Available online at www.sciencedirect.com

ScienceDirect

International Journal of Fatigue 00 (2016) 1–40

Journal Logo

A new strategy for fatigue analysis in presence of general multiaxial time varying loadings

Ma Zepeng[a,*], Patrick Le Tallec[b], Habibou Maitournam[c]

[a] *Laboratory of Solid Mechanics, Ecole Polytechnique, 91128 Palaiseau Cedex, France*
[b] *Laboratory of Solid Mechanics, Ecole Polytechnique, 91128 Palaiseau Cedex, France*
[c] *IMSIA, ENSTA ParisTech, CNRS, CEA, EDF, Université Paris-Saclay, 828 bd des Maréchaux, 91762 Palaiseau cedex France*

Abstract

The object of this paper is to propose an energy based fatigue approach which handles multidimensional time varying loading histories.

Our fundamental thought is to assume that the energy dissipated at small scales governs fatigue at failure. The basis of our model is to consider a plastic behavior at the mesoscopic scale with a dependence of the yield function not only on the deviatoric part of the stress but also on the hydro static part. A kinematic hardening under the assumptions of associative plasticity is also considered. We also follow the Dang Van paradigm at macro scale. The structure is elastic at the macroscopic scale. At each material points, there is a stochastic distribution of weak points which will undergo strong plastic yielding, which contribute to energy dissipation without affecting the overall macroscopic stress.

Instead of using the number of cycles, we use the concept of loading history. To accommodate real life loading history more accurately, mean stress effect is taken into account in mesoscopic yield function and non-linear damage accumulation law are also considered in our model. Fatigue will then be determined from the plastic shakedown cycle and from a phenomenological fatigue law linking lifetime and accumulated mesoscopic plastic dissipation.

Keywords: Fatigue; Energy; High cycle; Plasticity; Mean stress

*Corresponding author. Tel.: +33-634435338
Email address: zepeng.ma@polytechnique.edu

Nomenclature

S_{max}	maximum deviatoric stress during the loading cycles
σ_{-1}	fatigue limit for fully reversed condition
b	back stress
$\dot{w}$	energy dissipation rate at a certain scale
$\dot{W}$	energy dissipation rate at all scales
W	dissipated energy
W_{cyc}	dissipated energy per cycle
N	current number of cycles
N_F	number of cycles to failure
$\dot{\varepsilon}_p$	rate of effective plastic strain
$\dot{p}$	accumulated plastic strain rate given as $\sqrt{\frac{2}{3}}\|\dot{\varepsilon}_p\|$
W_F	dissipated energy to failure per unit volume
E	Young's modulus
$k = 500 \sim 800MPa$	hardening parameter
$\beta = 1 \sim 50$	weakening scales distribution exponent
$\gamma = 0 \sim 50$	material parameter from Chaboche law(Wohler curve exponent)
$\alpha = 1 - a\left\langle \frac{\max_t \sqrt{J_{2,a}}(t) + a_c P_{max}(t) - b_c}{\sigma_u - 2\max \sqrt{J_{2,a}}} \right\rangle$	characterizes non-linearity of damage accumulation
a	material parameter from Chaboche law
σ_y	macroscopic yield stress(normal or shear)
$\lambda = 0 \sim 5$	hydrostatic pressure sensitivity
$dev\dot{\underline{\underline{\Sigma}}}$	deviatoric part of the stress tensor
$\Sigma_H = P$	macroscopic hydrostatic pressure
$A_{II} = \tau_{oct,a} = \sqrt{\frac{1}{3}J_{2,a}}$	the amplitude of octahedral shear stress
$S_{max} = \sigma_{VM} = \sqrt{6J_{2,a}}$	Von Mises stress
s_{-1}	tensile fatigue limit for $R = -1$
$\langle\ \rangle$	Macaulay bracket symbol.$\langle\ \rangle$ is defined as $\langle m\rangle = 0$ if $m \leqslant 0$

1 Weakening scales and yield function

1.1. The concept of weakening scales

We follow the Dan Van paradigm. The structure is elastic at the macroscopic scale. At each material points, there is a stochastic distribution of weak points which will undergo strong plastic yielding, without contributing to the overall macroscopic stress. From a microscopic point of view, there is a distribution of weakening scales, namely $s \in [1, \infty)$. Let S_{max} be the macroscopic stress intensity at present time. Let σ_y be the yield limit before weakening. Then we imagine that for a given scale s:

• either $1 \leqslant s \leqslant \sigma_y/S_{max}$, then $S_{max} \leqslant \sigma_y/s$, the material stays in the elastic regime and there is no energy dissipation at this scale.

• or $\sigma_y/S_{max} \leqslant s \leqslant \infty$, then $S_{max} \geqslant \sigma_y/s$, the material is in the plastic regime and there is dissipated energy at scale s, contributing to the fatigue limit, which evolve through kinematic hardening.

In more details, at each scale s of a plastic evolution process there is a weakened yield limit σ_y/s, zero initial plastic strain $\underline{\underline{\varepsilon}}_p$ and zero initial backstress $\underline{\underline{b}}$ at initial time t_0.

1.2. Distribution of weakening scales

We assume the weakening scales have a probability distribution of power law:

$$P(s) = Cs^{-\beta},$$

where β is a material constant and C is hardening constant. The choice of a power law has two reasons: on one hand, this type of distribution corresponds to a scale invariant process, on the other hand it leads in cyclic loading to a prediction of a number of cycles to life limit as a power law function of the stress intensity. More general laws can also be proposed.

The integrated probability ranging from macroscopic to microscopic stress is unity. From this we can conclude:

$$\int_1^\infty P(s)ds = \left[\frac{Cs^{1-\beta}}{1-\beta}\right]_1^\infty = 0 - \frac{C}{1-\beta} = 1.$$

Then we know $C = \beta - 1$, so the distribution is given by:

$$P(s) = Cs^{-\beta} = (\beta - 1)s^{-\beta}$$

and it is shown in Figure 1.

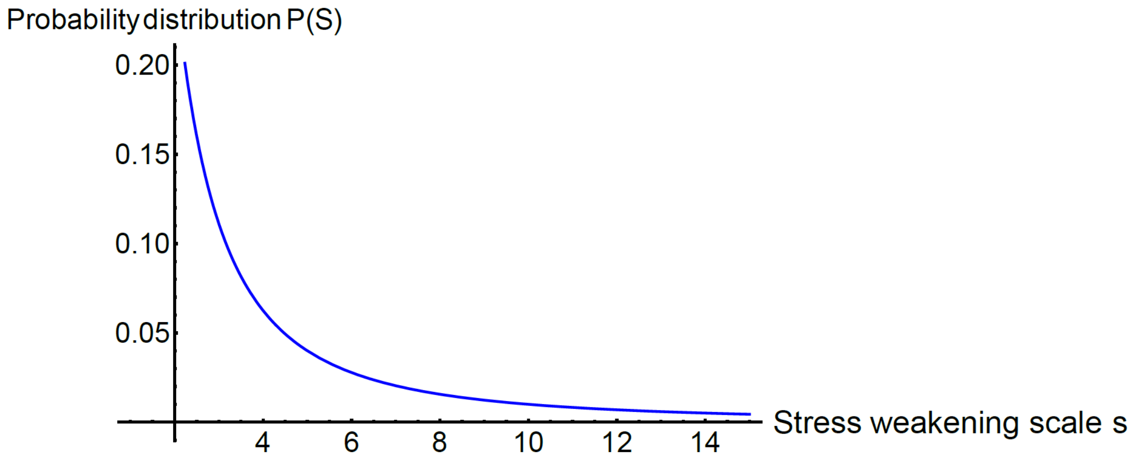

Figure 1: Weakening scales s probability distribution curve

1.3. Yield function with mean stress effect

Positive mean stress clearly reduces the fatigue life of the material. In design evaluation of multiaxial fatigue with mean stress, a simplified, conservative relation between mean stress and equivalent alternating stress is necessary. We can improve the model by modifying the yield function σ_y and the localization tensor.

The idea is to consider as in Maitournam and Krebs[1] that the yield limit σ_y can be reduced in presence of positive mean stress. The mesoscopic yield function can therefore be written as:

$$f(s) = \|\underline{\underline{S}}(s) - \underline{\underline{b}}(s)\| + (\lambda\Sigma_H - \sigma_y)/s \leqslant 0 \tag{1}$$

with $\underline{\underline{S}}$ denoting the deviatoric part of the stress tensor at microscale, and $\underline{\underline{b}}(s)$ the corresponding backstress at the same scale. The material remain in elastic regime when $f < 0$ and in plastic regime when $f = 0$.

1.4. Local plastic model

First we should describe the mesoscopic stress state. The model considers a plastic behavior at the mesoscopic scale. The mesoscopic evolution equations are thus:

$$\dot{\underline{\underline{S}}}(s, M, t) = dev\dot{\underline{\underline{\Sigma}}}(M, t) - \frac{E}{1+\nu}\dot{\underline{\underline{\varepsilon}}}^p(s, M, t), \tag{2}$$

which defines a Taylor-Lin scale transition model with unit localization tensor[2].

$$\dot{\underline{\underline{b}}}(s, M, t) = \frac{kE}{E-k}\dot{\underline{\underline{\varepsilon}}}^p(s, M, t), \tag{3}$$

which is our isotropic kinematic hardening model.

$$\dot{\underline{\underline{\varepsilon}}}^p(s, M, t) = \gamma\frac{\partial f(s, M, t)}{\partial \underline{\underline{S}}}, \tag{4}$$

which is the associated plastic flow rule assuming $\gamma = 0$ when $f < 0$ and $\gamma \geqslant 0$ when $f = 0$.

Here E denotes the Young's modulus and k the hardening parameter. The local dissipated energy rate per volume at weakening scales s is given by the local entropy dissipation:

$$\dot{w}(s, M, t) = (\underline{\underline{S}} - \underline{\underline{b}})(s, M, t) : \underline{\underline{\dot{\varepsilon}}}^p(s, M, t). \tag{5}$$

2 Construction of an energy based fatigue approach

In a preliminary step, we will consider a simple macroscopic loading history $\underline{\underline{\Sigma}}(M, t)$ which is uniaxial and time periodic of deviatoric amplitude S_{max}, and a Von Mises flow rule to see if we get a prediction of local failure for a number of cycles N_F varying as $\Sigma^{-\beta}$.

In uniaxial cyclic loading, there will be 3 kinds of loading patterns, as is shown in Figure 2:

1. Elastic regime, in phase 2 and 4,where $\underline{\underline{\dot{\varepsilon}}}^p(s, M, t) = 0$, and $|\underline{\underline{S}} - \underline{\underline{b}}| < (\sigma_y - \lambda\Sigma_H)/s$.
2. Plastic regime according to plastic flow rule, with increasing plastic deformation, in phase 5 and 1, where $\underline{\underline{\dot{\varepsilon}}}^p(s, M, t) = \gamma \dfrac{\underline{\underline{S}}(s) - \underline{\underline{b}}(s)}{\|\underline{\underline{S}}(s) - \underline{\underline{b}}(s)\|} > 0$ with $\gamma = \left(dev\dot{\Sigma}\right)\left(\dfrac{kE}{E-k} + \dfrac{E}{1+\nu}\right)^{-1}$, with $\underline{\underline{S}} - \underline{\underline{b}} = (\sigma_y - \lambda\Sigma_H)/s$ and $\underline{\underline{\dot{S}}} - \underline{\underline{\dot{b}}} = 0$.
3. Plastic regime in the other direction, in phase 3, there is $\underline{\underline{\dot{\varepsilon}}}^p(s, M, t) < 0$, then $\underline{\underline{S}} - \underline{\underline{b}} = -(\sigma_y - \lambda\Sigma_H)/s$ and $\underline{\underline{\dot{S}}} - \underline{\underline{\dot{b}}} = 0$

In phase 1, a direct analysis yields the energy dissipation at scale s:

$$dW = (S - b)d\varepsilon^p = \frac{(E-k)(1+\nu)}{E(E+k\nu)} \frac{\sigma_y - \lambda\Sigma_H}{s} \left(S_{max} - \frac{\sigma_y - \lambda\Sigma_H}{s}\right) \tag{6}$$

A similar analysis yields

$$dW(phase1) = dW(phase5) = \frac{1}{2} dW(phase3).$$

We can then calculate the local dissipated energy W at point M during one cycle by cumulating the input of all sub-scales with their probabilities [3].

$$\begin{aligned} W_{cyc} &= 4 \int_{(\sigma_y - \lambda\Sigma_H)/S_{max}}^{\infty} dW(s, M, t) P(s) ds \\ &= 4 \int_{(\sigma_y - \lambda\Sigma_H)/S_{max}}^{\infty} \frac{(E-k)(1+\nu)}{E(E+k\nu)} \frac{\sigma_y - \lambda\Sigma_H}{s} \left(S_{max} - \frac{\sigma_y - \lambda\Sigma_H}{s}\right) (\beta - 1)\, s^{-\beta} ds \\ &= \frac{4(E-k)(1+\nu)(\beta-1)}{E(E+k\nu)\beta(\beta+1)} \frac{S_{max}^{\beta+1}}{(\sigma_y - \lambda\Sigma_H)^{\beta-1}}. \end{aligned} \tag{7}$$

If the dissipated energy accumulates linearly until a failure value W_F, we can get directly the time to failure from Eq.(8):

$$T_{fail} = N_F t_{cyc} = \frac{W_F}{W_{cyc}} t_{cyc} = C(S_{max})^{-\beta-1}. \tag{8}$$

