Comportements élastoviscoplastiques mono et polycristallins

Résumé :

Le but de ce document est de décrire l'intégration des comportements mono et polycristallins.

On traite ici de l'intégration de ces lois de comportement associées à des systèmes de glissement correspondant aux familles cristallines habituelles ou spécifiées par l'utilisateur. Cette intégration peut se faire de façon explicite (méthode de Runge-Kutta avec contrôle de la précision et redécoupage local du pas de temps) ou de façon implicite (méthode de Newton avec redécoupage local du pas de temps).

Ces comportements peuvent être employés pour le calcul de microstructures (maillage d'un agrégat, avec représentation géométrique de chaque grain physique monocristallin) ou pour le calcul de type POLYCRISTAL, milieu « homogénéisé » possédant en tout point matériel (ou point d'intégration ou de calcul) plusieurs phases simultanées, dans des proportions variables.

On décrit aussi les types de calculs possibles avec ces comportements ainsi que la procédure de génération de maillages.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Table des Matières

1 Introduction....4
2 Formulation des comportements mono et polycristallins....4
2.1 Principe des lois de comportement du monocristal....4
2.2 Lois de comportement du monocristal disponibles....5
2.2.1 Lois phénoménologiques....5
2.2.1.1 Loi d'écoulement MONO_VISC1....5
2.2.1.2 Loi d'écoulement MONO_VISC2....5
2.2.1.3 Loi d'écrouissage cinématique MONO_CINE1....5
2.2.1.4 Loi d'écrouissage cinématique MONO_CINE2....6
2.2.1.5 Loi d'écrouissage isotrope MONO_ISOT1....6
2.2.1.6 Loi d'écrouissage isotrope MONO_ISOT2....6
2.2.1.7 Matrice d'interaction....6
2.2.2 Loi issue de la Dynamique des Dislocations : MONO_DD_KR....6
2.2.3 Lois issue de la Dynamique des Dislocations : MONO_DD_CFC / MONO_DD_CFC_IRRA /MONO_DD_FAT....7
2.2.4 Lois issues de la Dynamique des Dislocations : MONO_DD_CC / MONO_DD_CC_IRRA 11
Les lois d'évolution des densités de dislocations (et) sont :....12
2.3 Systèmes de glissement et comportement global du monocristal....12
2.3.1 Petites déformations, configuration initiale....13
2.3.2 Petites déformations, rotation du réseau cristallin....13
2.3.3 Grandes déformations....14
2.4 Comportement du polycristal homogénéisé....15
3 Intégration locale et mise en œuvre numérique....16
3.1 Système d'équations à résoudre....16
3.1.1 Comportement de type MONOCRISTAL....16
3.1.2 Comportement de type POLYCRISTAL....17
3.2 Résolution implicite du comportement MONOCRISTAL....18
3.2.1 Résolution implicite des lois phénoménologiques....18
3.2.2 Résolution implicite – Intégration du modèle MONO_DD_KR....21
3.2.3 Résolution implicite – modèles MONO_DD_CFC(IRRA) / MONO_DD_FAT....24
3.2.4 Résolution implicite – modèles MONO_DD_CC / MONO_DD_CC_IRRA....28
3.2.5 Algorithme d'intégration implicite en grandes déformations....32
3.2.6 Critères de convergence utilisés pour les résolutions implicites....34
3.3 Résolution explicite....34
4 Variables internes....35
4.1 Cas du monocristal....35
4.2 Cas du polycristal....36
5 Implantation numérique dans Code_Aster....37

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

6 Utilisation....38
6.1 Cas du monocristal....38
6.2 Cas du polycristal....39
6.3 Exemple....40
7 Bibliographie....43
8 Historique des versions du document....44
Annexe 1 Expression du Jacobien des équations élasto-visco-plastiques intégrées....45
Annexe 2 Évaluation de l'opérateur tangent cohérent....48
Annexe 3 Matrices tangentes en grandes déformations :....49
Annexe 4 Écriture générale des matrices jacobiennes....52
Annexe 5 Exemples de matrices d'interaction....54
Annexe 6 Procédure de création de maillage....57

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

1 Introduction

L'objectif général du développement des fonctionnalités « micro-macro » dans *Code_Aster* est de pouvoir intégrer de façon modulaire des modèles à plusieurs échelles (avec une possibilité de choix des lois de comportements, des règles de localisation, des types de microstructures). Ce qui peut mener à des types de calcul différents (calculs polycristallins, utilisation d'une loi de type *Berveiller-Zaoui* ou d'une loi type *« règle en* β *»*, calculs d'agrégats multi-cristallins avec maillage d'une microstructure, …).

La démarche présentée ici consiste à permettre le découplage, par modularité, des différents éléments qui constituent une loi de comportement. Cette souplesse est accessible directement à l'utilisateur. De plus, pour le développeur, il est possible d'ajouter facilement une loi de comportement cristalline en définissant simplement les dérivées partielles du problème, en termes de variables internes. Ceci est suffisant si on se contente d'une intégration explicite ; pour une intégration implicite, il faut définir en plus l'opérateur tangent.

Plus précisément, pour l'aspect comportement MONOCRISTAL, en chaque point d'intégration d'un élément fini donné, le comportement est celui d'un monocristal possédant une orientation donnée, et un certain nombre de systèmes de glissement. Chaque famille de systèmes de glissement possède sa propre loi de comportement locale.

Dans le cas d'un modèle polycristallin, on suppose qu'en un point matériel (point d'intégration d'un élément fini), plusieurs phases métallurgiques sont présentes simultanément, chaque phase pouvant être constituée de grains avec des orientations données, chaque grain possédant un certain nombre de systèmes de glissement (pas forcément les mêmes pour chaque phase). La représentation du matériau est fondée sur une approche auto-cohérente simplifiée 5 ou éventuellement étendue pour des chargements non radiaux ou cycliques 4. Chaque famille de systèmes de glissement possède sa propre loi de comportement locale. On trouve ainsi séparément la structure cristallographique, la loi de viscoplasticité cristalline et les règles de transition d'échelles. Ce mode de séparation est aussi étendu à la loi de viscoplasticité proprement dite, avec une séparation entre l'élasticité, le critère et la loi d'écoulement.

2 Formulation des comportements mono et polycristallins

2.1 Principe des lois de comportement du monocristal

Les comportements relatifs aux système de glissement d'un monocristal sont (dans l'ensemble des comportements envisagés) de type élasto-visco-plastique. Il peuvent être construits sur des bases phénoménologiques ou sur des bases physiques (dynamique des dislocations). Pour chacune des directions de glissement, le comportement est *mono dimensionnel*. Il peut se décomposer en trois types d'équations :

• Relation d'écoulement : $\dot{\gamma}_s$ représente le glissement plastique du système s

$\dot{\gamma}_s = g(\tau_s, \alpha_s, \gamma_s, p_s)$, avec $\dot{p}_s = |\dot{\gamma}_s|$ et :

- cas élastoplastique, un critère du type : $F(\tau_s, \alpha_s, \gamma_s, p_s) \leq 0$ et $F \cdot \dot{p}_s = 0$
- cas élasto-viscoplastique, $\dot{p}_s = f(\tau_s, \alpha_s, \gamma_s, p_s)$

• $\dot{\alpha}_s$ représente l'écrouissage cinématique dans le cas d'une loi phénoménologique et la densité de dislocations pour un comportement à base physique.

Son évolution est la forme : $\dot{\alpha}_s = h(\tau_s, \alpha_s, \gamma_s, p_s)$

• pour les lois phénoménologiques, l'écrouissage isotrope est défini par une fonction : $R(p_s)$.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Ces relations deviennent, après discrétisation en temps:

•Relation d'écoulement :

$\Delta\gamma_s = g(\tau_s, \alpha_s, \gamma_s, p_s)$, avec, $\Delta p_s = |\Delta\gamma_s|$

pour un comportement élastoplastique un critère du type : $F(\tau_s, \alpha_s, \gamma_s, p_s) \leq 0$ et $F \cdot \Delta p_s = 0$

et pour un comportement élasto-viscoplastique, $\Delta p_s = f(\tau_s, \alpha_s, \gamma_s, p_s)\Delta t$

•Évolution de $\dot{\alpha}_s$: $\Delta\alpha_s = h(\tau_s, \alpha_s, \gamma_s, p_s)$

•Évolution de l'écrouissage isotrope : $R(p_s)$

Les quantités $(\tau_s, \alpha_s, \gamma_s, p_s)$ sont évaluées à l'instant courant pour une discrétisation implicite et à l'instant précédent pour une discrétisation explicite.

Les lois monocristallines disponibles sont décrites ci-après. Les noms de ces relations correspondent à leur appellation dans la commande DEFI_MATERIAU [U4.43.01].

2.2 Lois de comportement du monocristal disponibles

2.2.1 Lois phénoménologiques

Elles sont construites à partir d'une loi d'écoulement, une loi d'écrouissage isotrope et une loi d'écrouissage cinématique. La combinaison des lois **MONO_VISC1, MONO_ISOT1, MONO_CINE1** correspond à la loi cristalline connue sous le nom de Meric-Cailletaud 1 .

2.2.1.1 Loi d'écoulement MONO_VISC1

$$\Delta\gamma_s = g(\tau_s, \alpha_s, \gamma_s, p_s) = \Delta p_s \frac{\tau_s - c\alpha_s}{|\tau_s - c\alpha_s|}$$

$$\Delta p_s = \Delta t \left\langle \frac{|\tau_s - c\alpha_s| - R_s(p_s)}{k} \right\rangle^n$$

Les paramètres sont : (c, k, n) .

2.2.1.2 Loi d'écoulement MONO_VISC2

$$\Delta\gamma_s = g(\tau_s, \alpha_s, \gamma_s, p_s) = \Delta p_s \frac{\tau_s - c\alpha_s - a\gamma_s}{|\tau_s - c\alpha_s - a\gamma_s|}$$

$$\Delta p_s = \Delta t \left\langle \frac{|\tau_s - c\alpha_s - a\gamma_s| - R_s(p_s) + \frac{d}{2c}(c\alpha_s)^2}{k} \right\rangle^n$$

Les paramètres sont : c, k, n, a, d .

2.2.1.3 Loi d'écrouissage cinématique MONO_CINE1

$$\Delta\alpha_s = h(\tau_s, \alpha_s, \gamma_s, p_s) = \Delta\gamma_s - d\alpha_s\Delta p_s$$

Le paramètre est : d .

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

2.2.1.4 Loi d'écrouissage cinématique MONO_CINE2

$$\Delta\alpha_s = h(\tau_s, \alpha_s, \gamma_s, p_s) = \Delta\gamma_s - d\,\alpha_s \Delta p_s - \left(\frac{|c\,\alpha_s|}{M}\right)^m \frac{\alpha_s}{|\alpha_s|}$$

Les paramètres étant alors : d, M, m, c.

2.2.1.5 Loi d'écrouissage isotrope MONO_ISOT1

$R_s(p) = R_0 + Q\left(\sum_{r=1}^{ns} h_{sr}\left(1 - e^{bp_r}\right)\right)$; les paramètres sont : R_0, Q, b, h_{sr}.

2.2.1.6 Loi d'écrouissage isotrope MONO_ISOT2

$R_s(p) = R_0 + Q_1 \sum_r h_{sr}\left(1 - e^{-b_1 p_r}\right) + Q_2\left(1 - e^{-b_2 p_s}\right)$; les paramètres sont : $R_0, Q_1, b_1, b_2, h_{sr}, Q_2$.

2.2.1.7 Matrice d'interaction

h_{sr} une matrice carrée symétrique d'interaction entre systèmes de glissement (voir annexe 4).
Dans le cas où un seul un coefficient H est fourni, la matrice h_{sr} prend une forme très simple :

$h_{rs} = 1\ si\ r = s$ et $h_{rs} = H\ si\ r \neq s$, avec par défaut $H = 0$ (matrice identité).

2.2.2 Loi issue de la Dynamique des Dislocations : MONO_DD_KR

Le modèle MONO_DD_KR (Kocks-Rauch) est applicable aux aciers de type CC (structure cristalline Cubique Centré). Sa variable interne principale est la densité de dislocations sur chaque système de glissement. Il offre par ailleurs l'avantage d'être valide pour une large gamme de températures [bib7]. L'écoulement se formule de la façon suivante :

Si $|\tau^s| > \tau_0$

$$\tau^s_{eff} = \left\langle |\tau^s| - \tau_0 - \tau^s_\mu \right\rangle \quad \langle\ \rangle \text{ partie positive}$$
$$\tau^s_\mu = \frac{(\mu)^2 \sum_u a^{su} \alpha^u}{|\tau^s| - \tau_0} \text{ avec } \alpha^u = \rho^u b^2$$

sinon $\tau^s_{eff} = 0$. Dans ces expressions, μ désigne le module de cisaillement du matériau, b est la constante de Burgers, ρ^u est la densité de dislocations, a^{su} est la matrice d'interaction entre les 24 systèmes de glissement (sa forme particulière, dépendant de 5 coefficients, est fournie en annexe). Les termes diagonaux représentent l'auto-écrouissage d'un système de glissement, les autres termes représentent l'écrouissage latent.

Si $|\tau^s| > \tau_0$ et $\tau^s_{eff} > 0$

$$\Delta\gamma_s = g(\tau_s, \gamma_s) = \dot{\gamma}_0 \exp\left(\frac{-\Delta G(\tau^s_{eff})}{k_B T}\right) \cdot \frac{\tau_s}{|\tau_s|} \cdot \Delta t$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

avec $\Delta G(\tau_{eff}^s)=\Delta G_0\left(1-\left(\frac{\langle \tau_{eff}^s \rangle}{\tau_R}\right)^p\right)^q$

sinon

$$\Delta \gamma_s=0$$
$$\Delta \alpha_s=0$$

$\Delta G(\tau_{eff}^s)$ est une activation d'enthalpie correspondant à une activation libre dans un domaine athermique. $\dot{\gamma}_0, \tau_R, p, q$ sont des données matériaux, k_B la constante de Boltzman et T la température.

L'évolution de la densité de dislocation est proportionnelle à la racine carrée de la somme des densités de dislocations de tous les autres systèmes de glissement :

$\dot{\rho}_s=\frac{|\dot{\gamma}^s|}{b}\left(\frac{1}{\Lambda^s}-g_c(T)\rho^s\right)$ avec $\frac{1}{\Lambda^s}=\frac{1}{d}+\frac{\sqrt{\sum_{u\neq s}\rho^u}}{K(T)}$

et $g_c(T)=g_{c0}\exp\left[-\frac{E_{gc}}{k_B T}\right]$ où $E_{gc}, k_B, b, d, g_{c0}, K(T)$ sont des paramètres matériaux,

d'où

$\Delta \rho^s=|\Delta \gamma^s|\left(\frac{1}{bd}+\frac{\sqrt{\sum_{u\neq s}\alpha^u}}{b^2 K}-\frac{g_c \alpha^s}{b^3}\right)$ et par conséquent :

$$\Delta \alpha^s=|\Delta \gamma^s|\left(\frac{b}{d}+\frac{\sqrt{\sum_{u\neq s}\alpha^u}}{K}-\frac{g_c \alpha^s}{b}\right)=|g(\tau_s,\alpha_s)|\cdot h(\alpha_s)$$

Les paramètres sont : $\dot{\gamma}_0, k_B, T, \tau_R, \Delta G_0, p, q, \tau_0, \mu, b, d, K, g_c$
k_B est la constante de Boltzman.

Remarque : La résolution de Newton peut amener à des valeurs négatives pour $\Delta \alpha^u$, il faut donc que les termes sous la racine soient protégés. Pour ce faire, nous choisissons la modification suivante : $\sqrt{\sum_{u\neq s}\langle \alpha^u \rangle}$ où $\langle . \rangle$ représente la parti positive : $\langle x \rangle=\begin{cases} x \text{ si } x>0 \\ 0 \text{ sinon} \end{cases}$

Remarquons aussi que l'expression de $\Delta \gamma_s$, qui est fonction de $\Delta G\left(\tau_{eff}^s\right)$, conduit à une discontinuité au voisinage de $\tau_{eff}^s=0$: pour $\tau_{eff}^s<0$, $\Delta \gamma_s=0$, et $\lim_{\tau_{eff}^s \to 0^*} \Delta \gamma_s=\dot{\gamma}_0\exp\left(\frac{-\Delta G_0}{k_B T}\right)\cdot\frac{\tau_s}{|\tau_s|}\cdot\Delta t$

2.2.3 Lois issue de la Dynamique des Dislocations : MONO_DD_CFC / MONO_DD_CFC_IRRA /MONO_DD_FAT

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

La formulation des lois DD_CFC [8], [9] et DD_FAT [18] est construite à partir de calculs de dynamique des dislocations. Elle s'applique aux matériaux à structure cristalline Cubique à Faces Centrées (CFC) tels que les aciers austénitiques. A priori le modèle DD_CFC n'est pas compatible avec un changement de trajet de chargement, en particulier les paramètres ne sont pas adaptés à une sollicitation cyclique (lorsqu'une dislocation "revient sur ses pas" sa cinétique est différente du fait d'une interaction différente avec les obstacles).

Les équations ci-dessous sont écrites pour chaque système de glissement s .

La direction de l'écoulement est donnée par :

$\dot{\gamma}_s = \dot{p}_s \frac{\tau_s}{|\tau_s|} = \dot{p}_s \text{signe}(\tau_s)$ et l'intensité de l'écoulement par :

$$\dot{p}_s = \dot{\gamma}_0 \left(\left(\frac{|\tau_s|}{\tau_f + \tau_s^{forest}} \right)^n - 1 \right) \quad \text{si} \quad |\tau_s| \geq \tau_f + \tau_s^{forest}, \quad \text{sinon} \quad \dot{p}_s = 0$$

avec n grand (pour représenter un modèle quasi élasto-plastique : pour le CFC, la dépendance en temps est en effet négligeable).

Par rapport au modèle initial [8], le seuil a été introduit ainsi que le terme $-\dot{\gamma}_0$ pour assurer la continuité lorsque le seuil est franchi.

En pratique, le paramètre $\dot{\gamma}_0$ est petit et le terme $\left(\frac{|\tau_s|}{\tau_f + \tau_s^{forest}} \right)^n$ devient très vite largement supérieur à 1 , à cause des grandes valeurs de l'exposant n .

τ_s^{forest} est une fonction d'écrouissage dont les coefficients sont fournis par le mot-clé MONO_DD_* (*=CFC,FAT ou CC) , et qui est décrite au § suivant.

Dans DD_CFC, l'évolution de la densité de dislocation est donnée par :

$$\dot{\rho}_s = \frac{\dot{p}_s}{b} \left(A \frac{\sum_{j \in forest(s)} \sqrt{a_{sj}^{eff}} \rho_j}{\sum_{j=1,12} \sqrt{a_{sj}^{eff} \rho_j}} + B \sum_{j \in copla(s)} \sqrt{a_{sj}^{eff} \rho_j} - y \rho_s \right)$$

- $j \in forest(s)$ sont les systèmes de normales distinctes $\mathbf{n_j} \neq \mathbf{n_s}$
- $j \in copla(s)$ sont les systèmes de normales identiques $\mathbf{n_j} = \mathbf{n_s}$

La valeur initiale de la densité de dislocation est : $\rho_s(t=0) = \rho_s^0$ (champ fourni par l'utilisateur. Bien vérifier s'il s'agit d'une densité globale (pour l'ensemble des 12 systèmes de glissement) ou bien d'une densité par système).

Comme $a_{ij}^{eff} = C(\rho)^2 . a_{ij}$, $\tau_s^{forest}(\boldsymbol{\rho}) = \mu b C(\boldsymbol{\rho}) \sqrt{\sum_{j=1,12} a_{sj} \rho_j}$ et

$$\dot{\rho}_s = \frac{\dot{p}_s}{b} \left(A \frac{\sum_{j \in forest(s)} \sqrt{a_{sj}} \rho_j}{\sum_{j=1,12} \sqrt{a_{sj} \rho_j}} + B C(\rho) \sum_{j \in copla(s)} \sqrt{a_{sj} \rho_j} - y \rho_s \right)$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Pour éviter tout problème de conditionnement de la matrice jacobienne, on choisit d'adimensionner le système en remplaçant la densité de dislocation par : $\omega_s = b^2 * \rho_s$ et résoudre en ω_s . Cela donne :

$$\dot{\omega}_s = \dot{p}_s h_s(\omega) \text{ avec } h_s(\omega) = \left(A \frac{\sum_{j \in forest(s)} \sqrt{a_{sj}}\omega_j}{\sum_{j=1,12} \sqrt{a_{sj}\omega_j}} + B\,C(\omega) \sum_{j \in copla(s)} \sqrt{a_{sj}\omega_j} - \frac{y}{b}\omega_s \right)$$

Remarque : physiquement, la densité de dislocation doit rester positive. Numériquement, pour que tous les termes en $\sqrt{\rho}$ soient licites, il faut tester la positivité de ρ et soit redécouper le pas de temps si cette valeur devient négative (par « hésitation » de l'algorithme au cours des itérations) soit introduire une partie positive dans les équations d'évolution. Cette dernière solution est choisie pour l'intégration explicite comme pour l'intégration implicite.

