mean due to residual stress effects.
- b) The Dirlik formula is semi-empirical and although it appears to work fairly well, it lacks a sound theoretical basis.

These drawbacks were addressed in the work of Tovo and Benasciutti culminating in the paper published as [2]; further theoretical details are given in [3] and [4]. Note, however, that the theoretical justifications given by Benasciutti in his PhD thesis [4] are for stationary Gaussian processes.

The method sums a weighted combination of two damage terms: a narrowband component, and a wider band range counting component. Both are Rayleigh distributions in amplitude, but with different variances, and the second also has a Gaussian PDF on the cycle mean.

The selection of Dirlik or Tovo-Benasciutti method is made by double clicking on the Algorithm tab in the Analysis Settings tab of fe-safe. This results in a PSD-specific algorithm dialog popping up as shown below.

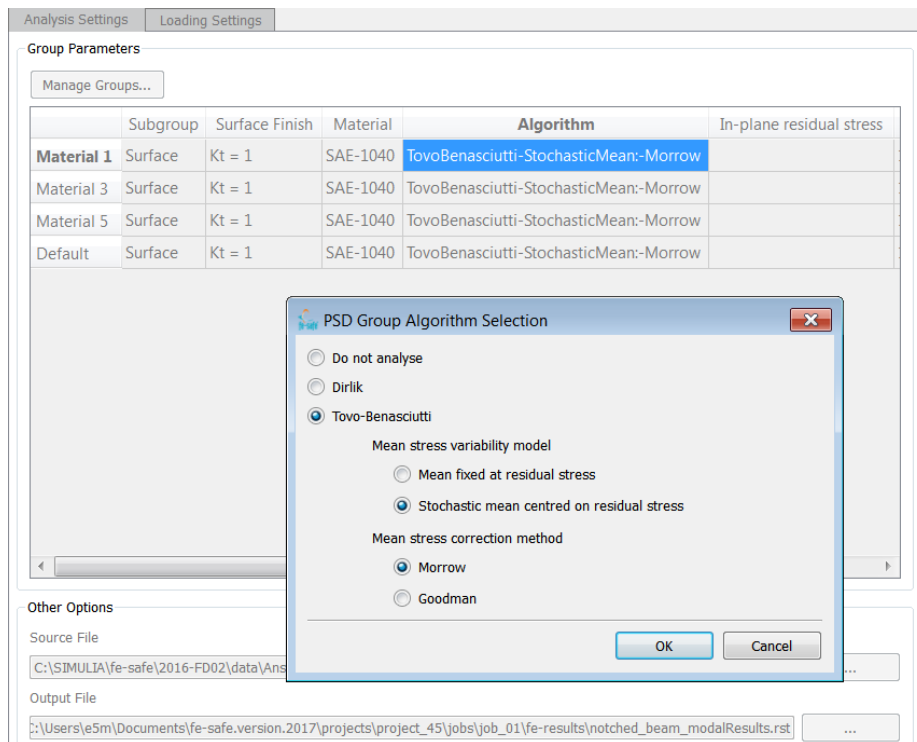


Figure 27.4-2 PSD Algorithm options

If a Tovo-Benasciutti algorithm is selected then the two radio buttons pairs for the mean stress variability model and the mean stress correction are activated. The mean stress used in Tovo-Benasciutti can either be set to a fixed value determined by the residual stress, or this can be used as the centre of a Gaussian distribution used to model the stochastic effect of random variation in individual cycles.

The employed mean stress correction takes the Goodman/Morrow form for positive mean m :

$$S' = S \left(1 - m/S_L \right)^{-1}$$

The limit stress S_L can be set to either the stress which gives damage of 1 on the SN curve (Morrow the default), or the UTS (Ultimate Tensile Strength), which is the (typically-over-conservative) Goodman correction.

When the stochastic mean form of Tovo-Benasciutti is used, then as well as integrating the expected damage over the Rayleigh distribution of stress, the (wide band) range counting component is also integrated over the Gaussian distribution of mean stress. This will produce more damage than using a fixed mean. Note that the stress correction can asymptote to infinity as the mean stress approaches the limit amplitude. This can be a problem in the stochastic mean form of Tovo-Benasciutti, where even if the process mean is below the limit, the random distribution can have a tail in excess of the limit. In these circumstances *fe-safe* always constrains the computed damage at 1 so that random mean contributions in the tail do not produce absurd contributions to the expected damage integral. The stress integral is always upper bounded at the SN curve intercept, and any remaining PDF tail is simply assigned an effective damage of 1 (i.e. the component can only be destroyed once).

