QUICK FATIGUE TOOL FOR MATLAB®

Appendices

Version Information

Documentation revision: 3 [19/08/2017]

Concurrent code release: 6.11-02

Contents

Appendix I. Fatigue analysis techniques

A1.1 Background

Quick Fatigue Tool is capable of assessing the fatigue life of entire FEA models. This is achieved by reading the stress tensor at each analysis item (node, integration point or centroid) and performing a damage calculation at each location in the model.

Before the analysis begins, the FEA definitions are combined with the load history to form a scaled time history of stresses at each location in the model. This is known as a *Scale and Combine* analysis.

The stresses at each location must be considered in multiple directions in order to account for multiaxial stress states. The plane which experiences the largest stress is isolated, and the load signal is cycle counted to obtain the stress amplitudes, so that a damage calculation can be performed.

The damage of each cycle in the loading is summed for each analysis item. The item with the largest damage is identified as the *worst* item and the fatigue life is quoted at this location. During the damage calculation process, the endurance limit may be modified to account for the effect of microscopic plasticity at the crack tip.

This section provides a concise description of how the code achieves the above steps.

A1.2 Combining FEA stresses with a loading

FE models usually comprise of multiple analysis steps, each representing a distinct load configuration for which the stress state is solved. Provided these stresses are elastic, each loading state may be combined with its respective load history to create a single scaled, combined loading, as depicted in Figure A1.1.

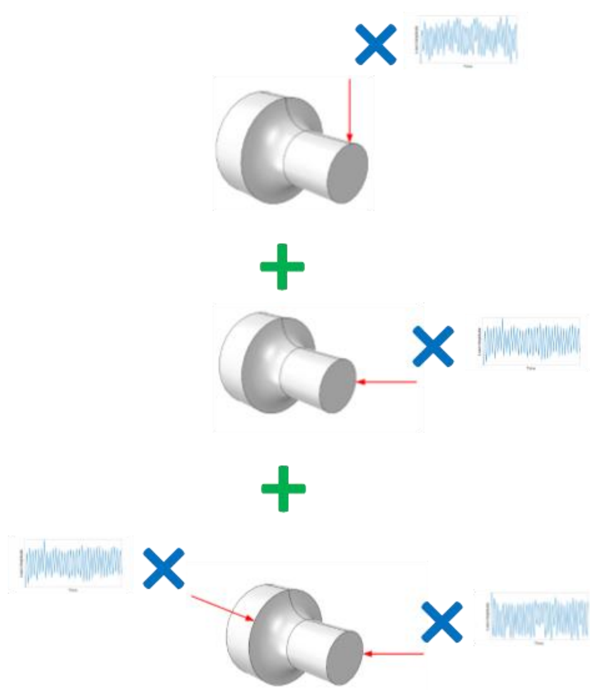

Figure A1.1: Illustration of a scale and combine loading

A1.3 Critical plane search algorithm

Critical plane analysis is a widely used engineering methodology to determine multiaxial fatigue damage where the most damaging stresses are not readily apparent [1] [2] [3] [4]. It has been argued that fatigue criteria cased on critical plane methods are the most effective [5].

Quick Fatigue Tool uses a critical plane approach based on a combination of work by R. Rabb and Luca Susmel.

The stress tensor history at each node is transformed onto a series of planes according to Figure A1.1 [6].

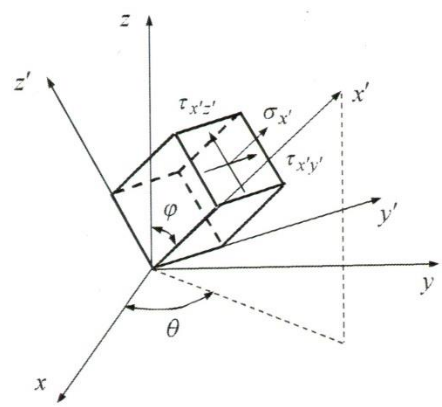

Figure A1.1: Tensor transformation

The rotated tensor, $\boldsymbol{\sigma}'$, is calculated from the original tensor by Equation A1.1.

$$\boldsymbol{\sigma}' = \boldsymbol{Q}^T \boldsymbol{\sigma} \boldsymbol{Q} \qquad \text{[A1.1]}$$

Where $\boldsymbol{Q}$ is the transformation matrix given by Equation A1.2.

$$\boldsymbol{Q} = \begin{bmatrix} \sin\varphi\cos\theta & -\sin\theta & -\cos\varphi\cos\theta \\ \sin\varphi\sin\theta & \cos\theta & -\cos\varphi\sin\theta \\ \cos\varphi & 0 & \sin\varphi \end{bmatrix} \qquad \text{[A1.2]}$$

The angles θ and φ correspond to the cylindrical coordinate system described by Figure A1.1. The rotated tensor is split into one normal and two shear components:

$$\sigma_n = \boldsymbol{\sigma}'(1,1)$$

$$\tau_{xy} = \boldsymbol{\sigma}'(1,2) \qquad \text{[A1.3]}$$

$$\tau_{xz} = \boldsymbol{\sigma}'(1,3)$$

Depending on the algorithm, the normal and/or shear stresses are cycle counted in order to calculate fatigue damage.

A1.4 Rainflow cycle counting

The rainflow counting method is used for the analysis of fatigue loadings which reduces a spectrum of variable amplitude stresses into a matrix of simple stress reversals. The advantage of such a method is that it allows for efficient summation of fatigue damage using Miner's rule [7] [8]. The original rainflow method was devised by Tatsuo Endo and M. Matsuishi [9]. However, as of 2008, several counting algorithms are available [10] [11] [12] [13].

Quick Fatigue Tool uses a custom-written algorithm which follows a similar logic to that of the original rainflow methodology. Cycles are counted on the basis of comparing adjacent pairs of stress ranges and removing these points from the loading until no more cycles are closed.

The cycle counting strategy is as follows:

1. Convert the load history into a sequence of peaks and valleys (no intermediate values between inflection points)
2. Re-arrange the history so that the absolute maximum stress is at the start of the signal. Perform the same re-arrangement on the indexes so that the true position of the data points is retained
3. Remove leading and trailing tails (identical values forming flat paths on either side of the load history
4. Remove plateaus (adjacent points with the same value) which can arise from step 2
    a. If the plateau lies between two diametrically opposed inflection points (a peak-valley pair), remove all plateaued points
    b. If the plateau lies between two diametrically similar inflection points (two peaks or two valleys), remove only the current plateau
5. For the current pair of history points, compare the current stress range to the previous stress range
    a. If the current stress range exceeds the previous stress range, count a cycle and remove the previous stress range from the loading
6. Repeat step 5 until no more cycles can be extracted
7. Move to the next pair of history points and repeat steps 5 and 6 until all cycles have been removed from the loading
8. Check for unmatched half cycles and add these to the cycle matrix until there is no more history data

The algorithm does not consider the nonlinear relationship between stress and strain. Therefore, the effect of cyclic hysteresis and material memory is not included. Since changes in the mean stress due to plasticity are ignored, the performance of the cycle counting algorithm will degrade as fatigue lives decrease below one million repeats.

The method described above is implemented from version 6.6 onwards. The previous version of the cycle counting algorithm can be enabled from the environment file. Values of 1.0 and 2.0 correspond to the old and new versions, respectively.

Environment file usage:

Variable	*Value*
rainflowAlgorithm	**{**1.0 | 2.0**}**

A1.5 Treatment of the endurance limit

A1.5.1 Overview

The endurance limit is the region on the stress-life curve where fatigue damage does not occur, irrespective of the number of applied stress cycles [14]. Ferrous and titanium alloys exhibit a distinct limit below which the fatigue life is observed to be infinite. However, other nonferrous alloys such as aluminium and copper do not have a distinct limit and will eventually fail regardless of the applied stress amplitude. The concept of the endurance limit is illustrated by the plateau in Figure A1.3.