L'ensemble des équations différentielles régissant l'écoulement viscoplastique du modèle DD_CFC s'écrit donc :

$$\dot{\gamma}_s = \dot{p}_s \frac{\tau_s}{|\tau_s|} \quad \text{(12 équations)}$$

$$\dot{p}_s = \dot{\gamma}_0 \left(\left(\frac{|\tau_s|}{\tau_f + \tau_s^{forest}} \right)^n - 1 \right) \text{ si } |\tau_s| \geq \tau_f + \tau_s^{forest}, \text{ sinon } \dot{p}_s = 0 \quad \text{(12 équations)}$$

$$\dot{\omega}_s = \dot{p}_s h_s(\langle \omega \rangle) \quad \text{(12 équations) avec : } \langle \omega_i \rangle = \begin{cases} \omega_i \text{ si } \omega_i > 0 \\ 0 \text{ sinon} \end{cases}$$

$$C(\omega) = 0.2 + 0.8 \frac{\ln\left(\alpha \sqrt{\sum_{i=1,12} \langle \omega_i \rangle}\right)}{\ln\left(\alpha b \sqrt{\rho_{ref}}\right)}$$

$$\tau_s^{forest}(\boldsymbol{\omega}) = \mu\, C(\boldsymbol{\omega}) \sqrt{\sum_{j=1,12} a_{sj} \langle \omega_j \rangle}$$

$$h_s(\omega) = \left(A \frac{\sum_{j \in forest(s)} \sqrt{a_{sj}} \langle \omega_j \rangle}{\sum_{j=1,12} \sqrt{a_{sj} \langle \omega_j \rangle}} + B\,C(\omega) \sum_{j \in copla(s)} \sqrt{a_{sj} \langle \omega_j \rangle} - \frac{y}{b} \langle \omega_s \rangle \right)$$

avec $\omega_s = b^2 * \rho_s$

Dans le cas MONO_DD_CFC_IRRA, l'évolution des variables liées à l'irradiation est donnée par :

$$\dot{\rho}_s^{loops} = -\xi \left(\sum_{j \in copla(s)} |\dot{\gamma}_j| \right) \left(\rho_s^{loops} - \rho^{sat} \right) \qquad \dot{\varphi}_s^{voids} = -\zeta \left(\sum_{j \in copla(s)} |\dot{\gamma}_j| \right) \left(\varphi_s^{voids} - \varphi^{sat} \right)$$

l'expression de τ_s^{forest} devient : $\tau_s^{forest} = \mu \sqrt{\sum_{j=1,12} a_{sj}^{eff} \omega_j + \alpha^{loops} \varphi^{loops} \rho_s^{loops} b^2 + \alpha^{voids} \varphi_s^{voids} \rho^{voids} b^2}$

ce qui, après multiplication par b^2 : $\omega_s^{loops} = b^2 * \rho_s^{loops}$ donne :

$$\dot{\omega}_s^{loops} = -\xi \left(\sum_{j \in copla(s)} |\dot{\gamma}_j| \right) \left(\omega_s^{loops} - \omega^{sat} \right) \text{ et } \tau_s^{forest} = \mu \sqrt{\sum_{j=1,12} a_{sj}^{eff} \omega_j + \alpha^{loops} \varphi^{loops} \omega_s^{loops} + \alpha^{voids} \varphi_s^{voids} \omega^{voids}}$$

on peut remarquer au passage que les équations d'évolution de ω_s^{loop} et φ_s^{voids} s'intègrent analytiquement :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

$$\omega_s^{loops}(t+\Delta t)=\omega^{sat}+\left(\omega_s^{loops}(t)-\omega^{sat}\right)\exp\left(-\xi\sum_{j\in copla(s)}|\Delta\gamma_j|\right)$$
$$\varphi_s^{voids}(t+\Delta t)=\varphi^{sat}+\left(\varphi_s^{voids}(t)-\varphi^{sat}\right)\exp\left(-\zeta\sum_{j\in copla(s)}|\Delta\gamma_j|\right)$$

Dans le cas DD_FAT, l'évolution de la densité de dislocation est donnée par :

$$\dot{\rho}_s=\frac{\dot{p}_s}{b}\left(\frac{1}{d}+\frac{\sqrt{\sum_{u\neq s}\rho_u}}{K}-g_{c0}\rho_s\right)$$

En adimensionnant comme ci-dessus, c'est à dire en posant :

$$\omega_s=b^2\times\rho_s$$

on obtient :

$$\dot{\omega}_s=\dot{p}_s h_s(\omega) \text{ avec } h_s(\omega)=\left(\frac{b}{d}+\frac{\sqrt{\sum_{u\neq s}\omega_u}}{K}-g_{c0}\frac{\omega_s}{b}\right)$$

D'où en introduisant les parties positives :

$$\dot{\gamma}_s=\dot{p}_s\frac{\tau_s}{|\tau_s|} \text{ (12 équations)}$$
$$\dot{p}_s=\dot{\gamma}_0\left(\left(\frac{|\tau_s|}{\tau_f+\tau_s^{forest}}\right)^n-1\right) \text{ si } |\tau_s|\geq\tau_f+\tau_s^{forest}, \text{ sinon } \dot{p}_s=0 \quad \text{(12 équations)}$$
$$\dot{\omega}_s=\dot{p}_s h_s(\langle\omega\rangle) \quad \text{(12 équations) avec : } \langle\omega_i\rangle=\begin{cases}\omega_i \text{ si } \omega_i>0\\ 0 \text{ sinon}\end{cases}$$
$$C(\omega)=1.$$
$$\tau_s^{forest}(\boldsymbol{\omega})=\mu\sqrt{\sum_{j=1,12}a_{sj}\langle\omega_j\rangle}$$
$$h_s(\omega)=\left(\frac{b}{d}+\frac{\sqrt{\sum_{u\neq s}\langle\omega_u\rangle}}{K}-g_{c0}\frac{\langle\omega_s\rangle}{b}\right)$$
avec $\omega_s=b^2*\rho_s$

La loi d'écrouissage des modèles DD_CFC est : $\tau_s^{forest}=\mu\, b\sqrt{\sum_{j=1,n_s}a_{sj}^{eff}\rho_j}$

et $a_{ij}^{eff}=C(\rho)^2.a_{ij}$ avec $C(\rho)=0.2+0.8\dfrac{\ln\left(\alpha b\sqrt{\sum_{i=1,12}\rho_i}\right)}{\ln\left(\alpha b\sqrt{\rho_{ref}}\right)}$

où a_{ij} représente la matrice d'interaction entre systèmes de glissement, dont la forme particulière est fournie en annexe 4.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

La loi d'écrouissage du modèle DD_FAT est : $\tau_s^{forest}(\rho)=\mu b\sqrt{\sum_{j=1,12} a_{sj}\rho_j}$, ce qui correspond dans le modèle précédent à $C(\rho)=1.$

2.2.4 Lois issues de la Dynamique des Dislocations : MONO_DD_CC / MONO_DD_CC_IRRA

La formulation des lois DD_CC (avec ou sans irradiation) est construite à partir de calculs de dynamique des dislocations [17]. Elle s'applique aux matériaux à structure cristalline Cubique Centrée (CC)

Les équations ci-dessous sont écrites pour chaque système de glissement s.

L'écoulement visco-plastique est obtenu par moyenne harmonique de deux régimes (basse température, et haute température).

On choisit ici de faire apparaître les inconnues ρ (représentant l'ensemble des densités de dislocation ρ^s et ρ_{irr}^s) et τ_{eff}^s (qui est une inconnue supplémentaire a priori. On verra que l'algorithme de résolution proposé permet de tout exprimer en fonction de ρ) .

ρ_{irr}^s désigne la densité de dislocations induite par l'irradiation (cas de **MONO_DD_CC_IRRA**), et n'est pas utilisée pour le comportement **MONO_DD_CC** afin de limiter le nombre de variables internes.

τ^s étant la cission résolue sur le système s :

- le glissement plastique est obtenue par : $\frac{1}{\dot{\gamma}^s}=\frac{1}{\dot{\gamma}_{nuc}^s}+\frac{1}{\dot{\gamma}_{prob}^s}$
- $\dot{\gamma}_{nuc}^S=\rho_m^s b\, H\cdot l^s(\rho,\tau_{eff}^s)\cdot\exp\left(\frac{-\Delta G(\tau_{eff}^s)}{k_B T}\right) sgn(\tau_s)$
- $\Delta G(\tau_{eff}^s)=\Delta G_0\left(1-\left(\frac{\langle\tau_{eff}^s\rangle}{\tau_0}\right)^{0.5}\right)$ (avec : $\tau_{eff}^s<\tau_0$ et $0<\Delta G<\Delta G_0$)
- $\tau_{eff}^s(\tau^s,\rho)=|\tau^s|-\tau_c \qquad \tau_c=\tau_F+\sqrt{(\tau_{LR}^s(\rho)^2+\tau_{\mathrm{LT}}^s(\rho,\tau_{eff}^s)^2)}$
- $\tau_{LR}^s(\rho)=\mu b\sqrt{a_{AT}^s(\rho)\rho^s}$
- $\tau_{\mathrm{LT}}^s(\rho,\tau_{eff}^s)=\max\left(0\,;\alpha_{AT}^s(\rho)\mu b\left(\frac{1}{\lambda^s(\rho,\tau_{eff}^s)}-\frac{1}{2\alpha_{AT}^s(\rho)R^s(\rho,\tau_{eff}^s)+l_c(T)}\right)\right)$
- $\rho_{tot}^s(\rho)=\sum_{j\neq s}\rho^j+\rho_{irr}^s$
- $\alpha_{AT}^s(\rho)=\sqrt{\sum_{j\neq s} a_{AT}^{sj}\frac{\rho^j}{\rho_{tot}^s}+a_{irr}\frac{\rho_{irr}^s}{\rho_{tot}^s}}$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

- $$\frac{1}{\lambda^s(\rho,\tau^s_{eff})+D}=min\left(\sqrt{\rho^s_{tot}(\rho)};(D+2R^s(\rho,\tau^s_{eff}))\,\rho^s_{tot}(\rho)\right)$$
- $$R^s(\rho,\tau^s_{eff})=\frac{\mu b}{2\tau_0\left(1-\frac{\Delta G(\tau^s_{eff})}{\Delta G_0}\right)^2}$$
- $$l^s(\rho,\tau^s_{eff})=max\left(\lambda^s(\rho,\tau^s_{eff})-2\alpha^s_{AT}(\rho)R^s(\rho,\tau^s_{eff}) \quad ; \quad l_c\right)$$
- $$\dot{\gamma}^s_{prob}=\dot{\gamma}_0\left(\frac{|\tau^s|}{\tau_c}\right)^n sgn(\tau^s)$$

Les paramètres sont : $\rho^s_m, b, H, \Delta G_0, \tau_0, \tau_F, \mu, a_{irr}, l_c(T), D, \dot{\gamma}_0, n$ et la matrice d'interaction α^{sr}_{AT}

Les lois d'évolution des densités de dislocations (ρ^s et ρ^s_{irr}) sont :

- $$\dot{\rho}^s=\frac{|\dot{\gamma}^s|}{b}\left(\frac{1}{d_{lath}}+\frac{c_{eff}\sqrt{a^{ss}_{AT}\rho^s}}{K_{self}}+\frac{\alpha^s_{AT}c_{eff}\lambda^s\rho^s}{K_f}-y^s\rho^s\right)$$
- $$\dot{\rho}^s_{irr}=-\xi\rho^s_{irr}|\dot{\gamma}^s|$$

avec :

$\frac{1}{y^s}=\frac{1}{y^s_{AT}}+\frac{2\pi\tau^s_{eff}(\rho)}{\mu b}$ et $c^s_{eff}=\left(1-\frac{\tau^s_{eff}}{\tau_0}\right)$,

avec les paramètres d_{lath} K_{self} K_f ξ y^s_{AT}

Remarques :

- Physiquement, la densité de dislocation doit rester positive. Numériquement, pour que tous les termes en $\sqrt{\rho}$ soient licites, il faut tester la positivité de ρ et soit redécouper le pas de temps si cette valeur devient négative (par « hésitation » de l'algorithme au cours des itérations) soit introduire une partie positive dans les équations d'évolution. Cette dernière solution est choisie pour l'intégration explicite comme pour l'intégration implicite.
- Dans le cas où l'élasticité n'est pas isotrope, la valeur de μ utilisée ci-dessus est indépendante des coefficient d'élasticité, car obtenue par homogénéisation.

2.3 Systèmes de glissement et comportement global du monocristal

Un monocristal est composé de une ou plusieurs familles de systèmes de glissement, (cubique, octaédrique, basal, prismatique,…), chaque famille comprenant un certain nombre de systèmes (12 pour la famille octaédrique par exemple).

A chaque famille de système de glissement sont associés une loi d'écoulement, un type d'écrouissage cinématique et isotrope, et des valeurs des paramètres pour ces lois. En d'autres termes, on ne prévoit pas de faire varier les relations de comportement ou les coefficients au sein d'une même famille de systèmes de glissement. Par contre, d'une famille à l'autre, les lois de comportement peuvent changer, ainsi que les valeurs des paramètres.

Un système de glissement est déterminé par un tenseur d'orientation (ou tenseur de Schmid) construit à partir des définitions cristallographiques de :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

•la direction de glissement du système s (définie par le vecteur unitaire $\boldsymbol{m}_s$)
•et de la normale au plan de glissement (définie par le vecteur unitaire $\boldsymbol{n}_s$).

2.3.1 Petites déformations, configuration initiale

En petites déformations, le tenseur d'orientation est défini par :

$$\mu_{ij}^s = \frac{1}{2}\left(\boldsymbol{m}_i \otimes \boldsymbol{n}_j + \boldsymbol{n}_j \otimes \boldsymbol{m}_i\right)$$

Du point de vue du comportement au point matériel, ce tenseur intervient pour le calcul de la cission résolue.

$$\tau^s = \boldsymbol{\Sigma} : \mu^s$$

et celui de la vitesse de déformation viscoplastique globale $\boldsymbol{E}^{vp}$, définie à partir de la connaissance des vitesses de glissement $\dot{\gamma}^s$ pour tous les systèmes de glissement :

$$\dot{\boldsymbol{E}}_{ij}^{vp} = \sum_{s \in g} \dot{\gamma}^s \mu_{ij}^s$$

De plus, le monocristal peut être orienté par rapport aux axes globaux de définition des coordonnées. Cette orientation est définie pour chaque maille ou groupe de mailles (typiquement pour chaque grain) par la donnée de 3 angles nautiques ou 3 angles d'Euler.

Les composantes du tenseur d'orientation μ^s , définies dans le repère lié au monocristal, sont alors exprimées dans le repère global en utilisant ces angles nautiques. En petites déformations, elles sont fixes dans la configuration initiale.

De plus, au niveau d'un point de Gauss, on postule une relation d'élasticité sur les tenseurs globaux:

•déformation totale macroscopique $\boldsymbol{E}$
•déformation viscoplastique macroscopique $\boldsymbol{E}^{vp}$
•contrainte macroscopique : $\boldsymbol{\Sigma}$

$$\boldsymbol{\Sigma} = \boldsymbol{\Lambda}\left(\boldsymbol{E} - \boldsymbol{E}^{th} - \boldsymbol{E}^{vp}\right)$$

où $\boldsymbol{\Lambda}$ représente l'opérateur d'élasticité (isotrope ou orthotrope)

2.3.2 Petites déformations, rotation du réseau cristallin

Une approximation de l'effet de la déformation sur le comportement du monocristal consiste à prendre en compte la rotation du réseau cristallin. Cette rotation est intéressante à deux titres :

- sa prise en compte effective dans les calculs est cohérente avec des déformations non infiniment petites (dans les calculs courants, il n'est pas rare d'atteindre 10%). Elle est un préliminaire à un modèle complet de grandes déformations sur le monocristal.
- C'est une information intéressante pour le post-traitement, car elle peut être confrontée à des résultats expérimentaux.

On décrit ici (suivant 9, 10, 11) la méthode de prise en compte de la rotation de réseau cristallin dans le cadre général des petites déformations. Les principales hypothèses physiques sont :

- connaissant le tenseur vorticité $\hat{\dot{\omega}}$ ou taux de rotation, qui est la partie antisymétrique du tenseur $\mathbf{L} = \dot{\mathbf{F}}\mathbf{F}^{-1}$), on postule une décomposition additive en une partie dite rotation plastique $\hat{\dot{\omega}}^p$ (partie antisymétrique de la rotation plastique) et une partie «élastique » $\hat{\dot{\omega}}^e$, qui permettra de calculer la rotation du réseau cristallin ;

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

• on admet pour le moment que l'on peut calculer le tenseur taux de rotation, tout en effectuant l'intégration du comportement avec un tenseur des déformations linéarisées. Cette limitation tombe naturellement avec le modèle complet en grandes déformations, mais on suppose ici que l'on peut raisonner de façon découplée. On a donc, sous forme incrémentale :

$\Delta\hat{\boldsymbol{\omega}}=\frac{1}{2}(\Delta\mathbf{F}\,\mathbf{F}^{-1}-(\Delta\mathbf{F}\,\mathbf{F}^{-1})^{T})$ et $\Delta\hat{\boldsymbol{\omega}}^{\mathbf{e}}=\Delta\hat{\boldsymbol{\omega}}-\Delta\hat{\boldsymbol{\omega}}^{\mathbf{p}}$ qui sont des tenseurs antisymétriques.

Le calcul de $\Delta\hat{\boldsymbol{\omega}}^{\mathbf{p}}$ est effectué en fin d'intégration : il est stocké et utilisé à l'itération suivante (un calcul purement implicite impliquerait de résoudre le système d'équations décrit au paragraphe 3 en ajoutant toutes les équations sur l'orientation du réseau, ce qui alourdirait beaucoup le calcul).

Une fois l'incrément de rotation $\Delta\hat{\boldsymbol{\omega}}^{\mathbf{e}}$ connu, on extrait le vecteur « axial » [voir par exemple R5.03.40] : $\Delta\boldsymbol{\omega}^{\mathbf{e}}$ tel que $\Delta\boldsymbol{\omega}^{\mathbf{e}}\wedge\mathbf{U}=\Delta\hat{\boldsymbol{\omega}}^{\mathbf{e}}\cdot\mathbf{U}$ et on calcule l'angle de rotation et le vecteur qui correspond à l'axe de rotation :

$\Delta\theta=\|\Delta\boldsymbol{\omega}^{\mathbf{e}}\|$ et $\mathbf{n}_{\boldsymbol{\omega}}=\frac{\Delta\boldsymbol{\omega}^{\mathbf{e}}}{\Delta\theta}$.

La matrice d'incrément de rotation est alors obtenue par la formule d'Euler-Rodrigues :

$\Delta Q=\mathbf{I}_{\mathbf{d}}+\sin(\Delta\theta)\hat{\mathbf{n}}_{\boldsymbol{\omega}}+(1-\cos(\Delta\theta))\hat{\mathbf{n}}_{\boldsymbol{\omega}}\cdot\hat{\mathbf{n}}_{\boldsymbol{\omega}}$ où $\hat{\mathbf{n}}_{\boldsymbol{\omega}}$ est la matrice antisymétrique issue de $\mathbf{n}_{\boldsymbol{\omega}}$

Ce qui permet de mettre à jour la matrice de rotation du réseau : $Q_{+}=\Delta Q\cdot Q_{-}$

Elle permet d'actualiser le tenseur d'orientation par : $\begin{matrix}\mathbf{n}^{\mathbf{s}}=Q_{+}\,\mathbf{n}_{\mathbf{0}}^{\mathbf{s}}\\ \mathbf{m}^{\mathbf{s}}=Q_{+}\,\mathbf{m}_{\mathbf{0}}^{\mathbf{s}}\end{matrix}$.

Le comportement est alors intégré avec cette définition du réseau cristallin.

Il reste alors à actualiser l'incrément de rotation plastique : $\Delta\hat{\boldsymbol{\omega}}^{\mathbf{p}}=\frac{1}{2}\sum_{s}\Delta\gamma^{s}(\mathbf{m}^{\mathbf{s}}\otimes\mathbf{n}^{\mathbf{s}}-\mathbf{n}^{\mathbf{s}}\otimes\mathbf{m}^{\mathbf{s}})$

Les variables internes supplémentaires sont : $\mathbf{Q}$, $\Delta\boldsymbol{\omega}^{\mathbf{p}}$, $\Delta\boldsymbol{\omega}^{\mathbf{e}}$, θ

En pratique la rotation de réseau est prise en compte via le mot-clé ROTA_RESEAU de la commande DEFI_COMPOR [U4.43.06]. Elle n'est effective que pour l'intégration implicite du comportement MONOCRISTAL.

2.3.3 Grandes déformations

Propriétés attendues : le formalisme choisi doit être objectif, et doit conserver l'incompressibilité plastique ($det(\boldsymbol{F}^{p})=1$). On utilise la décomposition multiplicative $\boldsymbol{F}=\boldsymbol{F}^{e}\,\boldsymbol{F}^{p}$ avec la configuration intermédiaire relâchée isocline de Mandel.

La loi d'(hyper-)élasticité choisie utilise le second tenseur de Piola-Kirchhoff $\boldsymbol{S}=J^{e}\,\boldsymbol{F}^{e-1}\boldsymbol{\sigma}\,\boldsymbol{F}^{e-T}$ et la déformation de Green-Lagrange associée au gradient de déformation élastique :

$$\boldsymbol{E}_{\boldsymbol{GL}}^{e}=\frac{1}{2}(\boldsymbol{F}^{eT}\boldsymbol{F}^{e}-\boldsymbol{I}_{d}) \qquad \boldsymbol{S}=\boldsymbol{\Lambda}:\boldsymbol{E}_{\boldsymbol{GL}}^{e}\ .$$

Sur chaque système de glissement s, la cission résolue est écrite avec le tenseur de Mandel $\boldsymbol{M}$ défini par $\boldsymbol{M}=J_{e}\,\boldsymbol{F}^{eT}\boldsymbol{\sigma}\,\boldsymbol{F}^{e\text{-}\mathrm{T}}$ de la façon suivante : $\tau_{s}=\boldsymbol{M}:\boldsymbol{m}_{s}\otimes\boldsymbol{n}_{s}$ (cf. 7, 14).

En effet, la puissance des efforts intérieurs peut s'écrire : $\frac{1}{\rho}\boldsymbol{\sigma}:\boldsymbol{D}=\frac{1}{\rho}\boldsymbol{\sigma}:\boldsymbol{L}=\frac{1}{\rho}\boldsymbol{\sigma}:\dot{\boldsymbol{F}}\,\boldsymbol{F}^{-1}$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

et $\dot{\boldsymbol{F}}\boldsymbol{F}^{-1}=\left(\dot{\boldsymbol{F}}^{\mathrm{e}}\boldsymbol{F}^{p}+\boldsymbol{F}^{\mathrm{e}}\dot{\boldsymbol{F}}^{p}\right)(\boldsymbol{F}^{p})^{-1}(\boldsymbol{F}^{e})^{-1}=\boldsymbol{L}^{\mathrm{e}}+\boldsymbol{F}^{\mathrm{e}}\boldsymbol{L}^{p}(\boldsymbol{F}^{e})^{-1}$

donc $\frac{1}{\rho}\boldsymbol{\sigma}:\boldsymbol{D}=\frac{1}{\rho}\left[\boldsymbol{\sigma}:\boldsymbol{L}^{\mathrm{e}}+\boldsymbol{\sigma}:\left(\boldsymbol{F}^{\mathrm{e}}\boldsymbol{L}^{p}\left(\boldsymbol{F}^{\mathrm{e}}\right)^{-1}\right)\right]=\frac{1}{\rho_0}\boldsymbol{S}:\dot{\boldsymbol{E}}^{\mathrm{e}}_{\boldsymbol{GL}}+\frac{1}{\rho_0}\boldsymbol{M}:\boldsymbol{L}^{p}$

La puissance dissipée plastiquement s'écrit : $\boldsymbol{M}:\boldsymbol{L}^{p}$ et aussi : $\sum_s \dot{\gamma}_s\tau_s$, donc

$\boldsymbol{M}:\boldsymbol{L}^{p}=\boldsymbol{M}:\sum_s \dot{\gamma}_s\boldsymbol{\mu}_{\boldsymbol{s}}=\sum_s \dot{\gamma}_s\boldsymbol{M}:\boldsymbol{\mu}_{\boldsymbol{s}}$, d'où $\tau_s=\boldsymbol{M}:\boldsymbol{\mu}_{\boldsymbol{s}}$

Pour chaque loi de comportement cristalline, les équations d'écoulement et d'évolution des variables internes s'écrivent classiquement sur chaque système de glissement :

$\dot{\gamma}_s=g(\tau_s,\alpha_s,\gamma_s,p_s)$ ou $\dot{\alpha}_s=h(\tau_s,\alpha_s,\gamma_s,p_s)$

Par contre la relation reliant la déformation plastique aux glissements des systèmes actifs s'écrit maintenant : $\dot{\boldsymbol{F}}^{p}=\left(\sum_s \dot{\gamma}_s\boldsymbol{m}_{\boldsymbol{s}}\otimes\boldsymbol{n}_{\boldsymbol{s}}\right)\boldsymbol{F}^{p}$

2.4 Comportement du polycristal homogénéisé

Dans le cas de polycristal homogénéisé, il faut définir chaque phase mono-cristalline par son orientation, sa proportion (fraction volumique) et le comportement associé. Une orientation globale du polycristal peut être définie. Il faut de plus définir une règle de localisation.
Le comportement monocristallin est construit comme précédemment à partir du comportement elasto-visco-plastique précédent et de la donnée de familles de systèmes de glissement.

En plus du comportement monocristallin décrit précédemment, on ajoute une échelle de modélisation, qui représente l'assemblage des phases.