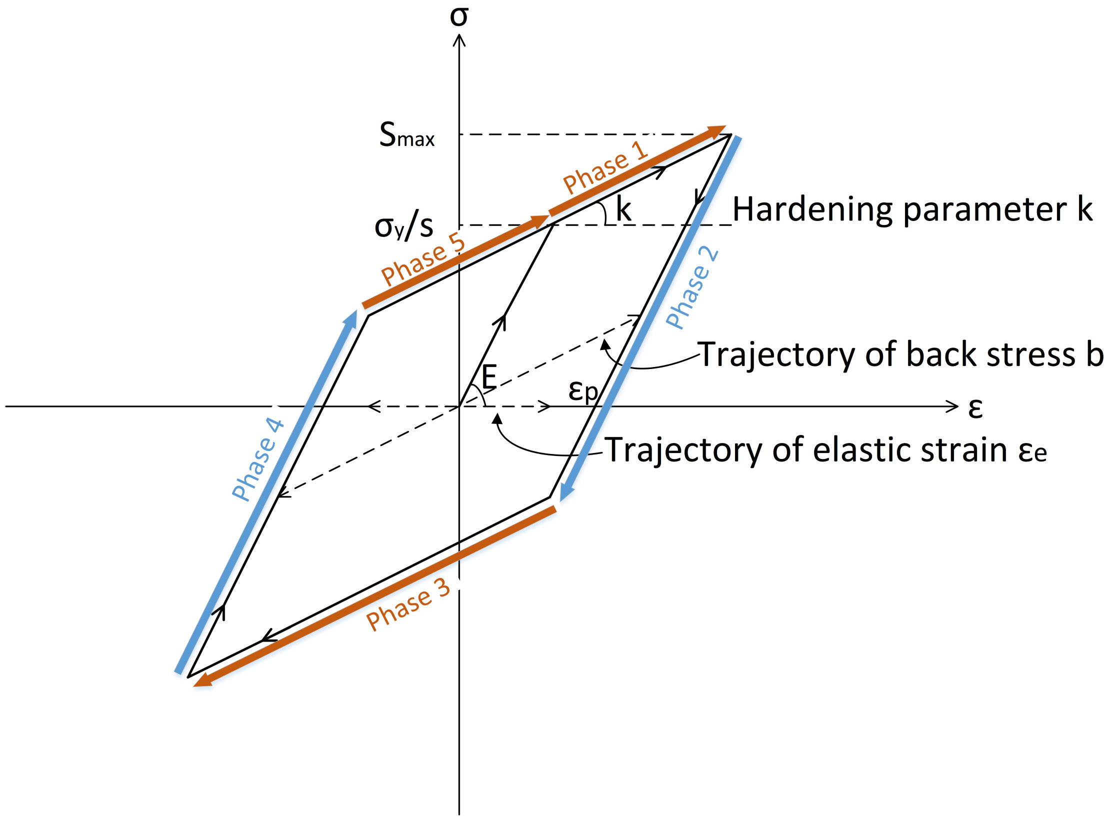

Figure 2: Uniaxial load with plastic dissipation

From Eq.(7), we then obtain that the model predicts as expected a power law dependence in function of S_{max}. However, experiments shows that the damage or the energy accumulation of a material evolves non-linearly in time. We should introduce below a method to handle such a nonlinearity.

3 Nonlinearity of damage accumulation

3.1. Energy approach with Chaboche law

The Chaboche law[4] is essentially a damage incremental law for cyclic loading of stress intensity σ with a deviatoric part A_{II} and hydrostatic part Σ_H, defining the damage increase by:

$$\delta D = \left(1 - (1 - D)^{\gamma+1}\right)^{\alpha} \left(\frac{A_{II}/\,(1 - D)}{M(\Sigma_H)}\right)^{\gamma} \delta N \tag{9}$$

which writes equivalently as Eq.(10)

$$\delta[1 - (1 - D)^{\gamma+1}]^{1-\alpha} = (1 - \alpha)(\gamma + 1) \left(\frac{A_{II}}{M(\Sigma_H)}\right)^{\gamma} \delta N = \frac{\delta N}{N_F(\sigma)}. \tag{10}$$

Here $N_F(\sigma)$ denotes the number of cycles at intensity σ to failure as obtained by integration of Eq.(10) from $D = 0$ to $D = 1$.

In our model, in case of a simple uniaxial cyclic loading, we propose to replace $\frac{1}{N_F(\sigma)}$ which is the relative unit increment of cycles by $\frac{W_{cyc}}{W_F}$, yielding the nonlinear damage incremental law:

$$\delta[1-(1-D)^{\gamma+1}]^{1-\alpha} = \frac{W_{cyc}}{W_F}\delta N. \tag{11}$$

This is a nonlinear law but used with a constant α, there will be no sequence effect. In other words, when applying two successive cycles of different intensities, the failure will occur at the same number of cycles whatever the order of the loading(high then low versus low then high).

3.2. Generalized damage accumulation

Formula (10) is a general accumulation law which can be applied for any cyclic loading sequence provided that we can identify the multiscale value of the dissipated energy per cycle.

But the notion of cycle itself may be hard to identify for general loadings. The idea is then to replace the relative increment of dissipated energy per cycle by the relative increment of dissipated energy per unit time, yielding:

$$\delta[1-(1-D)^{\gamma+1}]^{1-\alpha} = \frac{\dot{W}}{W_F}\delta t. \tag{12}$$

In a general loading case, $\dot{W}$ is to be computed. By integrating Eq.(5) over all microscales, we get:

$$\dot{W}(M,t) = \int_{s=1}^{\infty} \dot{w}(s,M,t)P(s)ds = \int_{s=1}^{\infty} \left(\underline{\underline{S}} - \underline{\underline{b}}\right)(s,M,t) : \underline{\underline{\dot{\varepsilon}^p}}(s,M,t)P(s)ds. \tag{13}$$

The evolution of $\underline{\underline{S}}$, $\underline{\underline{b}}$ and $\underline{\underline{\dot{\varepsilon}^p}}$ are given in section 1.4. Equation (12) and (13) are therefore our proposed damage law.

4 Loop on time and scales

4.1. Integration rules for $\dot{W}$ and δD

Our first approach takes one cycle as unit time. We compute analytically the energy dissipation at each scale during this cycle. The method is valid for simple loading history and which includes the integration on all weakening scales. The damage D is accumulated after each cycle.

However, there are certain limitations of this method. Firstly we need a load history decomposition in cycles. Secondly in real life the perfect close loop cycle is hardly applicable.

Thus we propose in Eq.(12) a more general method which can be integrated by a step by step strategy. We compute numerically the dissipation at different scales using an implicit Euler time integration of the constitutive laws of section 1.4. After which we make a numerical integration on different scales. Then we can update the damage and go to next time step.

Instead of doing the scale integration directly which can be difficult for complex loading, the Gaussian Quadrature rule with Legendre points is used to give the value of local dissipated energy rate.

To use the Gaussian quadrature rule the limit range of integral must be from −1 to 1, while the total dissipated energy is expressed by integrating all the weakening scale s ranging from 1 to infinity with their occurrence probabilities:

$$\dot{W} = \int_1^\infty \dot{w}(s)(\beta-1)(s)^{-\beta} ds.$$

To change the limit range of integral from $[1,\infty]$ to $[1,0]$ we take as new integration variable $u(s) = s^{-p}$ with $p = \beta - 1$, yielding $u(1) = 1$ and $u(\infty) = 0$ with

$$du = -ps^{-p-1}ds$$

that is

$$du = (-\beta+1)s^{-\beta}ds = (-\beta+1)s^{-\beta}ds.$$

Therefore the dissipated energy summed on all scales is:

$$\begin{aligned}
\dot{W} &= \int_1^\infty \dot{w}(s)(\beta-1)(s)^{-\beta} ds \\
&= \int_1^0 \dot{w}\left(u^{\frac{1}{1-\beta}}\right)(\beta-1)\frac{1}{-\beta+1}du \\
&= \int_0^1 \dot{w}\left(u^{\frac{1}{1-\beta}}\right)(\beta-1)\frac{1}{\beta-1}du \\
&= \int_0^1 \dot{w}\left(u^{\frac{1}{1-\beta}}\right)du \\
&= \frac{1}{2}\int_{-1}^1 \dot{w}\left[\left(\frac{x+1}{2}\right)^{\frac{1}{1-\beta}}\right]dx
\end{aligned} \tag{14}$$

if we set $u = \dfrac{x+1}{2}$.

So the dissipated energy rate integrated over all scales takes the form of Eq.(15):

$$\dot{W} = \frac{1}{2}\int_{-1}^1 \dot{w}\left[\left(\frac{x+1}{2}\right)^{\frac{1}{1-\beta}}, t\right]dx \approx \frac{1}{2}\sum_i \omega_i d\dot{w}\left[\left(\frac{x_i+1}{2}\right)^{\frac{1}{1-\beta}}, t\right], \tag{15}$$

where ω_i and x_i are respectively the weights and nodes of the Gauss Legerndre integration rule used for the numerical integration. In this work, we used 25 points[5].

After changing the integration limit, $\left(\dfrac{x+1}{2}\right)^{\frac{1}{1-\beta}}$ represents the weakening scale s.

Damage accumulation is deduced from Eq.(12):

$$g_{n+1} = g_n + \frac{\dot{W}dt}{W_F} \tag{16}$$

with $g_n = \left[1-(1-D_n)^{\gamma+1}\right]^{1-\alpha}$.

We upgrade the damage step by step following Eq.(16). When D reaches one, the material fails.

4.2. Regime determination under multiple scales

The material could be both in elastic and plastic regime under different scales. To be more elaborate, we reuse the fundamental equations in different regimes. At scale s, we have a dissipation rate given by:

$$\dot{w}(s) = \left(\underline{\underline{S}} - \underline{\underline{b}}\right) : \underline{\underline{\dot{\varepsilon}}}^p,$$

which differs between plastic and elastic regime.

Elastic regime:

There we have $\underline{\underline{\dot{\varepsilon}}}^p = 0$, $\underline{\underline{\dot{b}}} = 0$ and $\underline{\underline{\dot{S}}} = dev\underline{\underline{\dot{\Sigma}}}$, so

$$\underline{\underline{\dot{S}}} - \underline{\underline{\dot{b}}} = dev\underline{\underline{\dot{\Sigma}}},$$

yielding

$$\left(\underline{\underline{S}} - \underline{\underline{b}}\right)(t+dt) = \left(\underline{\underline{S}} - \underline{\underline{b}}\right)(t) + dev\underline{\underline{\dot{\Sigma}}}dt := \left(\underline{\underline{S}} - \underline{\underline{b}}\right)_{trial}(s, t+dt). \tag{17}$$

We are in elastic regime at scale s as long as we satisfy

$$\left(\underline{\underline{S}} - \underline{\underline{b}}\right)(t+dt) \leqslant (\sigma_y - \lambda\Sigma_H)/s.$$

Plastic regime:

When we leave elastic regime at scale s, we have:

$$\begin{cases} \underline{\underline{\dot{\varepsilon}}}^p = \gamma \dfrac{\underline{\underline{S}} - \underline{\underline{b}}}{\left\|\underline{\underline{S}} - \underline{\underline{b}}\right\|}, \gamma > 0, & \text{plastic flow,} \quad (18)\\ \left\|\underline{\underline{S}} - \underline{\underline{b}}\right\| = (\sigma_y - \lambda\Sigma_H)/s, & \text{yield limit,} \quad (19)\\ \left(\underline{\underline{S}} - \underline{\underline{b}}\right) : \left(\underline{\underline{\dot{S}}} - \underline{\underline{\dot{b}}}\right) = 0, & \text{yield limit time invariance,} \quad (20)\\ \underline{\underline{\dot{b}}} = \dfrac{kE}{E-k}\underline{\underline{\dot{\varepsilon}}}^p, & \text{kinematic hardening rule,} \quad (21)\\ \underline{\underline{\dot{S}}} = dev\underline{\underline{\dot{\Sigma}}} - \dfrac{E}{1+\nu}\underline{\underline{\dot{\varepsilon}}}^p, & \text{localisation rule.} \quad (22) \end{cases}$$

In all cases, we get(see annex 'Multi-dimensional analysis')

$$\left(\underline{\underline{S}} - \underline{\underline{b}}\right)(s, t+dt) = \frac{\left(\underline{\underline{S}} - \underline{\underline{b}}\right)_{trial}(s, t+dt)}{1+\eta}, \tag{23}$$

with

$$\eta = max\left\{ \underbrace{0}_{elastic\ regime}, \quad \underbrace{\frac{\left\|\underline{\underline{S}} - \underline{\underline{b}}\right\|_{trial}}{(\sigma_y - \lambda\Sigma_H)/s} - 1}_{plastic\ regime\ when\ this\ number\ is\ positive} \right\},$$

$$\left(\underline{\underline{S}}-\underline{\underline{b}}\right)_{trial}(s,t+dt)=\left(\underline{\underline{S}}-\underline{\underline{b}}\right)(s,t)+dev\dot{\underline{\underline{\Sigma}}}(t)dt.$$

That is to say, when the structure is in elastic regime at time t and scale s, we have $\left(\underline{\underline{S}}-\underline{\underline{b}}\right)(s,t)=\left(\underline{\underline{S}}-\underline{\underline{b}}\right)_{trial}(s,t)$. Otherwise, if the norm of $\left(\underline{\underline{S}}-\underline{\underline{b}}\right)_{trial}(s,t)$ is greater than the local yield limit $\left(\sigma_y-\lambda\Sigma_H\right)/s$, $\left(\underline{\underline{S}}-\underline{\underline{b}}\right)(s,t)$ will be projected on the yield limit.

Knowing the distinction between elastic and plastic regime under multiple scales, we compute the general expression of the dissipated energy rate.

$$\dot{w}=\left(\underline{\underline{S}}-\underline{\underline{b}}\right):\dot{\underline{\underline{\varepsilon}}}^p=\gamma\frac{\sigma_y-\lambda\Sigma_H}{s}. \tag{24}$$

From Eq.(A.5) and Eq.(A.8) in annex, we get:

$$E\gamma dt=\left\langle\left\|\underline{\underline{S}}-\underline{\underline{b}}\right\|_{trial}-\frac{\sigma_y-\lambda\Sigma_H}{s}\right\rangle/\left(\frac{1}{1+\nu}+\frac{k}{E-k}\right)=\left\langle\left\|\underline{\underline{S}}-\underline{\underline{b}}\right\|_{trial}-\frac{\sigma_y-\lambda\Sigma_H}{s}\right\rangle\frac{(E-k)(1+\nu)}{(E+k\nu)}, \tag{25}$$

where $\langle\ \rangle$ is Macaulay bracket symbol defined as $\langle m\rangle=0$ if $m\leqslant 0$, otherwise $\langle m\rangle=m$.