Note that when the shear algorithm is used, then the S-N curve used for the damage function is based on normal stress, but the shear is converted to an equivalent normal stress by doubling it so that in effect equivalent normal stress is given by (see [11]):

$$\sigma_{eqv} = 2\tau$$

If a non-default surface finish is specified ($K_t > 1$), then K_t is used to scale the stress integration axis, so for stress range S with probability density $p(S)$, the damage term is $D(K_t S)$. When performing a FOS analysis, the evaluated scale factor is applied to the stress axis of the damage integral in a similar way, rather than scaling the input loading (as that would be equivalent to a quadratic scaling. Note that the mean stress is not multiplied by K_t).

To allow *fe-safe* to calculate expected damage the following information must be provided [1]:

- a) Material parameters to define the S-N curve for the material (see section 8).
- b) PSD loading block exposure time, i.e. the amount of time that the component is exposed to the load case. This may be provided in the header of the PSD file, or specified later in the loading definition.
- c) Suitable settings for the granularity of the integration step and a value of k to define the maximum stress range¹, i.e. $k \cdot RMS$ (only used in Dirlik method).

If no S-N curve is provided then the strain-life curved may be used instead with an elastic conversion. Like other *fe-safe* stress algorithms, this depends on the setting of the **Use SF and b if no SN datapoints** checkbox (see Stress Analysis under the Algorithms tab of the Analysis Options dialog). Also note that multi-segment SN curves may be used. The damage function defined in references [1] and [2] is a fixed power law, equivalent to a single segment SN curve, but *fe-safe* will perform the PDF integral using a more general multi-segment SN curve if required. However this will result in a somewhat slower run-time, especially if the stochastic mean Tovo-Benasciutti option is used.

To calculate safety factors for infinite life, a FOS calculation at infinite life should be used, rather than the FRF calculation provided in some earlier versions of *fe-safe* (6.5-00 and 6.5-01). This has been removed because there were statistical difficulties in providing an accurate standard deviation scaling (a value for k) for the FRF over long

¹ A value ≥ 10 is recommended.

time scales, and the FOS calculation takes better account of smaller cycles. Note however that the FOS scaling produces the desired target life as the expected life, but due to random variability that may not be the life actually achieved in any specific instance. It is therefore recommended that a slightly conservative approach to FOS calculations be adopted.

If there are significant residual stresses present then one of the Tovo-Benasciutti algorithms should be used, as any overall mean effects will be ignored in Dirlik. The residual stress can be set on a group-wise basis by either using the Residual Stress column of the Analysis Settings tab (assumed isotropic), or by providing a residual stress dataset in an appended finite element model. The modal analysis datasets must always be loaded first using **Open Finite Element Model For PSD Analysis.....** Then if there is a dataset relating to a residual stress analysis, then that may be loaded using **Append Finite Element Model...** from the File menu. Then the residual dataset may be added to the Transitions Block on the Loading Settings tab using Replace Residual Dataset on the popup menu (the required dataset must be first selected). Note that this option was originally provided for elastic-plastic analyses, and therefore a limitation of the user interface is that an associated strain dataset must also be supplied, even though this will not be used in the PSD analysis (see section 13 for details of *Defining elastic-plastic residual stresses*). The residual stress tensor is projected onto the required critical plane when running critical plane searches to obtain the mean stress used in the Tovo-Benasciutti algorithm. When the stochasticmean option is selected the expected damage is integrated over both amplitude and randomised mean centred on the overall mean for the residual. The Tovo-Benasciutti algorithm defines a Gaussian distribution for the actual mean of a random cycle, but this is centered on the defined residual. If a Von Mises analysis is performed then there is no direction onto which the residual tensor should be projected, so the trace of the tensor is used instead.

27.5 Technical summary of the Tovo-Benasciutti algorithm.

The remainder of this chapter provides a technical summary of the Tovo-Benasciutti algorithm.

The Tovo-Benasciutti algorithm commences by computing distribution parameters for the range-mean counted stress amplitude PDF, and also mean stress PDF if random variability in mean was selected. These are derived from the response PSD spectral moments ($\{\lambda_v\}$), according to equation (21) in [2].

The damage is computed by summing a weighted combination of expected damage from this PDF, with expected damage from a narrowband PSD with an associated Rayleigh distribution of stress amplitude S according to the standard formula of Bendat [12] :

$$p_{nb}(S) = \frac{S}{\lambda_0} \exp\left(-\frac{S^2}{2\lambda_0}\right)$$

The weighting factor b between narrowband and range-mean damage is again a function of the spectral moments and is given in equation (17) of [2]. Note however that when combining narrowband and range-mean damage contributions for the signal, we need to account for the fact that they refer to different process rates (mean upcrossing rates ν_0 and rate of peak occurrence ν_p , respectively, derived from the moments via Rice's standard formulae). This means that the effective narrowband weighting is given by

$$b' = \alpha_2 b \text{ and } \alpha_2 = \frac{\lambda_2}{\sqrt{\lambda_0 \lambda_4}}$$

The damage integrals over amplitude and mean are limited by a stress limit set to the smaller of the UTS and the stress amplitude at which the damage is one. This is used to limit the damage integration at $S_L - |m|$.