Au niveau d'un point de Gauss, on postule une relation d'élasticité sur les tenseurs globaux:

- déformation totale macroscopique $\boldsymbol{E}$
- déformation viscoplastique macroscopique $\boldsymbol{E}^{\boldsymbol{VP}}$
- contrainte macroscopique : $\boldsymbol{\Sigma}$

$$\boldsymbol{\Sigma}=\boldsymbol{\Lambda}(\boldsymbol{E}-\boldsymbol{E}^{\boldsymbol{th}}-\boldsymbol{E}^{\boldsymbol{VP}})$$

$\boldsymbol{\Lambda}$ représente l'opérateur d'élasticité (isotrope ou orthotrope)

- De plus, connaissant l'ensemble des variables internes relatives aux systèmes de glissement de chaque phase, les paramètres de comportement de chaque phase, les orientations et fractions volumiques de chaque phase, et le type de méthode de localisation,

- pour chaque phase monocristalline (ou « grain »), définie par une orientation et une proportion f_g, une relation de localisation des contraintes, de la forme générale (à exprimer dans le repère local de chaque phase)

$$\sigma_g=L\left(\boldsymbol{\Sigma},\boldsymbol{E}^{vp},\boldsymbol{\varepsilon}^{vp}_{\boldsymbol{g}},\boldsymbol{\beta}_{\boldsymbol{g}}\right)$$

et pour chaque système de glissement de chaque phase, des relations de comportement relatives à chaque système de glissement, similaires au cas du monocristal :

- Relation d'écoulement :
  $\dot{\gamma}_s=g(\tau_s,\alpha_s,\gamma_s,p_s)$, avec, $\dot{p}_s=\left|\dot{\gamma}_s\right|$ et $\dot{p}_s=f(\tau_s,\alpha_s,\gamma_s,p_s)$
- Évolution de l'écrouissage cinématique ou de la densité de dislocations : $\dot{\alpha}_s=h(\tau_s,\alpha_s,\gamma_s,p_s)$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

•Évolution de l'écrouissage isotrope défini par une fonction : $R(p_s)$

•Déformations viscoplastiques de la phase : $\dot{\varepsilon}_{ij}^{vp_g} = \sum_{s \in g} \dot{\gamma}^s \mu_{ij}^s$

Il reste alors les équations d'homogénéisation : $\dot{\boldsymbol{E}}^{vp} = \sum_g f_g \dot{\boldsymbol{\varepsilon}}_g^{vp}$

Deux relations de localisation de type $\boldsymbol{\sigma}_g = L\left(\boldsymbol{\Sigma}, \boldsymbol{E}^{vp}, \boldsymbol{\varepsilon}_g^{vp}, \boldsymbol{\beta}_g\right)$ sont disponibles :

•La relation de Berveiller-Zaoui [bib5] établie sur la notion d'autocohérence. Cette relation est validée sous certaines conditions, à savoir : isotropie du matériau (des travaux récent mettent en oeuvre des extensions à l'élasticité cubique), comportement élastique homogène et chargement monotone :

$$\boldsymbol{\sigma}_{ij}^g = \boldsymbol{\Sigma}_{ij} + \alpha \mu^{loca}\left(\boldsymbol{E}_{ij}^{vp} - \boldsymbol{\varepsilon}_{ij}^{vp_g}\right) \quad \frac{1}{\alpha} = 1 + \frac{3}{2}\mu^{loca}\frac{\left\|E_{ij}^{vp}\right\|}{J_2(\Sigma_{ij})}$$

•La deuxième relation, inspirée de la précédente, et développée plus particulièrement pour des chargements cycliques [bib4] permet de donner une bonne description pour schématiser les interactions entre les grains :

• $\boldsymbol{\sigma}_{ij}^g = \boldsymbol{\Sigma}_{ij} + \mu^{loca}\left(\boldsymbol{B}_{ij} - \boldsymbol{\beta}_{ij}^g\right) \quad \boldsymbol{B}_{ij} = \sum_g f_g \boldsymbol{\beta}_{ij}^g$

• $\dot{\boldsymbol{\beta}}_{ij}^g = \dot{\boldsymbol{\varepsilon}}_{ij}^{vp_g} - D\left(\boldsymbol{\beta}_{ij}^g - \delta\, \boldsymbol{\varepsilon}_{ij}^{vp_g}\right)\left\|\dot{\boldsymbol{\varepsilon}}_{ij}^{vp_g}\right\|$

•

où μ^{loca} , D et δ sont des paramètres caractéristiques du matériau et de la température.

3 Intégration locale et mise en œuvre numérique

3.1 Système d'équations à résoudre

3.1.1 Comportement de type MONOCRISTAL

Le comportement local du monocristal est défini, à un instant donné de la discrétisation en temps et au niveau d'un point d'intégration d'un élément fini, par la donnée :

•du tenseur de contraintes macroscopiques à l'instant précédent $\Sigma(t_{i-1}) = \Sigma^-$,
•des variables internes à l'instant précédent, pour chaque système de glissement : $\alpha_s(t_{i-1}), \gamma_s(t_{i-1}), p_s(t_{i-1})$,
•et du tenseur d'accroissement de déformation totale fournie par l'itération n de l'algorithme global de résolution $\Delta \boldsymbol{E} = \Delta \boldsymbol{E}_i^n$ (avec les notations de [R5.03.01]).

L'intégration consiste à trouver :

•le tenseur de contraintes macroscopique $\Sigma = \Sigma(t_i)$
•et les variables internes $\alpha_s = \alpha_s(t_i)$, $\gamma_s = \gamma_s(t_i)$, $p_s = p_s(t_i)$

vérifiant les équations de comportement dans chaque système de glissement (qui sont des relations mono dimensionnelles), et les relations de passage entre les tenseurs macroscopique et l'ensemble des directions de glissement. Notation : on écrit les équations sous forme discrétisée de façon :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

•explicite, si les quantités notées $A^{+/-}$ sont évaluées à l'instant t_{i-1} : $A^{+/-}=A^{-}=A\left(t_{i-1}\right)$

•implicite, si elles sont évaluées à l'instant t_i : $A^{+/-}=A^{+}=A\left(t_i\right)$

Les équations à intégrer peuvent se mettre sous la forme générale suivante :

Etant donnés, en un point de Gauss, les tenseurs :
ΔE : variation de déformation à l'instant t_i ,
$E(t_{i-1})=E^{-}$: déformation à l'instant t_{i-1} ,
$\Sigma(t_{i-1})=\Sigma^{-}$: contrainte macroscopique à l'instant t_{i-1} ,
$\alpha_s\left(t_{i-1}\right), \gamma_s\left(t_{i-1}\right), p_s\left(t_{i-1}\right)$: variables internes pour chaque système de glissement à t_{i-1} ,

Il faut trouver :

$\Sigma=\Sigma(t_i)$: contrainte macroscopique à l'instant t_i , dans le repère correspondant à l'orientation globale
$\alpha_s=\alpha_s(t_i)$
$\gamma_s=\gamma_s(t_i)$
$p_s=p_s(t_i)$

vérifiant :

$\boldsymbol{D^{-1}\Sigma}=(\boldsymbol{D_{-}^{-1}})\boldsymbol{\Sigma^{-}}+(\Delta\boldsymbol{E}-\Delta\boldsymbol{E^{th}}-\Delta\boldsymbol{E^{vp}})$, où $\boldsymbol{\Lambda}$ peut dépendre de la température, et peut correspondre à une élasticité orthotrope.
$\Delta\boldsymbol{E^{vp}}=\sum_s \mu_s \Delta\gamma_s$
pour chaque système de glissement (de l'ensemble des familles de systèmes) :
n_s relations d'écoulement : soit en viscoplasticité $\Delta\gamma_s=g(\tau_s^{+/-},\alpha_s^{+/-},\gamma_s^{+/-},p_s^{+/-})$
avec $\Delta p_s=f(\tau_s^{+/-},\alpha_s^{+/-},\gamma_s^{+/-},p_s^{+/-})$
soit en plasticité $F(\tau_s^{+/-},\alpha_s^{+/-},\gamma_s^{+/-},p_s^{+/-})\leq 0$, $F\Delta p_s=0$
,
avec $\Delta p_s=|\Delta\gamma_s|$
n_s équations d'évolutions de l'écrouissage cinématique : $\Delta\alpha_s=h(\tau_s^{+/-},\alpha_s^{+/-},\gamma_s^{+/-},p_s^{+/-})$
n_s équations d'évolution de l'écrouissage isotrope : $R_s(p_s^{+/-})$

Ceci est résolu soit de façon explicite (*Runge_Kutta*), soit implicite (*Newton*).

3.1.2 Comportement de type POLYCRISTAL

Les relations de comportement discrétisées sont :

Etant donnés (en un point de Gauss) les tenseurs globaux :

•accroissement de déformation totale $\Delta\boldsymbol{E}$,

•déformation totale à l'instant précédent , $\boldsymbol{E}(t_{i-1})=\boldsymbol{E^{-}}$

•contrainte à l'instant précédent : $\boldsymbol{\Sigma}\left(t_{i-1}\right)=\boldsymbol{\Sigma^{-}}$,

•l'ensemble des variables internes $\alpha_s^{-}, \gamma_s^{-}, p_s^{-}$ relatives aux systèmes de glissement de chaque phase,

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

•les paramètres de comportement de chaque phase, les orientations et fractions volumiques de chaque phase, et le type de méthode de localisation.

Il faut trouver $\Sigma=\Sigma(t_i),\ \alpha_s=\alpha_s(t_i),\ \gamma_s=\gamma_s(t_i),\ p_s=p_s(t_i)$ vérifiant :

•au niveau du point de Gauss : $\boldsymbol{\Sigma}=\boldsymbol{\Lambda}(\boldsymbol{\Lambda}_{-}^{-1})\boldsymbol{\Sigma}^{-}+\boldsymbol{\Lambda}(\Delta\boldsymbol{E}-\Delta\boldsymbol{E}^{th}-\Delta\boldsymbol{E}^{vp})$, dans le repère global,

•pour chaque phase (ou « grain »), définie par une orientation et une proportion f_g , une relation de localisation des contraintes, de la forme générale (à exprimer dans le repère local de chaque phase)

$$\boldsymbol{\sigma}_{\boldsymbol{g}}=L\left(\boldsymbol{\Sigma},\boldsymbol{E}^{vp},\varepsilon_g^{vp},\beta_g\right)$$

•et pour chaque système de glissement de chaque phase :

- $\Delta\varepsilon_{g\,vp}=\sum_s m_s\Delta\gamma_s$
- n_s équations : $\tau^s=\boldsymbol{\Sigma}_{\mathbf{ij}}:\boldsymbol{\mu}_{\mathbf{ij}}^{\boldsymbol{s}}$
- n_s relations d'écoulement : $\Delta\gamma_s=g(\tau_s,\alpha_s,\gamma_s,p_s)$ avec $\Delta p_s=|\Delta\gamma_s|$
- n_s évolutions de l'écrouissage : $\Delta\alpha_s=h(\tau_s,\alpha_s,\gamma_s,p_s)$
- $F(\tau_s,\alpha_s,\gamma_s,p_s)\le 0,\ \ F\cdot\Delta p_s=0$, (en plasticité indépendante du temps)

Il reste alors les équations d'homogénéisation : $\Delta E^{vp}=\sum_g f_g\Delta\varepsilon_g^{vp}$

Les comportements viscoplastiques relatifs à chaque système de glissement sont identiques au cas de la microstructure.

Dans la version actuelle de *Code_Aster*, ces relations de comportement sont intégrés uniquement de façon explicite.

3.2 Résolution implicite du comportement MONOCRISTAL

3.2.1 Résolution implicite des lois phénoménologiques

Il faut résoudre un système de la forme générale suivante :

$$R(Y)=\begin{Bmatrix} s(\Sigma,\Delta E^{\mathrm{vp}},\alpha_s^+,\gamma_s^+,p_s^+)\\ e(\Sigma,\Delta E^{\mathrm{vp}},\alpha_s^+,\gamma_s^+,p_s^+)\\ n_s\begin{Bmatrix} a(\Sigma,\Delta E^{\mathrm{vp}},\alpha_s^+,\gamma_s^+,p_s^+)\\ g(\Sigma,\Delta E^{\mathrm{vp}},\alpha_s^+,\gamma_s^+,p_s^+)\\ p(\Sigma,\Delta E^{\mathrm{vp}},\alpha_s^+,\gamma_s^+,p_s^+)\end{Bmatrix}\end{Bmatrix}=\begin{Bmatrix} \boldsymbol{\Lambda}^{-1}\boldsymbol{\Sigma}-(\boldsymbol{\Lambda}_{-}^{-1})\boldsymbol{\Sigma}^{-}-(\Delta\boldsymbol{E}-\Delta\boldsymbol{E}^{\mathrm{th}}-\Delta\boldsymbol{E}^{\mathrm{vp}})\\ \Delta\boldsymbol{E}^{\mathrm{vp}}-\sum_s\boldsymbol{\mu}_s\Delta\gamma_s\\ n_s\begin{Bmatrix}\Delta\alpha_s-h(\tau_s^+,\alpha_s^+,\gamma_s^+,p_s^+)\\ \Delta\gamma_s-g(\tau_s^+,\alpha_s^+,\gamma_s^+,p_s^+)\\ \Delta p_s-f(\tau_s^+,\alpha_s^+,\gamma_s^+,p_s^+)\end{Bmatrix}\end{Bmatrix}=0$$

De manière plus contractée, on pose :

$$R(\Delta y)=0=\begin{bmatrix} s(\Delta y)\\ e(\Delta y)\\ a(\Delta y)\\ g(\Delta y)\\ p(\Delta y)\end{bmatrix}\quad\text{avec}\quad\Delta y=\begin{pmatrix}\Delta\Sigma\\ \Delta E^{vp}\\ \Delta\alpha_s\\ \Delta\gamma_s\\ \Delta p_s\end{pmatrix}$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Pour résoudre ce système de $6+6+3n_s$ équations non linéaires (en 3D), on utilise une méthode de Newton : on construit une suite de vecteurs solution de la façon suivante :

$$Y_{k+1}=Y_k-(\frac{dR}{dY_k})^{-1}R(Y_k)$$

Il faut donc définir les valeurs initiales Y_0 , et calculer la matrice jacobienne du système : $\frac{dR}{dY_k}$

(celle-ci est détaillée en annexe pour les comportements viscoplastiques décrits précédemment). Elle a la forme suivante :

$$J=\begin{bmatrix} \frac{\partial s}{\partial \Delta\Sigma} & \frac{\partial s}{\partial \Delta E^{vp}} & \frac{\partial s}{\partial \Delta\alpha_s} & \frac{\partial s}{\partial \Delta\gamma_s} & \frac{\partial s}{\partial \Delta p_s} \\ \frac{\partial e}{\partial \Delta\Sigma} & \frac{\partial e}{\partial \Delta E^{vp}} & \frac{\partial e}{\partial \Delta\alpha_s} & \frac{\partial e}{\partial \Delta\gamma_s} & \frac{\partial e}{\partial \Delta p_s} \\ \frac{\partial a}{\partial \Delta\Sigma} & \frac{\partial a}{\partial \Delta E^{vp}} & \frac{\partial a}{\partial \Delta\alpha_s} & \frac{\partial a}{\partial \Delta\gamma_s} & \frac{\partial a}{\partial \Delta p_s} \\ \frac{\partial g}{\partial \Delta\Sigma} & \frac{\partial g}{\partial \Delta E^{vp}} & \frac{\partial g}{\partial \Delta\alpha_s} & \frac{\partial g}{\partial \Delta\gamma_s} & \frac{\partial g}{\partial \Delta p_s} \\ \frac{\partial p}{\partial \Delta\Sigma} & \frac{\partial p}{\partial \Delta E^{vp}} & \frac{\partial p}{\partial \Delta\alpha_s} & \frac{\partial p}{\partial \Delta\gamma_s} & \frac{\partial p}{\partial \Delta p_s} \end{bmatrix}$$

On peut toutefois simplifier ce système d'équations en réduisant sa taille, en vue d'améliorer les performances de l'intégration.

Dans l'expression des 6 composantes du tenseur des contraintes, $\Delta \boldsymbol{E}^{\boldsymbol{vp}}$ peut être exprimée en fonction de $\sum_s \boldsymbol{\mu}_s \Delta\gamma_s$ donc une équation peut être éliminée du système global à résoudre.
Comme $\Delta p=|\Delta\gamma|$, l'équation en Δp peut être éliminée.

Dans le cas de **MONO_CINE1**, $\Delta\alpha$ s'exprime directement en fonction de $\Delta\gamma$. La taille du système à résoudre se réduit donc à $6+n_s$ équations en $\Delta\gamma$, avec n_s le nombre de systèmes de glissement.

Dans le cas de **MONO_CINE2**, on calcule numériquement $\Delta\alpha$ en fonction de $\Delta\gamma$, par une méthode de sécante.

L'équation se rapportant aux composantes des contraintes se réduit à :

$$R_1^i\left(\boldsymbol{\Sigma_i},\tau_s,\Delta\gamma_s,\Delta\alpha_s,R_s(p_r)\right)=\boldsymbol{\Lambda^{-1}\Sigma_i}-\left(\boldsymbol{\Lambda_-^{-1}\Sigma_{i_}}\right)-\Delta\boldsymbol{\varepsilon}+\Delta\boldsymbol{\varepsilon^{th}}+\sum_s \boldsymbol{\mu_s} g_s\left(\tau_s,\Delta\gamma_s,\Delta\alpha_s,R_s(p_r)\right)=0$$

L'équation se rapportant au paramètre d'écrouissage $\Delta\gamma$ s'écrit sous la forme :

$$R_2^s\left(\boldsymbol{\Sigma_i},\tau_s,\Delta\gamma_s,\Delta\alpha_s,R_s(p_r)\right)=\Delta\gamma_s-g_s\left(\tau_s,\Delta\gamma_s,\Delta\alpha_s,R_s(p_r)\right)=0$$

avec $i=1$ à 6 , $s=1$ à n_s et $r=1$ à n_s

Dans le cas de **MONO_VISC1** :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

$$g_s=\Delta t\left\langle\frac{|\tau_s-c\,\alpha_s|-R_s(p_r)}{k}\right\rangle^n\frac{\tau_s-c\alpha_s}{|\tau_s-c\,\alpha_s|}\quad\text{avec}\quad\tau_s=\boldsymbol{\Sigma}:\boldsymbol{\mu_s}$$

Dans le cas de **MONO_VISC2** :

$$g_s=\Delta t\left\langle\frac{|\tau_s-c\,\alpha_s-a\,\gamma_s|-R_s(p_r)+\frac{d}{2c}(c\,\alpha_s)^2}{k}\right\rangle^n\frac{\tau_s-c\,\alpha_s-a\,\gamma_s}{|\tau_s-c\,\alpha_s-a\,\gamma_s|}$$

La matrice jacobienne du système réduit s'écrit sous la forme :

$$J=\begin{bmatrix}\frac{\partial R_1^i}{\partial\Sigma_j} & \frac{\partial R_1^i}{\partial\Delta\gamma_s}\\ \frac{\partial R_2^s}{\partial\Sigma_i} & \frac{\partial R_2^s}{\partial\Delta\gamma_r}\end{bmatrix}\text{ de dimension }(6+n_s)\times(6+n_s)$$

Les différents termes de la matrice jacobienne sont :

$$\frac{\partial R_1^i}{\partial\Sigma_j}=\left(\boldsymbol{\Lambda^{-1}}\right)_{ij}+\sum_s(\mu_s)_i\frac{\partial g_s}{\partial\Sigma_j}=\left(\boldsymbol{\Lambda^{-1}}\right)_{ij}+\sum_s(\mu_s)_i\frac{\partial g_s}{\partial\tau_s}\frac{\partial\tau_s}{\partial\Sigma_j}=\left(\boldsymbol{\Lambda^{-1}}\right)_{ij}+\sum_s(\mu_s)_i(\mu_s)_j\frac{\partial g_s}{\partial\tau_s}$$

$$\frac{\partial R_2^s}{\partial\Sigma_i}=\frac{\partial\left(-g_s(\tau_s,\Delta\alpha_s,\Delta\gamma_s,R_s(p_r))\right)}{\partial\Sigma_i}=\frac{\partial(-g_s)}{\partial\tau_s}\cdot\frac{\partial\tau_s}{\partial\Sigma_i}=-\mu_s^i\cdot\frac{\partial g_s}{\partial\tau_s}$$

$$\frac{\partial R_1^i}{\partial\Delta\gamma_s}=\sum_s\mu_s^i\cdot\frac{\partial g_s}{\partial\Delta\gamma_s}$$

$$\frac{\partial R_2^s}{\partial\Delta\gamma_{rs}}=\delta_{sr}-\frac{\partial g_s}{\partial\Delta\gamma_r}\qquad\text{avec}\qquad\frac{\partial g_s}{\partial\Delta\gamma_r}=\frac{\partial g_s}{\partial\Delta\alpha_s}\cdot\frac{\partial\Delta\alpha_s}{\partial\Delta\gamma_s}\delta_{sr}+\frac{\partial g_s}{\partial R_s}\cdot\frac{\partial R_s}{\partial\Delta\gamma_r}$$

Dans les expressions intervenant ci-dessus , les termes $\frac{\partial g_s}{\partial\tau_s}$, $\frac{\partial g_s}{\partial\Delta\alpha_s}$, $\frac{\partial\Delta\alpha_s}{\partial\Delta\gamma_s}$, $\frac{\partial g_s}{\partial R_s}$, $\frac{\partial R_s}{\partial\Delta p_r}$ dépendent les lois d'écoulement et d'écrouissage :

- Dans le cas de **MONO_VISC1** :

$$\frac{\partial g_s}{\partial\tau_s}=\frac{n\Delta t}{k^n}\langle|\tau_s-c\,\alpha_s|-R_s(p_r)\rangle^{n-1}=H_s(\tau_s,\Delta\alpha_s,\Delta\gamma_s,R_s(p_r))$$

$$\frac{\partial g_s}{\partial\Delta\alpha_s}=-c\,.\,H_s(\tau_s,\Delta\gamma_s,R_s)$$

$$\frac{\partial g_s}{\partial R_s}=-H_s(\tau_s,\Delta\gamma_s,R_s)\operatorname{sgn}s\quad\text{avec}\quad\operatorname{sgn}s=\frac{\tau_s-c\,\alpha_s}{|\tau_s-c\,\alpha_s|}$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

- Dans le cas de **MONO_VISC2 :**

$$\frac{\partial g_s}{\partial \tau_s}=H_s\left(\tau_s,\Delta\alpha_s,\Delta\gamma_s,R_s\left(p_r\right)\right)=\frac{n}{k^n}\left\langle\left|\tau_s-c\alpha_s-a\gamma_s\right|-R_s\left(p_r\right)+\frac{d}{2c}\left(c\alpha_s\right)^2\right\rangle^{n-1}$$

$$\frac{\partial g_s}{\partial \Delta\alpha_s}=H_s\left(\tau_s,\Delta\gamma_s,R_s\right)\left(-c\,\mathrm{sgn}\,s+d\,\alpha_s\Delta\alpha_s\right)$$

$$\frac{\partial g_s}{\partial R_s}=-H_s\left(\tau_s,\Delta\gamma_s,R_s\right)\mathrm{sgn}\,s \qquad \text{avec } \mathrm{sgn}\,s=\frac{\tau_s-c\alpha_s-a\gamma_s}{\left|\tau_s-c\alpha_s-a\gamma_s\right|}$$

- Dans le cas de **MONO_CINE1 :**

$$\frac{\partial\Delta\alpha_s}{\partial\Delta\gamma_r}=\frac{1-d\,\alpha_s^-\mathrm{sgn}\,s}{\left(1+d\left|\Delta\gamma_s\right|\right)^2}$$

- Dans le cas de **MONO_CINE2 :**

$\frac{\partial\alpha_s}{\partial\Delta\gamma_r}$; Cette dérivation nécessite la résolution d'une équation non-linéaire par la méthode de Newton

- Dans le cas de **MONO_ISOT1 :**

$$\frac{\partial R_s}{\partial\Delta\gamma_r}=\frac{\partial\left[Q\sum_k h_{sk}\left(1-e^{-bp_k}\right)\right]}{\partial\Delta\gamma_r}=\frac{\partial}{\partial p_r}\left[Q\sum_k h_{sk}\left(1-e^{-bp_k}\right)\right]\frac{\partial p_r}{\partial\Delta\gamma_r}=bQh_{sr}e^{-bp_r}.\mathrm{sgn}\,r$$

$$\text{avec } \mathrm{sgn}\,r=\frac{\partial p_r}{\partial\Delta\gamma_r}=\frac{\partial\left|\Delta\gamma_r\right|}{\partial\Delta\gamma_r}=\frac{\Delta\gamma_r}{\left|\Delta\gamma_r\right|}$$