We replace γ deduced from Eq.(25) in Eq.(24) to give the expression of local energy dissipation rate at scale s:

$$\dot{w}dt=\frac{(E-k)(1+\nu)}{E(E+k\nu)}\left\langle\left\|\underline{\underline{S}}-\underline{\underline{b}}\right\|_{trial}-\frac{\sigma_y-\lambda\Sigma_H}{s}\right\rangle\frac{\sigma_y-\lambda\Sigma_H}{s}. \tag{26}$$

With Eq.(15), the final expression of energy dissipation W during time step dt writes:

$$\begin{aligned}W&=\dot{W}dt\\&=\frac{1}{2}\sum_i\omega_i\dot{w}\left[\left(\frac{x+1}{2}\right)^{\frac{1}{1-\beta}}\right]dt\\&=\frac{(E-k)(1+\nu)}{2E(E+k\nu)}\sum_i\omega_i\left\langle\left\|\underline{\underline{S}}-\underline{\underline{b}}\right\|_{trial}-\frac{\sigma_y-\lambda\Sigma_H}{\left(\frac{x_i+1}{2}\right)^{\frac{1}{1-\beta}}}\right\rangle\frac{\sigma_y-\lambda\Sigma_H}{\left(\frac{x_i+1}{2}\right)^{\frac{1}{1-\beta}}}.\end{aligned} \tag{27}$$

We have the damage accumulation deduced in Eq.(16):

$$g_{n+1}=g_n+\frac{\dot{W}dt}{W_F}=g_n+\frac{W}{W_F},$$

with $D_n=\left[1-\left(1-g_n^{\frac{1}{1-\alpha}}\right)^{\frac{1}{\gamma+1}}\right]$.

Now we are able to put these formula into numerical tests.

5 Test on different load histories

5.1. One dimensional application to simple cyclic data

The test is first performed on a sinusoidal axial load $\Sigma = C sin(t)$ with parameters in Table.1, giving a deviatoric amplitude $S_{max} = \sqrt{\frac{2}{3}}C$.

Parameters	**Value**
Load	$\Sigma = 5e8 sin(t)$ Pa
Young's modulus	$E = 2e11$ Pa
Hardening parameter	$k = 6e8$ Pa
Weakening scales distribution exponent	$\beta = 3$
Hydrostatic pressure sensitivity	$\lambda = 0.5$
Macroscopic yield stress	$\sigma_y = 6.38e8$ Pa
Material parameter from Chaboche law(Wohler curve exponent)	$\gamma = 0.5$
Non-linearity of damage accumulation	$\alpha = 0.5$
Initial damage	$D = 0$
Initial time	$t = 0$ s
Dissipated energy to failure per unit volume	$W_F = 3e6$ J
Looping step	1 s

Table 1: Material parameters in a simple cyclic load

We use matlab to realize our analytical method. We plot $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)_{trial}$ and $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)$ for two different scales($s_1 = 21.21657929229650$ and $s_8 = 2.176132808422946$) in Figure 3.

The nonlinearity is determined by

$$\alpha = 1 - a\left\langle \frac{\max_t \sqrt{J_{2,a}}(t) + a_c P_{max}(t) - b_c}{\sigma_u - 2\max\sqrt{J_{2,a}}} \right\rangle,$$

which is predominated by Crossland criterion, for simplicity we take α as a constant. The damage evolves like in Figure 4, where we compare the damage evolution as predicted by the cycle accumulation Eq.(7) and by the numerical strategy of section 4.

Now we compare the result to the one demonstrated in Figure 2. The first cycle has 3 phases which have the energy loss identical to phase 1. The following cycles each have 4 times energy loss as phase 1. We can see from Figure 5 and Figure 6 the difference between cyclic load calculation and numerical method as function of time(time step=1/5000s and 1/15000s separately). Because the step by step damage accumulation grows in a power law, so the amplitude of difference grows with time. However, the difference between the two

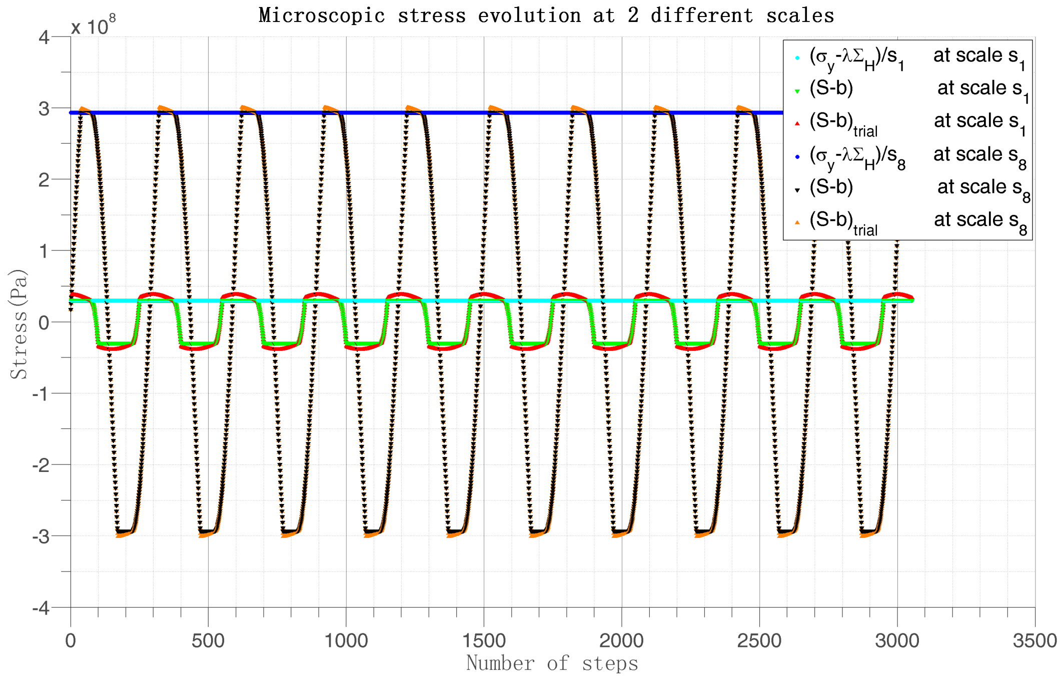

Figure 3: Microscopic $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)_{trial}$ and $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)$ evolution with time under different weakening scales in sinusoidal load

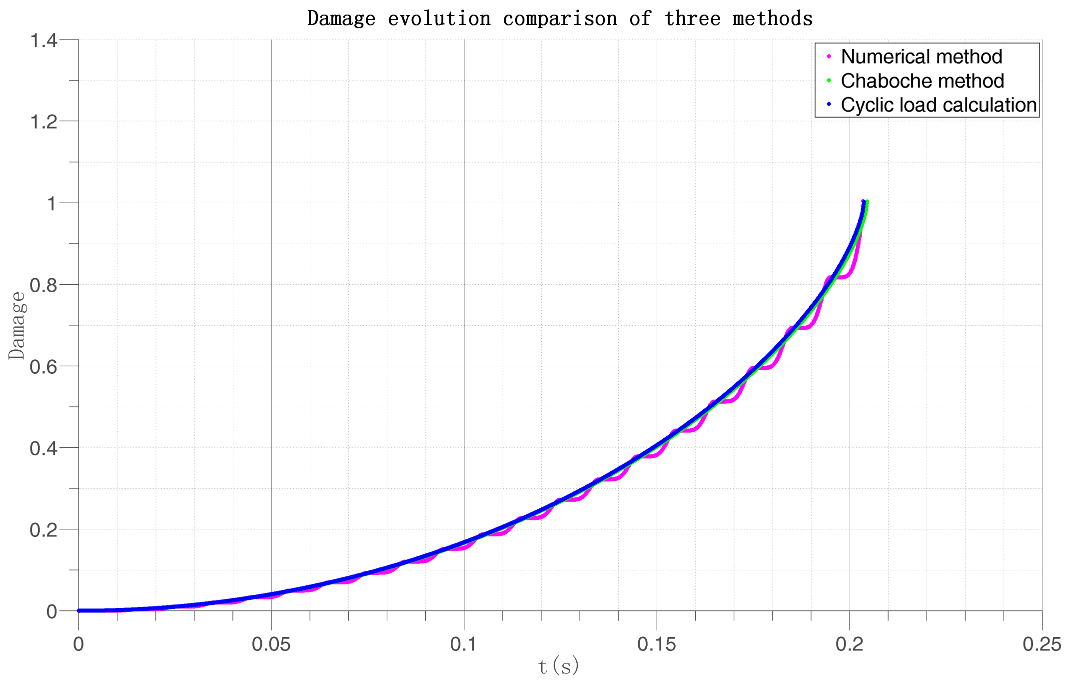

Figure 4: Damage evolution with time under sinusoidal load with two different methods

methods swing around 0 and from Figure 4 we can see the difference is not symmetrical, we could consider the numerical method converges in cyclic load calculation method.

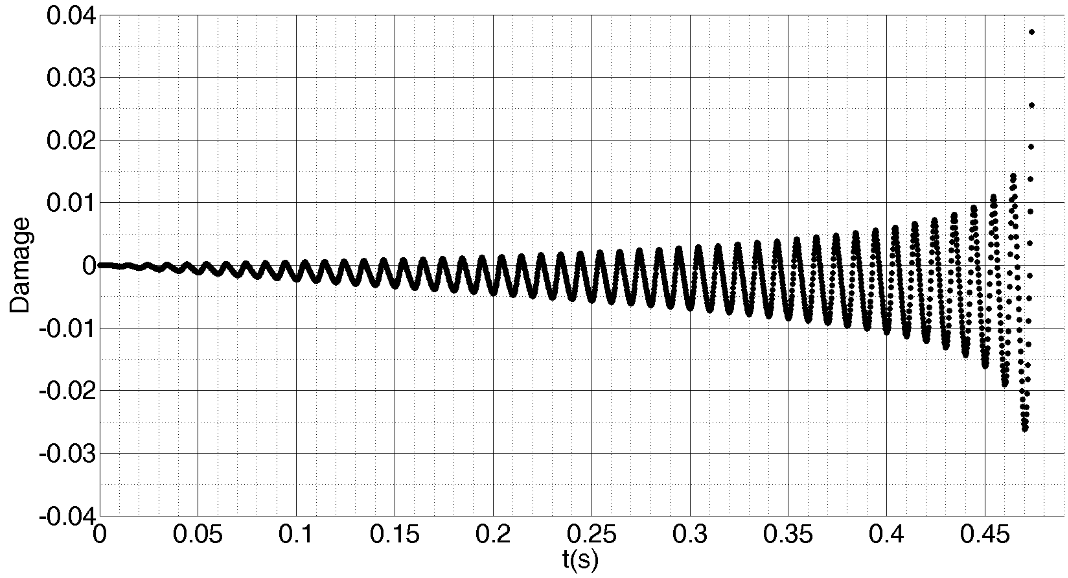

Figure 5: Difference between cyclic load calculation and numerical method as function of time(time step=1/5000s)

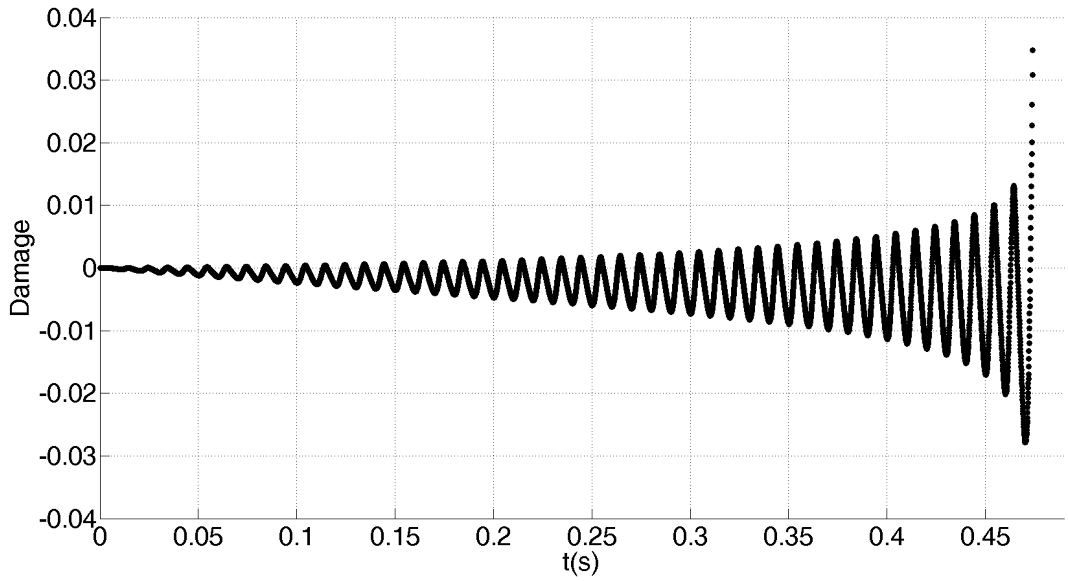

Figure 6: Difference between cyclic load calculation and numerical method as function of time(time step=1/15000s)

The cyclic load calculation is only valid for very simple such as proportional loading in fatigue, nevertheless it can still be used as a comparison group to verify the numerical results. The outcome seems satisfactory. Hence, to be more general for any loading history, we adopt the numerical method.

5.2. One dimensional application to PSA data

In this test, we reconstruct a unidimensional macroscopic stress history from recorded force data proposed by PSA group.

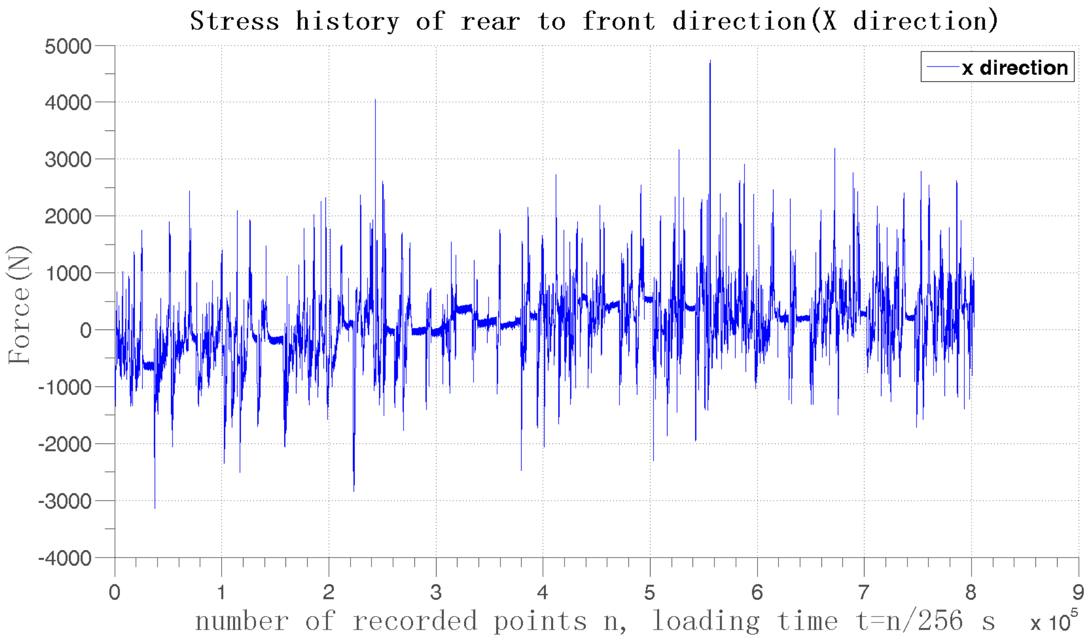

Figure 7: Loading history of X direction, force vs the record index n, with 256 sample recorded per second

The sample recording rate is 256 per second. In order to accumulate damage using very small steps, we have created 10 additional points between every 2 recorded points by linear interpolation. So the sample rate is $256 * 10$ per second.