For the fixed mean variant m is always m_c (derived from the appropriate residual if defined, otherwise zero). For randomised mean, an outer integration loop is performed over the mean (for range-mean damage term) using the

Gaussian PDF of mean stress (which is centred on m_c), see equation (42) in [2]. The general form for a signal of time length T seconds is:

$$D = \nu_p T \left[\alpha_2 b d_{nb} + (1 - b) \int_{-S_L}^{S_L} G_\mu(m, m_c) \left(\int_{S_0}^{S_L - |m|} D(S'(m)) R_a(S) ds + D_L (1 - \rho_a(S_L - |m|)) \right) dm \right]$$

where the first term represents the expected narrowband damage derived from integrating the narrowband Rayleigh distribution with the (mean stress corrected) damage function $D(S'(m))$; $G_\mu(m, m_c)$ is the Gaussian pdf of the mean stress; $R_a(S)$ is the Rayleigh pdf of the range counted damage for amplitude (see equation (21) in [2]) with cdf $\rho_a(S)$; and the limiting damage is D_L . Note that fe-safe does not accrue damage at low stress; a lower bound stress S_0 is calculated based on the CAEL (this is passed to PSD as a material property and is normally a fixed fraction of the CAEL stress).

The narrowband damage d_{nb} is obtained by a straightforward integration of the narrowband pdf, so

$$d_{nb} = \int_{S_0}^{S_L - |m_c|} D(S'(m_c)) p_{nb}(S) ds + D_L (1 - \rho_{nb}(S_L - |m|))$$

Above the limit the remaining part of the stress amplitude distribution does not use the damage function as the component cannot be destroyed more than once. So when integrating the amplitude tail, the limit damage D_L is set to the damage value at the limit $D(S_L - |m|)$, or 1 if:

$$3\sqrt{\lambda_2} > S_L - |m|$$

The latter condition is where the fe-safe implementation departs from the original Tovo-Benasciutti algorithm. This is because limiting the damage function value at $S_L - |m|$ (as in equations (38) and (45) in [2]) gives anomalous low damage and long life at large negative mean stress, even though this is supposed to result in component destruction. However the use of very large mean stress near or beyond the limit stress would be pushing the PSD fatigue analysis beyond its intended application, as it is really intended for medium to high cycle fatigue.

27.6 References

- [1] T. Dirlik, "Application of Computers in Fatigue Analysis", University of Warwick Thesis, 1985.
- [2] D Benasciutti and R Tovo, 2006. "On fatigue damage computation in random loadings with threshold level and mean value influence." *Struct. Durability Health Monitoring 2* (2006): 149-164.
- [3] D Benasciutti and R Tovo. *Rainflow cycle distribution and fatigue damage in Gaussian random loadings*. No. 129. Internal report, 2004, Dipartimento di Ingegneria ,Universita degli Studi di Ferrara, Italy.
- [4] D Benasciutti. Fatigue Analysis of Random Loadings. PhD Thesis, 2004, University of Ferrara, Italy.
- [5] Abaqus software documentation.
- [6] ANSYS software documentation.
- [7] H. Anton, "Elementary Linear Algebra", John Wiley & Sons, 2000.
- [8] C. Lalanne, "Mechanical Vibration and Shock Analysis, Random Vibration", Wiley, 2013.
- [9] G.M. Teixeira et al., "Random Vibration Fatigue – Frequency Domain Critical Plane Approaches", ASME, IMECE2013-62607, 2013.
- [10] G.M. Teixeira et al., "Random Vibration Fatigue-A Study Comparing Time Domain and Frequency Domain Approaches for Automotive Applications", No. 2014-01-0923, SAE Technical Paper, 2014.
- [11] E Macha and A Nieslony (2012). Critical plane fatigue life models of materials and structures under multiaxial stationary random loading: the state of the art in Opole Research Centre CESTI and directions of future activities. *International Journal of Fatigue*, 39:95-102, 2012.
- [12] Bendat JS. (1964). "Probability functions for random responses." NASA report on contract NAS-5-4590.

Technical Note 1: Approximating materials data

fe-safe has a materials approximation algorithm, accessible from the 'Options' button in the materials data base. This generates strain-life data for steels and for aluminium alloys, using the material's elastic modulus E and ultimate tensile strength. This algorithm has been shown to be reliable for a range of commonly used steels and aluminium alloys.

However, the user may have additional information available. In particular, a traditional S-N curve may be available for a cylindrical specimen tested at zero mean stress under axial loading. This note suggests a method for incorporating this information. Reference should be made to the Fatigue Theory Reference Manual section 3 for background information.