- Dans le cas de **MONO_ISOT2 :**

$$\frac{\partial R_s}{\partial\Delta\gamma_r}=\frac{\partial\left[Q_1\sum_k h_{sk}\left(1-e^{-b_1p_k}\right)+Q_2\left(1-e^{-b_2p_r}\right)\right]}{\partial\Delta\gamma_r}=$$
$$\frac{\partial}{\partial p_r}\left[Q_1\sum_k h_{sk}\left(1-e^{-b_1p_k}\right)\right]\frac{\partial p_r}{\partial\Delta\gamma_r}+\frac{\partial}{\partial p_r}\left[Q_2\left(1-e^{-b_2p_r}\right)\right]\frac{\partial p_r}{\partial\Delta\gamma_r}=$$
$$\left(b_1Q_1h_{sr}e^{-b_1p_r}+b_2Q_2e^{-b_2p_r}\delta_{sr}\right)\mathrm{sgn}\,r$$

$$\text{avec } \mathrm{sgn}\,r=\frac{\partial p_r}{\partial\Delta\gamma_r}=\frac{\partial\left|\Delta\gamma_r\right|}{\partial\Delta\gamma_r}=\frac{\Delta\gamma_r}{\left|\Delta\gamma_r\right|}$$

3.2.2 Résolution implicite – Intégration du modèle MONO_DD_KR

Le système à résoudre se réduit aux équations relatives aux composantes des contraintes

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

$$R_1^i\left(\Sigma_i,\tau_s,\alpha_s\right)=\left(\mathbf{\Lambda^{-1}}\Sigma\right)_i-\left(\mathbf{\Lambda_-^{-1}}\Sigma_-\right)_i-\Delta\varepsilon_i+\Delta\varepsilon_i^{th}+\sum_s \mu_i^s g_s\left(\tau_s,\alpha_s\right)=0$$

et celles relatives au paramètre d'écrouissage:

$$R_2^s\left(\sum_i,\tau_s,\alpha_s\right)=\Delta\alpha_s-\left|g_s\left(\tau_s,\alpha_s\right)\right|.h\left(\alpha_s\right)=0$$

avec i = 1 à 6 , s = 1 à ns et r =1 à ns

Ceci permet d'obtenir un système d'équations formulé de façon entièrement implicite, où R_1^i et R_2^s ne dépendent que de Σ_i,α_s (car τ_s se calcule explicitement à partir de Σ_i

$$g_s\left(\tau_s,\alpha_s\right)=\dot{\gamma}_0\exp\left(\frac{-\Delta G\left(\tau_{eff}^s\right)}{k_B T}\right).\frac{\tau_s}{\left|\tau_s\right|}.\Delta t$$

$$\Delta G\left(\tau_{eff}^s\right)=\Delta G_0\left(1-\left(\frac{\left\langle\tau_{eff}^s\right\rangle}{\tau_R}\right)^p\right)^q$$

$$h\left(\alpha_s\right)=\left(\frac{b}{d}+\frac{\sqrt{\sum_{u\neq s}\alpha^u}}{K}-\frac{g_c\alpha^s}{b}\right)$$

$\tau_{eff}^s=\left\langle\left|\tau^s\right|-\tau_0-\tau_\mu^s\right\rangle$ $\langle\rangle$: partie positive

$$\tau_\mu^s=\frac{\left(\mu\right)^2\sum_u a^{su}\alpha^u}{\left|\tau^s\right|-\tau_0}\quad\text{avec}\quad\alpha^u=\rho^u b^2$$

Remarque : pour chaque système de glissement s où les conditions $\tau_{eff}^s>0$ $\left|\tau^s\right|>\tau_0$ ne sont pas vérifiées, on obtient immédiatement : $g_s\left(\tau_s,\alpha_s\right)=0,\ \Delta\alpha_s=0$

La matrice jacobienne du système s'écrit sous la forme :

$$J=\begin{bmatrix}\dfrac{\partial R_1^i}{\partial\Sigma_j} & \dfrac{\partial R_1^i}{\partial\Delta\alpha_s}\\ \dfrac{\partial R_2^s}{\partial\Sigma_i} & \dfrac{\partial R_2^s}{\partial\Delta\alpha_r}\end{bmatrix}\quad\text{de dimension}\quad\left(6+n_s\right)\times\left(6+n_s\right)$$

- $$\frac{\partial R_1^i}{\partial\Sigma_j}=\left(\mathbf{\Lambda^{-1}}\right)_{ij}+\sum_s\left(\mu_s\right)_i\frac{\partial g_s}{\partial\Sigma_j}=\left(\mathbf{\Lambda^{-1}}\right)_{ij}+\sum_s\left(\mu_s\right)_i\frac{\partial g_s}{\partial\tau_s}\frac{\partial\tau_s}{\partial\Sigma_j}=\left(\mathbf{\Lambda^{-1}}\right)_{ij}+\sum_s\left(\mu_s\right)_i\left(\mu_s\right)_j\frac{\partial g_s}{\partial\tau_s}$$

=> Il faut calculer $\dfrac{\partial g^s}{\partial\tau_s}=\mathrm{sgn}\left(\tau_s\right).\dfrac{\partial\left|g^s\right|}{\partial\tau_s}$

- $$\frac{\partial R_2^s}{\partial\Sigma_i}=\frac{\partial\left(-\left|g_s\right|.h_s\right)}{\partial\tau_s}.\frac{\partial\tau_s}{\partial\Sigma_i}=-\mu_s^i.\left(\frac{\partial\left|g_s\right|}{\partial\tau_s}.h_s+\left|g_s\right|.\frac{\partial h_s}{\partial\tau_s}\right)$$

=> Le terme $\dfrac{\partial h_s}{\partial\tau_s}$ étant nul, il reste à calculer $\dfrac{\partial\left|g_s\right|}{\partial\tau_s}$.

- $$\frac{\partial R_1^i}{\partial\Delta\alpha_s}=\sum_s\mu_s^i\frac{\tau_s}{\left|\tau_s\right|}.\frac{\partial\left|g_s\right|}{\partial\alpha_s}$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

- $\dfrac{\partial R_2^s}{\partial \Delta\alpha_r}=\delta_{sr}-\dfrac{\partial|g_s|}{\partial \Delta\gamma_r}=\delta_{sr}-\left(\dfrac{\partial|g_s|}{\partial \Delta\alpha_r}h_s+|g_s|\dfrac{\partial h_s}{\partial \Delta\alpha_r}\right)$

=> Il faut aussi calculer $\dfrac{\partial|g_s|}{\partial\alpha_r}$ et $\dfrac{\partial|g_s|}{\partial\tau_s}$;

$h_s=\dfrac{\Delta\alpha_s}{|\Delta\gamma_s|}$ et $\dfrac{\partial\tau_\mu^s}{\partial\tau_s}=\dfrac{-\tau_\mu^s}{|\tau^s|-\tau_0}.\operatorname{sgn} s$

Calcul de $\dfrac{\partial|g_s|}{\partial\tau_r}$:

$$\frac{\partial|g_s|}{\partial\tau_s}=-\frac{\Delta t.\dot\gamma_0}{k_BT}\exp\left(-\frac{\Delta G\left(\tau_{eff}^s\right)}{k_BT}\right).\frac{\Delta G\left(\tau_{eff}^s\right)}{\Delta\tau_s}=\frac{-|g_s\left(\tau_s,\alpha_s\right)|}{k_BT}.\frac{\partial\Delta G\left(\tau_{eff}^s\right)}{\partial\tau_s}$$

où

$$\frac{\partial\Delta G\left(\tau_{eff}^s\right)}{\partial\tau_s}=-\Delta G_0\left(1-\left(\frac{\tau_{eff}^s}{\tau_R}\right)^p\right)^{q-1}.\frac{q.p}{\tau_R}.\left(\frac{\tau_{eff}^s}{\tau_R}\right)^{p-1}.\frac{\partial\tau_{eff}^s}{\partial\tau_s}$$

et

$$\frac{\partial\tau_{eff}^s}{\partial\tau_s}=\frac{\tau_s}{|\tau_s|}-\frac{\partial\tau_\mu^s}{\partial\tau_s}=\operatorname{sgn}\left(\tau_s\right)+\mu^2\left(\sum a^{su}\alpha^u\right).\frac{\operatorname{sgn}\left(\tau_s\right)}{\left(|\tau_s|-\tau_0\right)^2}=\operatorname{sgn}\left(\tau_s\right)+\tau_\mu^s.\frac{\operatorname{sgn}\left(\tau_s\right)}{|\tau_s|-\tau_0}$$

Calcul de $\dfrac{\partial|g_s|}{\partial\alpha_r}$:

$$\frac{\partial|g_s|}{\partial\alpha_r}=-\frac{\Delta t.\dot\gamma_0}{k_BT}\exp\left(-\frac{\Delta G\left(\gamma_{eff}^s\right)}{k_BT}\right).\frac{\Delta G\left(\tau_{eff}^s\right)}{\Delta\alpha_r}=\frac{-|g_s\left(\tau_s,\alpha_s\right)|}{k_BT}.\frac{\partial\Delta G\left(\tau_{eff}^s\right)}{\partial\alpha_r}$$

où

$$\frac{\partial\Delta G\left(\tau_{eff}^s\right)}{\partial\alpha_r}=-\frac{qp}{\tau_R}\Delta G_0\left(1-\left(\frac{\tau_{eff}}{\tau_R}\right)^p\right)^{q-1}.\left(\frac{\tau_{eff}}{\tau_R}\right)^{p-1}.\frac{\partial\tau_{eff}^s}{\partial\alpha_r}$$

et

$$\frac{\partial\tau_{eff}^s}{\partial\alpha_r}=-\frac{\partial\tau_\mu^s}{\partial\alpha_r}=-\frac{\mu^2}{|\tau_s|-\tau_0}\frac{\partial\left(\sum_u a^{su}\alpha^u\right)}{\partial\alpha_r}=-\frac{\mu^2}{|\tau_s|-\tau_0}a^{sr}$$

Calcul de $\dfrac{\partial|g_s|}{\partial\tau_s}$:

$$\frac{\partial h^s}{\partial\alpha_r}=\frac{1}{K}\frac{\partial}{\partial\alpha_r}\sqrt{\sum_{u\neq s}\alpha^u}-\frac{g_c}{b}\delta_{sr}=\frac{1}{2\mathrm{K}}\frac{1-\delta_{rs}}{\sqrt{\sum_{u\neq s}\alpha^u}}-\frac{g_c}{b}\delta_{sr}$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

3.2.3 Résolution implicite – modèles MONO_DD_CFC(IRRA) / MONO_DD_FAT

Expression du résidu Connaissant la solution à l'instant t_{i-1}, notée $\boldsymbol{\sigma}^-=\boldsymbol{\sigma}(t_{i-1})$, $\rho_s^-=\rho_s(t_{i-1})$, $\gamma_s^-=\gamma_s(t_{i-1})$, $p_s^-=p_s(t_{i-1})$,
il s'agit de trouver, à l'instant t_i, les quantités $\boldsymbol{\sigma}=\boldsymbol{\sigma}(t_i)$, $\rho_s=\rho_s(t_i)$, $\gamma_s=\gamma_s(t_i)$, $p_s=p_s(t_i)$ vérifiant :

$$\left(\boldsymbol{\Lambda}^{-1}\boldsymbol{\sigma}\right)_i-\left((\boldsymbol{\Lambda}^-)^{-1}\boldsymbol{\sigma}^-\right)_i=\Delta\varepsilon_i-\Delta\varepsilon_i^{th}-\sum_s\left((\boldsymbol{\mu_s})_i\Delta\gamma_s(\boldsymbol{\sigma},\boldsymbol{\omega})\right)=0$$

La cission résolue par système de glissement est : $\tau_s=\boldsymbol{\sigma}:\boldsymbol{\mu_s}=\left(\boldsymbol{\sigma}^-+\Delta\boldsymbol{\sigma}\right):\boldsymbol{\mu_s}$

La loi d'écoulement après discrétisation implicite devient :

$$\Delta\gamma_s=\dot{\gamma}_0\Delta t\left(\left(\frac{|\tau_s|}{\tau_f+\tau_s^{forest}}\right)^n-1\right).\frac{\tau_s}{|\tau_s|}=\Delta p_s\frac{\tau_s}{|\tau_s|}$$

$$\Delta p_s(\boldsymbol{\sigma},\boldsymbol{\omega})=\dot{\gamma}_0\Delta t\left(\left(\frac{|\tau_s|}{\tau_f+\tau_s^{forest}(\boldsymbol{\omega})}\right)^n-1\right)\ \text{ si }\ |\tau_s|\geq\tau_f+\tau_s^{forest}\ \text{ ; sinon }\ \Delta p_s=0$$

La loi d'écrouissage est : $\tau_s^{forest}(\omega)=\tau_s^{forest}(\langle\omega^-+\Delta\omega\rangle)=\mu C(\omega)\sqrt{\sum_{j=1,12}\langle a_{sj}\omega_j\rangle}$

En présence d'irradiation, la loi d'écrouissage devient :

$$\tau_s^{forest}=\mu\sqrt{\sum_{j=1,12}C(\omega)^2a_{sj}\omega_j+\alpha^{loops}\varphi^{loops}\omega_s^{loops}+\alpha^{voids}\varphi_s^{voids}\omega^{voids}}$$

avec $C(\omega)=0.2+0.8\dfrac{\ln\left(\alpha\sqrt{\sum_{i=1,12}\langle\omega_i\rangle}\right)}{\ln\left(\alpha b\sqrt{\rho_{ref}}\right)}$ ou, dans le cas DD_FAT, $C(\omega)=1.$

L'évolution de la densité de dislocation est donnée (toujours sous forme implicite) par :

$\Delta\omega_s=\Delta p_s h_s(\omega)=\Delta p_s h_s(\omega^-+\Delta\omega)$ avec :

$$h_s(\omega)=\left(A\ \frac{\sum_{j\in forest(s)}\sqrt{a_{sj}}\ \langle\omega_j\rangle}{\sum_{j=1,12}\sqrt{a_{sj}\langle\omega_j\rangle}}+BC(\boldsymbol{\omega})\sum_{j\in copla(s)}\sqrt{a_{sj}\langle\omega_j\rangle}-\frac{y}{b}\omega_s\right)$$

ou, dans le cas DD_FAT, $h_s(\omega)=\left(\dfrac{b}{d}+\dfrac{\sqrt{\sum_{u\neq s}\langle\omega_u\rangle}}{K}-g_{c0}\dfrac{\langle\omega_s\rangle}{b}\right)$

L'évolution des densités de dislocations liées à l'irradiation est de la forme suivante :

$$\omega_s^{loops}(t+\Delta t)=\omega^{sat}+\left(\omega_s^{loops}(t)-\omega^{sat}\right)\exp\left(-\xi\sum_{j\in copla(s)}|\Delta\gamma_j|\right)$$

$$\varphi_s^{voids}(t+\Delta t)=\varphi^{sat}+\left(\varphi_s^{voids}(t)-\varphi^{sat}\right)\exp\left(-\zeta\sum_{j\in copla(s)}|\Delta\gamma_j|\right)$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

En utilisant la notation vectorielle de Kelvin [chapitre 5 de 15) pour les tenseurs symétriques ($\boldsymbol{\sigma}$, $\boldsymbol{\mu_s}$), et en prenant en compte les éventuelles variations des coefficients de l'élasticité avec la température, on aboutit au système à résoudre, comportant 18 équations ; $i=1,6$, $s=1,12$, les 18 inconnues étant les 6 composantes de $\boldsymbol{\sigma}$ et les 12 valeurs $\boldsymbol{\omega}$.

Système à résoudre en $\boldsymbol{\sigma}$ (6 composantes) et $\Delta\omega$ (12 composantes) :

$$R_i^{(1)}(\boldsymbol{\sigma},\boldsymbol{\omega})=\left(\boldsymbol{\Lambda^{-1}\sigma}\right)_i-\left((\Lambda^-)^{-1}\boldsymbol{\sigma}^-\right)_i-\Delta\varepsilon_i+\Delta\varepsilon_i^{th}+\sum_s\left((\boldsymbol{\mu_s})_i\Delta p_s(\boldsymbol{\sigma},\boldsymbol{\omega})\cdot\frac{\tau_s}{|\tau_s|}\right)=0$$

$$R_s^{(2)}(\boldsymbol{\sigma},\boldsymbol{\omega})=\Delta\omega_s-\Delta p_s h_s(\omega)=0$$

avec : $\tau_s=\boldsymbol{\sigma}:\boldsymbol{\mu_s}=(\boldsymbol{\sigma}^-+\Delta\sigma):\boldsymbol{\mu_s}$

$$\Delta p_s(\boldsymbol{\sigma},\boldsymbol{\omega})=\dot{\gamma}_0\Delta t\left(\left(\frac{|\tau_s|}{\tau_f+\tau_s^{forest}(\boldsymbol{\omega})}\right)^n-1\right)\quad\text{si}\quad|\tau_s|\geq\tau_0+\tau_s^f\ ;\ \text{sinon}\ \Delta p_s=0$$

et si $\Delta p_s>0$:

$$\tau_s^{forest}(\omega)=\tau_s^{forest}(\omega^-+\Delta\omega)=\mu\,C(\omega)\sqrt{\sum_{j=1,12}a_{sj}\langle\omega_j\rangle}$$

ou $\tau_s^{forest}=\mu\sqrt{\sum_{j=1,12}C(\omega)^2a_{sj}\omega_j+\alpha^{loops}\varphi^{loops}\omega_s^{loops}+\alpha^{voids}\varphi_s^{voids}\omega^{voids}}$

$$C(\omega)=0.2+0.8\frac{\ln\left(\alpha\sqrt{\sum_{i=1,12}\langle\omega_i\rangle}\right)}{\ln\left(\alpha\,b\sqrt{\rho_{ref}}\right)}$$

$$h_s(\omega)=\left(A\ \frac{\sum_{j\in forest(s)}\sqrt{a_{sj}}\ \langle\omega_j\rangle}{\sum_{j=1,12}\sqrt{a_{sj}\langle\omega_j\rangle}}+B\,C(\boldsymbol{\omega})\sum_{j\in copla(s)}\sqrt{a_{sj}\langle\omega_j\rangle}-\frac{y}{b}\langle\omega_s\rangle\right)$$

Dans le cas DD_FAT, ce système devient :

$$R_i^{(1)}(\boldsymbol{\sigma},\boldsymbol{\omega})=\left(\boldsymbol{\Lambda^{-1}\sigma}\right)_i-\left((\Lambda^-)^{-1}\boldsymbol{\sigma}^-\right)_i-\Delta\varepsilon_i+\Delta\varepsilon_i^{th}+\sum_s\left((\boldsymbol{\mu_s})_i\Delta p_s(\boldsymbol{\sigma},\boldsymbol{\omega})\cdot\frac{\tau_s}{|\tau_s|}\right)=0$$

$$R_s^{(2)}(\boldsymbol{\sigma},\boldsymbol{\omega})=\Delta\omega_s-\Delta p_s h_s(\omega)=0$$

avec : $\tau_s=\boldsymbol{\sigma}:\boldsymbol{\mu_s}=(\boldsymbol{\sigma}^-+\Delta\sigma):\boldsymbol{\mu_s}$

$$\Delta p_s(\boldsymbol{\sigma},\boldsymbol{\omega})=\dot{\gamma}_0\Delta t\left(\left(\frac{|\tau_s|}{\tau_f+\tau_s^{forest}(\boldsymbol{\omega})}\right)^n-1\right)\quad\text{si}\quad|\tau_s|\geq\tau_0+\tau_s^f\ ;\ \text{sinon}\ \Delta p_s=0$$

et si $\Delta p_s>0$:

$$\tau_s^{forest}(\omega)=\tau_s^{forest}(\omega^-+\Delta\omega)=\mu\sqrt{\sum_{j=1,12}a_{sj}\langle\omega_j\rangle}$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

$$h_s(\omega)=\left(\frac{b}{d}+\frac{\sqrt{\sum_{u\neq s}\langle\omega_u\rangle}}{K}-g_{c0}\frac{\langle\omega_s\rangle}{b}\right)$$

La matrice jacobienne du système peut être alors calculée pour l'intégration par la méthode de Newton (R5.03.14).