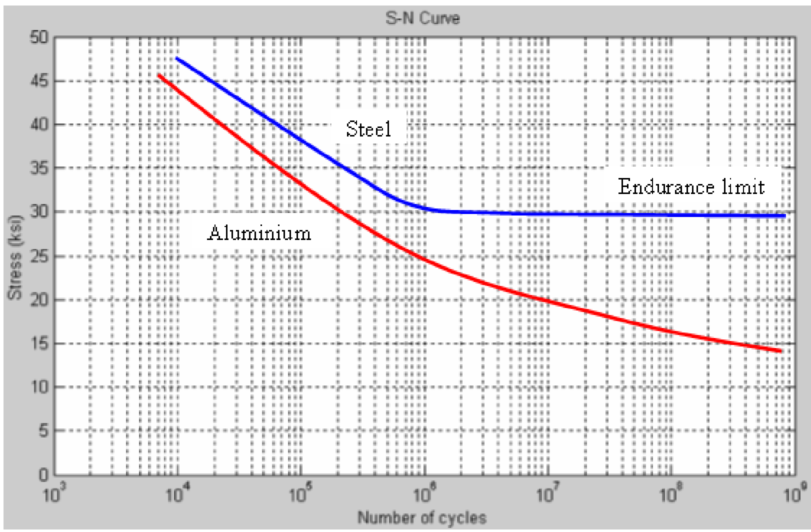

Figure A1.3: Representative Stress-Life curves for steel and aluminium, indicating the presence (and lack thereof) an endurance limit stress below which no fatigue damage occurs.

Image courtesy of Andrew Dressel (Wikimedia Commons: CC-BY-SA-3.0)

A1.5.2 Enforcement of the endurance limit

Quick Fatigue Tool may ignore fatigue damage for cycles below the endurance limit depending on the setting of the following environment variable:

Environment file usage:

Variable	*Value*
ndEndurance	**{**0.0 | 1.0 | 2.0**}**

If ndEndurance is set to 0.0, Quick Fatigue Tool determines from the table below whether to account for the endurance limit based on the material classification defined in the material *.mat* file. Values of 1.0 and 2.0 disable or enable treatment of the endurance limit, respectively.

Material Behaviour	Ignore damage below endurance limit?
Aluminium/copper alloys	NO
All other material classifications	YES

A1.5.3 Endurance limit of non-ferrous metals

Due to the difficulties associated with testing specimens to very long lives, it is not always feasible to determine the exact endurance limit of the material. Furthermore, for some nonferrous metals as well as very strong steels, the stress-life curve is observed to continuously decrease as far as the test data are available. As a result, materials are often considered to cross into infinite life at a pre-determined limit (usually 10^7 cycles), beyond which it is usually safe to assume that no damage will occur. Thus, if the material is classified as *Other* in the material definition, the endurance limit is enforced.

A1.5.4 The "bouncing endurance limit"

Investigations have shown that cycles below the endurance limit can result in fatigue damage from otherwise non-damaging cycles [15]. This phenomenon can be explained by considering a cycle just above the endurance limit. Since the cycle is large enough to cause fatigue damage, it acts to open the crack tip resulting in a very small plastic zone ahead of the crack. This microscopic plasticity temporarily reduces the fatigue strength of the specimen at the crack tip, until the crack propagates through the plastic zone. The phenomenon of a "bouncing endurance limit" has been investigated in considerable detail [16] [17] [18] [19] [20].

This behaviour is modelled in Quick Fatigue Tool such that the fatigue limit is reduced by a factor, k_{scale}, if the current cycle is damaging and only applies when enforcement of the endurance limit is enabled.

Environment file usage:

Variable	*Value*
ndEndurance	**{**0.0[1] | 1.0 |2.0**}**
modifyEnduranceLimit	**{**0.0 | 1.0**}**
enduranceScaleFactor	k_{scale}

[1] If ndEndurance = 0.0, the "bouncing endurance limit" is enabled unless the material behaviour is aluminium, in which case damage below the endurance limit is always allowed.

Over the course of the next $n_{recover}$ consecutive non-damaging cycles, the fatigue limit is linearly incremented back to its original value. By default, $k_{scale} = 0.25$ and $n_{recover} = 50$ [21].

Environment file usage:

Variable	*Value*
cyclesToRecover	$n_{recover}$

If any subsequent cycle exceeds the fatigue limit, the limit will be scaled back to a factor of k_{scale} of its original value and the recovery process will start from the beginning.

Appendix II. Materials data generation

A2.1 Background

Despite extensive efforts over years in characterizing the fatigue behaviour of metals, obtaining fatigue materials data is often time consuming and expensive. Therefore, it is important to be aware of methods which are capable of producing reasonable estimates of material data based on readily available information.

Quick Fatigue Tool offers several approximation algorithms which can estimate the fatigue materials data:

1. Uniform Law (Bäumel-Seeger)
2. Universal Slopes (Manson)
3. Modified Universal Slopes (Muralidharan)
4. 90/50 Rule

A2.2 Bäumel-Seeger Method

The Bäumel-Seeger method approximates the fatigue materials data as follows [22] [23] [24]:

Fatigue Property	Plain/Alloy Steel	Aluminium
Fatigue strength coefficient $\boldsymbol{\sigma_f}'$	$1.5\sigma_U$	$1.67\sigma_U$
Fatigue strength exponent $\boldsymbol{b}$	-0.087	-0.095
Fatigue ductility coefficient $\boldsymbol{\varepsilon_f}'$	$0.59a$ $a = 1, \frac{\sigma_U}{E} < 3^{-3}$ $a = 1.375 - 125\frac{\sigma_U}{E}, \frac{\sigma_U}{E} > 3^{-3}$	0.35
Fatigue ductility exponent $\boldsymbol{c}$	-0.58	-0.69
Cyclic strain hardening coefficient $\boldsymbol{K}'$	$1.65\sigma_U$	$1.61\sigma_U$
Cyclic strain hardening exponent $\boldsymbol{n}'$	0.15	0.11

A2.3 Manson

The Manson method approximates the fatigue materials data as follows [25]:

Fatigue Property	**Plain/Alloy Steel**	**Aluminium**	**Other**
Fatigue strength coefficient $\boldsymbol{\sigma_f}'$	$1.9\sigma_U$	$1.9\sigma_U$	$1.9\sigma_U$
Fatigue strength exponent $\boldsymbol{b}$	-0.12	-0.12	-0.12
Fatigue ductility coefficient $\boldsymbol{\varepsilon_f}'$	$0.76\log{\frac{1}{1-0.1541}}^{0.6}$	$0.76\log{\frac{1}{1-0.4394}}^{0.6}$	$0.76\log{\frac{1}{1-0.1541}}^{0.6}$
Fatigue ductility exponent $\boldsymbol{c}$	-0.6	-0.6	-0.6
Cyclic strain hardening coefficient $\boldsymbol{K'}$	$\frac{\sigma_f{}'}{{\varepsilon'_f}^{0.2}}$	$\frac{\sigma_f{}'}{{\varepsilon'_f}^{0.2}}$	$\frac{\sigma_f{}'}{{\varepsilon'_f}^{0.2}}$
Cyclic strain hardening exponent $\boldsymbol{n'}$	-0.2	-0.2	-0.2

A2.4 Modified Manson

The Modified Manson method approximates the fatigue materials data as follows [26]:

Fatigue Property	**Plain/Alloy Steel**	**Aluminium**	**Other**
Fatigue strength coefficient $\boldsymbol{\sigma_f}'$	$0.623{\sigma_U}^{0.823}E^{0.168}$	$0.623{\sigma_U}^{0.823}E^{0.168}$	$0.623{\sigma_U}^{0.823}E^{0.168}$
Fatigue strength exponent $\boldsymbol{b}$	-0.09	-0.09	-0.09
Fatigue ductility coefficient $\boldsymbol{\varepsilon_f}'$	$0.196\log\frac{1}{1-0.1541}^{0.155}\frac{{\sigma_U}^{-0.53}}{E}$	$0.196\log\frac{1}{1-0.4394}^{0.155}\frac{{\sigma_U}^{-0.53}}{E}$	$0.196\log\frac{1}{1-0.155}^{0.155}\frac{{\sigma_U}^{-0.53}}{E}$
Fatigue ductility exponent $\boldsymbol{c}$	-0.56	-0.56	-0.56
Cyclic strain hardening coefficient $\boldsymbol{K'}$	$\frac{{\sigma_f}'}{{\varepsilon'_f}^{0.2}}$	$\frac{{\sigma_f}'}{{\varepsilon'_f}^{0.2}}$	$\frac{{\sigma_f}'}{{\varepsilon'_f}^{0.2}}$
Cyclic strain hardening exponent $\boldsymbol{n'}$	-0.2	-0.2	-0.2

A2.5 90/50 Rule

The *90/50* rule is an informal approximation of the stress-life relationship. It assumes the following two points on the endurance curve if only the ultimate tensile strength is known:

Life (N_f)	**Stress (S)**
1000	$0.9\sigma_U$
N_∞	$0.5\sigma_U$

Basquin's exponent is then the gradient of the straight line defined by the above two points when plotted on log-log axes:

$$b = \frac{\log 1.8}{\log \frac{1e3}{N_\infty}} \tag{A2.1}$$

where N_∞ is the constant amplitude endurance limit. Note that the value of b depends only on the endurance limit and not on the ultimate tensile strength.

The relationship between the ultimate tensile strength and the fatigue limit becomes increasingly nonlinear with increasing ultimate tensile strength. Therefore, if the UTS exceeds 1000MPa, the endurance limit stress is fixed to 500MPa and Equation A2.2 is used to approximate Basquin's exponent:

$$b_{\sigma_U > 1000MPa} = \frac{\log \frac{0.9\sigma_U}{500}}{\log \frac{1e3}{N_\infty}} \tag{A2.2}$$

If the stress-life relationship is given by Equation A2.3, then the value of the tensile fatigue strength coefficient, $\sigma_f{}'$, may be derived using Equation A2.4 which assumes that $S(1e3) = 0.9\sigma_U$.

$$S = \sigma_f{}'\left(2N_f\right)^b \tag{A2.3}$$

$$\sigma_f{}' = \frac{0.9\sigma_U}{(2 \times 10^3)^b} \tag{A2.4}$$

If $N_\infty = 1e6$ cycles, then the stress life equation is given by Equation A2.5:

$$S = 1.72\sigma_U {N_f}^{-0.085} \tag{A2.5}$$

In general, the stress-life relationship is given by Equation A2.6:

$$S = \frac{0.9\sigma_U}{(2 \times 10^3)^b}\left(2N_f\right)^b, \qquad b = \frac{\log 1.8}{\log \frac{1e3}{N_\infty}} \qquad \text{[A2.6]}$$

Appendix III. Gauge fatigue toolbox

A3.1 Background

A3.1.1 Overview

The Gauge Fatigue Toolbox is a set of MATLAB apps which facilitate the analysis of measured strain data. The following apps are included:

- *Multiaxial Gauge Fatigue*
- *Rosette Analysis*
- *Virtual Strain Gauge*

In addition to the above, the toolbox also includes *Mohr Solver* for calculating principal stresses and strains for a given stress tensor.

The toolbox is located in *Application_Files\toolbox*. To install the app, double click on the *.mlappinstall* file. The app appears in the *Apps* tab of the MATLAB ribbon UI.

A3.1.2 Gauge conventions

The strain gauge is defined as a three-armed rosette and is placed relative to the reference $X - Y$ coordinate system shown in Figure A3.1.

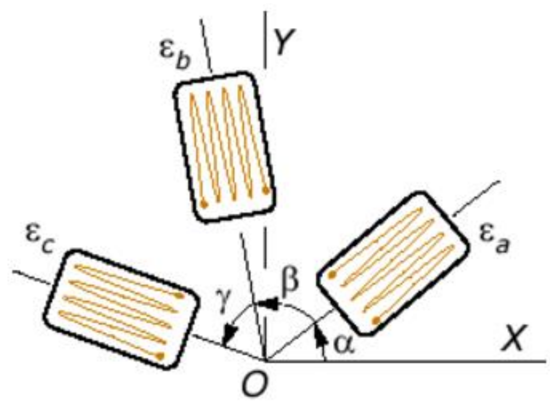

Figure A3.1: Gauge orientation relative to the reference $X - Y$ axis. Rosette arms are separated by angles α, β and γ

The user must define the gauge orientation relative to the reference coordinate system. Where appropriate, the GUI contains an  icon which displays the active gauge orientation. The orientation can be defined in three ways:

1. Rectangular
2. Delta
3. User-defined

Rectangular

The rectangular orientation is the special case where the strains ε_A, ε_B and ε_C from the rosette arms A, B and C, respectively, coincide with the reference axes.

$$\begin{aligned} \varepsilon_{11} &= \varepsilon_A \\ \varepsilon_{22} &= \varepsilon_C \\ \varepsilon_{12} &= 2\varepsilon_B - \varepsilon_A - \varepsilon_C \end{aligned} \qquad \text{[A3.1]}$$

The principal strains, ε_1 and ε_2, are readily calculated from the gauge strains:

$$\varepsilon_{1,2} = \frac{\varepsilon_A + \varepsilon_C}{2} \pm \frac{1}{\sqrt{2}}\sqrt{(\varepsilon_A - \varepsilon_B)^2 + (\varepsilon_B - \varepsilon_C)^2} \qquad \text{[A3.2]}$$

Delta

The delta orientation is the special case where the rosette arms are separated by 60 degrees and rosette arm A is rotated 30 degrees counter-clockwise from the positive X reference direction.

The principal strains are readily calculated from the gauge strains:

$$\varepsilon_{1,2} = \frac{\varepsilon_A + \varepsilon_B + \varepsilon_C}{3} \pm \frac{\sqrt{2}}{3}\sqrt{(\varepsilon_A - \varepsilon_B)^2 + (\varepsilon_B - \varepsilon_C)^2 + (\varepsilon_C - \varepsilon_A)^2} \qquad \text{[A3.3]}$$

User-defined

User-defined gauge orientations allow the user to specify the angles α, β and γ directly. The reference strains are then solved as a system of linear equations.

$$\begin{aligned} \chi_1 &= \frac{\varepsilon_{11} + \varepsilon_{22}}{2} + \frac{\varepsilon_{11} - \varepsilon_{22}}{2}\cos 2\alpha + \frac{\varepsilon_{12}}{2}\sin 2\alpha = \varepsilon_A \\ \chi_2 &= \frac{\varepsilon_{11} + \varepsilon_{22}}{2} + \frac{\varepsilon_{11} - \varepsilon_{22}}{2}\cos 2(\alpha + \beta) + \frac{\varepsilon_{12}}{2}\sin 2(\alpha + \beta) = \varepsilon_B \\ \chi_3 &= \frac{\varepsilon_{11} + \varepsilon_{22}}{2} + \frac{\varepsilon_{11} - \varepsilon_{22}}{2}\cos 2(\alpha + \beta + \gamma) + \frac{\varepsilon_{12}}{2}\sin 2(\alpha + \beta + \gamma) = \varepsilon_C \end{aligned} \qquad \text{[A3.4]}$$

The principal strains are calculated from the reference strains:

$$\varepsilon_{1,2} = \frac{\varepsilon_{11} + \varepsilon_{22}}{2} \pm \sqrt{\left(\frac{\varepsilon_{11} - \varepsilon_{22}}{2}\right)^2 + \left(\frac{\varepsilon_{12}}{2}\right)^2} \qquad \text{[A3.5]}$$

Stress determination

Assuming an elastic relationship, the principal stresses are determined from the principal strains:

$$\sigma_{1,2} = \frac{E}{(1 - \nu^2)}\left(\varepsilon_{1,2} + \nu\varepsilon_{2,1}\right) \qquad \text{[A3.6]}$$

where E is Young's Modulus and ν is Poisson's ratio. The reference stresses σ_{11}, σ_{22} and σ_{12} may be determined from Equation A3.6 by substituting ε_1 and ε_2 with ε_{11} and ε_{22}, respectively.