First run the materials approximation algorithm in the materials data base, using the appropriate values of E and Ultimate Tensile Strength.

The stress-life curve may be defined as shown in *Figure 1.1*. In the high cycle regime, (say) between 10^5 and 10^7 cycles, the slope of the S-N curve and the slope of the local stress-life curve will be very similar. The parameter b may therefore be obtained from the S-N curve and will replace the value calculated from the approximation algorithm.

The S-N curve may also define the stress amplitude at 10^7 cycles, or some other high cycle endurance. With reference to Figure 1, adjust the stress-life curve to pass through the known data point, keeping the slope b calculated in the previous paragraph. This will produce a revised value of σ'_f

These parameters can replace the values generated by the materials approximation program.

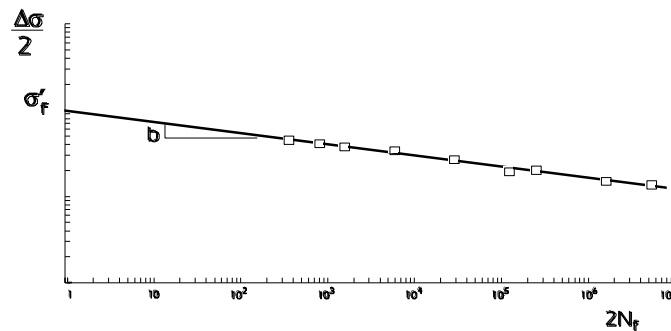


Figure 1-1 Local stress-life curve

The remaining parameters for the strain-life curve generated from the materials approximation routine can be accepted.

An adjustment to the value of σ'_f implies that the relative values of elastic and plastic strain have changed. The value of n' should be re-calculated using

$$n' = \frac{b}{c}$$

The value of K' should be replaced by

$$K' = \frac{\sigma'_f}{(\varepsilon'_f)^{n'}}$$

Technical Note 2: Elastic-plastic FEA for fatigue analysis

1 Introduction

Most fatigue analysis is performed using stresses from an elastic FEA. The conversion from elastically-calculated FEA stresses to elastic-plastic stress-strains is carried out in the fatigue software. The two essential features of the fatigue modelling process are (a) an elastic-plastic conversion routine, and (b) a kinematic hardening model. A common elastic-plastic conversion routine is Neuber's rule, and although other methods are available, these will be all be referred to as Neuber's rule in this document.

In implementing Neuber's rule, each node is treated as a separate entity. The elastic to elastic-plastic conversion cannot therefore allow for the fact that stresses may redistribute from one node to another as a result of yielding. Normally this is an acceptable approximation, because yielding generally occurs in notches. However, there may be instances where gross yielding occurs on a component, and stresses redistribute from one area to another. This may require an elastic-plastic FEA.

In order to set up an elastic-plastic FEA correctly, it is important to appreciate the methods used in the fatigue software. These are described below.

2 Kinematic hardening

The Fatigue Theory Reference Manual, pages 2-20 to 2-22 show an example of the stress-strain response to a sequence of elastic-plastic strains, for uniaxial stresses. The response has been calculated using a kinematic hardening model.

The example is reproduced below (retaining the Figure numbers from the user manual)

Example 2.1

Figure 2.31 shows a short time history of local strain.

The strain values are:

POINT	STRAIN	POINT	STRAIN
A	0.003	D	-0.0025
B	-0.001	E	0.0014
C	0.0014	F	-0.001

A materials data set for SAE1045 steel will be used:

$$\begin{aligned} \sigma'_f &= 948 \text{ MPa} & b &= -0.092 & K' &= 1258 \text{ MPa} \\ E &= 202000 \text{ MPa} & \epsilon'_f &= 0.26 & c &= -0.445 \\ n' &= 0.208 \end{aligned}$$

The stress-strain and endurance curves are shown in *Figure 2.32*.