The force on wheel is firstly considered as under uniaxial loading F_x. Here we temporally set $\Sigma_x = F_x/A$ where $A = \dfrac{1}{1e6}m^2$ is the area of force, and $W_F = 3e6J$. The other data are as Table.1. The plot of $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)_{trial}$ and $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)$ under 2 different scales(s_1 = 21.21657929229650 and s_8 = 2.176132808422946)are shown in Figure 8. The damage evolves like Figure 10.

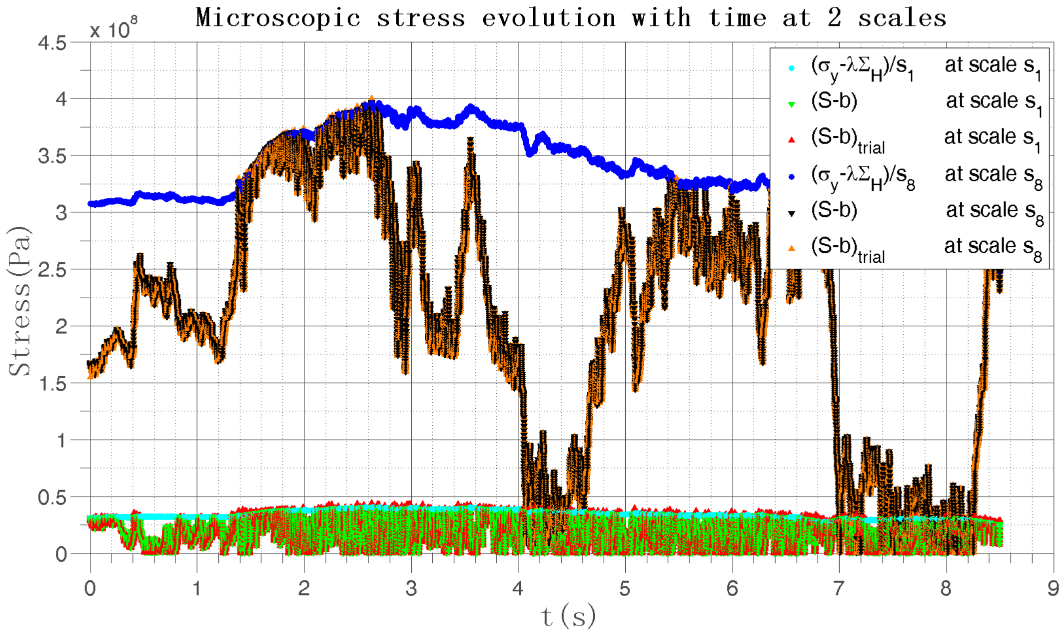

Figure 8: $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)_{trial}$ and $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)$ evolution with time under different weakening scales in PSA load history

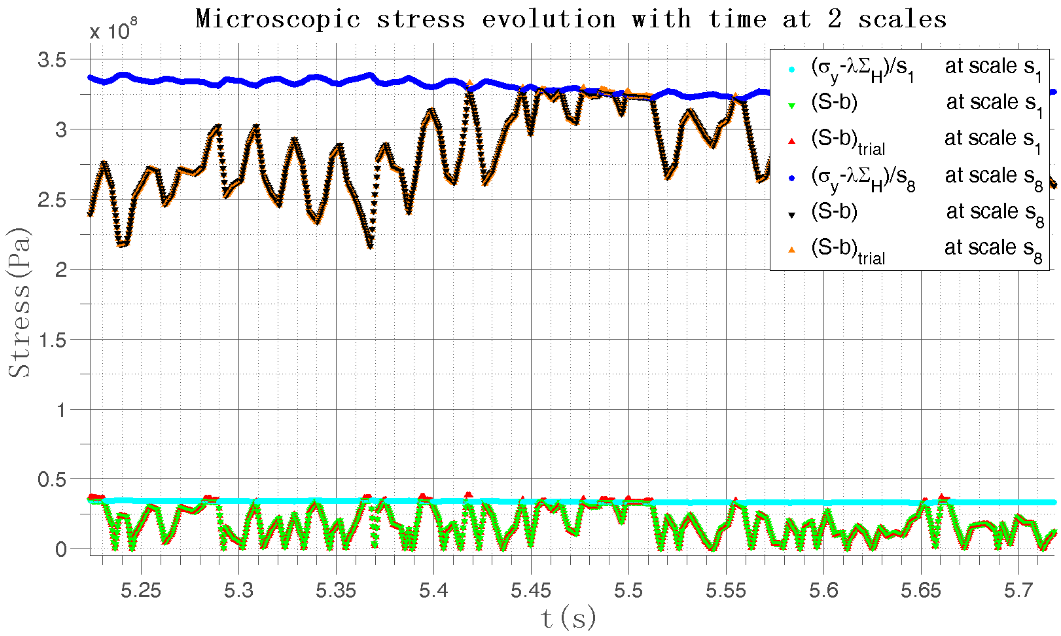

Figure 9: Circled area magnification in Figure 8 where there is more $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)_{trial} > \sigma_y$(plasticity) at s_1 than at s_8

5.3. Multi-dimensional application to PSA data

We now consider a situation where we have force recorded measured in 3 different directions as shown in Figure 11. In real case, the vertical force F_z is much larger than the axial and horizontal forces F_x and F_y,

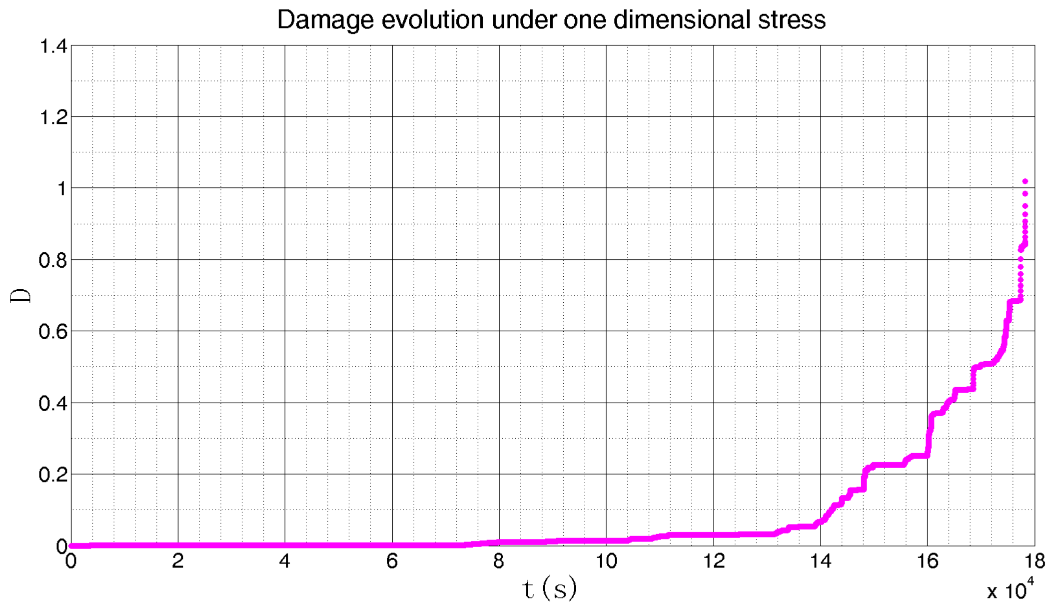

Figure 10: Damage evolution with time at one dimension PSA load history

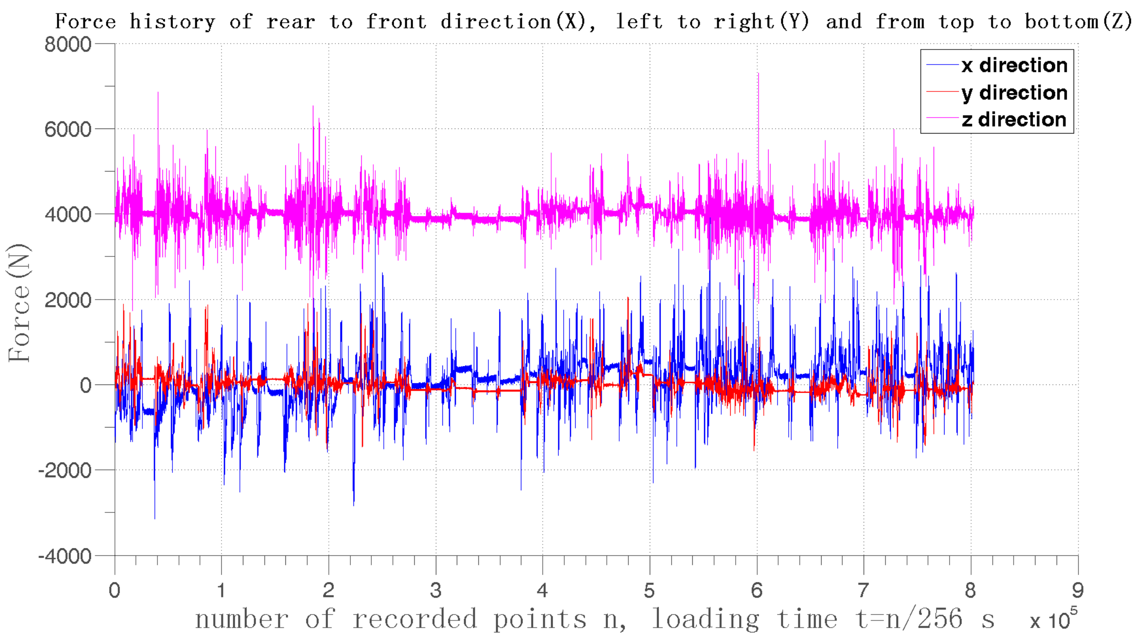

Figure 11: Loading history of 3 different directions

as shown in Figure 11. However, in order to investigate large domains of interest, we first scale the axial and horizontal forces to reach comparable impact and transform them in principal stresses $c_x \frac{F_x}{A}$ applied along the stress principle vector $\underline{e}_\alpha$(respectively $\underline{e}_\beta$) that we choose randomly(Figure 12). We therefore consider

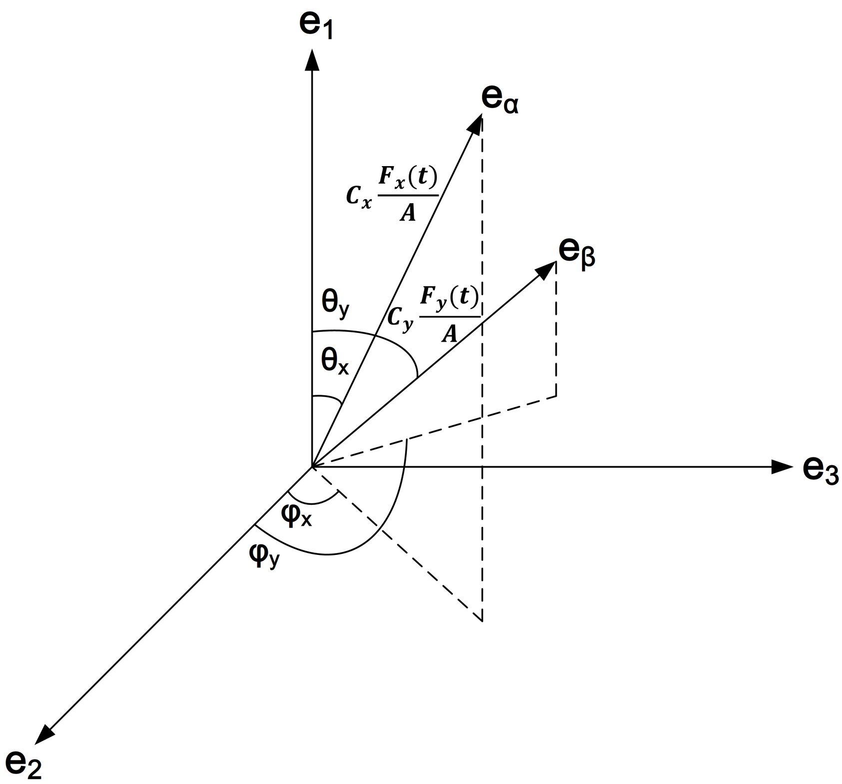

Figure 12: Loading in 3 different directions

the following macroscopic stress tensor:

$$\underline{\underline{\Sigma}} = \frac{F_z(t)}{A}\underline{e}_1 \otimes \underline{e}_1 + c_x \frac{F_x(t)}{A}\underline{e}_\alpha \otimes \underline{e}_\alpha + c_y \frac{F_y(t)}{A}\underline{e}_\beta \otimes \underline{e}_\beta \tag{28}$$

where $\underline{e}_\alpha$ and $\underline{e}_\beta$ are principal vectors whose spherical coordinate are θ_x, φ_x, θ_y and φ_y respectively:

$$\underline{e}_\alpha = cos\theta_x \underline{e}_1 + sin\theta_x cos\varphi_x \underline{e}_2 + sin\theta_x sin\varphi_x \underline{e}_3,$$

$$\underline{e}_\beta = cos\theta_y \underline{e}_1 + sin\theta_y cos\varphi_y \underline{e}_2 + sin\theta_y sin\varphi_y \underline{e}_3.$$

Here $F_x(t)$, $F_y(t)$, $F_z(t)$ are from test data, and θ_x, φ_x, θ_y, φ_y are structural parameters to be chosen randomly.The physical data are the same with parameters in Table.1. The structural data we choose is shown in Table.2.