Matrice jacobienne

Ce paragraphe ne concerne que les lois DD_CFC et DD_CC : pour la loi DD_FAT, la matrice jacobienne n'a pas été programmée et on utilise la matrice calculée par perturbation. La matrice jacobienne du système s'écrit sous la forme :

$$J=\begin{bmatrix}\left(\frac{\partial R_i^{(1)}}{\partial\sigma_j}\right)_{i=1,6;\ j=1,6} & \left(\frac{\partial R_i^{(1)}}{\partial\Delta\omega_s}\right)_{i=1,6;\ s=1,12}\\ \left(\frac{\partial R_s^{(2)}}{\partial\sigma_j}\right)_{s=1,12;\ j=1,6} & \left(\frac{\partial R_s^{(2)}}{\partial\Delta\omega_r}\right)_{s=1,12;\ r=1,12}\end{bmatrix}$$ de dimension 18×18

- $$\frac{\partial R_i^{(1)}}{\partial\sigma_j}=\left(\Lambda^{-1}\right)_{ij}+\sum_s\left(\mu_s\right)_i\frac{\partial\left(\Delta p_s\frac{\tau_s}{|\tau_s|}\right)}{\partial\sigma_j}=\left(\Lambda^{-1}\right)_{ij}+\sum_s\left(\mu_s\right)_i\frac{\tau_s}{|\tau_s|}\frac{\partial\Delta p_s}{\partial\tau_s}\frac{\partial\tau_s}{\partial\sigma_j}=\left(\Lambda^{-1}\right)_{ij}+\sum_s\left(\mu_s\right)_i\left(\mu_s\right)_j\frac{\tau_s}{|\tau_s|}\frac{\partial\Delta p_s}{\partial\tau_s}$$

avec $\frac{\partial\Delta p_s}{\partial\tau_s}=n\frac{\left(\Delta p_s+\dot\gamma_0\Delta t\right)}{\tau_s}$

en effet : $\frac{\partial\Delta p_s}{\partial\tau_s}=\frac{n\dot\gamma_0\Delta t}{\tau_f+\tau_s^{forest}(\omega)}\left(\frac{|\tau_s|}{\tau_f+\tau_s^{forest}(\omega)}\right)^{n-1}\cdot\frac{\tau_s}{|\tau_s|}=n\frac{\left(\Delta p_s+\dot\gamma_0\Delta t\right)}{|\tau_s|}\cdot\frac{\tau_s}{|\tau_s|}$

car $\tau_s^{forest}(\omega)$ ne dépend pas de τ_s et $\Delta p_s+\dot\gamma_0\Delta t=\dot\gamma_0\Delta t\left(\frac{|\tau_s|}{\tau_f+\tau_s^{forest}(\omega)}\right)^n$

- $$\frac{\partial R_i^{(1)}}{\partial\Delta\omega_s}=\sum_{u=1,12}\left(\mu_u\right)_i\frac{\tau_u}{|\tau_u|}\frac{\partial\Delta p_u}{\partial\Delta\omega_s}$$

avec $\frac{\partial\Delta p_u}{\partial\Delta\omega_s}=-n\dot\gamma_0\Delta t|\tau_u|\left(\frac{|\tau_u|}{\tau_f+\tau_u^{forest}(\omega)}\right)^{n-1}\frac{\frac{\partial\tau_u^{forest}}{\partial\Delta\omega_s}}{\left(\tau_f+\tau_u^{forest}(\omega)\right)^2}$ ou encore

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

$$\frac{\partial \Delta p_u}{\partial \Delta \omega_s} = -n\dot{\gamma}_0 \Delta t \left(\frac{|\tau_u|}{\tau_f + \tau_u^{forest}(\boldsymbol{\omega})} \right)^n \frac{\frac{\partial \tau_u^{forest}}{\partial \Delta \omega_s}}{\left(\tau_f + \tau_u^{forest}(\boldsymbol{\omega}) \right)}$$

$$\boxed{\frac{\partial \Delta p_u}{\partial \Delta \omega_s} = -n \frac{\left(\Delta p_u + \dot{\gamma}_0 \Delta t \right)}{\left(\tau_f + \tau_u^{forest}(\boldsymbol{\omega}) \right)} \frac{\partial \tau_u^{forest}}{\partial \Delta \omega_s}}$$

Il reste à exprimer : $\frac{\partial \tau_u^{forest}}{\partial \Delta \omega_s}$. Sans irradiation, on obtient :

$$\frac{\partial \tau_u^{forest}}{\partial \Delta \omega_s} = \mu \frac{\partial C(\boldsymbol{\omega})}{\partial \Delta \omega_s} \cdot \sqrt{\sum_{j=1,12} a_{uj} \omega_j} + \mu\, C(\boldsymbol{\omega}) \frac{a_{us}}{2\sqrt{\sum_{j=1,12} a_{uj} \omega_j}} \frac{\langle \omega_s \rangle}{\omega_s}$$

et avec irradiation : $\frac{\partial \left(\tau_u^{forest} \right)}{\partial \left(\Delta \omega_s \right)} = \frac{\mu^2 C(\omega)}{2 \tau_u^{forest}} \left(2 \frac{\partial C(\omega)}{\Delta \omega_s} \sum_{j=1,12} a_{uj} \omega_j + C(\omega) a_{us} \right)$

avec $\boxed{\frac{\partial C(\boldsymbol{\omega})}{\partial \Delta \omega_s} = \frac{0.8}{2 \ln(\alpha b \sqrt{\rho_{ref}})} \frac{1}{\sum_{k=1,12} \omega_k} \frac{\langle \omega_s \rangle}{\omega_s}}$

ce qui fournit la dérivée : $\boxed{\frac{\partial R_i^{(1)}}{\partial \Delta \omega_s} = -\sum_{u=1,12} (\mu_u)_i \frac{\tau_u}{|\tau_u|} n \frac{\left(\Delta p_u + \dot{\gamma}_0 \Delta t \right)}{\left(\tau_f + \tau_u^{forest}(\boldsymbol{\omega}) \right)} \frac{\partial \tau_u^{forest}}{\partial \Delta \omega_s}}$

- $\boxed{\frac{\partial R_s^2}{\partial \sigma_i} = -(\mu_s)_i \frac{\partial \Delta p_s}{\partial \tau_s} h_s(\omega)}$

  avec $\frac{\partial \Delta p_s}{\partial \tau_s} = n \frac{\left(\Delta p_s + \dot{\gamma}_0 \Delta t \right)}{\tau_s}$

- $\boxed{\frac{\partial R_s^{(2)}}{\partial \omega_r} = \delta_{sr} - \Delta p_s \frac{\partial h_s(\omega)}{\partial \omega_r} - h_s(\omega) \frac{\partial \Delta p_s}{\partial \omega_r}}$

• On a déjà calculé $\frac{\partial \Delta p_s}{\partial \Delta \omega_r}$. Il reste à dériver : $\frac{\partial h_s(\omega)}{\partial \Delta \omega_r}$:

$$\frac{\partial}{\partial \Delta \omega_r} \left(A \frac{\sum_{j \in forest(s)} \sqrt{a_{sj}}\ \omega_j}{\sum_{j=1,12} \sqrt{a_{sj} \omega_j}} + B\, C(\boldsymbol{\omega}) \sum_{j \in copla(s)} \sqrt{a_{sj} \omega_j} - \frac{y}{b} \omega_s \right) = A \frac{\partial T_1(\boldsymbol{\omega})}{\partial \Delta \omega_r} + B \frac{\partial T_2(\boldsymbol{\omega})}{\partial \Delta \omega_r} - \frac{y}{b} \delta_{sr} \frac{\langle \omega_r \rangle}{\omega_r}$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

$$\frac{\partial}{\partial \Delta\omega_r}\left(T_1(\boldsymbol{\omega})\right)=\frac{\sqrt{a_{sr}}}{\sum_{j=1,12}\sqrt{a_{sj}\omega_j}}\cdot I_s^{forest}(r)\frac{\langle\omega_r\rangle}{\omega_r}-\frac{a_{sr}\sum_{j\in forest(s)}\sqrt{a_{sj}}\omega_j}{2\sqrt{a_{sr}\omega_r}\left(\sum_{j=1,12}\sqrt{a_{sj}\omega_j}\right)^2}\frac{\langle\omega_r\rangle}{\omega_r}$$

$$\frac{\partial}{\partial \Delta\omega_r}T_2(\boldsymbol{\omega})=\frac{\partial C}{\partial \Delta\omega_r}\sum_{j\in copla(s)}\sqrt{a_{sj}\omega_j}+C(\omega)\frac{a_{sr}}{2\sqrt{a_{sr}\omega_r}}I_s^{copla}(r)\frac{\langle\omega_r\rangle}{\omega_r}$$

$I_s^{copla}(r)=1$ si $r\in copla(s)$, $=0$ sinon et $I_s^{forest}(r)=1$ si $r\in forest(s)$, 0 sinon

3.2.4 Résolution implicite – modèles MONO_DD_CC / MONO_DD_CC_IRRA

On choisit d'adimensionner les inconnues en posant $\omega=\rho b^2$. L'algorithme proposé (cf. 17) consiste à éliminer la dépendance de R^s à τ_{eff}^s en approchant ΔG par :

1. $\Delta G_{app}=min\left(\Delta G_0\,;kT\ln\left[\frac{\omega_m^s H}{\sqrt{\omega_{tot}^s}\dot{\varepsilon}_{eq}}\right]\right)$ avec $\omega_{tot}^s(\boldsymbol{\omega})=\omega_F^s(\boldsymbol{\omega})=\sum_{j\neq s}\omega^j+\omega_{irr}^s$
   dans cette expression, $\dot{\varepsilon}_{eq}$ est un paramètre fixé.
2. $R^s(\boldsymbol{\omega})=\frac{\mu b}{2\tau_0\left(1-\frac{\Delta G_{app}}{\Delta G_0}\right)^2}$ en vérifiant que $\Delta G_{app}\leq\Delta G_0$ (sinon, R^s est choisi grand)
3. $\frac{1}{\lambda^s(\boldsymbol{\omega})+D}=min\left(\frac{\sqrt{\omega_{tot}^s(\boldsymbol{\omega})}}{b}\,;(D+2R^s(\boldsymbol{\omega}))\frac{\omega_{tot}^s(\boldsymbol{\omega})}{b^2}\right)$
4. $l^s(\boldsymbol{\omega})=max\left(\lambda^s(\boldsymbol{\omega})-2\alpha_{AT}^s(\boldsymbol{\omega})R^s(\boldsymbol{\omega})\quad;\quad l_c\right)$
5. $\alpha_{AT}^s(\boldsymbol{\omega})=\sqrt{\sum_{j\neq s}h^{sj}\frac{\omega^j}{\omega_{tot}^s}+a_{irr}\frac{\omega_{irr}^s}{\omega_{tot}^s}}$ avec h_{sj} le terme courant de la matrice d'interaction
6. $\tau_{\mathrm{LT}}^s(\boldsymbol{\omega})=\max\left(0\,;\alpha_{AT}^s(\boldsymbol{\omega})\mu b\left(\frac{1}{\lambda^s(\boldsymbol{\omega})}-\frac{1}{2\alpha_{AT}^s(\boldsymbol{\omega})R^s(\boldsymbol{\omega})+l_c(T)}\right)\right)$
7. $\tau_{LR}^s(\boldsymbol{\omega})=\mu\sqrt{h^{ss}\omega^s}$
8. $\tau_{eff}^s(\boldsymbol{\omega},\tau^s)=|\tau^s|-\tau_c(\boldsymbol{\omega})$ et $\tau_c^s=\tau_F+\sqrt{\left(\tau_{LR}^s(\boldsymbol{\omega})^2+\tau_{\mathrm{LT}}^s(\boldsymbol{\omega})^2\right)}$
9. $\Delta\gamma_{nuc}^S(\boldsymbol{\omega},\tau^s)=\frac{\Delta t\,\omega_m^s H}{b}\cdot l^s(\boldsymbol{\omega})\exp\left(\frac{-\Delta G(\boldsymbol{\omega},\tau^s)}{k_B T}\right)sgn(\tau_s)$ avec
   $\Delta G(\boldsymbol{\omega},\tau^s)=\Delta G_0\left(1-\sqrt{\frac{<\tau_{eff}^s(\boldsymbol{\omega},\tau^s)>}{\tau_0}}\right)$ si $\tau_{eff}^s\leq\tau_0$, sinon $\Delta G=\Delta G_0$
10. $\Delta\gamma_{prob}^s(\boldsymbol{\omega},\tau^s)=\Delta t\,\dot{\gamma}_0\left(\frac{|\tau^s|}{\tau_c^s(\boldsymbol{\omega})}\right)^n sgn(\tau^s)$
11. $\frac{1}{\Delta\gamma^s(\boldsymbol{\omega},\tau^s)}=\frac{1}{\Delta\gamma_{nuc}^s(\boldsymbol{\omega},\tau^s)}+\frac{1}{\Delta\gamma_{prob}^s(\boldsymbol{\omega},\tau^s)}$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

12.

$$\Delta\omega^s(\omega,\tau^s)=\left|\Delta\gamma^s(\omega,\tau^s)\right|\left(\frac{b}{d_{lath}}+\frac{c^s_{eff}(\omega,\tau^s)\sqrt{h^{ss}\omega^s}}{K_{self}}+\frac{\alpha^s_{AT}(\omega)c^s_{eff}(\omega,\tau^s)\lambda_s(\omega)\omega^s_{tot}(\omega)}{b\,K_f}-y^s(\omega,\tau^s)\frac{\omega^s}{b}\right)$$

avec

$$c^s_{eff}(\omega,\tau^s)=\left(1-\frac{<\tau^s_{eff}(\omega,\tau^s)>}{\tau_0}\right)\quad\frac{1}{y^s}(\omega,\tau^s)=\frac{1}{y^s_{AT}}+\frac{2\pi\tau^s_{eff}(\omega,\tau^s)}{\mu b}$$

Dans le cas DD_CC_IRRA , $\dot{\omega}^s_{irr}=-\xi\omega^s_{irr}\left|\dot{\gamma}^s(\omega)\right|$, soit : $\omega^s_{irr}(t+\Delta t)=\omega^s_{irr}(t)\exp(-\xi|\Delta\gamma_s|)$

Il faut donc résoudre un système de 12 équations en $\Delta\omega^s$

Remarque : Si on veut résoudre complètement les équations ci-dessus sans l'approximation ΔG_{app} , il faut alors ajouter les 12 inconnues τ_{eff} . En effet la dépendance à τ_{eff} est implicite dans l'expression de R_s si on remplace ΔG_{app} par ΔG .

La matrice jacobienne du système peut être alors calculée pour l'intégration par la méthode de Newton .

Matrice jacobienne

Avec l'approximation choisie, le système d'équations à résoudre est :

$$R^{(1)}_i(\sigma,\omega)=\left(\Lambda^{-1}\sigma\right)_i-\left((\Lambda^{-})^{-1}\sigma^{-}\right)_i-\Delta\varepsilon_i+\Delta\varepsilon^{th}_i+\sum_s\left((\mu_s)_i\Delta p_s(\sigma,\omega).\frac{\tau_s}{|\tau_s|}\right)=0 \quad \text{ou encore}$$

$$R^{(1)}_i(\Delta\varepsilon^{el},\omega)=\Delta\varepsilon^{el}_i-\Delta\varepsilon_i+\Delta\varepsilon^{th}_i+\sum_s\left(\mu^s_i\Delta\gamma^s(\tau^s,\omega)\right)=0 \quad \text{avec} \quad \tau^s=\sigma:\mu^s=\Lambda\varepsilon^{el}:\mu^s$$

$$R^{(2)}_s(\tau^s,\omega)=\Delta\omega_s-\Delta p_s(\omega,\tau^s)H_s(\omega,\tau^s)=0 \quad \text{avec} \quad \Delta p_s(\omega,\tau^s)=\left|(\Delta\gamma_s(\omega,\tau^s))\right| \quad \text{et}$$

$$H^s(\omega,\tau^s)=\left(\frac{b}{d_{lath}}+\frac{c^s_{eff}(\omega,\tau^s)\sqrt{h^{ss}\omega^s}}{K_{self}}+\frac{\alpha^s_{AT}(\omega)c^s_{eff}(\omega,\tau^s)\lambda_s(\omega)\omega^s_{tot}(\omega)}{b\,K_f}-y^s(\omega,\tau^s)\frac{\omega^s}{b}\right)$$

La matrice jacobienne du système s'écrit sous la forme :

$$J=\begin{bmatrix}\left(\frac{\partial R^{(1)}_i}{\partial\varepsilon^{el}_j}\right)_{i=1,6;\,j=1,6} & \left(\frac{\partial R^{(1)}_i}{\partial\Delta\omega_s}\right)_{i=1,6;\,s=1,12}\\ \left(\frac{\partial R^{(2)}_s}{\partial\varepsilon^{el}_j}\right)_{s=1,12;\,j=1,6} & \left(\frac{\partial R^{(2)}_r}{\partial\Delta\omega_s}\right)_{r=1,12;\,s=1,12}\end{bmatrix}$$ de dimension 18×18

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

$$\frac{\partial R_i^{(1)}}{\partial \varepsilon_j^{el}} = \delta_{ij} + \sum_s \mu_i^s \frac{\partial \tau^s}{\partial \sigma_j} = \delta_{ij} + \sum_s \frac{\partial \Delta\gamma^s}{\partial \tau^s} \mu_i^s \mu_k^s \Lambda_{kj}$$

avec $\frac{\partial \Delta\gamma^s}{\partial \tau^s} = (\Delta\gamma^s)^2 \left[\frac{1}{(\Delta\gamma_{nuc}^s)^2} \frac{\partial \Delta\gamma_{nuc}^s}{\partial \tau_s} + \frac{1}{(\Delta\gamma_{prob}^s)^2} \frac{\partial \Delta\gamma_{prob}^s}{\partial \tau_s} \right]$

$$\frac{\partial \Delta\gamma_{nuc}^s}{\partial \tau_s} = \frac{\Delta\gamma_{nuc}^s \Delta G_0}{2 k_b T \sqrt{\tau_0 \tau_{eff}}} sgn(\tau^s) \quad si\ \tau_{eff} > 0 \ , \qquad 0 \text{ sinon}$$

$$\frac{\partial \Delta\gamma_{prob}^s}{\partial \tau_s} = \frac{n \Delta\gamma_{prob}^s}{|\tau_s|}$$

$$\frac{\partial R^{s(2)}}{\partial \varepsilon_i^{el}} = -\left[\frac{\partial \Delta p^s}{\partial \tau^s} H^s + \Delta p^s \frac{\partial H^s}{\partial \tau^s} \right] \mu_k^s \Lambda_{kj}$$

avec $\frac{\partial \Delta p_s}{\partial \tau_s} = \frac{\partial \Delta p^s}{\partial \Delta\gamma^s} \frac{\partial \Delta\gamma^s}{\partial \tau_s} = \frac{\Delta\gamma^s}{|\Delta\gamma^s|} \frac{\partial \Delta\gamma^s}{\partial \tau_s}$ $\frac{\partial \Delta\gamma^s}{\partial \tau_s}$ déjà calculé précédemment.

et $\frac{\partial H^s}{\partial \tau^s} = \left(\frac{\partial c_{eff}^s}{\partial \tau^s} \frac{\sqrt{h^{ss} \omega^s}}{K_{self}} + \frac{\partial c_{eff}^s}{\partial \tau^s} \frac{\alpha_{AT}^s(\omega) \lambda_s(\omega) \omega_{tot}^s(\omega)}{b K_f} - \frac{\partial y^s}{\partial \tau^s} \frac{\omega^s}{b} \right)$

$$\frac{\partial c_{eff}^s}{\partial \tau^s} = \frac{-\partial \tau_{eff}^s}{\partial \tau^s} \frac{1}{\tau_0} = \frac{-\tau^s}{|\tau^s|} \frac{1}{\tau_0} = \frac{-sgn(\tau^s)}{\tau_0} \quad \text{si} \quad \tau_{eff}^s > 0 \ , \ 0 \text{ sinon}$$

$$\frac{\partial y^s}{\partial \tau^s} = -(y^s)^2 \frac{2\pi}{\mu b} \frac{\partial \tau_{eff}^s}{\partial \tau^s} = -(y^s)^2 \frac{2\pi}{\mu b} sgn(\tau^s)$$

$$\frac{\partial R_i^{(1)}}{\partial \Delta\omega^s} = \sum_{r=1,12} \mu_i^r \frac{\partial \Delta\gamma^r}{\partial \Delta\omega^s}$$

avec $\frac{\partial \Delta\gamma^r}{\partial \Delta\omega^s} = (\Delta\gamma^r)^2 \left[\frac{1}{(\Delta\gamma_{nuc}^r)^2} \frac{\partial \Delta\gamma_{nuc}^r}{\partial \Delta\omega^s} + \frac{1}{(\Delta\gamma_{prob}^r)^2} \frac{\partial \Delta\gamma_{prob}^r}{\partial \Delta\omega^s} \right]$

Les différents termes intervenant ci-dessus sont :

$$\frac{\partial \Delta\gamma_{nuc}^r}{\partial \Delta\omega^s} = \Delta\gamma_{nuc}^r \left[\frac{1}{l^r} \frac{\partial l^r}{\partial \Delta\omega^s} - \frac{1}{k_b T} \frac{\partial \Delta G}{\partial \Delta\omega^s} \right]$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

avec

$$\frac{\partial l^r}{\partial \Delta\omega^s}=\frac{\partial \lambda^r}{\partial \Delta\omega^s}-2\frac{\partial \alpha^r_{AT}}{\partial \Delta\omega^s}R^r\ -2\alpha^r_{AT}\frac{\partial R^r}{\partial \Delta\omega^s}\qquad si\ \lambda^s(\boldsymbol{\omega})-2\alpha^s_{AT}(\boldsymbol{\omega})R^s(\boldsymbol{\omega})>l_c\,;0\ sinon$$

Le calcul de $\frac{\partial \lambda^r}{\partial \Delta\omega^s}$ se décline en deux possibilités :

si $min\left(\frac{\sqrt{\omega^r_{tot}(\boldsymbol{\omega})}}{b}\,;(D+2R^r(\boldsymbol{\omega}))\frac{\omega^r_{tot}(\boldsymbol{\omega})}{b^2}\right)=\frac{\sqrt{\omega^r_{tot}(\boldsymbol{\omega})}}{b}$ alors $\frac{\partial \lambda^r}{\partial \Delta\omega^s}=\frac{-b}{2(\omega^r)^{3/2}}(1-\delta_{rs})$

sinon $\frac{\partial \lambda^r}{\partial \Delta\omega^s}=\frac{-b^2}{(D+2R^r)^2(\omega^r_{tot})^2}\left[2\omega^r_{tot}\frac{\partial R^r}{\partial \Delta\omega^s}+(D+2R^r)(1-\delta_{rs})\right]$

$$\frac{\partial \alpha^r_{AT}}{\partial \Delta\omega_s}=\frac{1}{2\alpha^r_{AT}}\frac{1-\delta_{rs}}{\omega^r_{tot}}\left[h^{rs}-(\alpha^r_{AT})^2\right]$$

$$\frac{\partial R^r}{\partial \Delta\omega_s}=\frac{2R^r}{\Delta G_0-\Delta G_{app}}\frac{\partial \Delta G_{app}}{\partial \Delta\omega_s}$$

$$\frac{\partial \Delta G_{app}}{\partial \Delta\omega_s}=\frac{-kT(1-\delta_{rs})}{2\omega^r_{tot}}$$

$$\frac{\partial \Delta G}{\partial \Delta\omega^s}=\frac{\Delta G_0}{2\sqrt{\tau_0\tau_{eff}}}\frac{\partial \tau^r_c}{\partial \Delta\omega^s}$$

$$\frac{\partial \Delta\gamma^r_{prob}}{\partial \Delta\omega^s}=-\Delta\gamma^r_{prob}\frac{n}{|\tau^r_c|}\frac{\partial \tau^r_c}{\partial \Delta\omega^s}$$

avec $\frac{\partial \tau^r_c}{\partial \Delta\omega^s}=\frac{\tau^r_{LR}\frac{\partial \tau^r_{LR}}{\partial \Delta\omega^s}+\tau^r_{\mathrm{LT}}\frac{\partial \tau^r_{\mathrm{LT}}}{\partial \Delta\omega^s}}{\tau^r_c-\tau_F}$

$$\frac{\partial \tau^r_{LR}}{\partial \Delta\omega^s}=\delta_{rs}\frac{\mu}{2}\sqrt{\frac{h^{rr}}{\omega^r}}=\frac{\delta_{rs}}{2}\frac{\tau^r_{LR}}{\omega^r}$$

$$\frac{\partial \tau^r_{\mathrm{LT}}}{\partial \Delta\omega^s}=\frac{\partial \alpha^r_{AT}}{\partial \Delta\omega_s}\frac{\tau^r_{\mathrm{LT}}}{\alpha^r_{AT}}-\mu b\frac{\alpha^r_{AT}}{(\lambda^r)^2}\frac{\partial \lambda^r}{\partial \Delta\omega^s}+2\mu b\frac{\alpha^r_{AT}}{(2\alpha^r_{AT}R^r+L_c)^2}\left(\frac{\partial \alpha^r_{AT}}{\partial \Delta\omega_s}R^r+\alpha^r_{AT}\frac{\partial R^r}{\partial \Delta\omega_s}\right)\ \text{si}\ \tau^s_{\mathrm{LT}}>0\ ,\ 0$$

sinon

$$\boxed{\frac{\partial R^{(2)}_r}{\partial \Delta\omega^s}=\delta_{rs}-\Delta p_r\frac{\partial H_r}{\partial \Delta\omega_s}-H_r\frac{\partial \Delta p_r}{\partial \Delta\omega_s}}$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

$\frac{\partial \Delta p_r}{\partial \Delta \omega_s} = \frac{\gamma^r}{|\gamma^r|} \frac{\partial \Delta \gamma^r}{\partial \Delta \omega_s}$ $\frac{\partial \Delta \gamma^s}{\partial \Delta \omega_s}$ a déjà été calculé précédemment.

Il reste à calculer :

$$\frac{\partial H_r}{\partial \Delta \omega_s} = \frac{\partial c_{eff}^r}{\partial \Delta \omega_s} \frac{\sqrt{h^{rr} \omega^r}}{K_{self}} + \frac{c_{eff}^r \sqrt{h^{rr}} \delta_{rs}}{2\sqrt{\omega^r} K_{self}} - \frac{\partial y^r}{\partial \Delta \omega_s} \frac{\omega^s}{b} - \frac{y^r}{b} \delta_{rs}$$

$$\frac{+1}{b K_f} \left(\frac{\partial c_{eff}^r}{\partial \Delta \omega_s} \alpha_{AT}^r \lambda^r \omega_{tot}^r + c_{eff}^r \frac{\partial \alpha_{AT}^r}{\partial \Delta \omega_s} \lambda^r \omega_{tot}^r + c_{eff}^r \alpha_{AT}^r \frac{\partial \lambda^r}{\partial \Delta \omega_s} \omega_{tot}^r + c_{eff}^r \alpha_{AT}^r \lambda^r (1 - \delta_{rs}) \right)$$

avec :

$$\frac{\partial c_{eff}^s}{\partial \Delta \omega^s} = \frac{1}{\tau_0} \frac{\partial \tau_c^r}{\partial \Delta \omega^s}$$

$$\frac{\partial y^r}{\partial \Delta \omega^s} = (y^r)^2 \frac{2\pi}{\mu b} \frac{\partial \tau_c^r}{\partial \Delta \omega^s}$$

3.2.5 Algorithme d'intégration implicite en grandes déformations

A partir de $\boldsymbol{U}_n$ et $\Delta \boldsymbol{U}$, on peut calculer $\boldsymbol{F}_n$, $\Delta \boldsymbol{F}$ et $\boldsymbol{F}_{n+1} = \Delta \boldsymbol{F} \boldsymbol{F}_n$. de plus, on connait les résultats de l'incrément de temps n : contraintes de Cauchy $\boldsymbol{\sigma}_n$ variables internes.