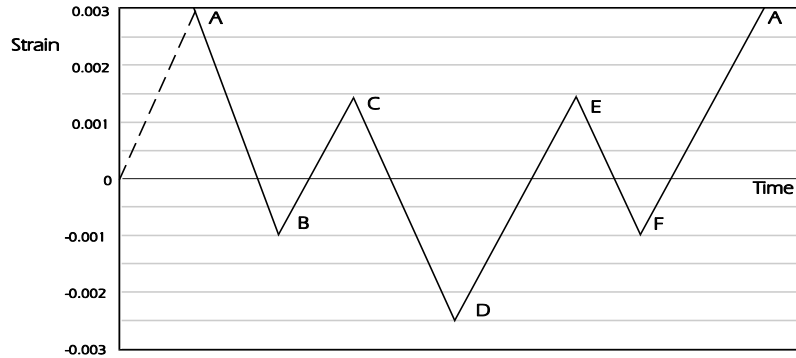


Figure 2.31 Local strain history for example 2.1

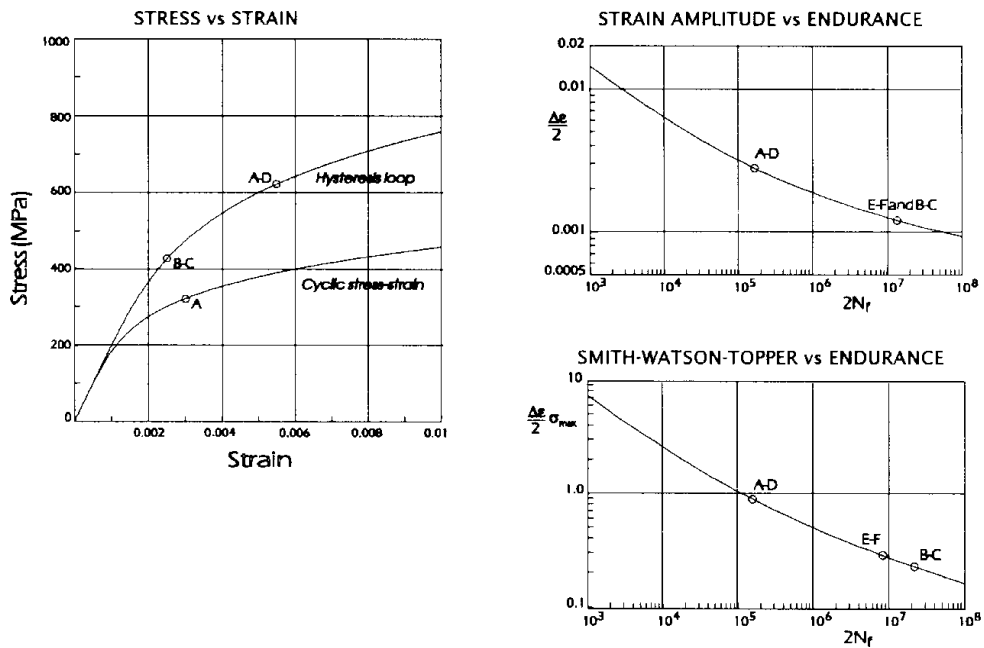


Figure 2.32 Materials data for SAE1045 steel

Referring to *Figure 2.31*.

The strain at point A lies on the cyclic stress-strain curve. A strain of 0.003 is found by iteration to correspond to a stress of 321.1 MPa.

The strain range from A to B follows the hysteresis loop curve, with its origin at A. The strain range is $(0.003 - (-0.001)) = 0.004$. By iteration, the stress range from A to B is 546.3 MPa. The stress at B is therefore $(321.1 - 546.3) = -225.2$ MPa.

The strain range from B to C is $(-0.001 + 0.0014) = 0.0024$. On the hysteresis loop curve, with its origin at point B, this represents a stress range from B to C of 415.1 MPa. The stress at C is $(-225.2 + 415.1) = 189.9$ MPa.

The strain range from C to D closes a hysteresis loop, because the strain range C-D is greater than the strain range B-C. The cycle B-C has a strain range of 0.0024, and a maximum stress at C of 189.9 MPa. Because of the material memory effect, the stress at D is calculated by using the strain range from A to D. The strain range is $(0.003 - (-0.0025)) = 0.0055$. On a hysteresis loop curve with origin at A, the stress range from A is 622.2 MPa. The stress at D is then $(321.1 - 622.2) = -301.1$ MPa.

The strain range from D to E is $(-0.0025 - 0.0014) = 0.0039$. On a hysteresis loop curve with origin at point D, this represents a stress range from D of 540.1 MPa, so the stress at E is $(-301.1 + 540.1) = 239$ MPa.

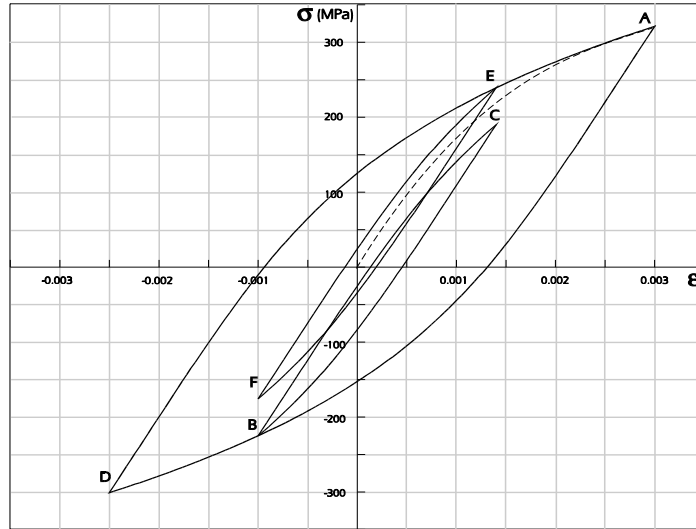


Figure 2.33 Solution to example 2.1

The strain range from E to F is $(0.0014 - (-0.0011)) = 0.0024$. On the hysteresis loop curve with its origin at point E, the stress range is 415.1 MPa, and the stress at F is $(239.1 - 415.1) = -176$ MPa.