Parameter	A(m^2)	c_x	c_y	θ_x	φ_x	θ_y	φ_y
Value	1/6e4	10	60	0.5	0.3	0.6	0.4

Table 2: The structural data in 3D analysis

The underlying assumption is that a unit load on wheel in direction $\underline{e}_x$ creates a stress tensor at point $\boldsymbol{M}$ given by:

$$c_x \frac{F_x(t)}{A} \underline{e}_\alpha \otimes \underline{e}_\alpha,$$

where $\underline{e}_\alpha \otimes \underline{e}_\alpha$ defines the local structural response of the vehicle.

Replacing $\underline{e}_\alpha$ and $\underline{e}_\beta$ in Eq.(28) we get the stress tensor in Eq.(A.9) in the annex.

The plot of $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)_{trial}$ and $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)$ under 2 different scales are shown in Figure 13.

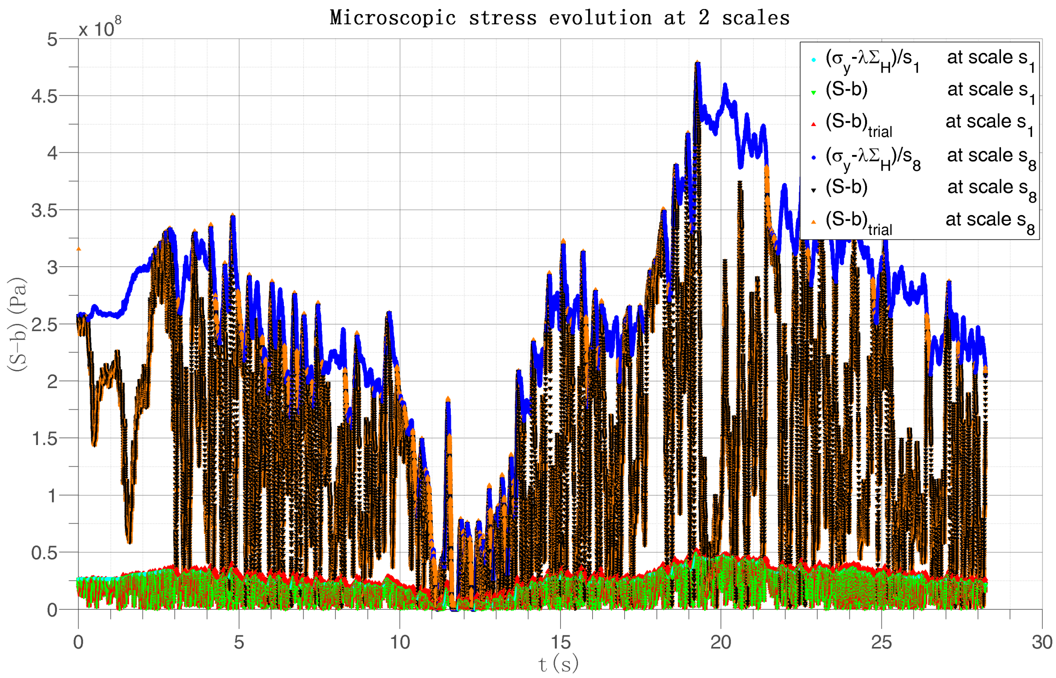

Figure 13: $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)_{trial}$ and $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)$ evolution with time under different weakening scales in PSA load history

In the load history, when $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)_{trial} > \sigma_y$, the damage accumulates. However, under scale s_{10}, there are much less damage accumulation than under scale s_1. In this way we do not neglect the small influences in load history and the big fluctuation in stress is magnified which reflects the real situation.

The damage evolves like in Figure 14.

We can improve the result by inserting more arithmetic sequence points between every 2 recorded points. As is shown in Table.3 :

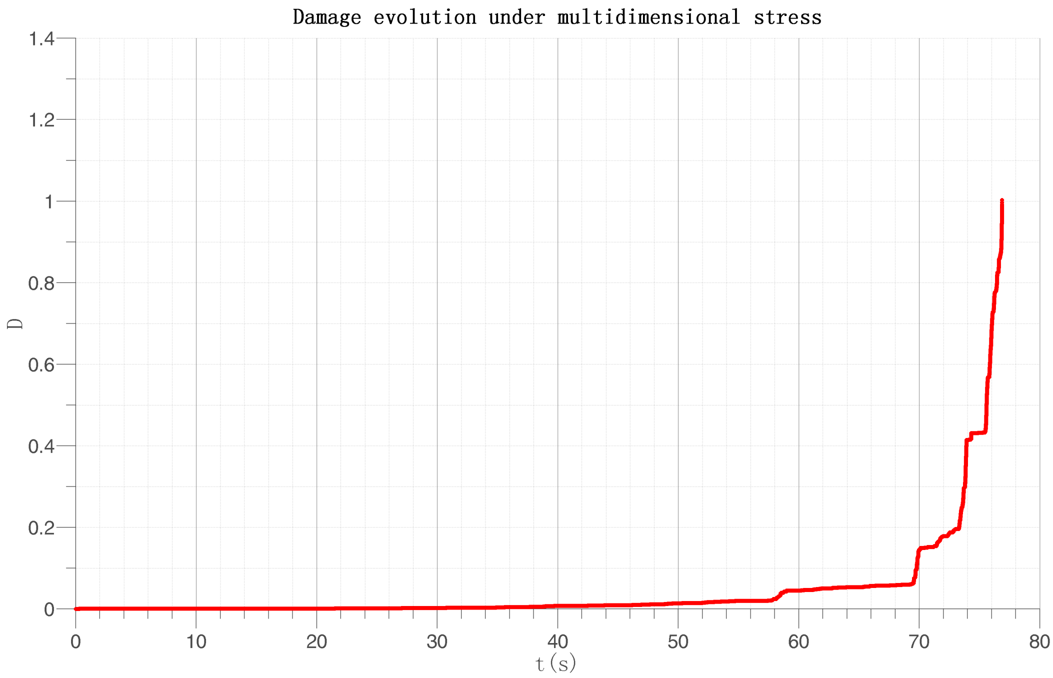

Figure 14: Damage evolution under multidimensional stress

Table 3: Arithmetic sequence points density effect

Arithmetic sequence points between every two points	**Total time to failure(s)**
10	78.63711
20	72.24630
30	70.25793
50	68.69148
100	67.49223

6 Discussion

The strategy can be made more complex by introducing a local space averaging process in the calculation of the local damage, and by taking more general plastic flows. The energy based fatigue approach takes into account impurities and hardness in the material which affect the fatigue life. The load sequence effects for complex multiaxial loading history are included in damage accumulation process. The small step-by-step strategy does not ignore small fluctuations in the load history. In addition, it can take into account any type of micro plasticity law and multiaxial load geometry.

Further research of energy based failure criteria should be focused on the following aspects:

1. The accommodation law might be more elaborate than kinematic hardening.
2. The differentiation of shear stress and normal stress effect on fatigue life should be clarified.
3. The non-linearity parameter α contains the stress σ, so it can evolve with time. But for complex loading history, should it change at every time step?

Acknowledgments

We are grateful for the financial and technical support of Chaire PSA.

References

[1] M. Maitournam, C. Krebs, A. Galtier, A multiscale fatigue life model for complex cyclic multiaxial loading, International Journal of Fatigue 33 (2) (2011) 232 – 240. doi:http://dx.doi.org/10.1016/j.ijfatigue.2010.08.017.
URL http://www.sciencedirect.com/science/article/pii/S0142112310001933

[2] S. Bosia, A. Constantinescu, Fast time-scale average for a mesoscopic high cycle fatigue criterion, International Journal of Fatigue 45 (2012) 39 – 47. doi:http://dx.doi.org/10.1016/j.ijfatigue.2012.06.015.
URL http://www.sciencedirect.com/science/article/pii/S0142112312002162

[3] M. Zepeng, Structures under multiaxial fatigue taking consideration of gradients of the constraints with time and space variation.

[4] J. Lemaitre, J.-L. Chaboche, Mechanics of solid materials, Cambridge university press, 1990.

[5] Legendre Gauss Quadrature weights and nodes, https://www.mathworks.com/matlabcentral/fileexchange/4540-legendre-gauss-quadrature-weights-and-nodes, accessed: 2004-05-11.

Appendices

Appendix A DETAILED EXPLOITATION

****

A DETAILED DESCRIPTION OF ANALYTICAL EXPLOITATION ON UNIAXIAL CYCLE

****

Phase 1: The deviatoric stress amplitude increases from σ_y/s to S_{max}.

The material is in local plastic regime, then $\dot{\varepsilon}^p > 0$ and $\dot{\sigma} - \dot{b} = 0 \Rightarrow \dot{\Sigma} - \frac{E}{1+\nu}\dot{\varepsilon}^p = \frac{kE}{E-k}\dot{\varepsilon}^p \Rightarrow$

$$\dot{\varepsilon}^p = \frac{(E-k)(1+\nu)}{E(E+k\nu)}\dot{\Sigma}.$$

$\Rightarrow \dot{\varepsilon}^p$ varies from 0 to $\frac{(E-k)(1+\nu)(S_{max}-\sigma_y/s)}{E(E+k\nu)}$.

From Taylor-Lin scale transition model:

$$\dot{\sigma} = \dot{\Sigma} - \frac{E}{1+\nu}\dot{\varepsilon}_p = \dot{\Sigma} - \frac{E-k}{E-\nu k}\dot{\Sigma} = \frac{k(1-\nu)}{E-k\nu}\dot{\Sigma}.$$

$\Rightarrow \sigma$ varies from σ_y/s to $\sigma_y/s + \frac{k(1-\nu)(S_{max}-\sigma_y/s)}{E-k\nu}$.

$$\dot{b} = \dot{\Sigma} - \frac{E}{1+\nu}\dot{\varepsilon}_p = \dot{\Sigma} - \frac{E-k}{E-\nu k}\dot{\Sigma} = \frac{k(1-\nu)}{E-k\nu}\dot{\Sigma}.$$

$\Rightarrow b$ varies from 0 to $\frac{k(1-\nu)(S_{max}-\sigma_y/s)}{E-k\nu}$.

So the energy dissipation rate is:

$$(\sigma - b)\dot{\varepsilon}^p = \frac{\sigma_y}{s}\dot{\varepsilon}^p = \frac{\sigma_y}{s}\frac{(E-k)(1+\nu)}{E(E+k\nu)}\dot{\Sigma}.$$

The energy dissipation is:

$$(\sigma - b)\Delta\varepsilon^p = \frac{\sigma_y}{s}\frac{(E-k)(1+\nu)(S_{max}-\sigma_y/s)}{E(E+k\nu)}.$$

Phase 2: The deviatoric stress amplitude decreases from S_{max} to $S_{max} - 2\sigma_y/s$.

The material is in local elastic regime, then $\dot{\varepsilon}^p = 0$ and $\dot{\sigma} - \dot{b} = 0 \Rightarrow$

$\dot{b} = 0$, $\dot{\sigma} = \dot{\Sigma} - \frac{E}{1+\nu}\dot{\varepsilon}_p = \dot{\Sigma}$.

σ varies from $\sigma_y/s + \frac{k(1-\nu)(S_{max}-\sigma_y/s)}{E-k\nu}$ to $-\sigma_y/s + \frac{k(1-\nu)(S_{max}-\sigma_y/s)}{E-k\nu}$.

$\sigma - b$ varies from σ_y/s to $-\sigma_y/s$.

The energy dissipation rate is:

$$(\sigma - b)\dot{\varepsilon}^p = 0.$$

Phase 3: The deviatoric stress amplitude decreases from $S_{max} - 2\sigma_y/s$ to $-S_{max}$.

The material is in local plastic regime, then $\dot{\varepsilon}^p > 0$ and $\dot{\sigma} - \dot{b} = 0 \Rightarrow$

$$\dot{\varepsilon}^p = \frac{(E-k)(1+\nu)}{E(E+k\nu)}\dot{\Sigma}$$

as opposite to phase 1 for $\dot{\Sigma} < 0$.

$\Rightarrow \varepsilon^p$ varies from $\dfrac{(E-k)(1+\nu)(S_{max} - \sigma_y/s)}{E(E+k\nu)}$ to
$\dfrac{(E-k)(1+\nu)(S_{max} - \sigma_y/s - S_{max} - (S_{max} - 2\sigma_y/s))}{E(E+k\nu)} = -\dfrac{(E-k)(1+\nu)(S_{max} - \sigma_y/s)}{E(E+k\nu)}$.

From Taylor-Lin scale transition model:

$$\dot{\sigma} = \dot{\Sigma} - \frac{E}{1+\nu}\dot{\varepsilon}_p = \dot{\Sigma} - \frac{E-k}{E-\nu k}\dot{\Sigma} = \frac{k(1-\nu)}{E-k\nu}\dot{\Sigma}.$$

$\Rightarrow \sigma$ varies from $-\sigma_y/s + \dfrac{k(1-\nu)(S_{max} - \sigma_y/s)}{E-k\nu}$ to $-\sigma_y/s - \dfrac{k(1-\nu)(S_{max} - \sigma_y/s)}{E-k\nu}$.

$$\dot{b} = \dot{\Sigma} - \frac{E}{1+\nu}\dot{\varepsilon}_p = \dot{\Sigma} - \frac{E-k}{E-\nu k}\dot{\Sigma} = \frac{k(1-\nu)}{E-k\nu}\dot{\Sigma}.$$

$\Rightarrow b$ varies from $\dfrac{k(1-\nu)(S_{max} - \sigma_y/s)}{E-k\nu}$ to $-\dfrac{k(1-\nu)(S_{max} - \sigma_y/s)}{E-k\nu}$.

So the energy dissipation rate is:

$$(\sigma - b)\dot{\varepsilon}^p = -\frac{\sigma_y}{s}\dot{\varepsilon}^p = -\frac{\sigma_y}{s}\frac{(E-k)(1+\nu)}{E(E+k\nu)}\dot{\Sigma}.$$

The energy dissipation is:

$$(\sigma - b)\Delta\varepsilon^p = -\frac{\sigma_y}{s}\frac{(E-k)(1+\nu)(-2S_{max} + 2\sigma_y/s)}{E(E+k\nu)} = \frac{2\sigma_y}{s}\frac{(E-k)(1+\nu)(S_{max} - \sigma_y/s)}{E(E+k\nu)}.$$

Phase 4: The deviatoric stress amplitude increases from $-S_{max}$ to $-S_{max} + 2\sigma_y/s$.