Reference orientations

The reference orientation is the angle between the reference axes and the principal/shear axes. The orientations are calculated as follows:

$$\phi_D = \frac{1}{2}\tan^{-1}\frac{\varepsilon_{12}}{\varepsilon_{11} - \varepsilon_{22}}$$

$$\phi_S = -\frac{1}{2}\tan^{-1}\frac{\varepsilon_{11} - \varepsilon_{22}}{\varepsilon_{12}} \qquad \text{[A3.7]}$$

where ϕ_D is the angle between the reference axes and the principal axes, and ϕ_S is the angle between the reference axes and the direction of maximum shear stress.

A3.1.3 Strain units

The *Multiaxial Gauge Fatigue* and *Uniaxial Strain-Life* applications allow the user to select between strain [ε] and microstrain [$\mu\varepsilon$]. The *Rosette Analysis* and *Virtual Strain Gauge* applications both assume microstrain as the input units.

A3.2 Multiaxial Gauge Fatigue

Multiaxial Gauge Fatigue is a strain-life analysis application which takes measured strain gauge data as the fatigue loading input. The application GUI is shown in Figure A3.2.

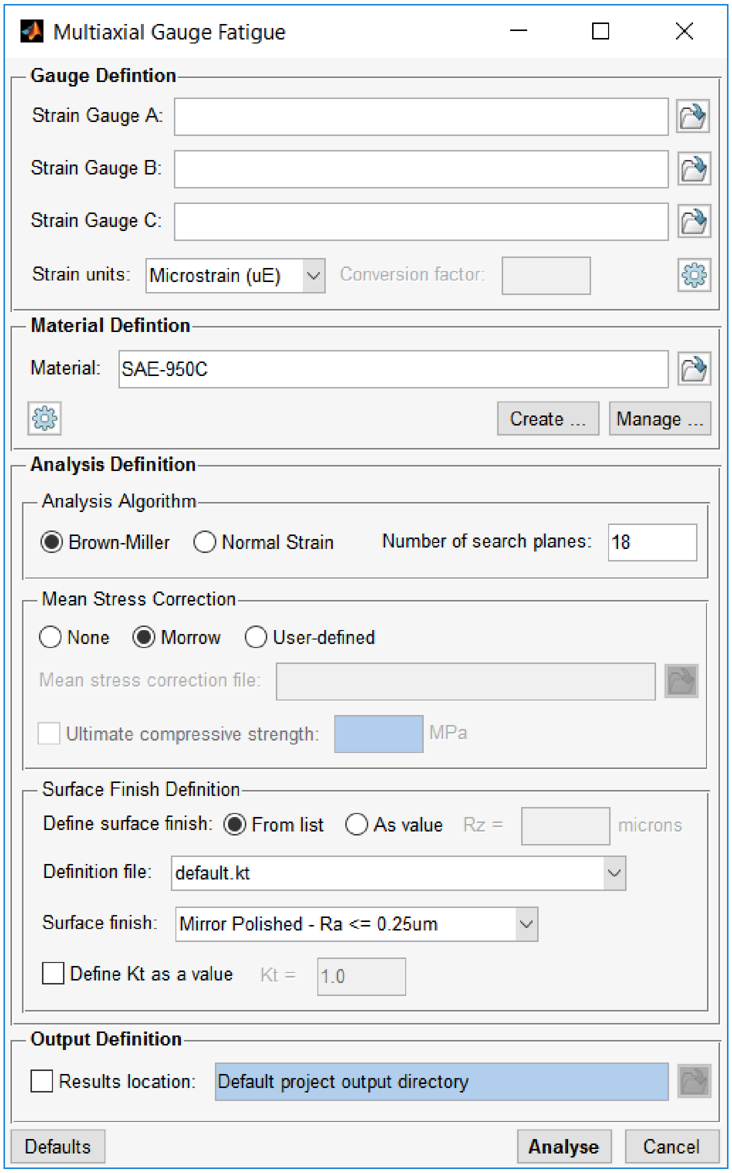

Figure A3.2: Multiaxial Gauge Fatigue GUI

A3.2.1 Gauge Definition

By default, the gauge is assumed to be orientated in a rectangular fashion, i.e. $\alpha = 0°$, $\beta = 45°$ and $\gamma = 45°$. The gauge orientation can be modified by selecting the button. This displays the gauge orientation dialogue shown in Figure A3.3.

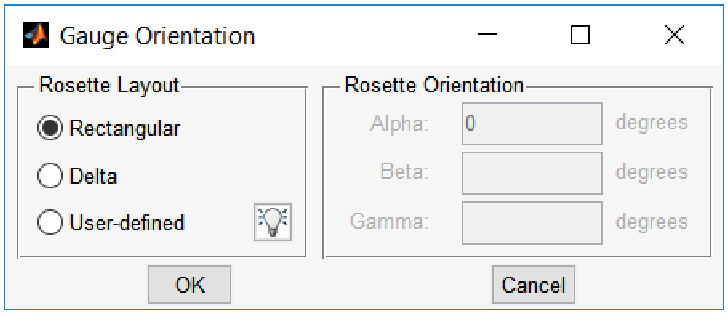

Figure A3.3: Gauge orientation dialogue

The gauge input is defined as a text file containing the measured strain history from its respective rosette arm. Absolute (full) and relative paths may be used. If no path is specified, the current working directory will be searched. The file extension must be included.

If the sample rate is not the same for all three gauges, an optional time history may be included for each gauge. The data file is formatted as follows:

Strain history	Time history (optional)	
ε_1	t_1	← First strain point
ε_2	t_2	
.	.	
.	.	
ε_N	t_N	← Last strain point

The data can be formatted as $N \times a$ or $a \times N$, where N is the number of strain samples. $a = 1$ if no time history is specified and $a = 2$ if a time history is specified.

If time histories are not specified, Quick Fatigue Tool assumes that the lapsed time between samples is equal. If the three signals contain the same number of samples, the phase will be preserved. If the strain histories are not the same length, zeros will be added to the end of the signals where applicable so that they all have the same number of sample points.

If the histories contain a different number of samples, and/or the lapsed time between measurements is not constant, the phase between the signals will be lost if time points are not specified, and inaccurate fatigue results may be obtained. When time histories are defined alongside the strain measurements, Quick Fatigue Tool interpolates the signals so that all three gauges contain measurements at the exact same time points. This ensures that the phase between the signals is preserved.

Time points do not have to be specified for all three gauges. Gauges which do not have time points will be assumed to be increasing monotonically through time:

Strain	ε_1	ε_2	...	ε_n
Time	0	1	...	n

Time points do not have to be specified for every sample point. Missing time points will be interpolated automatically. When using time points, the first point must always be zero, and at least the first and last points must be defined.

The user should exercise caution when using time points. The underlying assumption remains that the strain measurements were obtained at identical points in time. The correction applied by Quick Fatigue Tool is intended for signals with small measurement delay. If the time points between the three gauges differ greatly, the resulting strain history may suffer from aliasing. This happens because of interpolating linearly a strain history which varies nonlinearly.