Il s'agit de déterminer $\boldsymbol{\sigma}_{n+1}$ et les variables internes $(\gamma_s)_{n+1}$ (ou $(\alpha_s)_{n+1}$) solutions du système de $6 + n_s$ équations :

$$\boldsymbol{S}_{n+1} = \boldsymbol{\Lambda} . \frac{1}{2} (\boldsymbol{F}_{n+1}^{e\,T} \boldsymbol{F}_{n+1}^e - \boldsymbol{I}_d)$$

et pour chaque système de glissement :

$(\Delta \gamma_s)_{n+1} = \Delta t \, g(\tau_s, \alpha_s, \gamma_s, p_s)$ ou $\Delta \alpha_s = \Delta t \, h(\tau_s, \alpha_s, \gamma_s, p_s)$

avec $\tau_s(\boldsymbol{S}) = \boldsymbol{M}_{n+1} : \boldsymbol{m}_s \otimes \boldsymbol{n}_s = \left(\boldsymbol{F}^{eT}_{n+1} \boldsymbol{F}^e_{n+1} (\boldsymbol{S})_{n+1} \right) : \boldsymbol{m}_s \otimes \boldsymbol{n}_s = \left[\left(2 \boldsymbol{\Lambda}^{-1} \boldsymbol{S} + \boldsymbol{I}_d \right) \boldsymbol{S} \right] : \boldsymbol{m}_s \otimes \boldsymbol{n}_s$

en effet suivant 7et 13, on considère ici des contraintes de Piola-Kirchhoff mesurées à partir de la configuration intermédiaire, donc seule la partie élastique du gradient de transformation intervient.

$$\boldsymbol{F}^{\mathbf{e}}_{n+1} = \boldsymbol{F}_{n+1} (\boldsymbol{F}^p_{n+1})^{-1} = \Delta \boldsymbol{F} \, \boldsymbol{F}^{\mathbf{e}}_n (\Delta \boldsymbol{F}^p)^{-1}$$

Plusieurs choix sont possibles pour le calcul de $(\Delta \boldsymbol{F}^p)^{-1}$:

- $(\Delta \boldsymbol{F}^p)^{-1} = \exp\left(-\sum \Delta \gamma_s (\boldsymbol{S}, \omega_s) \boldsymbol{m}_s \otimes \boldsymbol{n}_s \right)$ 12 qui a l'avantage de conserver la trace nulle des déformations plastiques, mais qui peut présenter un surcoût ;

- $(\Delta \boldsymbol{F}^p)^{-1} = \left(\boldsymbol{I}_d + \sum \Delta \gamma_s (\boldsymbol{S}, \omega_s) \boldsymbol{m}_s \otimes \boldsymbol{n}_s \right)^{-1} \left(det \left(\boldsymbol{I}_d + \sum \Delta \gamma_s (\boldsymbol{S}, \omega_s) \boldsymbol{m}_s \otimes \boldsymbol{n}_s \right) \right)^{-\frac{1}{3}}$ 13

- $(\Delta \boldsymbol{F}^p)^{-1} = \left(\boldsymbol{I}_d - \sum \Delta \gamma_s (\boldsymbol{S}, \omega_s) \boldsymbol{m}_s \otimes \boldsymbol{n}_s \right) \left(det \left(\boldsymbol{I}_d - \sum \Delta \gamma_s (\boldsymbol{S}, \omega_s) \boldsymbol{m}_s \otimes \boldsymbol{n}_s \right) \right)^{-\frac{1}{3}}$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Les gradients F^e sont stockés comme variables internes (en ôtant le tenseur identité)

En post-traitement on calcule les contraintes de Cauchy : $\sigma_{n+1}=\frac{1}{det\, F^e_{n+1}} F^e_{n+1}(S)_{n+1} F^{e\,T}_{n+1}$

ou les contraintes de Kirchhoff : $\tau_{n+1}=det\, F_{n+1}\sigma_{n+1}=F^e_{n+1}(S)_{n+1} F^{e\,T}_{n+1}$

car $det\, F_{n+1}=det\, F^e_{n+1} det\, F^p_{n+1}=det\, F^e_{n+1}$

Le problème à résoudre en chaque point d'intégration a donc pour but, connaissant les variables internes. $(\gamma_s)_n$ $(\alpha_s)_n$ $(p_s)_n$ F^e_n de déterminer σ_{n+1} et les variables internes $(\gamma_s)_{n+1}$ ($(\alpha_s)_{n+1}$, F^e_{n+1}) solutions du système à $6+n_s$ équations :

$$R_1=(\Lambda)^{-1} S_{n+1}-\frac{1}{2}(F^{e\,T}_{n+1} F^e_{n+1}-I_d)=0$$

et pour chaque système de glissement :

$R_2=(\Delta\gamma_s)_{n+1}-\Delta t\, g(\tau_s,\alpha_s,\gamma_s,p_s)=0$ ou $R_2=\Delta\alpha_s-\Delta t\cdot h(\tau_s,\alpha_s,\gamma_s,p_s)=0$

avec $\tau_s(S)=\left[\left(2\Lambda^{-1}S+I_d\right)S\right]:m_s\otimes n_s$

$F^e_{n+1}=\Delta F\, F^e_n\left(\Delta F^p\right)^{-1}$

Le résolution de ce système d'équation non linéaires peut se faire classiquement par l'algorithme de Newton [R5.03.14].

Le vecteur des inconnues est composé de : S_{n+1} (6 composantes) et $\Delta\alpha_s$ (n_s composantes).

Le calcul de la matrice jacobienne est décrit en annexe.

Initialisation,

L'algorithme est initialisé de la façon suivante :

On pose $F^{\text{e-trial}}_{n+1}=F_{n+1}(F^p_n)^{-1}$ ce qui revient à supposer que $F^p_{n+1}=\Delta F^p F^p_n=I_d F^p_n$

On calcule R_1 avec $F^{\text{e}}_{n+1}=F^{\text{e-trial}}_{n+1}$ puis toutes les équations R_2 Si aucun seuil n'est atteint, alors $F^{\text{e}}_{n+1}=F^{\text{e-trial}}_{n+1}$. Sinon il faut résoudre le système R_1 R_2 .

On peut choisir de garder comme variable interne le tenseur F^e_{n+1} . pour calculer plus aisément $F^e_{n+1}=\Delta F\, F^e_n\left(\Delta F^p\right)^{-1}$

En pratique on stocke F^e-I_d dans les variables internes $V(6+3*n_s+1)$ à $V(6+3*n_s+9)$

On conserve aussi F^p-I_d dans les variables internes $V(6+3*n_s+10)$ à $V(6+3*n_s+19)$

En post-traitement, les tenseurs m_s et n_s peuvent être mis à jour en utilisant la rotation issue de la décomposition polaire $F^e=R^e U^e$: $m_s=\mathrm{R}^{\text{e}} m_{s0}$ $n_s=\mathrm{R}^{\text{e}} n_{s0}$

Pour le calcul de la décomposition polaire, un algorithme exact est proposé dans 12.

Suivant 13, il peuvent aussi être transportés dans la configuration actuelle par :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

$\boldsymbol{m}_s = \boldsymbol{F}^e \boldsymbol{m}_{s0}$ et $\boldsymbol{n}_s = \mathbf{F}^{\mathbf{e}-\mathbf{T}} \boldsymbol{n}_{s0}$

3.2.6 Critères de convergence utilisés pour les résolutions implicites

Le critère d'arrêt des itérations porte sur la nullité relative du résidu : $\frac{\|R(Y)\|_h}{\|R_{ref}\|_h} < \varepsilon_c$. où ε_c représente la tolérance donnée par RESI_INTE_RELA . Le problème consiste à bien choisir R_{ref} ainsi que la norme $\|.\|_h$

Par homogénéité avec l'algorithme global de Newton utilisé dans STAT_NON_LINE [R5.03.02], et pour éviter des itérations inutiles quand le résidu initial est très petit, on choisit :

$$\frac{\max_{i=1,6}|R_i(Y)|}{\max_{i=1,6}|E_i^{tr}|} < \varepsilon_c \text{ avec } \boldsymbol{E}^{tr} = \boldsymbol{\Lambda}^{-1}\boldsymbol{\Sigma}^- + \Delta\boldsymbol{E} - \Delta\boldsymbol{E}^{th}$$

$$\frac{\max_{i=7,n}|R_i(Y)|}{\max_{i=7,n}|Y_i^- + \Delta Y_i|} < \varepsilon_c$$

Les deux critères précédents doivent être vérifiés pour obtenir la convergence.

Si la convergence n'est pas atteinte après le nombre maximum d'itérations, on teste également la stationnarité de la solution :

$$|Y_{k+1} - Y_k| < \varepsilon$$

La méthode utilisée permet un re-découpage local du pas de temps, soit systématique, soit en cas de non convergence (mot-clé ITER_INTE_PAS) : ceci permet une intégration plus facile, mais en perdant alors le bénéfice d'un opérateur tangent cohérent à l'issue de l'intégration du comportement.

Il est souvent préférable, en cas de non convergence locale au bout de ITER_INTE_MAXI itérations, de procéder à un redécoupage global du pas de temps (en utilisant DEFI_LIST_INST), afin de conserver une matrice tangente cohérente.

3.3 Résolution explicite

Une autre méthode de résolution, très simple à mettre en œuvre pour résoudre les équations différentielles du comportement monocristallin est la résolution explicite. Pour qu'elle soit efficace numériquement, il est indispensable de lui associer un contrôle de pas automatique. Comme dans [R5.03.14], on utilise la méthode de Runge et Kutta. Le calcul des variables internes à l'instant $t+h$ n'est fonction que des valeurs de leurs dérivés $\frac{dY}{dt} = F(Y,t)$:

$$\Delta Y = \begin{cases} \Delta\alpha_s \\ \Delta\gamma_s \\ \Delta p_s \\ \Delta E^{vp} \end{cases} = \begin{cases} h(\tau_s^-, \alpha_s^-, \gamma_s^-, p_s^-) \\ g(\tau_s^-, \alpha_s^-, \gamma_s^-, p_s^-) \\ |\Delta\gamma_s| = f(\tau_s^-, \alpha_s^-, \gamma_s^-, p_s^-) \\ \sum_s m_s \Delta\gamma_s \end{cases}$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

avec $\tau_s = \boldsymbol{\Sigma} : \boldsymbol{\mu}_s = \left(\boldsymbol{\Sigma}^- + \boldsymbol{\Lambda} \left(\Delta \boldsymbol{E} - \Delta \boldsymbol{E}^{th} - \Delta \boldsymbol{E}^{vp} \right) \right) : \boldsymbol{\mu}_s$

On intègre selon le schéma suivant :

$Y_{t+h} = Y^{(2)}$ si le critère de précision est satisfait

$Y^{(2)} = Y + \frac{h}{2} \left[F(Y,t) + F(Y^{(1)}, t+h) \right]$ avec $Y^{(1)} = Y + h F(Y,t)$

La différence entre $Y^{(2)}$ (schéma d'ordre 2) et $Y^{(1)}$ (schéma d'ordre 1, Euler) fournit une estimation de l'erreur d'intégration et permet de contrôler la taille du pas de temps h qui est initialisé à Δt_i pour la première tentative. La stratégie du contrôle du pas est définie à partir d'une norme de l'écart entre les deux méthodes d'intégration : $\left\| Y^{(2)} - Y^{(1)} \right\|$ et de la précision requise par l'utilisateur η (mot-clé : RESI_INTE_RELA). Le critère retenu est le suivant, où l'on note $Y = (y_1, y_2, ..., y_N)$:

$$\delta Y(t) = \sup_{j=1,N} \left\{ \frac{\left| y_i^{(2)} - y_i^{(1)} \right|}{max \left[\varepsilon, \left| y_j^{(t)} \right| \right]} \right\} < \eta$$

Le paramètre ε est fixé à 0,001. La précision d'intégration souhaitée η doit être cohérente avec le niveau de précision requis pour l'étape globale.

Si le critère n'est pas vérifié, le pas de temps est re-découpé selon une méthode heuristique (nombre de sous-pas défini par l'utilisateur via le mot-clé ITER_INTER_PAS). Lorsque le pas de temps devient trop faible ($h < 1.10^{-20}$), le calcul est arrêté avec un message d'erreur.

Algorithme d'intégration explicite en grandes déformations

A chaque instant et en chaque point d'intégration, on peut calculer $\boldsymbol{F}_n$, $\Delta \boldsymbol{F}$ et $\boldsymbol{F}_{n+1} = \Delta \boldsymbol{F} \boldsymbol{F}_n$. On connait les contraintes de Cauchy $\boldsymbol{\sigma}_n$ et les variables internes au temps n

Il s'agit de déterminer $\boldsymbol{S}$ et les variables internes (γ_s), (α_s) $\boldsymbol{F}^p$ au temps $n+1$. On peut utiliser pour ce faire une méthode de Runge-Kutta [cf. R5.03.14] sur le système d'équations différentielles :

$$\dot{\boldsymbol{F}}^p = \left(\sum_s \dot{\gamma}_s \boldsymbol{m}_s \otimes \boldsymbol{n}_s \right) \boldsymbol{F}^p$$

$\dot{\gamma}_s = g(\tau_s, \alpha_s, \gamma_s, p_s)$ pour chaque système de glissement

$\dot{\alpha}_s = h(\tau_s, \alpha_s, \gamma_s, p_s)$ pour chaque système de glissement

avec : $\tau_s = \boldsymbol{M} : \boldsymbol{m}_s \otimes \boldsymbol{n}_s = \left(\boldsymbol{F}^{eT} \boldsymbol{F}^e \boldsymbol{S} \right) : \boldsymbol{m}_s \otimes \boldsymbol{n}_s = \left[\left(2 \boldsymbol{\Lambda}^{-1} \boldsymbol{S} + \boldsymbol{I}_d \right) \boldsymbol{S} \right] : \boldsymbol{m}_s \otimes \boldsymbol{n}_s$

$(\boldsymbol{S}) = \boldsymbol{\Lambda} . \frac{1}{2} (\boldsymbol{F}^{eT} \boldsymbol{F}^e - \boldsymbol{I}_d)$ et $\boldsymbol{F}^e = \boldsymbol{F} (\boldsymbol{F}^p)^{-1}$

4 Variables internes

4.1 Cas du monocristal

Les variables internes dans *Code_Aster* sont dénommées V_1, V_2, ... V_p.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Les six premières sont les 6 composantes de la déformation visco-plastique : $\boldsymbol{E}_{ij}^{vp}$:

$$\boldsymbol{E}^{vp}=\sum_{t}\left(\Delta \boldsymbol{E}^{vp}\right) \text{ avec } \Delta \boldsymbol{E}^{vp}=\sum_{s}\boldsymbol{\mu}_{s}\Delta\gamma_{s}$$

$$V_1=E_{xx}^{vp},\ V_2=E_{yy}^{vp},\ V_3=E_{zz}^{vp},\ V_4=\sqrt{(2)}\,E_{xy}^{vp},\ V_5=\sqrt{(2)}\,E_{xz}^{vp},\ V_6=\sqrt{(2)}\,E_{yz}^{vp}$$

V_7 , V_8 , V_9 sont les valeurs de α_1 γ_1 p_1 pour le système de glissement $s=1$
V_{10} , V_{11} , V_{12} correspondent au système $s=2$, et ainsi de suite,où :

- α_s représente la variable cinématique du système s dans le cas des modèles phénoménologiques, et la densité de dislocations dans un modèle issu de la DD ;
- γ_s représente le glissement plastique du système s
- p_1 représente le glissement plastique cumulé du système s

Prise en compte de l'irradiation :

- dans le cas DD_CC_IRRA, il faut ajouter $n_{irra}=12$ variables internes : V_{6+3n_s+1} à V_{6+3n_s+12} contiennent pour chaque système de glissement la densité de dislocations liée à l'irradiation ρ_s^{irr}
- dans le cas DD_CFC_IRRA, il faut ajouter $n_{irra}=24$ variables internes : V_{6+3n_s+1} à V_{6+3n_s+12} contiennent pour chaque système de glissement $\rho_s^{loops}*b^2$ V_{6+3n_s+13} à V_{6+3n_s+24} contiennent pour chaque système de glissement ϕ_s^{voids}

On stocke ensuite les cissions pour chaque système de glissement : τ_1 , … τ_{n_s}

Dans le cas où on prend en compte la rotation du réseau cristallin, il faut ajouter $n_{rota}=16$ variables internes :

- V_{6+3n_s+1} à V_{6+3n_s+9} sont les 9 composantes de la matrice de rotation $\mathbf{Q}$,
- V_{6+3n_s+10} à V_{6+3n_s+12} sont les 3 composantes de $\Delta\boldsymbol{\omega}^{p}$,
- V_{6+3n_s+13} à V_{6+3n_s+15} sont les 3 composantes de $\Delta\boldsymbol{\omega}^{e}$,
- V_{6+3n_s+16} représente Θ

L'antépénultième variable interne est la contrainte de clivage : $\max_{s}\left(\Sigma.n\right):n$
L'avant dernière variable interne contient la déformation plastique cumulée globale, définie par :

$$V_{p-1}=\sum\Delta E_{eq}^{vp} \text{ avec } \Delta E_{eq}^{vp}=\sqrt{\frac{2}{3}\left(\Delta\mathbf{E}^{vp}:\Delta\mathbf{E}^{vp}\right)}$$

La dernière variable interne, Vp , ($p=6+3n_s+n_{rota}+3$, n_s étant le nombre total de systèmes de glissement) est un indicateur de plasticité (seuil dépassé en au moins un système de glissement au pas de temps courant). S'il est nul, il n'y a pas eu d'accroissement de variables internes à l'instant courant. Sinon, il contient le nombre d'itérations de Newton local (pour une résolution implicite) qui ont été nécessaires pour obtenir la convergence.

4.2 Cas du polycristal

Les variables internes dans *Code_Aster* sont dénommées V_1 , V_2 ,… V_p .

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Le nombre de variables internes est $p=7+6\mathrm{m}+\sum_{g=1,m}\left(3\mathrm{n}_{s(g)}\right)+6\mathrm{m}+1$, m étant le nombre de phases et $n_s(g)$ étant le nombre de systèmes de glissement de la phase g) .
Dans le cas où on prend en compte l'irradiation, le nombre total de variables internes est :
$p=7+6\mathrm{m}+\sum_{g=1,m}\left(3\mathrm{n}_{s(g)}\right)+12\mathrm{m}+6\mathrm{m}+1$

- Les six premières variables internes sont les composantes de la déformation viscoplastique macroscopique $\boldsymbol{E^{vp}}$:
  $V_1=E_{xx}^{vp}$, $V_2=E_{yy}^{vp}$, $V_3=E_{zz}^{vp}$, $V_4=\sqrt{(2)}\,E_{xy}^{vp}$, $V_5=\sqrt{(2)}E_{xz}^{vp}$, $V_6=\sqrt{(2)}E_{yz}^{vp}$;
- la septième est la déformation viscoplastique équivalente cumulée macroscopique P :
  $V_7=\sum\Delta E_{eq}^{vp}$ avec $\Delta E_{eq}^{vp}=\sqrt{\frac{2}{3}\left(\Delta\mathbf{E}^{vp}:\Delta\mathbf{E}^{vp}\right)}$;
- puis, pour chaque phase, on trouve les 6 composantes des déformations viscoplastiques ou du tenseur β de la phase :
  $\left\{\varepsilon_{xx}^{vp}(g),\varepsilon_{yy}^{vp}(g),\varepsilon_{zz}^{vp}(g),\sqrt{(2)}\varepsilon_{xy}^{vp}(g),\sqrt{(2)}\varepsilon_{xz}^{vp}(g),\sqrt{(2)}\varepsilon_{yz}^{vp}(g)\right\}_{g=1,m}$;
- ensuite, pour chaque phase, et pour chaque système de glissement de la phase, on trouve les valeurs de α_s γ_s p_s ;
- si l'irradiation est prise en compte, pour chaque phase, on trouve les 12 densités de dislocation liées à l'irradiation ρ_{irr}^{s} ;
- puis, pour chaque phase, on trouve les 6 composantes des contraintes de la phase :
  $\left\{\sigma_{xx}(g),\sigma_{yy}(g),\sigma_{zz}(g),\sqrt{(2)}\sigma_{xy}(g),\sqrt{(2)}\sigma_{xz}(g),\sqrt{(2)}\sigma_{yz}(g)\right\}_{g=1,m}$;
- la dernière variable interne est un indicateur de plasticité (seuil dépassé en au moins un système de glissement au pas de temps courant).

5 Implantation numérique dans *Code_Aster*

D'une façon générale, les comportements monocristallins sont intégrés aux méthodes de Runge-Kutta pour l'intégration explicite, et à l'environnement « `plasti` » pour l'intégration implicite [R5.03.14]. Les tenseurs d'orientation des systèmes de glissement sont quant à eux tous définis dans la routine `LCMMSG` fournissant le tenseur en repère global pour le nième système de la famille fournie dont le nom est fourni par la routine appelante.

Pour ajouter un nouveau comportement de monocristal, ou simplement une nouvelle loi d'écoulement ou d'écrouissage, il convient de définir ses paramètres dans `DEFI_MATERIAU`. Suivant le cas (écoulement, écrouissage isotrope ou cinématique), il faut ajouter la lecture de ces paramètres dans les routines `LCMAFL`, `LCMAEI`, `LCMAEC`. Pour l'intégration, il suffit d'écrire la définition des accroissements de variables internes dans les routines `LCMMFE` (écoulement), `LCMMFC`(écrouissage cinématique) et `LCMMFI`(écrouissage isotrope), pour que l'intégration explicite fonctionne.

L'intégration implicite utilise également les routines `LCMMFE`, `LCMMFC` et `LCMMFI`. Elle demande en plus de définir les dérivées des équations par rapport aux différentes variables. Les dérivées sont à écrire dans les routines `LCMMJF` (dérivées de l 'équation d'écoulement), `LCMMJI` (dérivées de la relation d'écrouissage isotrope) et `LCMMJC` (dérivées de la relation d'écrouissage cinématique).

Pour plus de détails sur l'architecture de la résolution des comportements cristallins, voir [D5.04.02].

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

6 Utilisation

Ces modèles sont accessibles dans *Code_Aster* en **3D**, déformations planes (**D_PLAN**), contraintes planes (**C_PLAN,**par la méthode de De Borst [R5.03.03]) et axisymétrie (**AXIS**).

6.1 Cas du monocristal

Dans le cas de microstructures maillées, les différents grains d'un monocristal étant représentés par des groupes de mailles, il faut affecter les paramètres des matériaux et les comportements des monocristaux ainsi que leurs orientations aux différents grains.