The material is in local elastic regime, then $\dot{\varepsilon}^p = 0$ and $\dot{\sigma} - \dot{b} = 0 \Rightarrow$

$\dot{b} = 0$, $\dot{\sigma} = \dot{\Sigma} - \dfrac{E}{1+\nu}\dot{\varepsilon}_p = \dot{\Sigma}$.

σ varies from $-\sigma_y/s - \dfrac{k(1-\nu)(S_{max} - \sigma_y/s)}{E-k\nu}$ to $\sigma_y/s - \dfrac{k(1-\nu)(S_{max} - \sigma_y/s)}{E-k\nu}$.

$\sigma - b$ varies from $-\sigma_y/s$ to σ_y/s.

So the energy dissipation rate is:

$$(\sigma - b)\dot{\varepsilon}^p = 0.$$

Phase 5: The deviatoric stress amplitude increases from $-S_{max} + 2\sigma_y/s$ to σ_y/s.

The material is in local plastic regime, then $\dot{\varepsilon}^p > 0$ and $\dot{\sigma} - \dot{b} = 0 \Rightarrow$

$$\dot{\varepsilon}^p = \frac{(E-k)(1+\nu)}{E(E+k\nu)}\dot{\Sigma}$$

as in phase 1.

$\Rightarrow \dot{\varepsilon}^p$ varies from $-\dfrac{(E-k)(1+\nu)(S_{max}-\sigma_y/s)}{E(E+k\nu)}$ to 0.

$$\dot{\sigma} = \dot{\Sigma} - \frac{E}{1+\nu}\dot{\varepsilon}_p = \dot{\Sigma} - \frac{E-k}{E-\nu k}\dot{\Sigma} = \frac{k(1-\nu)}{E-k\nu}\dot{\Sigma}.$$

$\Rightarrow \sigma$ varies from $\sigma_y/s - \dfrac{k(1-\nu)(S_{max}-\sigma_y/s)}{E-k\nu}$ to σ_y/s.

$$\dot{b} = \dot{\Sigma} - \frac{E}{1+\nu}\dot{\varepsilon}_p = \dot{\Sigma} - \frac{E-k}{E-\nu k}\dot{\Sigma} = \frac{k(1-\nu)}{E-k\nu}\dot{\Sigma}.$$

$\Rightarrow b$ varies from $-\dfrac{k(1-\nu)(S_{max}-\sigma_y/s)}{E-k\nu}$ to 0.

So the energy dissipation rate is:

$$(\sigma - b)\dot{\varepsilon}^p = \frac{\sigma_y}{s}\dot{\varepsilon}^p = \frac{\sigma_y}{s}\frac{(E-k)(1+\nu)}{E(E+k\nu)}\dot{\Sigma}.$$

The energy dissipation is:

$$(\sigma - b)\Delta\varepsilon^p = \frac{\sigma_y}{s}\frac{(E-k)(1+\nu)(S_{max}-\sigma_y/s)}{E(E+k\nu)}.$$

From the three phase analysis in local plastic regime, the dissipated energy is like $dW(phase1) = \frac{1}{2}dW(phase3) = dW(phase5)$ and the dissipation rate is like $d\dot{W}(phase1) = d\dot{W}(phase3) = d\dot{W}(phase5)$.

$$d\dot{W} = \frac{(E-k)(1+\nu)}{E(E-k\nu)}\left(\frac{\sigma_y}{s}\right)|\dot{\Sigma}| \tag{A.1}$$

****

MULTI-DIMENSIONAL PLASTIC AND ELASTIC REGIME ANALYSIS

****

At a certain scale s_i, after elimination of $\dot{\underline{\underline{\varepsilon}}}^p$, there are

$$\dot{\underline{\underline{S}}} - \dot{\underline{\underline{b}}} = dev\dot{\underline{\underline{\Sigma}}} - E\gamma\left(\frac{1}{1+\nu} + \frac{k}{E-k}\right)\frac{\underline{\underline{S}} - \underline{\underline{b}}}{\left\|\underline{\underline{S}} - \underline{\underline{b}}\right\|}.$$

If we are at yield limit at (t+dt), we get on the other hand:

$$\left(\underline{\underline{S}} - \underline{\underline{b}}\right)(t+dt) = \left(\underline{\underline{S}} - \underline{\underline{b}}\right)(t) + \left(\dot{\underline{\underline{S}}} - \dot{\underline{\underline{b}}}\right)dt,$$

$$\left\|\left(\underline{\underline{S}} - \underline{\underline{b}}\right)(t+dt)\right\| = \left(\sigma_y - \lambda\sigma_m\right)/s_i. \tag{A.2}$$

Replacing $\left(\dot{\underline{\underline{S}}} - \dot{\underline{\underline{b}}}\right)$ in the integration by its expression we get:

$$\left(\underline{\underline{S}} - \underline{\underline{b}}\right)(t+dt) = \left(\underline{\underline{S}} - \underline{\underline{b}}\right)(t) + dev\dot{\underline{\underline{\Sigma}}}dt - E\gamma dt\left(\frac{1}{1+\nu} + \frac{k}{E-k}\right)\frac{\left(\underline{\underline{S}} - \underline{\underline{b}}\right)(t+dt)}{\left\|\underline{\underline{S}} - \underline{\underline{b}}\right\|(t+dt)} \tag{A.3}$$

Putting all terms with $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)(t+dt)$ on the left hand side, we get:

$$\left(\underline{\underline{S}} - \underline{\underline{b}}\right)(t+dt)\left(1+\eta\right) = \left(\underline{\underline{S}} - \underline{\underline{b}}\right)(t) + dev\dot{\underline{\underline{\Sigma}}}dt = \left(\underline{\underline{S}} - \underline{\underline{b}}\right)_{trial}(t+dt) \tag{A.4}$$

with

$$\eta = \frac{E\gamma dt}{\left\|\underline{\underline{S}} - \underline{\underline{b}}\right\|(t+dt)}\left(\frac{1}{1+\nu} + \frac{k}{E-k}\right). \tag{A.5}$$

To see whether the structure is in elastic or plastic regime at each time step, we use $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)_{trial}(t+dt)$ to compare with the yield stress at the same scale s_i, thus to give a value to $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)(t+dt)$.

Since $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)(t+dt)$ is in the same direction as $\left(\underline{\underline{S}} - \underline{\underline{b}}\right)_{trial}(t+dt)$, we have

$$\left(\underline{\underline{S}} - \underline{\underline{b}}\right)(t+dt) = \left(\sigma_y - \lambda\sigma_m\right)/s\frac{\left(\underline{\underline{S}} - \underline{\underline{b}}\right)_{trial}(t+dt)}{\left\|\underline{\underline{S}} - \underline{\underline{b}}\right\|_{trial}(t+dt)} \tag{A.6}$$

We now compare Eq.(A.4) and Eq.(A.6), the only solution is to have:

$$1+\eta = \frac{\left\|\underline{\underline{S}} - \underline{\underline{b}}\right\|_{trial}}{\left(\sigma_y - \lambda\sigma_m\right)/s} \tag{A.7}$$

that is:

$$\eta = \frac{\left\|\underline{\underline{S}} - \underline{\underline{b}}\right\|_{trial}}{\left(\sigma_y - \lambda\sigma_m\right)/s} - 1 \tag{A.8}$$

which is positive in plastic regime.

****

3D STRESS TENSOR

****

$$
\begin{aligned}
\underline{\underline{\Sigma}} =& \frac{F_z}{A}\underline{e}_1 \otimes \underline{e}_1 + c_x\frac{F_x}{A}\underline{e}_\alpha \otimes \underline{e}_\alpha + c_y\frac{F_y}{A}\underline{e}_\beta \otimes \underline{e}_\beta \\
=& \frac{F_z}{A}\underline{e}_1 \otimes \underline{e}_1 + c_x\frac{F_x}{A}\left(cos\theta_x\underline{e}_1 + sin\theta_x cos\varphi_x\underline{e}_2 + sin\theta_x sin\varphi_x\underline{e}_3\right) \otimes \left(cos\theta_x\underline{e}_1 + sin\theta_x cos\varphi_x\underline{e}_2 + sin\theta_x sin\varphi_x\underline{e}_3\right) \\
&+ c_y\frac{F_y}{A}\left(cos\theta_y\underline{e}_1 + sin\theta_y cos\varphi_y\underline{e}_2 + sin\theta_y sin\varphi_y\underline{e}_3\right) \otimes \left(cos\theta_y\underline{e}_1 + sin\theta_y cos\varphi_y\underline{e}_2 + sin\theta_y sin\varphi_y\underline{e}_3\right) \\
=& \left(\frac{F_z}{A} + c_x\frac{F_x}{A}cos^2\theta_x + c_y\frac{F_zy}{A}cos^2\theta_y\right)\underline{e}_1 \otimes \underline{e}_1 \\
&+ \left(c_x\frac{F_x}{A}cos\theta_x sin\theta_x cos\varphi_x + c_y\frac{F_y}{A}cos\theta_y sin\theta_y cos\varphi_y\right)\left(\underline{e}_1 \otimes \underline{e}_2 + \underline{e}_2 \otimes \underline{e}_1\right) \\
&+ \left(c_x\frac{F_x}{A}cos\theta_x sin\theta_x sin\varphi_x + c_y\frac{F_y}{A}cos\theta_y sin\theta_y sin\varphi_y\right)\left(\underline{e}_1 \otimes \underline{e}_3 + \underline{e}_3 \otimes \underline{e}_1\right) \\
&+ \left(c_x\frac{F_x}{A}sin^2\theta_x cos^2\varphi_x + c_y\frac{F_y}{A}sin^2\theta_y cos^2\varphi_y\right)\underline{e}_2 \otimes \underline{e}_2 \\
&+ \left(c_x\frac{F_x}{A}sin^2\theta_x cos\varphi_x sin\varphi_x + c_y\frac{F_y}{A}sin^2\theta_y cos\varphi_y sin\varphi_y\right)\left(\underline{e}_2 \otimes \underline{e}_3 + \underline{e}_3 \otimes \underline{e}_2\right) \\
&+ \left(c_x\frac{F_x}{A}sin^2\theta_x sin^2\varphi_x + c_y\frac{F_y}{A}sin^2\theta_y sin^2\varphi_y\right)\underline{e}_3 \otimes \underline{e}_3
\end{aligned}
\tag{A.9}
$$