The default gauge units are microstrain [$\mu\varepsilon$]. User-defined units may be specified with a conversion factor such that:

$$[User\ units] \times Conversion\ Factor = [strain] \qquad \text{[A3.8]}$$

A3.2.2 Material definition

The material is selected either by browsing for an existing material (*.mat) file, or by creating a new material with the Material Manager application. Absolute (full) or relative paths may be used. The file extension does not have to be included. For more information on defining material data, consult Section 5 of the document *Quick Fatigue Tool User Guide*.

Strain-life analysis requires the fatigue ductility (Coffin-Manson) parameters: The fatigue ductility coefficient ($\varepsilon_f{}'$) and the fatigue ductility exponent (c).

Additional options relating to the material behaviour are accessed by selecting **Options…** from the Material Definition region of the user interface. The options are shown in Figure A3.4.

Ignore damage for fully-compressive cycles

By default, the analysis calculates damage for all cycles above the fatigue limit. Selecting this option will ignore fatigue damage for any cycle whose maximum and minimum points are less than-or-equal-to zero. This behaviour is appropriate for various irons and plastics.

Include out-of-plane strains

Rosette strain gauges can only detect plane strain and the analysis ignores out-of-plane strains by default. The assumption is that the thickness of the specimen is sufficient to constrain out-of-plane displacements, and that through-thickness strains would not cause fatigue damage. Selecting this option will enable calculation of the out-of-plane principal strain according to Equation A3.7.

$$\varepsilon_3 = \frac{-\nu}{1-\nu}(\varepsilon_1 + \varepsilon_2) \qquad \text{[A3.9]}$$

This calculation assumes a constant section Poisson's ratio and should only be used in cases where such an estimation is valid.

Ignore damage below endurance limit

Most steels exhibit a clearly defined knee-point, below which fatigue is not observed. In such cases, selecting this option will ignore cycles whose stress amplitude is below the pre-determined fatigue limit. The fatigue limit is defined as the strain amplitude which would result in a fatigue life equal to the material's constant amplitude endurance limit.

Materials such as aluminium do not exhibit a fatigue limit. For these materials, this option should not be selected.

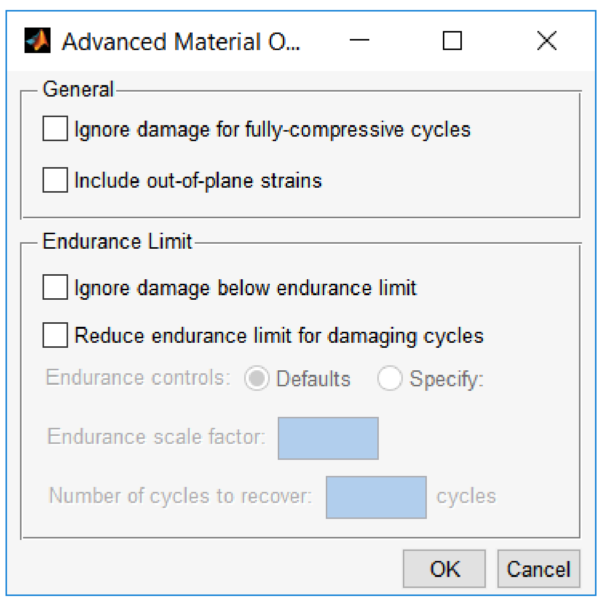

Figure A3.4: Material options

Reduce endurance limit for damaging cycles

If the user selects to ignore damage below the endurance limit, this option can be additionally selected which will automatically reduce the endurance limit in response to a damaging cycle. The resulting behaviour is that previously non-damaging cycles (below the original fatigue limit), will become damaging.

By default, the endurance limit is scaled by a factor of 0.25, and recovers linearly to its original value over the course of 50 non-damaging cycles. If any cycle exceeds its instantaneous fatigue limit during the recovery process, the endurance limit is reset to the scaled value. The user may change these defaults by selecting the **Specify:** radio button and manually entering the values of the endurance limit scale factor and the number of cycles to recover.

A3.2.3 Analysis definition

The gauge fatigue analysis can be run with either the Brown-Miller or the Principal Strain algorithm. These two algorithms work in a similar fashion to the Stress-based Brown-Miller and Normal Stress algorithms described in Sections 6.2 and 6.3, respectively. Strains are resolved onto a series of planes and fatigue damage is calculated on the plane which experiences the largest combination of normal and shear strain (Brown-Miller) or the largest normal strain (Principal Strain).

The number of search planes can be selected from the GUI. The use of 18 search planes is found to be sufficient in most cases. Increasing the number of search panes drastically increases the analysis time.

A3.2.4 Mean Stress Correction

The analysis can be run without mean stress correction, or with either the Morrow correction or a user-defined mean stress correction file.

The Morrow mean stress correction modifies the elastic region of the strain-life equation:

$$\frac{\Delta\varepsilon}{2} = \frac{\sigma_f{}' - \sigma_m}{E} {N_f}^{b} + \varepsilon_f{}' {N_f}^{c} \qquad \text{[A3.10]}$$

The mean stress has a more significant effect on the material's endurance at higher lives. The Morrow correction subtracts the mean stress from the fatigue strength coefficient.

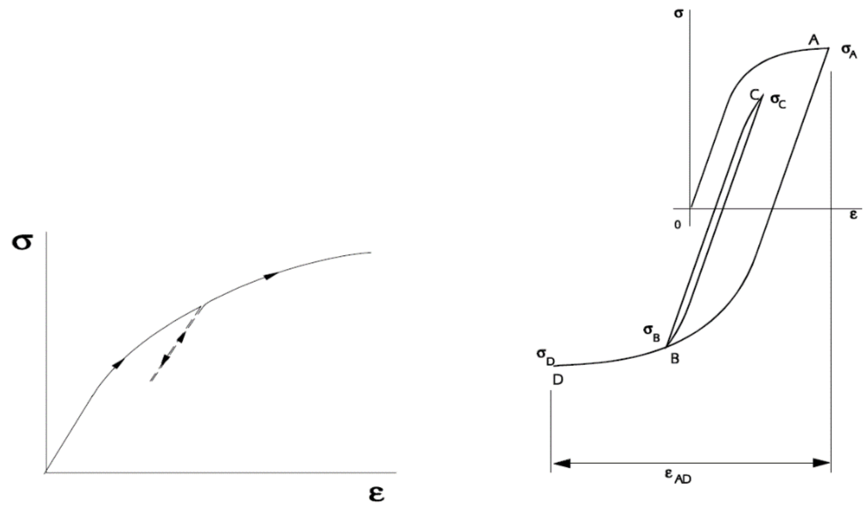

Figure A3.5: The material memory effect. After cycle closure, the hysteresis response resumes from that of the previous excursion

User-defined mean stress correction can be performed by selecting a mean stress correction (*.msc) file. Absolute (full) and relative paths may be used. If no path is specified, the current working directory will be searched. The file extension must be included. User-defined mean stress corrections are discussed in Section 7.9. If data are specified in the compressive regime, the ultimate compressive strength may also be defined by selecting the *Ultimate compressive strength* checkbox. If the compressive strength is not specified, the tensile strength is used by default.

To perform mean stress correction, the true stresses must be obtained from the measured strain history. If mean stress correction is specified, the program converts the principal strain history into principal stresses before the analysis, and uses the stress and strain for the critical plane analysis.

The relationship between true stress and strain is assumed to obey a multilinear cyclic strain-hardening law. This is accomplished with the Ramberg-Osgood relationship for cyclic stress and strain:

$$\frac{\Delta\varepsilon}{2}=\frac{\Delta\sigma}{E}+2\left(\frac{\Delta\sigma}{2K'}\right)^{1/n'} \quad \text{[A3.11]}$$

Where $\frac{\Delta\varepsilon}{2}$ and $\frac{\Delta\sigma}{2}$ is the true strain and stress range, respectively. Since the true strain is used, elastic-plastic correction such as Neuber’s rule is not required. The effect of material memory is considered whereby if a cycle is closed, the material stress-strain response continues as though the reversal had not occurred. This behaviour is illustrated by Figure A3.5. This allows for the effect of ratcheting to be considered.