Les valeurs des paramètres des relations de comportement sont fournies à l'aide de la commande DEFI_MATERIAU. Ceci se définit à partir des mots clés MONO_VISC1, MONO_VISC2, MONO_DD_KR, MONO_DD_CFC pour l'écoulement, MONO_ISOT1, MONO_ISOT2, MONO_DD_CFC pour l'écrouissage isotrope et MONO_CINE1, MONO_CINE2 pour l'écrouissage cinématique [U4.43.01]. Par exemple [V6.04.172] :

```
MATER1=DEFI_MATERIAU(
/ ELAS      =_F(  E=..., NU=...,
/ ELAS_ORTH=_F(  E_L=192500,
                 E_T=128900,
                 NU_LT=0.23,
                 G_LT=74520,),

# RELATIONS D'ECOULEMENT
  / MONO_VISC1=_F(N=10, K=40,C=6333),
  / MONO_VISC2=_F(N=10, K=40,C=6333,D=37,A=121),
  / MONO_DD_KR=_F(...),

# ECROUISSAGE ISOTROPE
 /  MONO_ISOT1=_F(R_0=75.5,Q=9.77,B=19.34,H=2.54),
 /  MONO_ISOT2=_F(R_0=75.5,Q1=9.77,B1=19.34,H=2.54,Q2=-33.27, B2=5.345,),

# ECROUISSAGE CINEMATIQUE

 / MONO_CINE1=_F(D=36.68),
/  MONO_CINE2=_F(D=36.68, GM=, PM=,),
                    );
```

On peut ainsi dissocier, au niveau des données, l'écoulement de l'écrouissage isotrope et de l'écrouissage cinématique.

Il faut maintenant définir le (ou les) type de monocristal étudié. Pour cela, on définit le comportement de façon externe à STAT_NON_LINE, par l'intermédiaire de l'opérateur DEFI_COMPOR, par exemple :

```
MONO1=DEFI_COMPOR(MONOCRISTAL =(_F( MATER=MATER1,
                                    ECOULEMENT=MONO_VISC1,
                                    MONO_ISOT=MONO_ISOT1,
                                    MONO_CINE=MONO_CINE1,
                                    FAMI_SYST_GLIS=('CUBIQUE1',),

                                  _F( MATER=MATER1,
                                    ECOULEMENT=MONO_VISC1,
                                    MONO_ISOT=MONO_ISOT2,
                                    MONO_CINE=MONO_CINE2,
                                    FAMI_SYST_GLIS='CUBIQUE2',),
                                    ),
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
                    _F( MATER=MATER2,
                        ECOULEMENT=MONO_VISC1,
                        MONO_ISOT=MONO_ISOT2,
                        MONO_CINE=MONO_CINE2,
                        FAMI_SYST_GLIS='PRISMATIQUE',),

                    _F( MATER=MATER2,
                        ECOULEMENT=MONO_VISC1,
                        MONO_ISOT=MONO_ISOT2,
                        MONO_CINE=MONO_CINE2,
                        FAMI_SYST_GLIS='BCC24',),
                        ),
                        )
```

L'opérateur DEFI_COMPOR calcule le nombre total de variables internes associé au monocristal.

Enfin, pour réaliser un calcul de micro-structure, il faut donner, grain par grain, ou par groupe de mailles (représentant des ensembles de grains) une orientation, à l'aide du mot-clé MASSIF de AFFE_CARA_ELEM. Par exemple :

```
ORIELEM = AFFE_CARA_ELEM ( MODELE = MO_MECA,MASSIF =(
          _F ( GROUP_MA='GRAIN1', ANGL_REP=(348.0,24.0,172.0),),
          _F ( GROUP_MA='GRAIN2', ANGL_REP=( 327.0, 126.0, 335.0),),
          _F ( GROUP_MA='GRAIN3', ANGL_REP=( 235.0, 7.0, 184.0),),
          _F ( GROUP_MA='GRAIN4', ANGL_REP=( 72.0, 338.0, 73.0), ,
…)
```

Remarque 1 :
Les orientations des systèmes de glissement peuvent être renseignées en angles nautiques sous le mot clé de ANGL_REP ou en angles d'Euler sous le mot clé ANGL_EULER.

Remarque 2 :

- *Pour un même monocristal, les valeurs des paramètres peuvent être différentes d'une famille de systèmes de glissement à l'autre. C'est pourquoi on peut définir un matériau différent par occurrence du mot-clé facteur MONOCRISTAL . Mais dans ce cas, comment fournir transmettre à STAT_NON_LINE l'information stipulant qu'en un point de gauss (tous ceux du groupe de mailles concerné), on a plusieurs matériaux présents ? Ceci est possible grâce à une évolution de AFFE_MATERIAU [U4.43.03] et de la structure de données matériau [D4.06.18]) :*

```
MAT=AFFE_MATERIAU(MAILLAGE=MAIL,
                  AFFE    =_F(GROUP_MA='GRAIN1',

MATER=(MATER1,MATER2),),
                  );
```

Les autres données du calcul sont identiques à un calcul de structure habituel.

Enfin, dans STAT_NON_LINE, le comportement issu de DEFI_COMPOR est fourni, sous le mot clé COMPORTEMENT via le mot clé COMPOR, obligatoire avec le mot-clé RELATION='MONOCRISTAL'.
COMPORTEMENT = _F(RELATION ='MONOCRISTAL', COMPOR = COMP1

Précisions que pour l'intégration explicite, (ALGO_INTE='RUNGE_KUTTA'), il est inutile de demander la réactualisation de la matrice tangente puisque celle-ci n'est pas calculée. Pour débuter des itérations de Newton de l'algorithme global, il peut être utile de spécifier PREDICTION='EXTRAPOLE' [U4.51.03].

On pourra trouver des exemples d'utilisation dans les tests : SSNV171, SSNV172, SSNV194.

6.2 Cas du polycristal

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Dans le cas de polycristaux multiphasés, chaque phase correspond à un monocristal. On utilisera donc les paramètres définis précédemment dans DEFI_MATERIAU pour le monocristal. Ici, il s'agit de définir, pour chaque phase, l'orientation, la fraction volumique, le monocristal utilisé, et le type de loi de localisation. Ceci est effectué sous le mot-clé facteur POLYCRISTAL de DEFI_COMPOR.

```
MONO1=DEFI_COMPOR(MONOCRISTAL=_F(MATER=MATPOLY,
                                 ECOULEMENT='MONO_VISC2',
                                 MONO_ISOT='MONO_ISOT2',
                                 MONO_CINE='MONO_CINE1',
                                 ELAS='ELAS',
                                 FAMI_SYST_GLIS='OCTAEDRIQUE',),);

POLY1=DEFI_COMPOR(POLYCRISTAL=(_F(MONOCRISTAL=MONO1,
                                  FRAC_VOL=0.025,
                                  ANGL_REP=(-149.676,15.61819,154.676,),),
                               _F(MONOCRISTAL=MONO1,
                                  FRAC_VOL=0.025,
                                  ANGL_REP=(-150.646,33.864,55.646,),),
                               _F(MONOCRISTAL=MONO1,
                                  FRAC_VOL=0.025,
                                  ANGL_REP=(-137.138,41.5917,142.138,),),
                              ……
                               _F(MONOCRISTAL=MONO1,
                                  FRAC_VOL=0.025,
                                  ANGL_REP=(-481.729,35.46958,188.729,),),),),
                  LOCALISATION='BETA',
                   MU_LOCA=80000,,
                  DL=321.5,
                  DA=0.216,);
```

Le mot-clé POLYCRISTAL permet de définir chaque phase par la donnée d'une orientation, (fournie par ANGL_REP ou ANGL_EULER), d'une fraction volumique, d'un mono-cristal (c'est-à-dire un modèle de comportement et des systèmes de glissement).
Le mot-clé LOCALISATION permet de choisir la méthode de localisation pour l'ensemble des phases du polycristal.

Le mot clé MASSIF permet de choisir une orientation globale du polycristal sous le mot clé ANGL_REP si c'est en angles nautiques ou bien sous le mot clé ANGL_EULER si c'est en angles d'Euler.

Enfin, dans STAT_NON_LINE, le comportement issu de DEFI_COMPOR est fourni, sous le mot clé COMPORTEMENT via le mot clé COMPOR, obligatoire avec le mot-clé RELATION='POLYCRISTAL'.

```
COMPORTEMENT = _F(   RELATION ='POLYCRISTAL',
                    COMPOR = COMP1 )
```

6.3 Exemple

A titre d'exemple de mis en œuvre, on présente ici succinctement un calcul d'agrégat, de forme cubique (volume élémentaire) comprenant 100 grains monocristallins, définis chacun par un groupe de mailles. Le nombre total d'éléments est 86751. Avec des mailles d'ordre 1 (TETRA4) il comporte 15940 nœuds. Avec des mailles d'ordre 2 (TETRA10), il en comporte 121534.
Le chargement consiste en une déformation homogène, appliquée par l'intermédiaire d'un déplacement normal imposé sur une face du cube (direction z). On atteint une déformation de 4% en 1s et 50 incréments.

```
ACIER=DEFI_MATERIAU(ELAS=_F(E =145200.0,NU=0.3,),
                    MONO_VISC1=_F( N=10., K=40.,C=10.,),
                    MONO_ISOT2=_F(R_0=75.5,
                                  B1 =19.34,
                                  B2 =5.345,
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
                                        Q1 =9.77,
                                        Q2 =33.27,
                                        H=0.5),
                              MONO_CINE1=_F(D=36.68,),);

COEF=DEFI_FONCTION(NOM_PARA ='INST', VALE =(0.0,0.0,1.0,1.0,),);
MAT=AFFE_MATERIAU(MAILLAGE=MAIL, AFFE=_F(TOUT ="OUI", MATER=(ACIER),),);

COMPORT=DEFI_COMPOR(MONOCRISTAL=( _F(MATER =ACIER,
    ECOULEMENT="MONO_VISC1",
    MONO_ISOT ="MONO_ISOT2",
    MONO_CINE ="MONO_CINE1",
    ELAS="ELAS",
    FAMI_SYST_GLIS='OCTAEDRIQUE',),
),);


ORIELEM = AFFE_CARA_ELEM ( MODELE = MO_MECA, MASSIF =(
_F ( GROUP_MA='GRAIN1', ANGL_REP=(348.0,24.0,172.0), ),
_F ( GROUP_MA='GRAIN2', ANGL_REP=( 327.0, 126.0, 335.0), ),
_F ( GROUP_MA='GRAIN3', ANGL_REP=( 235.0, 7.0, 184.0), ),
……..
_F ( GROUP_MA='GRAIN99', ANGL_REP=( 201.0, 198.0, 247.0), ),
_F ( GROUP_MA='GRAIN100', ANGL_REP=( 84.0, 349.0, 233.0), ),
))

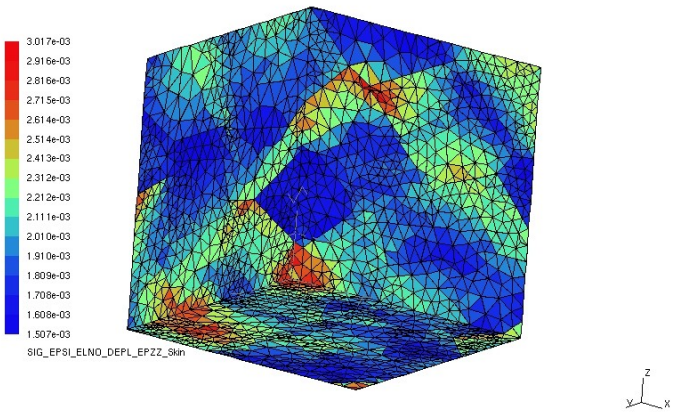
FO_UZ = DEFI_FONCTION ( NOM_PARA = 'INST',VALE = ( 0.0, 0.0, 1.0, 0.04, ),)

CHME4=AFFE_CHAR_MECA_F(MODELE=MO_MECA, DDL_IMPO=_F(GROUP_NO='HAUT', DZ=FO_UZ,),)

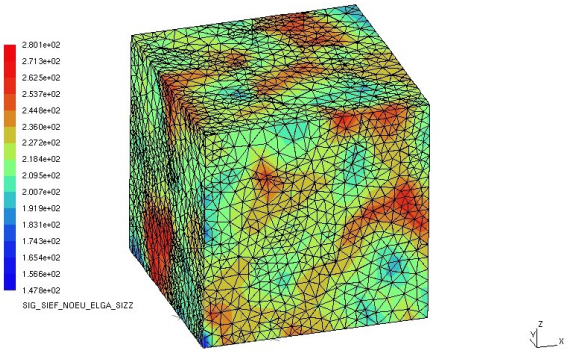
LINST = DEFI_LIST_REEL (DEBUT= 0.,INTERVALLE =(_F ( JUSQU_A = 1.,NOMBRE= 50 ),) )

SIG=STAT_NON_LINE(MODELE =MO_MECA,CARA_ELEM=ORIELEM,CHAM_MATER =MAT,
       EXCIT=(_F(CHARGE=CHME1),
              _F(CHARGE=CHME2),
              _F(CHARGE=CHME3),
              _F(CHARGE=CHME4),),
       COMPORTEMENT=_F(  RELATION ='MONOCRISTAL',COMPOR =COMPORT),
       INCREMENT=_F(LIST_INST=LINST),))
```

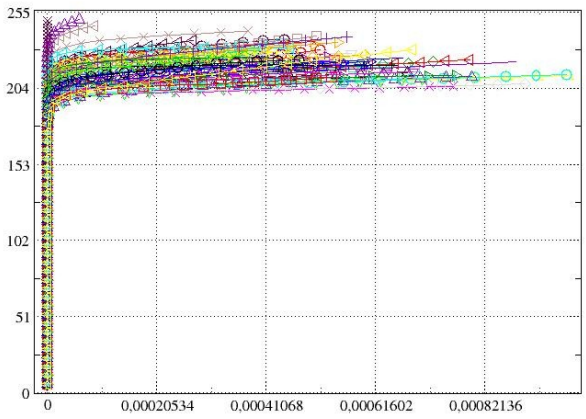
Les figures suivantes représentent des isovaleurs des déformations les contraintes suivant z. On note la non homogénéité des valeurs, et on peut même discerner le contour des grains.



Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)



Pour pouvoir exploiter ce type de résultats, on peut par exemple calculer des champs moyens par grains. Sur la figure suivante, on a représenté les contraintes équivalentes en fonction des déformations plastiques équivalentes pour l'ensemble des grains.



Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

7 Bibliographie

1 MERIC L., CAILLETAUD G. : « single crystal modeling fort structural calculations » in Journal of Engineering Material and Technology, janvier 1991, vol 113, pp171-182.

2 LECLERCQ S., DIARD, O., PROIX J.M. : “Etude d’impact de l’implantation d’une bibliothèque de lois de comportement et de règles de transition d’échelles » Note EDF R&D HT-26/03/053/A.

3 CAILLETAUD G. : “A micromechanical approach to inelastic behaviour of metals”, Int. J. of Plasticity, 8, pp. 55-73, 1992.

4 PILVIN P. : “The contribution of micromechanical approaches to the modelling of inelastic behaviour of polycrystals”, Int. Conf. on Biaxial / Multiaxial fatigue, France, ESIS/SF2M, pp. 31-46, 1994.

5 BERVEILLER M., ZAOUI A. : “An extension of the self-consistant scheme to plasticity flowing polycrystal” J. Mech. Phys. Solids, 6, pp. 325-344, 1979.

6 G.MONNET “A crystalline plasticity law for austenitic stainless steels”, Note EDF R&D H-B60-2008-04690-EN

7 N.Rupin « implementation of a new constitutive law based on dislocation dynamics for FCC materials» Note EDF-R&D : Note EDF R&D HT24-2010-01128-EN.

8 C.Petry “Procédure de génération de maillages d'agrégats polycristallins” Note EDF R&D H-T24-2008-03481

9 A.ZEGHADI : « Effet de la morphologie 3D et de la taille de grain sur le comportement mécanique d'agrégats polycristallins ». Thèse de l’Ecole des Mines de Paris, 2005.

10 N.Rupin : « Déformation à chaud de métaux biphasés. Modélisations théoriques et confrontations expérimentales ». Thèse de l'Ecole Polytechnique, 2007.

11 M.Fivel, S.Forest “Plasticité cristalline et transition d'échelle : cas du monocristal”. Techniques de l'ingénieur M4 016

12 « Computational methods for plasticity » Wiley 2008, EA de Souza Neto, D. Peric, DRJ. Owen

13 « A computational procedure for rate-independent crystal plasticity » L.ANAND & M.KOTHARI, J.Mech.Phys.Solids, vol 44, n°4, pp.525-558, 1996

14 C. Stolz. Milieux continus en transformations finies : hyperélasticité, rupture, élasto-plasticité. Les éditions de l’Ecole Polytechnique, 2002.

15 Homogénéisation en mécanique des matériaux”, tome 2. M.Bornert, P.Gilormini, T.Bertheau

16 J. SCHWARTZ : “Approche non locale en plasticité cristalline : application à l'étude du comportement mécanique de l'acier AISI 316LN en fatigue oligocyclique”. Thèse de l'Ecole Centrale de Paris, Juin 2011.

17 G.Monnet : “DELIVERABLE D1-2.9. Crystal plasticity constitutive law for irradiated RPV steel” Note EDF R&D H-T27-2011-02738-EN, Décembre 2011.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

8 Historique des versions du document

Version Aster	Auteur(s) ou contributeur(s), organisme	Description des modifications
7.4	J.M.PROIX, O.DIARD, T.KANIT EDF/R&D	Version initiale
8.4	J.EL-GHARIB,J.M.PROIX, EDF/R&D	Compléments
9.2	J.EL-GHARIB,J.M.PROIX, EDF/R&D	Ajout du comportement MONO_DD_KR.
9.4	J.EL-GHARIB,J.M.PROIX, EDF/R&D	Optimisation du comportement MONO_DD_KR.
10.2	J.M.PROIX	remarque sur la matrice d'interaction
10.3	N.RUPIN,J.M.PROIX, EDF/R&D	Ajout du comportement DD_CFC
10.5	N.RUPIN,J.M.PROIX, EDF/R&D	Ajout de la rotation du réseau cristallin
11.1	N.RUPIN, F.LATOURTE, J.M.PROIX, EDF/R&D	Ajout des grandes déformations
11.1	N.RUPIN,J.M.PROIX, EDF/R&D	Fiches 16373 (critère de convergence) et 17422 : correction variables internes du POLYCRISTAL.
11.1	P. DE BONNIERES, J.M. PROIX	Fiche 14586 : ajout du comportement DD_FAT
11.2	J.M. PROIX	Fiche 18398 : résorption de ALGO_C_PLAN
11.2	J.M. PROIX	Fiche 18692 ajout de DD_CC
11.3	J.M. PROIX	Fiche 19021 ajout des variables internes pour DD_CC_IRRA
12.2	J.M. PROIX	Fiches 22563 : simplification de la loi DD_CC et 22491 coefficient μ^{loca} pour l'homogénéisation,

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Annexe 1 Expression du Jacobien des équations élasto-visco-plastiques intégrées

Le système à résoudre est de la forme :

$$R(Y)=R(\Sigma,\Delta E^{vp},\alpha_s^+,\gamma_s^+,p_s^+)=\begin{Bmatrix} s(\Sigma,\Delta E^{vp},\alpha_s^+,\gamma_s^+,p_s^+) \\ e(\Sigma,\Delta E^{vp},\alpha_s^+,\gamma_s^+,p_s^+) \\ n_s\begin{Bmatrix} a(\Sigma,\Delta E^{vp},\alpha_s^+,\gamma_s^+,p_s^+) \\ g(\Sigma,\Delta E^{vp},\alpha_s^+,\gamma_s^+,p_s^+) \\ p(\Sigma,\Delta E^{vp},\alpha_s^+,\gamma_s^+,p_s^+) \end{Bmatrix} \end{Bmatrix}$$

$$=\begin{Bmatrix} \boldsymbol{\Lambda}^{-1}\boldsymbol{\Sigma}-(\boldsymbol{\Lambda}_-^{-1})\boldsymbol{\Sigma}^--(\Delta\boldsymbol{E}-\Delta\boldsymbol{E}^{th}-\Delta\boldsymbol{E}^{vp}) \\ \Delta\boldsymbol{E}^{vp}-\sum_s \boldsymbol{\mu_s}\Delta\gamma_s \\ n_s\begin{Bmatrix} \Delta\alpha_s-h(\tau_s^+,\alpha_s^+,\gamma_s^+,p_s^+) \\ \Delta\gamma_s-g(\tau_s^+,\alpha_s^+,\gamma_s^+,p_s^+) \\ \Delta p_s-f(\tau_s^+,\alpha_s^+,\gamma_s^+,p_s^+) \end{Bmatrix} \end{Bmatrix}$$

$$=0 \quad \text{avec} \quad \tau_s^+=\boldsymbol{\Sigma}^+:\boldsymbol{\mu_s}$$

Soit donc à évaluer les termes de l'hypermatrice jacobienne J à l'instant $t+\Delta t$

$$J=\begin{bmatrix} \frac{\partial s}{\partial\Delta\Sigma} & \frac{\partial s}{\partial\Delta E^{vp}} & \frac{\partial s}{\partial\Delta\alpha_s} & \frac{\partial s}{\partial\Delta\gamma_s} & \frac{\partial s}{\partial\Delta p_s} \\ \frac{\partial e}{\partial\Delta\Sigma} & \frac{\partial e}{\partial\Delta E^{vp}} & \frac{\partial e}{\partial\Delta\alpha_s} & \frac{\partial e}{\partial\Delta\gamma_s} & \frac{\partial e}{\partial\Delta p_s} \\ \frac{\partial a}{\partial\Delta\Sigma} & \frac{\partial a}{\partial\Delta E^{vp}} & \frac{\partial a}{\partial\Delta\alpha_s} & \frac{\partial a}{\partial\Delta\gamma_s} & \frac{\partial a}{\partial\Delta p_s} \\ \frac{\partial g}{\partial\Delta\Sigma} & \frac{\partial g}{\partial\Delta E^{vp}} & \frac{\partial g}{\partial\Delta\alpha_s} & \frac{\partial g}{\partial\Delta\gamma_s} & \frac{\partial g}{\partial\Delta p_s} \\ \frac{\partial p}{\partial\Delta\Sigma} & \frac{\partial p}{\partial\Delta E^{vp}} & \frac{\partial p}{\partial\Delta\alpha_s} & \frac{\partial p}{\partial\Delta\gamma_s} & \frac{\partial p}{\partial\Delta p_s} \end{bmatrix}$$

En ce qui concerne la première ligne de la matrice, indépendamment des équations d'écrouissage et d'écoulement, on a :

$$\frac{\partial s}{\partial\Delta\Sigma}=D^{-1} \quad \frac{\partial s}{\partial\Delta E^{vp}}=Id \quad \frac{\partial s}{\partial\Delta\alpha_s}= \quad \frac{\partial s}{\partial\Delta\gamma_s}=\frac{\partial s}{\partial\Delta p_s}=0$$

La deuxième ligne peut s'écrire également indépendamment de l'écoulement et des écrouissages :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

$$\frac{\partial e}{\partial \Delta\Sigma}=0 \quad \frac{\partial e}{\partial \Delta E^{vp}}=Id \quad \frac{\partial e}{\partial \Delta\alpha_s}=0 \quad \frac{\partial e}{\partial \Delta\gamma_s}=-\boldsymbol{\mu}_s \quad \frac{\partial e}{\partial \Delta p_s}=0$$

La première colonne des lignes correspondant aux équations (a), (g) et (p) s'écrit :

$$\frac{\partial a}{\partial \Delta\Sigma}=\frac{\partial a}{\partial \Delta\tau_s}\frac{\Delta\tau_s}{\Delta\Sigma}$$
$$\frac{\partial g}{\partial \Delta\Sigma}=\frac{\partial g}{\partial \Delta\tau_s}\frac{\Delta\tau_s}{\Delta\Sigma}$$
$$\frac{\partial p}{\partial \Delta\Sigma}=\frac{\partial p}{\partial \Delta\tau_s}\frac{\Delta\tau_s}{\Delta\Sigma}$$

avec

$$\frac{\Delta\tau_s}{\Delta\Sigma}=\left(\boldsymbol{\mu}_s\right)^T$$

La deuxième colonne est identiquement nulle (du fait de l'équation (e) : les relations d'écoulement et d'écrouissage ne peuvent s'exprimer qu'en fonction de $\Delta\gamma_s$ et non pas de ΔE^{vp} .