Appendix B MATLAB CODE LISTING

```
***********************************************************************
*
*   CODING OF DAMAGE AND STRESS EVOLUTION OF A SINUSOIDAL LOAD (3 METHODS)
*
***********************************************************************
% Program to get the Gauss-Legendre Quadrature results (Vectorized)
clear;clc;
tic;
dbstop if error
format long e
[x]= [-0.99555697 -0.976663921 -0.942974571 -0.894991998 -0.833442629 -0.759259263
    -0.673566368...
-0.57766293 -0.473002731 -0.361172306 -0.243866884 -0.122864693 0 0.122864693
    0.243866884 0.361172306...
0.473002731 0.57766293 0.673566368 0.759259263 0.833442629 0.894991998 0.942974571
    0.976663921...
0.99555697];
[weight]=[0.011393799   0.026354987    0.040939157    0.054904696    0.068038334
        0.0801407     0.091028262...
0.100535949    0.108519624    0.114858259    0.119455764    0.122242443
    0.123176054    0.122242443    0.119455764...
0.114858259    0.108519624    0.100535949    0.091028262    0.0801407
    0.068038334    0.054904696    0.040939157...
0.026354987    0.011393799];
% [x]=xlsread('Gauss-Legendre Quadrature','Sheet1','b1:z1');
% [weight]=xlsread('Gauss-Legendre Quadrature','Sheet1','b2:z2');

E=2e11;                   %'Youngs modulus
k=6e8;                      %hardening parameter
b=3;                          %weakening scales distribution exponent
nu=0.3;                     %poisson's ratio
tt=2e8;                      %torsion fatigue limit
ff=2.5e8;                  %bending fatigue limit
ac=(tt-ff/sqrt(3))/(ff/3); %crossland criterial constant
bc=tt;                        %crossland criterial constant
sigu=8e8;                 %ultimite stress
gam=0.5;                  %material parameter from Chaboche law(Wohler curve exponent)
```

```
y=6.38e8;               %macroscopic yield stress
WF=3e6;                 %dissipated energy to failure per unit volume
load=5e8;               %cyclic load
loadtensor= [load 0 0;0 0 0;0 0 0];
stepnumber=300;         %devide one cycle in 200 parts
f=50;                           %frequency of load

%---------------------numerical method----------------------------
alp=0.5;
D=0;                %initial damage
n=1;        %initial recording point
G = (1 - (1 - D).^(gam + 1)).^(1-alp);
%---------------------to get the the first Sb----------------------------
stress11=load*sin(2*pi/stepnumber);
m=1/3*sum(stress11+0+0);
dev1=[stress11 0 0 ;0 0 0 ;0 0 0 ]-m*diag([1,1,1]);
dev11=dev1(1,1); dev12=dev1(1,3); dev13=dev1(1,3);
dev21=dev1(2,1); dev22=dev1(2,2); dev23=dev1(2,3);
dev31=dev1(3,1); dev32=dev1(3,2); dev33=dev1(3,3);
[s]= ([x]/2+1/2).^(1/(1-b)); %1*25

trial11=dev11; trial12=dev12; trial13=dev13;
trial21=dev21; trial22=dev22; trial23=dev23;
trial31=dev31; trial32=dev32; trial33=dev33;

normtrial(1)=norm([trial11, trial12, trial13; trial21, trial22, trial23;trial31,
    trial32, trial33],'fro');
[eta]=bsxfun(@minus,bsxfun(@times,normtrial(1)/y,s),1); %1*25
eta(eta<0)=0;

Sb11=bsxfun(@rdivide,trial11,bsxfun(@plus,[eta],1));Sb12=bsxfun(@rdivide,trial12,
    bsxfun(@plus,[eta],1));Sb13=bsxfun(@rdivide,trial13,bsxfun(@plus,[eta],1));
Sb21=bsxfun(@rdivide,trial21,bsxfun(@plus,[eta],1));Sb22=bsxfun(@rdivide,trial22,
    bsxfun(@plus,[eta],1));Sb23=bsxfun(@rdivide,trial23,bsxfun(@plus,[eta],1));
Sb31=bsxfun(@rdivide,trial31,bsxfun(@plus,[eta],1));Sb32=bsxfun(@rdivide,trial32,
    bsxfun(@plus,[eta],1));Sb33=bsxfun(@rdivide,trial33,bsxfun(@plus,[eta],1));
%1*25 for each Sb element
```

```
Sbtensor=[Sb11; Sb12; Sb13; Sb21; Sb22; Sb23;Sb31; Sb32; Sb33];
normSb=sqrt(sum(Sbtensor.^2));

while G<1
stress11=load*sin((n)*2*pi/stepnumber);
m=1/3*sum(stress11+0+0);
dev1=[stress11 0 0 ;0 0 0 ;0 0 0 ]-m*diag([1,1,1]);
dev11=dev1(1,1); dev12=dev1(1,3); dev13=dev1(1,3);
dev21=dev1(2,1); dev22=dev1(2,2); dev23=dev1(2,3);
dev31=dev1(3,1); dev32=dev1(3,2); dev33=dev1(3,3);

stress11=load*sin((n+1)*2*pi/stepnumber);
m=1/3*sum(stress11+0+0);
devn=[stress11 0 0;0 0 0;0 0 0]-m*diag([1,1,1]);
dev11g=devn(1,1); dev12g=devn(1,3); dev13g=devn(1,3);
dev21g=devn(2,1); dev22g=devn(2,2); dev23g=devn(2,3);
dev31g=devn(3,1); dev32g=devn(3,2); dev33g=devn(3,3);

trial11=bsxfun(@plus,Sb11,(dev11g-dev11)); trial12=bsxfun(@plus,Sb12,(dev12g-dev12));
    trial13=bsxfun(@plus,Sb13,(dev13g-dev13));
trial21=bsxfun(@plus,Sb21,(dev21g-dev21)); trial22=bsxfun(@plus,Sb22,(dev22g-dev22));
    trial23=bsxfun(@plus,Sb23,(dev23g-dev23));
trial31=bsxfun(@plus,Sb31,(dev31g-dev31)); trial32=bsxfun(@plus,Sb32,(dev32g-dev32));
    trial33=bsxfun(@plus,Sb33,(dev33g-dev33));
trialtensor=[trial11; trial12; trial13; trial21; trial22; trial23;trial31; trial32;
    trial33];
normtrial=sqrt(sum(trialtensor.^2));
[eta]=bsxfun(@minus,bsxfun(@times,normtrial/y,s),1); %1*25
eta(eta<0)=0;

Sb11=bsxfun(@rdivide,trial11,bsxfun(@plus,[eta],1));Sb12=bsxfun(@rdivide,trial12,
    bsxfun(@plus,[eta],1));Sb13=bsxfun(@rdivide,trial13,bsxfun(@plus,[eta],1));
Sb21=bsxfun(@rdivide,trial21,bsxfun(@plus,[eta],1));Sb22=bsxfun(@rdivide,trial22,
    bsxfun(@plus,[eta],1));Sb23=bsxfun(@rdivide,trial23,bsxfun(@plus,[eta],1));
Sb31=bsxfun(@rdivide,trial31,bsxfun(@plus,[eta],1));Sb32=bsxfun(@rdivide,trial32,
    bsxfun(@plus,[eta],1));Sb33=bsxfun(@rdivide,trial33,bsxfun(@plus,[eta],1));
%1*25 for each Sb element
```

```
Sbtensor=[Sb11; Sb12; Sb13; Sb21; Sb22; Sb23;Sb31; Sb32; Sb33];
normSb=sqrt(sum((Sbtensor.^2)));

Ws=(bsxfun(@minus,normtrial,bsxfun(@rdivide, y,[s]))<=0).*...
(0)+...
(bsxfun(@minus,normtrial,bsxfun(@rdivide, y,[s]))>0).*...
((E-k)*(1+nu)/(2*E*(E+k*nu))*bsxfun(@times,[weight],bsxfun(@rdivide,bsxfun(@times,bsxfun(@minus,normtrial,bsxfun(@rdivide, y,[s])),y),[s])));

W= sum(Ws);
G = G+W/WF;
D=1-(1-G.^(1/(1-alp))).^(1/(gam + 1));
%   t=n/stepnumber*1/f;
hold on;
yield1=plot (n,y*s(1).^-1, 'LineStyle', 'none','LineWidth', 1, 'Marker', 'o', 'MarkerSize', 6, ...
'MarkerEdgeColor', 'none', 'MarkerFaceColor' , 'c');
Trial1=plot (n,sign(trial11(1))*normtrial(1),'LineStyle', 'none','LineWidth', 1,'Marker', '^', 'MarkerSize', 6, ...
'MarkerEdgeColor','r', 'MarkerFaceColor','r');
Sb1=plot (n,sign(Sb11(1))*normSb(1),'LineStyle', 'none','LineWidth', 1,'Marker', 'v', 'MarkerSize', 6, ...
'MarkerEdgeColor','g', 'MarkerFaceColor','g');
yield8=plot (n,y*s(8).^-1,'LineStyle', 'none','LineWidth', 1,'Marker', 'o', 'MarkerSize', 6, ...
'MarkerEdgeColor', 'none', 'MarkerFaceColor', 'b');
Trial8=plot (n,sign(trial11(8))*normtrial(8),'LineStyle', 'none','LineWidth', 1,'Marker', '^', 'MarkerSize', 6, ...
'MarkerEdgeColor', [1 0.5 0], 'MarkerFaceColor',[1 0.5 0]);
Sb8=plot (n,sign(Sb11(8))*normSb(8),'LineStyle', 'none','LineWidth', 1,'Marker', 'v', 'MarkerSize', 6, ...
'MarkerEdgeColor','k', 'MarkerFaceColor','k');

%   DamageN=plot (t,D,'LineStyle', 'none','LineWidth', 1, 'Marker', 'o', 'MarkerSize', 6, ...
%    'MarkerEdgeColor', 'none', 'MarkerFaceColor' , 'm');

```

```
%---------------------Difference between cyclic load calculation and numerical method
    as function of time----------------------------
% Gcyc = Gcyc+Wcyc/stepnumber/WF
% Dcyc=1-(1-Gcyc.^(1/(1-alp))).^(1/(gam + 1));
% hold on
% Damagecyc=plot (t,D-Dcyc,'LineStyle', 'none','LineWidth', 1, 'Marker', 'o', '
    MarkerSize', 6, ...
%     'MarkerEdgeColor', 'k', 'MarkerFaceColor' , 'k');
n=n+1;
end;
toc;
disp(['Number of points to failure is ' num2str(n) ' points.']);
% %---------------------chaboche method----------------------------
% alp=0.5;
% D=0;              %initial damage
% n=1;       %initial recording point
%   G = (1 - (1 - D).^(gam + 1)).^(1-alp);
%   m=1/3*sum(diag(loadtensor));
%   S1=loadtensor-m*diag([1,1,1]);
%   sqrj1=1/2*sqrt(1/2)*norm(S1,'fro');
%   M=ff^1.233*(1-3*m/sigu);
%   while G<1
%   NF=1/((gam+1)*(1-alp))*(sqrj1/M)^(-gam);
%   G = G+1/stepnumber/NF
%   D=1-(1-G.^(1/(1-alp))).^(1/(gam + 1));
%   t=n/stepnumber*1/f;
%   hold on;
%   DamageC=plot (t,D, 'LineStyle', 'none','LineWidth', 1, 'Marker', 'o', 'MarkerSize
    ', 6, ...
%     'MarkerEdgeColor', 'none', 'MarkerFaceColor' , 'g');
%   n=n+1;
%   end
% %---------------------Cyclic load calculation----------------------------
% Dcyc=0;
% n=1;
% Gcyc = (1 - (1 - Dcyc).^(gam + 1)).^(1-alp);
% Wcyc=4*(E-k)*(1+nu)*(b-1)/(E*(E+k*nu)*b*(b+1))*norm(loadtensor-(1/3*sum(diag(
```

```
    loadtensor)))*diag([1,1,1]),'fro').^(b+1)*y.^(1-b) ;
% while Gcyc< 1
%  Gcyc = Gcyc+Wcyc/stepnumber/WF
%  Dcyc=1-(1-Gcyc.^(1/(1-alp))).^(1/(gam + 1));
%  t=n/stepnumber*1/f;
%  hold on
%  Damagecyc=plot (t,Dcyc,'LineStyle', 'none','LineWidth', 1, 'Marker', 'o', '
    MarkerSize', 6, ...
%    'MarkerEdgeColor', 'none', 'MarkerFaceColor' , 'b');
%  n=n+1;
% end

%---------------------plot settings-----------------------------
grid on;
grid minor;
% axis([0 0.49 -0.04 0.04]);
set(gca ,'FontSize',30);
hXLabel = xlabel('Number of steps' ,'Fontsize' ,30);

hTitle =title('Microscopic stress evolution at 2 different scales ' ,'Fontsize' ,30);
hYLabel =ylabel('Stress(Pa)', 'Fontsize' ,30);
hLegend=legend([yield1,Sb1,Trial1,yield8,Sb8,Trial8],'(\sigma_y-\lambda\Sigma_H)/s_1
       at scale s_1','(S-b)               at scale s_1',...
'(S-b)_{trial}          at scale s_1', '(\sigma_y-\lambda\Sigma_H)/s_8     at scale s_
    {8}','(S-b)                at scale s_{8}','(S-b)_{trial}          at scale s_{8}')
    ;


% hTitle = title('Damage evolution comparison of three methods' ,'Fontsize' ,30);
% hYLabel = ylabel('Damage', 'Fontsize' ,30);
%  hLegend=legend([DamageN,DamageC,Damagecyc],'Numerical method','Chaboche method',...
%    'Cyclic load calculation');

% hTitle = title('Difference between cyclic load calculation and numerical method as
    function of time(time step=1/15000)' ,'Fontsize' ,30);

% Adjust font
```

```
set(gca, 'FontName', 'Helvetica')
set([hTitle, hXLabel, hYLabel], 'FontName', 'AvantGarde')

set([hXLabel, hYLabel], 'FontSize', 30)
set(hTitle, 'FontSize', 30, 'FontWeight' , 'bold')
set([hLegend, gca], 'FontSize', 30)
% Adjust axes properties
set(gca, 'Box', 'off', 'TickDir', 'out', 'TickLength', [.02 .02], ...
'XMinorTick', 'on', 'YMinorTick', 'on', 'YGrid', 'on', ...
'XColor', [.3 .3 .3], 'YColor', [.3 .3 .3], ...
'LineWidth', 1)

set(gcf,'color','w'); %set figure background transparent
set(gca,'color','w'); %set axis transparent
% Maximize print figure
set(gcf,'outerposition',get(0,'screensize'));
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'points'); %[ {inches} | centimeters | normalized | points ]
set(gcf, 'PaperPosition', [0 0 1920 1080]); %set(gcf,'PaperPosition',[left,bottom,
    width,height])
saveas(gcf,'trialsin.png');
% saveas(gcf,'damagesin.png');
```

```
**********************************************************************
*
*   CODING OF DAMAGE AND ENERGY EVOLUTION OF PSA LOAD (3 DIMENSIONAL)
*
**********************************************************************
clear;clc;
dbstop if error
format long e

load('FX_RAVG.mat');
signal.data=double(signal.data);
forcex= transpose(signal.data);
load('FY_RAVG.mat');
signal.data=double(signal.data);
forcey= transpose(signal.data);
load('FZ_RAVG.mat');
signal.data=double(signal.data);
forcez= transpose(signal.data);
copy=3;
forcex=repmat(forcex,copy,1);
forcey=repmat(forcey,copy,1);
forcez=repmat(forcez,copy,1);

%--------------------Arithmetic sequence between every recorded points
    --------------------
ari=10;
for i=2:(1*802805)
%force(1+ari*(i-1):1+ari*i)=linspace(forceorigin(i),forceorigin(i+1),ari+1);
forcelx(1+ari*(i-2):1+ari*(i-1))=linspace(forcex(i-1),forcex(i),ari+1);
forcely(1+ari*(i-2):1+ari*(i-1))=linspace(forcey(i-1),forcey(i),ari+1);
forcelz(1+ari*(i-2):1+ari*(i-1))=linspace(forcez(i-1),forcez(i),ari+1);
end;
%  ari*(i-1)+1 ;%the number of points

%----------------------build the stress tensor--------------------
A=1/6e4;
cx=10;
```

```
cy=60;
thetax=0.5;
thetay=0.6;
phix=0.3;
phiy=0.4;
stress11=1/A*(forcelz+cx*forcelx*cos(thetax)^2+cy*forcely*cos(thetay)^2);
stress12=1/A*(cx*forcelx*cos(thetax)*sin(thetax)*cos(phix)+cy*forcely*cos(thetay)*sin(
    thetay)*cos(phiy));
stress13=1/A*(cx*forcelx*cos(thetax)*sin(thetax)*sin(phix)+cy*forcely*cos(thetay)*sin(
    thetay)*sin(phiy));
stress21=stress12;
stress22=1/A*(cx*forcelx*sin(thetax)^2*cos(phix)^2+cy*forcely*sin(thetay)^2*cos(phiy)
    ^2);
stress23=1/A*(cx*forcelx*sin(thetax)^2*cos(phix)*sin(phix)+cy*forcely*sin(thetay)^2*
    cos(phiy)*sin(phiy));
stress31=stress13;
stress32=stress23;
stress33=1/A*(cx*forcelx*sin(thetax)^2*sin(phix)^2+cy*forcely*sin(thetay)^2*sin(phiy)
    ^2);
% [max(stress11) max(stress12) max(stress13);
% max(stress12) max(stress22) max(stress23);
% max(stress23) max(stress13) max(stress33);]
% [mean(stress11) mean(stress12) mean(stress13);
% mean(stress12) mean(stress22) mean(stress23);
% mean(stress23) mean(stress13) mean(stress33);]

[x]= [-0.99555697 -0.976663921 -0.942974571 -0.894991998 -0.833442629 -0.759259263
    -0.673566368...
-0.57766293 -0.473002731 -0.361172306 -0.243866884 -0.122864693 0 0.122864693
    0.243866884 0.361172306...
0.473002731 0.57766293 0.673566368 0.759259263 0.833442629 0.894991998 0.942974571
    0.976663921...
0.99555697];
[weight]=[0.011393799    0.026354987    0.040939157    0.054904696    0.068038334
        0.0801407    0.091028262...
0.100535949    0.108519624    0.114858259    0.119455764    0.122242443
    0.123176054    0.122242443    0.119455764...
```

```
0.114858259      0.108519624      0.100535949      0.091028262      0.0801407
    0.068038334      0.054904696      0.040939157...
0.026354987      0.011393799];
% [x]=xlsread('Gauss-Legendre Quadrature','Sheet1','b1:z1');
% [weight]=xlsread('Gauss-Legendre Quadrature','Sheet1','b2:z2');
y=6.38e8;              %macroscopic yield stress
lam=0.5;               %hydrostatic pressure sensitivity
E=2e11;                %'Youngs modulus
k=6e8;                 %hardening parameter
b=3;                   %weakening scales distribution exponent
nu=0.3;                    %poisson's ratio
tt=2e8;                %torsion fatigue limit
ff=2.5e8;              %bending fatigue limit
ac=(tt-ff/sqrt(3))/(ff/3); %crossland criterial constant
bc=tt;                     %crossland criterial constant
sigu=8e8;              %ultimite stress
gam=0.5;               %material parameter from Chaboche law(Wohler curve exponent)
samplerate=256;   %recorded samples per second

%---------------------Vecterization----------------------------
tic;
WF=3e7;                %dissipated energy to failure per unit volume
alp=0.8;
D=0;               %initial damage
n=1;                       %initial recording point
step=1/samplerate/ari;
t=n*step;
G = (1 - (1 - D).^(gam + 1)).^(1-alp);
%---------------------to get the the first Sb------------------------------
m=1/3*sum(stress11(1)+stress22(1)+stress33(1));
yield(1)=y-lam*m; %macro yield strength considering mean stress effect
dev1=[stress11(1) stress12(1) stress13(1);stress21(1) stress22(1) stress23(1);stress31
    (1) stress32(1) stress33(1)]-m*diag([1,1,1]);
dev11=dev1(1,1); dev12=dev1(1,2); dev13=dev1(1,3);
dev21=dev1(2,1); dev22=dev1(2,2); dev23=dev1(2,3);
dev31=dev1(3,1); dev32=dev1(3,2); dev33=dev1(3,3);