The current approach for calculating the stresses is limited by the assumption that the strain history is already cyclically stable. Therefore, the analysis does not make additional correction for the effect of kinematic hardening. Depending on the hardening characteristics of the material, fatigue results obtained from the Multiaxial Gauge Fatigue application may be on the non-conservative side. In such cases, additional care and consideration should be taken by the user.

A3.2.5 Surface Finish Definition

Surface finish is specified either directly or by surface finish type. The effect of surface finish is discussed in Section 4.3.

A3.2.6 Output Definition

In addition to a fatigue analysis summary, the Multiaxial Gauge Fatigue application writes the following history output:

ST, NT, DPP, DP, LP	Maximum shear strain, maximum normal strain, damage parameter, damage and life as a function of plane angle θ
HD	Mean stress and strain amplitude for each cycle in the load history
CN, CS	Normal and resultant shear strain history on the critical plane
EST	Principal tensor history. Includes the principal stress if mean stress correction is used

The following MATLAB figures are produced:

CN + CS	Normal and shear strain history on the critical plane
CPS	Normal strain and resultant shear stress vs. plane angle
CSS*n*	Cyclic stress-strain curves for maximum, middle and minimum principal stress-strain histories
DP	Damage vs. angle
DPP	Damage parameter vs. angle
LP	Life vs. angle
PS	Time-correlated principal stresses
PE	Time-correlated principal strains
RHIST	Rainflow cycle histogram

A3.3 Rosette Analysis

The Rosette Analysis GUI is shown in Figure A3.6.

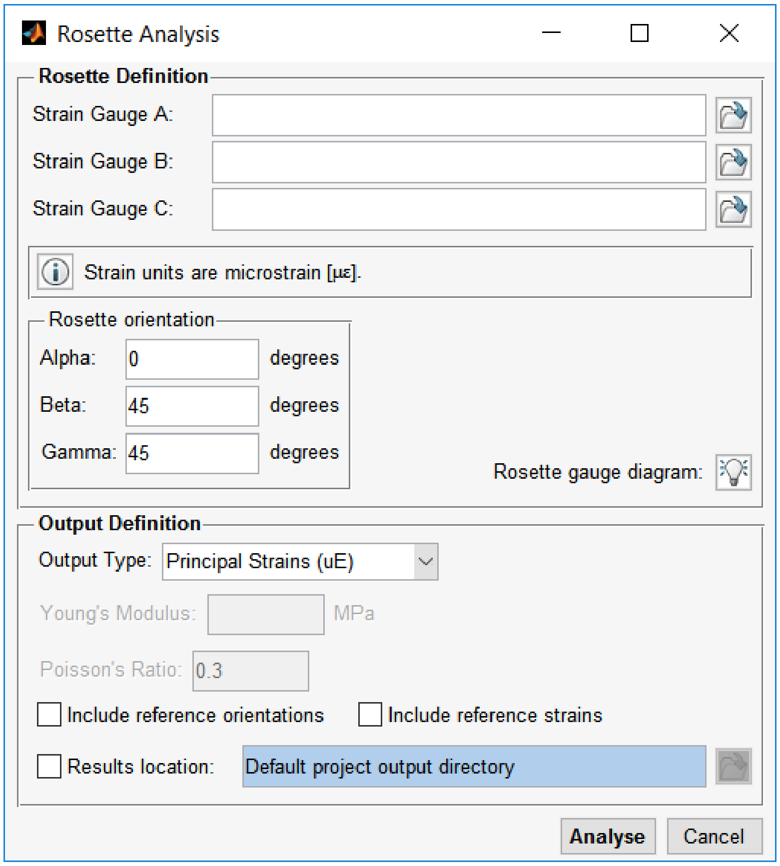

Figure A3.6: Rosette Analysis GUI

The purpose of this application is to process data from a strain gauge to convert it into principal stresses and strains. Strain data is specified for the three gauge arms. The strain is defined in units of microstrain $[\mu\varepsilon]$.

Input is specified either by selecting an ASCII text file containing the strain gauge history, or by defining the history directly in the edit field. Histories defined directly must follow MATLAB syntax of a $1 \times N$ vector, e.g. [2000, -2000].

The gauge orientation must be defined. A rectangular orientation corresponds to $\alpha = 0°$, $\beta = 45°$ and $\gamma = 45°$, whereas a delta layout would be defined as $\alpha = 30°$, $\beta = 60°$ and $\gamma = 60°$. However, any user gauge orientation may be specified.

The user may request the output in terms of principal strain or principal stress. Additionally, the reference orientations and stress/strain may be requested. The reference orientations and stress/strain are given by Equation A3.7 and Equation A3.4, respectively.

All available output variables and their identifiers are listed below.

PE1	Maximum principal strain
PE2	Minimum principal strain
E12$^{(MAX)}$	Maximum shear strain
E11$^{(R)}$	Normal strain in reference X-direction
E22$^{(R)}$	Normal strain in reference Y-direction
E12$^{(R)}$	Shear strain in reference XY-direction
PS1	Maximum principal stress
PS2	Minimum principal stress
S12$^{(MAX)}$	Maximum shear stress
S11$^{(R)}$	Normal stress in reference X-direction
S22$^{(R)}$	Normal stress in reference Y-direction
S12$^{(R)}$	Normal stress in reference XY-direction
ϕ_D	Angle between principal and reference axes

ϕ_s	Angle between maximum shear and reference axes

Output stress, strain and angle units are MPa, $\mu\varepsilon$ and $degrees$, respectively.

A3.4 Virtual Strain Gauge

The Virtual Strain Gauge GUI is shown in Figure A3.7.

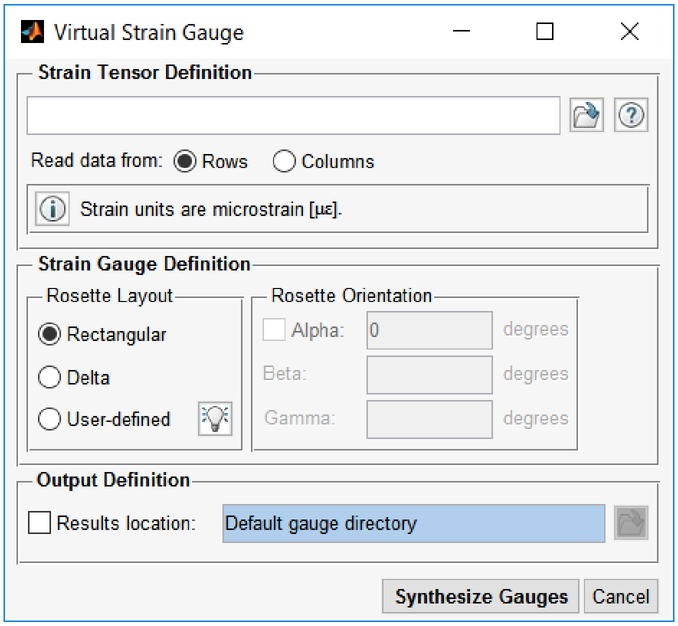

Figure A3.7: Virtual Strain Gauge GUI

The purpose of this application is to mimic the output of a virtual strain gauge in response to a strain tensor. The strain tensor represents the response of the structure onto which the virtual gauge would be adhered. This tool is useful for determining the theoretical response of a rosette strain gauge at a specific orientation.