The strain range from F to A closes the cycle E-F. Its strain range is 0.0024 and the maximum stress at E is 239.1 MPa. Using material memory, the stress at A is calculated using a hysteresis loop curve with its origin at D. The strain range from D to A is 0.0055, and the stress at A is 321.1 MPa. This strain range has closed the largest cycle in the signal, that from A-D-A. Its strain range is 0.0055, and the maximum stress at A is 321.1 MPa.

A summary of the three cycles is shown in *Figure 2.33*, and in the table below.

CYCLE	$\Delta\epsilon$	σ_{max}
B-C	0.0024	189.9
E-F	0.0024	239.1
A-D	0.0055	321.1

Important features of kinematic hardening are illustrated in *Figure 2.33*. These are

1. Once a closed hysteresis loop has occurred, for example the loop B-C, the 'material memory' phenomena occurs, in that the materials stress-strain response from A to D is calculated as though the closed loop B-C had not occurred.
2. Subsidiary loops (B-C and E-F) have some plasticity associated with them. Isotropic hardening would not produce this effect, because with isotropic hardening the materials yield stress increases to encompass the largest event experienced so far, and so subsidiary cycles would be elastic.

Kinematic hardening is illustrated further in the Fatigue Theory Reference Manual, pages 7-40 to 7-43.

Note that in fatigue analysis, 'yielding' is considered to occur at stresses much lower than the 0.2% proof stress. In *fe-safe*, the yield stress is taken to be the stress at which the difference between the elastically-calculated stress and the elastic-plastic stress is 1% of the elastically-calculated stress.

3 Effect of repeating the loading sequence

The Fatigue Theory Reference Manual, page 2-23, shows the effect that repeating the load history has on the stress values. Extracts from this section are reproduced in *Figure 2.34*.

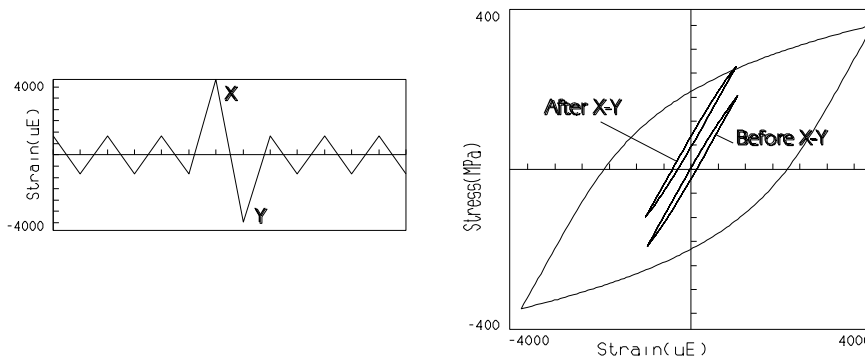
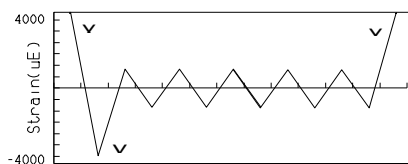


Figure 2.34 Residual stresses after a large strain cycle.

Before the large event X-Y, the small cycles have a zero mean stress. After X-Y, the mean-stress for the smaller cycles has been increased. If the loading represents a 'day in the life' of the component, this effect will only occur on the first 'day'. After this, all the small cycles will have the higher mean-stress.

Fatigue software simulates this effect by starting and finishing the analysis at the numerically largest strain (or stress). The sequence would be analysed as though it consisted of the strain history shown below, i.e. starting and finishing at point X.



Assuming that the fatigue life will be many repeats of this loading, the procedure produces the correct mean stresses for all repeats except the first part of the first repeat. This is considered an acceptable approximation.

In modelling a fatigue loading sequence in elastic-plastic FEA, it is important that this procedure is followed. In the example above, it may be necessary to model the sequence up to point X in *Figure 2.34*, or to model an initial occurrence of point X. The sequence up to the next occurrence of point X should then be modelled. The sequence of stress/strain from X to X (as shown above) is required for the fatigue analysis.

4 Materials data

Many materials cyclically harden or cyclically soften during the first few cycles of fatigue loading, until a stable cyclic stress-strain response is attained (see the Fatigue Theory Reference Manual, page 3-3). Fatigue analysis is carried out using the stable cyclic properties, and it is important that these stable cyclic properties are also used in the elastic-plastic FEA. Conventional monotonic properties should not be used.