```

```
trial11=dev11; trial12=dev12; trial13=dev13;
trial21=dev21; trial22=dev22; trial23=dev23;
trial31=dev31; trial32=dev32; trial33=dev33;
trialtensor=[trial11; trial12; trial13; trial21; trial22; trial23;trial31; trial32; trial33];
normtrial(1,1:length(x))=sqrt(sum(trialtensor.^2));
[s]= ([x]/2+1/2).^(1/(1-b)); %1*25
[eta]=bsxfun(@minus,bsxfun(@times,normtrial(1,1:length(x))/yield(1),s),1); %1*25
eta(eta<0)=0;

Sb11=bsxfun(@rdivide,trial11,bsxfun(@plus,[eta],1));Sb12=bsxfun(@rdivide,trial12,bsxfun(@plus,[eta],1));Sb13=bsxfun(@rdivide,trial13,bsxfun(@plus,[eta],1));
Sb21=bsxfun(@rdivide,trial21,bsxfun(@plus,[eta],1));Sb22=bsxfun(@rdivide,trial22,bsxfun(@plus,[eta],1));Sb23=bsxfun(@rdivide,trial23,bsxfun(@plus,[eta],1));
Sb31=bsxfun(@rdivide,trial31,bsxfun(@plus,[eta],1));Sb32=bsxfun(@rdivide,trial32,bsxfun(@plus,[eta],1));Sb33=bsxfun(@rdivide,trial33,bsxfun(@plus,[eta],1));
%1*25 for each Sb element
Sbtensor=[Sb11; Sb12; Sb13; Sb21; Sb22; Sb23;Sb31; Sb32; Sb33];
normSb(1,:)=sqrt(sum(Sbtensor.^2));
Ws=(bsxfun(@minus,normtrial(1,1:length(x)),bsxfun(@rdivide, yield(1),[s]))<=0).*...
(0)+...
(bsxfun(@minus,normtrial(1,1:length(x)),bsxfun(@rdivide, yield(1),[s]))>0).*...
((E-k)*(1+nu)/(2*E*(E+k*nu))*bsxfun(@times,[weight],bsxfun(@rdivide,bsxfun(@times,bsxfun(@minus,normtrial(1,1:length(x)),bsxfun(@rdivide, yield(1),[s])),yield(1)),[s])));
W= sum(Ws);
G = G+W/WF; %1.322163316411401e-03
D(1)=1-(1-G.^(1/(1-alp))).^(1/(gam + 1));
while G<1
m=1/3*sum(stress11(n)+stress22(n)+stress33(n));
dev1=[stress11(n) stress12(n) stress13(n);stress21(n) stress22(n) stress23(n);stress31(n) stress32(n) stress33(n)]-m*diag([1,1,1]);
dev11=dev1(1,1); dev12=dev1(1,2); dev13=dev1(1,3);
dev21=dev1(2,1); dev22=dev1(2,2); dev23=dev1(2,3);
dev31=dev1(3,1); dev32=dev1(3,2); dev33=dev1(3,3);

m=1/3*sum(stress11(n+1)+stress22(n+1)+stress33(n+1));
```

```
yield(n+1)=y-lam*m; %macro yield strength considering mean stress effect
yield(yield<0)=0;
devn=[stress11(n+1) stress12(n+1) stress13(n+1);stress21(n+1) stress22(n+1) stress23(n+1);stress31(n+1) stress32(n+1) stress33(n+1)]-m*diag([1,1,1]);
dev11g=devn(1,1); dev12g=devn(1,2); dev13g=devn(1,3);
dev21g=devn(2,1); dev22g=devn(2,2); dev23g=devn(2,3);
dev31g=devn(3,1); dev32g=devn(3,2); dev33g=devn(3,3);

trial11=bsxfun(@plus,Sb11,(dev11g-dev11)); trial12=bsxfun(@plus,Sb12,(dev12g-dev12)); trial13=bsxfun(@plus,Sb13,(dev13g-dev13));
trial21=bsxfun(@plus,Sb21,(dev21g-dev21)); trial22=bsxfun(@plus,Sb22,(dev22g-dev22)); trial23=bsxfun(@plus,Sb23,(dev23g-dev23));
trial31=bsxfun(@plus,Sb31,(dev31g-dev31)); trial32=bsxfun(@plus,Sb32,(dev32g-dev32)); trial33=bsxfun(@plus,Sb33,(dev33g-dev33));
trialtensor=[trial11; trial12; trial13; trial21; trial22; trial23;trial31; trial32; trial33];
normtrial(n+1,:)=sqrt(sum(trialtensor.^2));
[eta]=bsxfun(@minus,bsxfun(@times,normtrial(n+1,:)/yield(n+1),s),1); %1*25
eta(eta<0)=0;

Sb11=bsxfun(@rdivide,trial11,bsxfun(@plus,[eta],1));Sb12=bsxfun(@rdivide,trial12,bsxfun(@plus,[eta],1));Sb13=bsxfun(@rdivide,trial13,bsxfun(@plus,[eta],1));
Sb21=bsxfun(@rdivide,trial21,bsxfun(@plus,[eta],1));Sb22=bsxfun(@rdivide,trial22,bsxfun(@plus,[eta],1));Sb23=bsxfun(@rdivide,trial23,bsxfun(@plus,[eta],1));
Sb31=bsxfun(@rdivide,trial31,bsxfun(@plus,[eta],1));Sb32=bsxfun(@rdivide,trial32,bsxfun(@plus,[eta],1));Sb33=bsxfun(@rdivide,trial33,bsxfun(@plus,[eta],1));
%1*25 for each Sb element
Sbtensor=[Sb11; Sb12; Sb13; Sb21; Sb22; Sb23;Sb31; Sb32; Sb33];
normSb(n+1,:)=sqrt(sum((Sbtensor.^2)));

Ws=(bsxfun(@minus,normtrial(n+1,:),bsxfun(@rdivide, yield(n+1),[s]))<=0).*...
(0)+...
(bsxfun(@minus,normtrial(n+1,:),bsxfun(@rdivide, yield(n+1),[s]))>0).*...
((E-k)*(1+nu)/(2*E*(E+k*nu))*bsxfun(@times,[weight],bsxfun(@rdivide,bsxfun(@times,bsxfun(@minus,normtrial(n+1,:),bsxfun(@rdivide, yield(n+1),[s])),yield(n+1)),[s]))
);
W= sum(Ws);
```

```
G = G+W/WF;
D(n+1)=1-(1-G.^(1/(1-alp))).^(1/(gam + 1));
t=n*step;
%       hold on;
%      yield1=plot (t,yield(n)*s(1).^-1, 'LineStyle', 'none','LineWidth', 1, 'Marker', 'o', 'MarkerSize', 6, ...
%      'MarkerEdgeColor',  'none', 'MarkerFaceColor' , 'c');
%      Trial1=plot (t,normtrial(n,1),'LineStyle', 'none','LineWidth', 1,'Marker', '^', 'MarkerSize', 6, ...
%      'MarkerEdgeColor','r', 'MarkerFaceColor','r');
%      Sb1=plot (t,normSb(n,1),'LineStyle', 'none','LineWidth', 1,'Marker', 'v', 'MarkerSize', 6, ...
%      'MarkerEdgeColor','g', 'MarkerFaceColor','g');
%      yield8=plot (t,yield(n)*s(8).^-1,'LineStyle', 'none','LineWidth', 1,'Marker', 'o', 'MarkerSize', 6, ...
%      'MarkerEdgeColor', 'none', 'MarkerFaceColor','b');
%      Trial8=plot (t,normtrial(n,8),'LineStyle', 'none','LineWidth', 1,'Marker', '^', 'MarkerSize', 6, ...
%      'MarkerEdgeColor', [1 0.5 0], 'MarkerFaceColor',[1 0.5 0]);
%      Sb8=plot (t,normSb(n,8),'LineStyle', 'none','LineWidth', 1,'Marker', 'v', 'MarkerSize', 6, ...
%      'MarkerEdgeColor','k', 'MarkerFaceColor','k');
% DamageN=plot (t,D,'LineStyle', 'none','LineWidth', 1, 'Marker', 'o', 'MarkerSize', 6, ...
%     'MarkerEdgeColor',  'none', 'MarkerFaceColor' , 'r');
n=n+1;
end;
toc;
time=num2str(toc)
NF=num2str(n)
testtime=num2str(t)
%
%  hold on;
%    yield1=plot ((1:n)*step,yield(1:n)*s(1).^-1, 'LineStyle', 'none','LineWidth', 1, 'Marker', 'o', 'MarkerSize', 6, ...
%      'MarkerEdgeColor',  'none', 'MarkerFaceColor' , 'c');
%    Trial1=plot ((1:n)*step,normtrial(1:n,1),'LineStyle', 'none','LineWidth', 1,'
```

```
    Marker', '^', 'MarkerSize', 6, ...
%      'MarkerEdgeColor','r', 'MarkerFaceColor','r');
%      Sb1=plot ((1:n)*step,normSb(1:n,1),'LineStyle', 'none','LineWidth', 1,'Marker', 'v', 'MarkerSize', 6, ...
%      'MarkerEdgeColor','g', 'MarkerFaceColor','g');
%    yield8=plot ((1:n)*step,yield(1:n)*s(8).^-1,'LineStyle', 'none','LineWidth', 1,'Marker', 'o', 'MarkerSize', 6, ...
%      'MarkerEdgeColor', 'none', 'MarkerFaceColor','b');
%    Trial8=plot ((1:n)*step,normtrial(1:n,8),'LineStyle', 'none','LineWidth', 1,'Marker', '^', 'MarkerSize', 6, ...
%      'MarkerEdgeColor', [1 0.5 0], 'MarkerFaceColor',[1 0.5 0]);
%     Sb8=plot ((1:n)*step,normSb(1:n,8),'LineStyle', 'none','LineWidth', 1,'Marker', 'v', 'MarkerSize', 6, ...
%      'MarkerEdgeColor','k', 'MarkerFaceColor','k');
% %
% DamageN=plot ((1:n)*step,D(1:n),'LineStyle', 'none','LineWidth', 1, 'Marker', 'o', 'MarkerSize', 6, ...
%     'MarkerEdgeColor', 'none', 'MarkerFaceColor' , 'r');

%--------------------plot settings---------------------------
grid on;
grid minor;
set(gca ,'FontSize',30);
hXLabel = xlabel('t(s)' ,'Fontsize' ,30);

hTitle =title('Damage evolution under multidimensional stress' ,'Fontsize' ,25);
hYLabel =ylabel('D', 'Fontsize' ,25);

% hTitle = title('Microscopic stress evolution at 2 scales' ,'Fontsize' ,30);
% hYLabel = ylabel('(S-b)(Pa)', 'Fontsize' ,30);
% hLegend=legend([yield1,Sb1,Trial1,yield8,Sb8,Trial8],'(\sigma_y-\lambda\Sigma_H)/s_1 at scale s_1','(S-b)                  at scale s_1',...
%     '(S-b)_{trial}              at scale s_1', '(\sigma_y-\lambda\Sigma_H)/s_8       at scale s_{8}','(S-b)                  at scale s_{8}','(S-b)_{trial}            at scale s_{8}');
% set([hLegend, gca], 'FontSize', 30)

```

```
% Adjust font
set(gca, 'FontName', 'Helvetica')
set([hTitle, hXLabel, hYLabel], 'FontName', 'AvantGarde')
set([hXLabel, hYLabel], 'FontSize', 30)
set(hTitle, 'FontSize', 30, 'FontWeight' , 'bold')

% Adjust axes properties
set(gca, 'Box', 'off', 'TickDir', 'out', 'TickLength', [.02 .02], ...
'XMinorTick', 'on', 'YMinorTick', 'on', 'YGrid', 'on', ...
'XColor', [.3 .3 .3], 'YColor', [.3 .3 .3], ...
'LineWidth', 1)

set(gcf,'color','w'); %set figure background transparent
set(gca,'color','w'); %set axis transparent
% Maximize print figure
set(gcf,'outerposition',get(0,'screensize'));
set(gcf, 'PaperPositionMode', 'manual');
set(gcf, 'PaperUnits', 'points'); %[ {inches} | centimeters | normalized | points ]
set(gcf, 'PaperPosition', [0 0 1920 1080]); %set(gcf,'PaperPosition',[left,bottom,
    width,height])
%  saveas(gcf,'damage3d.png');
saveas(gcf,'trialreal3d.png');
```