Input is specified by selecting an ASCII text file containing the strain gauge data. The text file has the following format:

$\boldsymbol{\varepsilon_{11}}$	$\boldsymbol{\varepsilon_{22}}$	$\boldsymbol{\varepsilon_{12}}$	
.	.	.	← First strain point
.	.	.	
ε_{11N}	ε_{22N}	ε_{12N}	← Last strain point

In the above example, the strain components are defined by three separate columns representing the reference X, Y and shear XY directions, respectively. However, the strain data may be defined by rows or columns, provided that the user specified this with the *Read data from* radio buttons. The strain is defined in units of microstrain [$\mu\varepsilon$].

The process of converting the input data in strain gauge histories is describes in Section 4.9.

A3.5 Mohr Solver

The Mohr Solver GUI is shown in Figure A3.8.

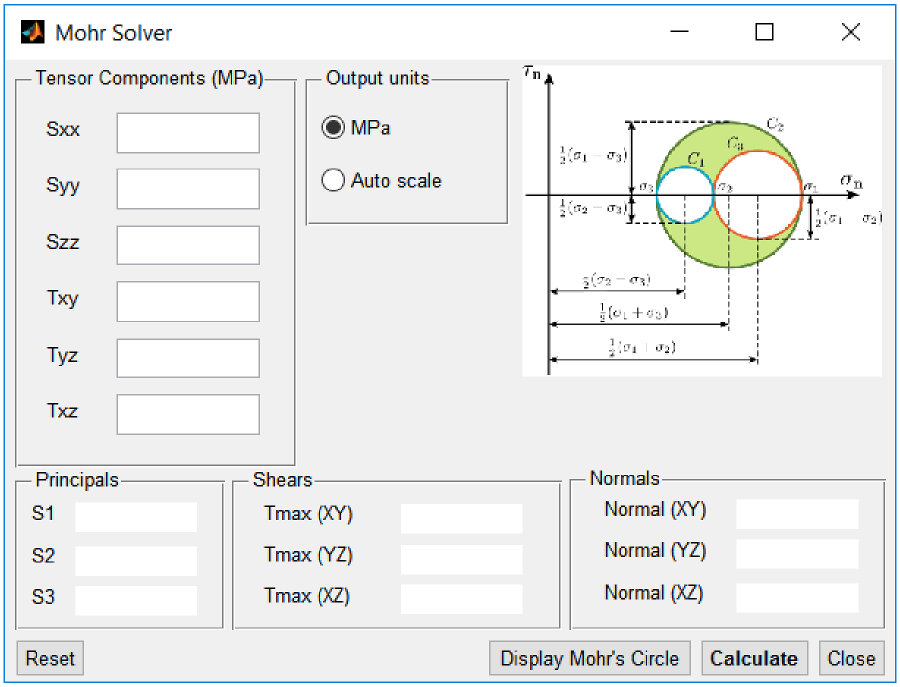

Figure A3.8: Mohr's Circle Solver GUI

The principal stresses are given by Equation A3.12.

$$[\sigma_1\ \sigma_2\ \sigma_3] = \begin{vmatrix} \sigma_{xx} & \tau_{xy} & \tau_{xz} \\ . & \sigma_{yy} & \tau_{yz} \\ . & . & \sigma_{zz} \end{vmatrix} \qquad [A3.12]$$

The maximum shear stresses are given by Equation A3.13.

$$\tau_{max,xy,yz,xz} = \frac{1}{2}\left(\sigma_{11,22,11} - \sigma_{22,33,33}\right) \qquad [A3.13]$$

The maximum normal stresses are given by Equation A3.14.

$$\sigma_{N_{max,xy,yz,xz}} = \frac{1}{2}\left(\sigma_{11,22,11} + \sigma_{22,33,33}\right) \qquad [A3.14]$$

A3.6 Uniaxial Strain-Life

Uniaxial Strain-Life is a strain-life analysis application which takes a stress or strain history as the fatigue loading input. The application GUI is shown in Figure A3.9.

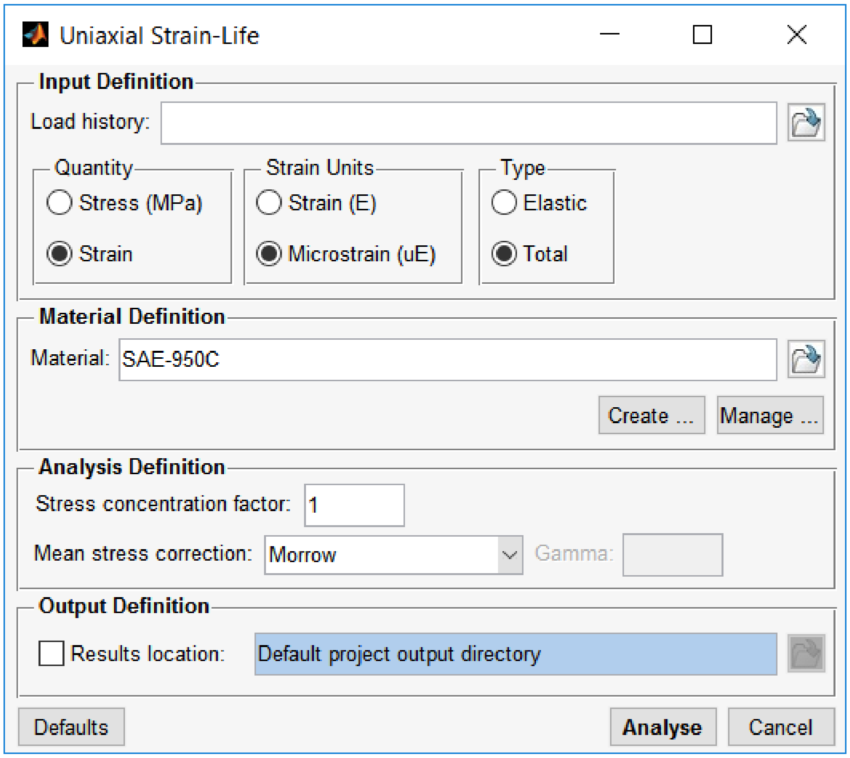

Figure A3.9: Uniaxial Strain-Life GUI

A3.6.1 Input definition

The input is specified as a stress or strain history. The stress is assumed to be elastic, and the strain can be elastic or inelastic.

The application converts the input history into the inelastic stress and strain, which is then rainflow cycle counted so that the strain amplitude can be obtained. The elastic stress is converted into inelastic stress and strain using the Ramberg-Osgood nonlinear elastic cyclic strain hardening relationship (Equation A.15) with Neuber's correction (Equation A.16).

$$\frac{\Delta\varepsilon}{2} = \frac{\Delta\sigma}{E} + 2\left(\frac{\Delta\sigma}{2K'}\right)^{1/n'} \quad \text{[A.15]}$$

$$\Delta\varepsilon\Delta\sigma = \Delta\varepsilon^{e}\Delta\sigma^{e} \quad \text{[A.16]}$$

The user of the Ramberg-Osgood model is described in the table below.

Input	Ramberg-Osgood	Neuber's Rule?
Elastic stress (σ^e)	Solve for $\Delta\varepsilon$ and $\Delta\sigma$	Yes
Elastic strain (ε^e)	Convert ε^e to σ^e Solve for $\Delta\varepsilon$ and $\Delta\sigma$	Yes
Inelastic strain (ε)	Solve for $\Delta\sigma$	No

Input is specified either by selecting an ASCII text file containing the history data, or by defining the history directly in the edit field. Histories defined directly must follow MATLAB syntax of a $1 \times N$ vector, e.g. [2000, -2000].

A3.6.2 Material definition

The material is selected either by browsing for an existing material (*.mat) file, or by creating a new material with the Material Manager application. Absolute (full) or relative paths may be used. The file extension does not have to be included. For more information on defining material data, consult Section 5 of the document *Quick Fatigue Tool User Guide*.