5 Discussion

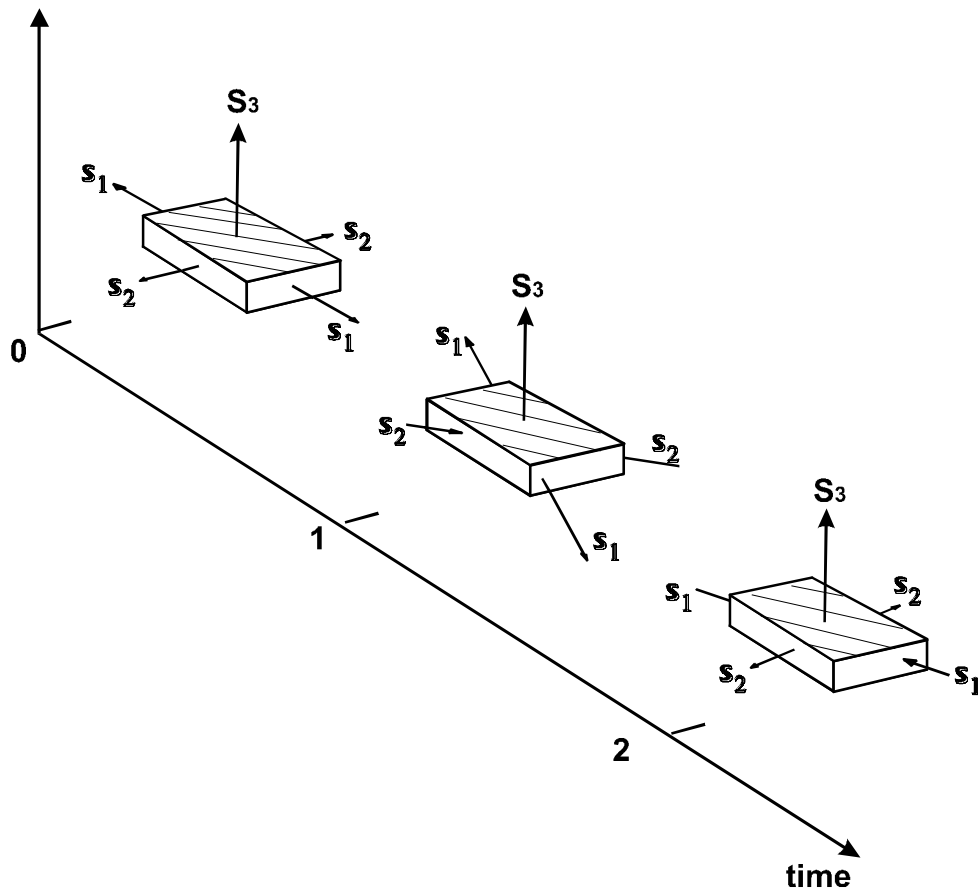
It is clear from the above that care is needed when setting up elastic-plastic FEA for subsequent fatigue analysis. Even when this is done, a series of presentations at user conferences has suggested that elastic-plastic FEA does not generate stress/strain sequences that match those generated by fatigue analysis software. This seems to be related to the way that kinematic hardening for cyclic loading is implemented in the FEA software. As a result, users may see a lack of comparability between the fatigue lives calculated from an elastic-plastic FEA and those calculated from elastic FEA.

Technical Note 3: Treatment of triaxial stresses

1 Introduction

This technical note provides an outline of how *fe-safe* deals with triaxial stress states. This can happen on the surface of components where contact occurs.

fe-safe uses the stress tensor history built by combining the stresses from the Finite Element datasets and load histories to identify the orientation of the surface of the component. The assumption is that two of the principal stresses will lie in the surface of the component and the third will be perpendicular to the surface. The two in surface principals may change direction within the surface of the component during the whole loading sequence but the out-of-plane principal will not. This is shown in the 3 sample dataset sequence figure below. NOTE: The surface is hatched.



Where the third principal is insignificant the stress state is identified as 2 dimensional.

Where the out-of-plane principal stress is significant but the surface shear stresses are not significant, *fe-safe* treats this as a two-dimensional stress state.

Otherwise, the stress tensor is marked as triaxial and the fatigue calculations are performed on 3 planes. In each of

the 3 planes the critical plane approach is used and the worst damage on any of the critical planes is stored.

Prior to version 5.00 the treatment of triaxial stresses was not supported when performing fatigue from elastic-plastic FEA results, and although it was implemented for fatigue from elastic FEA stresses the treatment has been further refined. This may result in the reduction of fatigue lives at nodes where the stress tensor history is treated as triaxial.