A3.6.3 Analysis definition

The fatigue analysis uses the Strain-Life relationship in Equation A.17.

$$\frac{\Delta\varepsilon}{2} = \frac{{\sigma_f}'}{E}\left(2N_f\right)^b + {\varepsilon_f}'\left(2N_f\right)^c \qquad \text{[A.17]}$$

where $\frac{\Delta\varepsilon}{2}$ is the uniaxial strain amplitude.

A stress concentration factor can be specified, which scales the elastic stress and strain in the Neuber correction.

The Morrow, Smith-Watson-Topper and Walker mean stress corrections are available. For more information on mean stress correction, consult Section 7 of the document *Quick Fatigue Tool User Guide*.

A3.6.4 Output definition

In addition to a fatigue analysis summary, the Uniaxial Strain-Life application writes the following history output:

LH	Input load history before and after gating
HD	Mean strain and strain amplitude for each cycle in the load history
CSS	Inelastic cyclic stress-strain history

Appendix IV. List of supported elements for surface detection

The surface detection algorithm described in Section 4.5.3 "Specifying the analysis region" of the document *Quick Fatigue Tool User Guide* supports the following Abaqus elements.

3D continuum hexahedron (brick) elements

C3D8, C3D8H, C3D8I, C3D8IH, C3D8R, C3D8RH, C3D8S, C3D8HS, C3D20, C3D20H, C3D20R, C3D20RH, C3D8T, C3DHT, C3D8RT, C3D8RHT, C3D20T, C3D20HT, C3D20RT, C3D20RHT.

3D continuum tetrahedral elements

C3D4, C3D4H, C3D10, C3D10H, C3D10HS, C3D10M, C3D10MH, C3D4T, C3D10T, C3D10HT, C3D10MT, C3D10MHT.

3D continuum wedge (triangular prism) elements

C3D6, C3D6H, C3D15, C3D15H.

3D continuum pyramid elements

C3D5, C3D5H.

3D conventional triangular shell elements

STRI3, S3, S3R, S3RS, STRI65, S3T, S3RT.

3D conventional quadrilateral shell elements

S4, S4R, S4RS, S4RSW, S4R5, S8R, S8R5, S4T, S4RT, S8RT.

3D continuum triangular shell elements

SC6R, SC6RT.

3D continuum hexahedral shell elements

SC8R, SC8RT

2D continuum triangular elements:

CPE3, CPE3H, CPE6, CPE6H, CPE6M, CPE6MH, CPS3, CPS6, CPS6M, CPEG3, CPEG3H, CPEG6, CPEG6H, CPEG6M, CPEG6MH.

2D continuum quadrilateral elements:

CPE4, CPE4H, CPE4I, CPE4IH, CPE4R, CPE4RH, CPE8, CPE8H, CPE8R, CPE8RH, CPS4, CPS4I, CPS4R, CPS8, CPS8R, CPEG4, CPEG4H, CPEG4RH, CPEG8, CPEG8H, CPEG8R, CPEG8RH.

References

[1] A. Fatemi and D. F. Socie, “A Critical Plane Approach to Multiaxial Fatigue Damage Including Out-Of-Phase Loading,” *Fatigue & Fracture of Engineering Materials & Structures,* vol. 11, no. 3, pp. 149-165, 1988.

[2] J. Park and D. Nelson, “Evaluation of an energy-based approach and a critical plane approach for predicting constant amplitude multiaxial fatigue life,” *International Journal of Fatigue,* vol. 22, no. 1, pp. 23-39, 2000.

[3] L. Susmel, “A simple and efficient numerical algorithm to determine the orientation of the critical plane in multiaxial fatigue problems,” *International Journal of Fatigue,* vol. 32, no. 11, pp. 1875-1883, 2010.

[4] E. Macha, “Generalization of Fatigue Fracture Criteria for Multiaxial Sinnusoidal Loadings in the Range of Random Loadings,” Technical University of Opole, Opole, 1989.

[5] D. Socie and G. Marquis, Multiaxial Fatigue, SAE International, 1999.

[6] R. Rabb, “Todennakoisyysteoriaan Pohjautuva Vasymisanalyysi,” *Rakenteiden Mekaniikka,* vol. 45, no. 3, pp. 162-187, 2012.

[7] A. Palmgren, “Die lebensdauer von kugellagern,” *Zeitschrift des Vereins Deutscher Inginieure,* vol. 68, no. 14, pp. 339-341, 1924.

[8] M. Miner, “Cumulative damage in fatigue,” *Journal of Applied Mechanics,* vol. 12, no. 3, pp. 159-164, 1945.

[9] M. Matsuishi and T. Endo, “Fatigue of metals subjected to varying stress,” *Japan Society of Mechanical Engineers,* pp. 37-40, 1968.

[10] S. D. Downing and D. F. Socie, “Simple rainflow counting algorithms,” *International Journal of Fatigue,* vol. 4, pp. 31-40, January 1982.

[11] ASTM-E 1049-85, “Standard Practices for Cycle Counting in Fatigue Analysis,” West Conshohocken, 1990.

[12] L. Schluter, “Programmer's Guide for LIFE2's Rainflow Counting Algorithm,” *Sandia Report SAND90-2260,* 1991.

[13] SAE International, SAE Fatiuge Design Handbook (AE-22), 3rd ed., Library of Congress Cataloging-in-Publication Data, 1997, p. 16.

[14] F. P. Beer and E. Russell Johnston, Jr., Mechanics of Materials, 2nd ed., McGraw-Hill, Inc., 1992, p. 51.

[15] H. F. Moore and J. B. Kommers, The Fatigue of Metals, New York, London: McGraw Hill, 1927.

[16] M. H. El Haddad, K. N. Smith and T. H. Topper, "Fatigue crack propagation of short cracks," *Journal of Engineering Materials and Technology,* vol. 101, no. 1, pp. 42-46, 1979.

[17] T. H. Topper and T. S. Lam, "Effective strain-fatigue life data for variable amplitude fatigue," *International Journal of Fatigue,* vol. 19, 1997.

[18] D. L. DuQuesnay, M. A. Pompetzki, T. H. Topper and M. T. Yu, "Effects of Compression and Compressive Overloads on the Fatigue Behaviour of a 2014-T351 Aluminium Alloy and a SAE 1045 Steel," *ASTM STP,* vol. 942, 1988.

[19] M. A. Pompetzki, T. H. Topper and D. L. DuQuesnay, "The effect of compressive underloads and tensile overloads on fatigue damage accumulation in SAE 1045 steel," *Internation Journal of Fatigue,* vol. 12, no. 3, 1990.

[20] R. Jurcevic, D. L. DuQuesnay, T. H. Topper and M. A. Pompetzki, "Fatigue damage accumulation in 2014-T351 Aluminium subjected to periodic reversed overloads," *International Journal of Fatigue,* vol. 12, no. 4, 1990.

[21] D. L. DuQuesnay, M. A. Pompetzki and T. H. Topper, "Fatigue life prediction for variable amplitude strain histories," *SAE Technical Paper No. 930400,* 1993.

[22] A. J. Bäumel and T. Seeger, "Materials data for cyclic loading, supplement 1," *Elsevier Science Publishers BV,* 1987.

[23] T. M. I. Fonseca Jr. and R. Magnabosco, "Evaluation of Methods for Estimating Fatigue Properties Applied to Stainless Steels and Aluminium Alloys," *Tecnologia em Metalurgia, Materiais e Mineracao,* vol. 9, no. 4, pp. 284-293, 2012.

[24] J. Morrow, "Cyclic plastic strain energy and fatigue of metals," *ASTM International,* 1965.

[25] M. Jono and T. Inoue, "Mechanical Behaviour of Materials - VI," *Proceedings of the Sixth International Conference,* 29 July 1991.

[26] S. S. Manson, "A modified universal slopes equation for estimation of fatigue characteristics of metals," *Journal of Engineering Materials and Technology,* vol. 110, p. 55, January 1988.