2 Detailed algorithm discussion

In more detail *fe-safe* runs through the following procedure to classify the stress tensor history at the node. If the loading comprises of multiple blocks then this is repeated for each block individually.

- Identify a reference sample to use to evaluate the surface orientation. This is usually the sample within the stress tensor history with the largest magnitude principal stress. If the other two principals at this sample are small then the maximum principal sample is not used and the tensor history is scanned sample by sample to find a tensor with at least two significant principals. This reference sample can be exported to the diagnostics log (see section 3). The orientation of the surface is calculated assuming that 2 principals will be in the surface and the third will be perpendicular to it, for this reference sample.
- Transform the whole stress tensor history from it's original axes (XYZ) onto the surface orientated axes. We call these new axes X' Y' Z'.
- Scan the transformed tensors to see if the stress state is 2 dimensional. If it is, the non-zero 2D stresses will be in the surface of the component.

For X'Y' this would occur if X'Z' Y'Z' and Z'Z' stresses are near zero. For X'Z' this would occur if X'Y', Y'Z' and Y'Y' stresses are near zero. For Y'Z' this would occur if X'Y', X'Z' and X'X' stresses are near zero.

- If the stress state is not 2 dimensional *fe-safe* checks if one of the shears is significant and the others are near zero. If this is the case the standard critical plane procedure is used.
- If the stress tensor has not been classified in the previous two steps we have a triaxial stress state.
- For non-triaxial stress states evaluate the fatigue damage using a critical plane approach in the evaluated surface orientation.

It should be noted that for some algorithms the number of planes that are scanned can be reduced from 19 if the orientation of the in surface principals does not change or if the stress state is proportional.

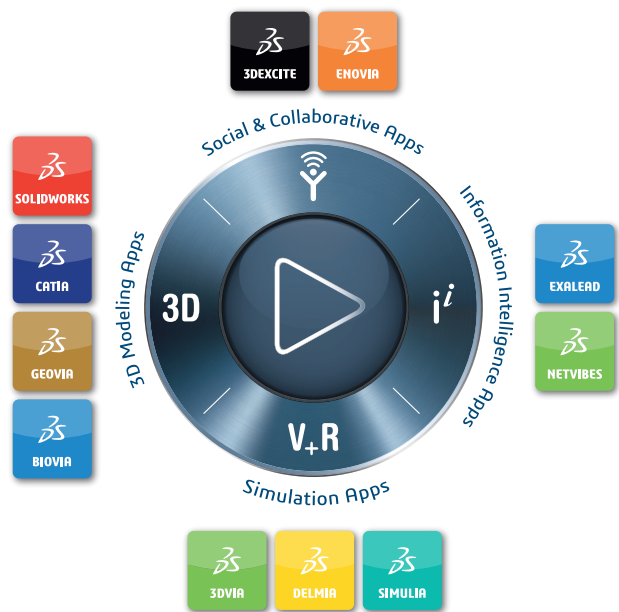
- For triaxial stress states evaluate the fatigue damage using a critical plane approach as though the surface is X'Y, X'Z' and Y'Z' in turn. The worst damage is stored for the node.

About SIMULIA

Dassault Systèmes SIMULIA applications, including Abaqus, Isight, Tosca, and Simulation Lifecycle Management, enable users to leverage physics-based simulation and high-performance computing to explore real-world behavior of products, nature, and life. As an integral part of Dassault Systèmes' 3DEXPERIENCE® platform, SIMULIA applications accelerate the process of making highly informed, mission-critical design and engineering decisions before committing to costly and time-consuming physical prototypes. www.3ds.com/simulia

Our 3DEXPERIENCE® platform powers our brand applications, serving 12 industries, and provides a rich portfolio of industry solution experiences.

Dassault Systèmes, the 3DEXPERIENCE® Company, provides business and people with virtual universes to imagine sustainable innovations. Its world-leading solutions transform the way products are designed, produced, and supported. Dassault Systèmes' collaborative solutions foster social innovation, expanding possibilities for the virtual world to improve the real world. The group brings value to over 210,000 customers of all sizes in all industries in more than 140 countries. For more information, visit www.3ds.com.



©2018 Dassault Systèmes. All rights reserved. 3DEXPERIENCE®, the Compass icon, the 3DS logo, CATIA, SOLIDWORKS, ENOVIA, DELMIA, SIMULIA, GEOVIA, EXALEAD, 3D VIA, 3DSWMM, BIOVIA, NETVIBES, IPWE and 3DEXCITE are commercial trademarks or registered trademarks of Dassault Systèmes, a French "société européenne" (Versailles Commercial Register # B 322 306 440), or its subsidiaries in the United States and/or other countries. All other trademarks are owned by their respective owners. Use of any Dassault Systèmes or its subsidiaries trademarks is subject to their express written approval.