Le dernier bloc d'équations, dépend quant à lui des comportements choisis :

$$\begin{matrix}\frac{\partial a}{\partial \Delta\alpha_s} & \frac{\partial a}{\partial \Delta\gamma_s} & \frac{\partial a}{\partial \Delta p_s}\\ \frac{\partial g}{\partial \Delta\alpha_s} & \frac{\partial g}{\partial \Delta\gamma_s} & \frac{\partial g}{\partial \Delta p_s}\\ \frac{\partial p}{\partial \Delta\alpha_s} & \frac{\partial p}{\partial \Delta\gamma_s} & \frac{\partial p}{\partial \Delta p_s}\end{matrix}$$

Exemple

Choisissons la relation d'écoulement viscoplastique **MONO_VISC1**

$$(g)\ \Delta\gamma_s-\Delta p_s\frac{\tau_s-c\alpha_s}{|\tau_s-c\alpha_s|}=0$$
$$(p)\ \ \Delta p_s-\Delta t.\left\langle\frac{|\tau_s-c\alpha_s|-R_s(p_s)}{k}\right\rangle^n=0$$

avec l'écrouissage isotrope **MONO_ISOT1** : $R_s(p_s)=R_0+Q\left(\sum_{r=1}^{N}h_{sr}\left(1-e^{-bp_r}\right)\right)$, $h_{sr}=(1-\delta_{sr})+\delta_{sr}$

et un écrouissage cinématique défini par **MONO_CINE1**

$(a)\ \ \Delta\alpha_s-\Delta\gamma_s+d\alpha_s\,\Delta p_s=0$

alors :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

$$\frac{\partial a}{\partial \Delta\tau_s}=0$$
$$\frac{\partial g}{\partial \Delta\tau_s}=0$$
$$\frac{\partial p}{\partial \Delta\tau_s}=\frac{-n\Delta t}{K^n}\langle|\tau_s-c\alpha_s|-R_s(p_s)\rangle^{n-1}\frac{\tau_s-c\alpha}{|\tau_s-c\alpha_s|}$$
$$\frac{\partial a}{\partial \Delta\alpha_s} = 1+d\Delta p_s$$
$$\frac{\partial g}{\partial \Delta\alpha_s} = 0$$
$$\frac{\partial p}{\partial \Delta\alpha_s} = \frac{nc\,\Delta t}{K^n}\langle|\tau_s-c\alpha_s|-R_s(p_s)\rangle^{n-1}\frac{\tau_s-c\alpha_s}{|\tau_s-c\alpha_s|}$$
$$\frac{\partial a}{\partial \Delta\gamma_s} = -1$$
$$\frac{\partial g}{\partial \Delta\gamma_s} = 1$$
$$\frac{\partial p}{\partial \Delta\gamma_s} = 0$$
$$\frac{\partial a}{\partial \Delta p_s} = d\alpha_s$$
$$\frac{\partial g}{\partial \Delta p_s} = \frac{\tau_s-c\alpha_s}{|\tau_s-c\alpha_s|}$$
$$\frac{\partial p}{\partial \Delta p_s} = 1+\frac{n\Delta t}{K^n}\langle|\tau_s-c\alpha_s|-R_s(p_s)\rangle^{n-1}\frac{dR_s(p_s)}{d\Delta p_s}$$
$$\frac{dR_s(p_s)}{d\Delta p_s}=Qbh_{ss}e^{-bp_s}$$

et, concernant l'interaction entre systèmes de glissement, il y un seul terme non nul :

$$\frac{\partial p}{\partial \Delta p_r}=1+\frac{n\Delta t}{K^n}\langle|\tau_s-c\alpha_s|-R_s(p_s)\rangle^{n-1}\frac{dR_s(p_s)}{d\Delta p_r}$$
$$\frac{dR_s(p_s)}{d\Delta p_r}=Qbh_{sr}e^{-bp_r}$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Annexe 2 Évaluation de l'opérateur tangent cohérent

Il s'agit de trouver l'opérateur tangent cohérent, c'est à dire calculé à partir de la solution de $\left(R\left(Y\right)=0\right)$ en fin d'incrément. Pour une petite variation de R , en considérant cette fois ΔE comme variable et non comme paramètre, on obtient :

$$\frac{\partial R}{\partial \Sigma}\delta\Sigma+\frac{\partial R}{\partial \Delta E}\delta\Delta E+\frac{\partial R}{\partial \Delta \gamma_s}\delta\Delta\gamma_s=0$$

Ce système peut s'écrire :

$$\frac{\partial R}{\partial Y}\delta\left(Y\right)=X \text{ avec } Y=\begin{bmatrix}\Sigma\\ \Delta\gamma_s\end{bmatrix} \text{ et } X=\begin{bmatrix}\delta\Delta E\\ 0\end{bmatrix}$$

En écrivant la matrice jacobienne sous la forme :

$$J.\delta Y=\begin{bmatrix}Y_0 & Y_1\\ Y_2 & Y_3\end{bmatrix}\begin{bmatrix}\Sigma\\ \Delta Z\end{bmatrix}$$

Avec :

$$Y_0=\mathbf{\Lambda^{-1}}$$
$$\Delta Z=\left(\Delta\gamma_s\right)\times n_s$$

Les sous-matrices ont pour dimensions :

$$\dim\left(Y_0=\mathbf{\Lambda^{-1}}\right)=\left[6,6\right]$$
$$\dim Y_1=\left[6,n_s\right]$$
$$\dim Y_2=\left[n_s,6\right]$$
$$\dim Y_3=\left[n_s,n_s\right]$$

En opérant par éliminations et substitutions successives, le troisième bloc du système d'équation donne :

$$\Delta Z=-\left(Y_3\right)^{-1}Y_2\Sigma$$
$$\left(Y_0-Y_1\left(Y_3\right)^{-1}Y_2\right)\Sigma=\Delta E$$

l'opérateur tangent recherché peut donc s'écrire directement :

$$\left(\frac{\partial\Sigma}{\partial E}\right)_{t+\Delta t}=\left(\frac{\partial\Sigma}{\partial\Delta E}\right)_{t+\Delta t}=\left(Y_0-Y_1Y_3^{-1}Y_2\right)^{-1}$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Annexe 3 Matrices tangentes en grandes déformations :

Système local résolu par Newton :

$$\boldsymbol{R_1}(\boldsymbol{S},\Delta\omega)=\boldsymbol{\Lambda}^{-1}.\boldsymbol{S}-\frac{1}{2}\left(\boldsymbol{F}^{\mathrm{eT}}\boldsymbol{F}^{\mathrm{e}}-\boldsymbol{I}_d\right)=0$$
$$\boldsymbol{R_2}(\boldsymbol{S},\Delta\omega)=\Delta\omega_s-h_s(\tau_s(\boldsymbol{S}),\Delta\omega_s)=0$$

avec $\tau_s(\boldsymbol{S})=\left[\left(2\boldsymbol{\Lambda}^{-1}\boldsymbol{S}+\boldsymbol{I}_d\right)\boldsymbol{S}\right]:\boldsymbol{m}_s\otimes\boldsymbol{n}_s$ et $\boldsymbol{F}^e_{n+1}=\Delta\boldsymbol{F}\,\boldsymbol{F}^e_n\left(\Delta\boldsymbol{F}^p\right)^{-1}$

Calcul de la matrice tangente :

En exprimant la variation de $\boldsymbol{R_1}$ et $\boldsymbol{R_2}$ par rapport à $\boldsymbol{S},\Delta\omega$ et $\Delta\boldsymbol{F}$, on obtient le système :

$$\frac{\partial\boldsymbol{R_1}}{\partial\boldsymbol{S}}\delta\boldsymbol{S}+\frac{\partial\boldsymbol{R_1}}{\partial\Delta\omega}\delta\Delta\omega+\frac{\partial\boldsymbol{R_1}}{\partial\boldsymbol{\Delta F}}\delta\Delta\boldsymbol{F}=0$$
$$\frac{\partial\boldsymbol{R_2}}{\partial\boldsymbol{S}}\delta\boldsymbol{S}+\frac{\partial\boldsymbol{R_2}}{\partial\Delta\omega}\delta\Delta\omega=0\quad car\quad\frac{\partial\boldsymbol{R_2}}{\partial\boldsymbol{\Delta F}}=0$$

en replaçant $\delta\Delta\omega$ issu du deuxième groupe d'équations dans le premier (condensation statique) on obtient :

$$\left(\frac{\partial\boldsymbol{R_1}}{\partial\boldsymbol{S}}-\frac{\partial\boldsymbol{R_1}}{\partial\Delta\omega}\left(\frac{\partial\boldsymbol{R_2}}{\partial\Delta\omega}\right)^{-1}\frac{\partial\boldsymbol{R_2}}{\partial\boldsymbol{S}}\right)\delta\boldsymbol{S}=\frac{-\partial\boldsymbol{R_1}}{\partial\boldsymbol{\Delta F}}\delta\Delta\boldsymbol{F}$$ donc :

$$\frac{\delta\boldsymbol{S}}{\delta\Delta\boldsymbol{F}}=-\left(\frac{\partial\boldsymbol{R_1}}{\partial\boldsymbol{S}}-\frac{\partial\boldsymbol{R_1}}{\partial\Delta\omega}\left(\frac{\partial\boldsymbol{R_2}}{\partial\Delta\omega}\right)^{-1}\frac{\partial\boldsymbol{R_2}}{\partial\boldsymbol{S}}\right)^{-1}\frac{\partial\boldsymbol{R_1}}{\partial\boldsymbol{\Delta F}}$$

ceci permet de calculer cette expression intervenant dans l'opérateur tangent uniquement en fonction de la matrice jacobienne du système d'équations local.

L'opérateur tangent complet peut s'écrire, en configuration actuelle (cf. R5.03.21) : $\frac{\delta\boldsymbol{\tau}}{\delta\Delta F}$ avec $\boldsymbol{\tau}$ le tenseur des contraintes de Kirchhoff, déduit de $\boldsymbol{S}$ par : $\boldsymbol{\tau}=\boldsymbol{F}^{\mathrm{e}}\boldsymbol{S}\,\boldsymbol{F}^{\mathrm{eT}}$

on en déduit : $\frac{\delta\boldsymbol{\tau}}{\delta\Delta F}=\frac{\partial\boldsymbol{F}^{\mathrm{e}}}{\delta\Delta F}\boldsymbol{S}\,\boldsymbol{F}^{\mathrm{eT}}+\boldsymbol{F}^{\mathrm{e}}\frac{\partial\boldsymbol{S}}{\delta\Delta F}\boldsymbol{F}^{\mathrm{eT}}+\boldsymbol{F}^{\mathrm{e}}\boldsymbol{S}\frac{\partial\boldsymbol{F}^{\mathrm{eT}}}{\delta\Delta F}$

il reste à calculer $\frac{\partial\boldsymbol{F}^{\mathrm{e}}}{\delta\Delta F}=\frac{\partial\Delta\boldsymbol{F}\,\boldsymbol{F}^{\mathrm{e}}_n(\Delta\boldsymbol{F}^p)^{-1}}{\delta\Delta F}$

$$\frac{\partial\boldsymbol{F}^{\mathrm{e}}_{ij}}{\delta\Delta F_{kl}}=\delta_{ik}F^{\mathrm{e}}(t_{i-1})_{lb}(\Delta\boldsymbol{F}^p)^{-1}_{bj}\quad\text{et}\quad\frac{\partial\boldsymbol{F}^{\mathrm{eT}}_{ij}}{\delta\Delta F_{kl}}=\frac{\partial\boldsymbol{F}^{\mathrm{e}}_{ji}}{\delta\Delta F_{kl}}=\delta_{jk}F^{\mathrm{e}}(t_{i-1})_{la}(\Delta\boldsymbol{F}^p)^{-1}_{ai}$$

$$\frac{\partial\boldsymbol{R}_{1ij}}{\delta\Delta F_{kl}}=\frac{-1}{2}\left[\frac{\partial\boldsymbol{F}^{\mathrm{eT}}}{\delta\Delta F}\boldsymbol{F}^{\mathrm{e}}+\boldsymbol{F}^{\mathrm{eT}}\frac{\partial\boldsymbol{F}^{\mathrm{e}}}{\delta\Delta F}\right]_{ijkl}=\frac{-1}{2}\left[\frac{\partial\boldsymbol{F}^{\mathrm{e}}_{mi}}{\delta\Delta F_{kl}}\boldsymbol{F}^{\mathrm{e}}_{mj}+\boldsymbol{F}^{\mathrm{e}}_{mi}\frac{\partial\boldsymbol{F}^{\mathrm{e}}_{mj}}{\delta\Delta F_{kl}}\right]$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Il reste à calculer les termes de la matrice jacobienne, spécifiques aux grandes déformations.

- $\frac{\partial \boldsymbol{R_1}(\boldsymbol{S},\Delta\omega)}{\partial \boldsymbol{S}}=\Lambda^{-1}-\frac{1}{2}\left(\frac{\partial \boldsymbol{F}^{eT}}{\partial \boldsymbol{S}}\boldsymbol{F}^{e}+\boldsymbol{F}^{eT}\frac{\partial \boldsymbol{F}^{e}}{\partial \boldsymbol{S}}\right)$

$$\frac{\partial \boldsymbol{F}^{e}}{\partial \boldsymbol{S}}=\Delta \boldsymbol{F}\,\boldsymbol{F}_{n}^{e}\frac{\partial(\Delta \boldsymbol{F}^{p})^{-1}}{\partial \boldsymbol{S}}=\Delta \boldsymbol{F}\,\boldsymbol{F}_{n}^{e}\sum_{s}\left[\frac{\partial(\Delta \boldsymbol{F}^{p})^{-1}}{\partial \Delta\gamma_s}\frac{\partial \Delta\gamma_s}{\partial\tau_s}\frac{\partial\tau_s}{\partial \boldsymbol{S}}\right]$$

avec $\frac{\partial\tau_s}{\partial \boldsymbol{S}}=\left[2\Lambda^{-1}(\mu_s\boldsymbol{S})+2(\Lambda^{-1}\boldsymbol{S})\mu_s+\mu_s\right]$ *où* $\mu_s=\boldsymbol{m}_s\otimes\boldsymbol{n}_s$

ce qui s'écrit en composantes : $\frac{\partial\tau_s}{\partial S_{ab}}=\left[2\,\Lambda^{-1}_{abmn}(\mu_s\boldsymbol{S})_{mn}+2\left(\Lambda^{-1}_{akmn}S_{mn}\right)(\mu_s)_{kb}+(\mu_s)_{ab}\right]$

$$\frac{\partial(\boldsymbol{R_1})_{ij}}{\partial \boldsymbol{S}_{kl}}=\Lambda^{-1}_{ijkl}-\frac{1}{2}\left(\frac{\partial \boldsymbol{F}^{e}_{mi}}{\partial \boldsymbol{S}_{kl}}\boldsymbol{F}^{e}_{mj}+\boldsymbol{F}^{e}_{mi}\frac{\partial \boldsymbol{F}^{e}_{mj}}{\partial \boldsymbol{S}_{kl}}\right)\qquad \frac{\partial \boldsymbol{F}^{e}_{ij}}{\partial \boldsymbol{S}_{kl}}=\Delta \boldsymbol{F}_{im}\boldsymbol{F}^{en}_{mn}\sum_{s}\left[\frac{\partial(\Delta \boldsymbol{F}^{p})^{-1}_{nj}}{\partial \Delta\gamma_s}\frac{\partial \Delta\gamma_s}{\partial\tau_s}\frac{\partial\tau_s}{\partial \boldsymbol{S}_{kl}}\right]$$

- $\frac{\partial \boldsymbol{R_1}(\boldsymbol{S},\Delta\omega)}{\partial \Delta\omega_s}=\frac{-1}{2}\left(\frac{\partial \boldsymbol{F}^{eT}}{\partial \Delta\omega_s}\boldsymbol{F}^{e}+\boldsymbol{F}^{eT}\frac{\partial \boldsymbol{F}^{e}}{\partial \Delta\omega_s}\right)$

$$\frac{\partial \boldsymbol{F}^{e}}{\partial \Delta\omega_s}=\Delta \boldsymbol{F}\,\boldsymbol{F}_{n}^{e}\frac{\partial(\Delta \boldsymbol{F}^{p})^{-1}}{\partial \Delta\omega_s}=\Delta \boldsymbol{F}\,\boldsymbol{F}_{n}^{e}\sum_{r}\left[\frac{\partial(\Delta \boldsymbol{F}^{p})^{-1}}{\partial \Delta\gamma_r}\frac{\partial \Delta\gamma_r}{\partial \Delta\omega_s}\right]$$

- $\frac{\partial \boldsymbol{R_2}(\boldsymbol{S},\Delta\omega)}{\partial \boldsymbol{S}}=-\frac{\partial h_s}{\partial \boldsymbol{S}}=-\sum_r\frac{\partial h_s}{\partial\tau_r}\frac{\partial\tau_r}{\partial \boldsymbol{S}}=-\frac{\partial h_s}{\partial\tau_s}\frac{\partial\tau_s}{\partial \boldsymbol{S}}$. Les termes $\frac{\partial\Delta\gamma_s}{\partial\tau_s}$, $\frac{\partial\Delta\gamma_r}{\partial\Delta\omega_s}$, $\frac{\partial h_s}{\partial\tau_s}$ sont spécifiques à chaque comportement monocristallin, et sont calculés de façon identique aux petites déformations.

- Il reste à calculer $\frac{\partial(\Delta \boldsymbol{F}^{p})^{-1}}{\partial\Delta\gamma_r}$. Suivant les expressions choisies pour le calcul de $(\Delta \boldsymbol{F}^{p})^{-1}$:

1. $(\Delta \boldsymbol{F}^{p})^{-1}=\exp(-\sum\Delta\gamma_s\mu_s)$ conduit à calculer la dérivée de l'exponentielle :

$$\frac{\partial(\Delta \boldsymbol{F}^{p})^{-1}}{\partial\Delta\gamma_r}=-\frac{\partial\exp(-\sum_s\Delta\gamma_s\mu_s)}{\partial(-\sum\Delta\gamma_s\mu_s)}\mu_r$$

Ceci peut être calculé grâce à un développement en séries, décrit dans 12 .

2. $(\Delta \boldsymbol{F}^{p})^{-1}=\left(\frac{\boldsymbol{A}}{(det\,\boldsymbol{A})^{\frac{1}{3}}}\right)^{-1}$ avec $\boldsymbol{A}=\boldsymbol{I}_d+\sum\Delta\gamma_s\boldsymbol{m}_s\otimes\boldsymbol{n}_s$

$$\frac{\partial(\Delta \boldsymbol{F}^{p})^{-1}}{\partial\Delta\gamma_r}=-(\Delta \boldsymbol{F}^{p})^{-1}\otimes(\Delta \boldsymbol{F}^{p})^{-T}\frac{\partial\Delta \boldsymbol{F}^{p}}{\partial\Delta\gamma_r}$$ ou

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

$$\frac{\partial((\Delta \boldsymbol{F}^{p})^{-1})_{ij}}{\partial \Delta \gamma_r} = -((\Delta \boldsymbol{F}^{p})^{-1})_{ik}((\Delta \boldsymbol{F}^{p})^{-T})_{jl}\frac{\partial(\Delta \boldsymbol{F}^{p})_{kl}}{\partial \Delta \gamma_r}$$

avec :

$$\frac{\partial \Delta \boldsymbol{F}^{p}}{\partial \Delta \gamma_r} = (det\, \boldsymbol{A})^{\frac{-1}{3}}\left(\boldsymbol{m}_r \otimes \boldsymbol{n}_r - \frac{1}{3}(\boldsymbol{A}^{-T} : \boldsymbol{m}_r \otimes \boldsymbol{n}_r)\boldsymbol{A}\right)$$

ou en composantes :

$$\frac{\partial(\Delta \boldsymbol{F}^{p})_{ij}}{\partial \Delta \gamma_r} = (det\, \boldsymbol{A})^{\frac{-1}{3}}\left((\boldsymbol{m}_r \otimes \boldsymbol{n}_r)_{ij} - \frac{1}{3}(\boldsymbol{A})^{-T}_{kl}(\boldsymbol{m}_r \otimes \boldsymbol{n}_r)_{kl}\boldsymbol{A}_{ij}\right)$$

3, $(\Delta \boldsymbol{F}^{p})^{-1} = (\boldsymbol{B})(det\, \boldsymbol{B})^{-\frac{1}{3}}$ avec $\boldsymbol{B} = \boldsymbol{I}_d - \sum \Delta \gamma_s \boldsymbol{m}_s \otimes \boldsymbol{n}_s$ conduit à

$$\frac{\partial(\Delta \boldsymbol{F}^{p})^{-1}}{\partial \Delta \gamma_r} = -(det\, \boldsymbol{B})^{-\frac{1}{3}}\left(\boldsymbol{m}_r \otimes \boldsymbol{n}_r - \frac{1}{3}\left((\boldsymbol{B})^{-T} : \boldsymbol{m}_r \otimes \boldsymbol{n}_r\right)\boldsymbol{B}\right)$$

ou en composantes :

$$\frac{\partial((\Delta \boldsymbol{F}^{p})^{-1})_{ij}}{\partial \Delta \gamma_r} = -(det\, \boldsymbol{B})^{-\frac{1}{3}}\left((\boldsymbol{m}_r \otimes \boldsymbol{n}_r)_{ij} - \frac{1}{3}(\boldsymbol{B})^{-T}_{kl}(\boldsymbol{m}_r \otimes \boldsymbol{n}_r)_{kl}\boldsymbol{B}_{ij}\right)$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Annexe 4 Écriture générale des matrices jacobiennes

Formes générales du système local résolu par Newton :

Petites déformations:

$$\boldsymbol{R_1}(\sigma,\Delta\beta)=\Lambda^{-1}.\Delta\sigma-\Delta\varepsilon+\Delta\varepsilon^{th}+\sum_s \Delta\gamma_s\mu_s=0$$
$$\boldsymbol{R_2}(\sigma,\Delta\beta)=\Delta\beta_s-k_s(\tau_s(\sigma),\Delta\beta)=0$$

où l'inconnue est : $\boldsymbol{Y}=\begin{bmatrix}\sigma\\ \Delta\beta\end{bmatrix}$

avec $\tau_s(\sigma)=\sigma:\mu_s$

Grandes déformations

$$\boldsymbol{R_1}(\boldsymbol{S},\Delta\beta)=\Lambda^{-1}.\boldsymbol{S}-\frac{1}{2}\left(\boldsymbol{F}^{eT}\boldsymbol{F}^{e}-\boldsymbol{I_d}\right)=0$$
$$\boldsymbol{R_2}(\boldsymbol{S},\Delta\beta)=\Delta\beta_s-k_s(\tau_s(\boldsymbol{S}),\Delta\beta)=0$$

où l'inconnue est : $\boldsymbol{Y}=\begin{bmatrix}\boldsymbol{S}\\ \Delta\beta\end{bmatrix}$

avec $\tau_s(\boldsymbol{S})=\left[\left(2\Lambda^{-1}\boldsymbol{S}+\boldsymbol{I_d}\right)\boldsymbol{S}\right]:\boldsymbol{m_s}\otimes\boldsymbol{n_s}$ et $\boldsymbol{F}^{e}_{n+1}=\Delta\boldsymbol{F}\,\boldsymbol{F}^{e}_{n}\left(\Delta\boldsymbol{F}^{p}(\Delta\gamma_s)\right)^{-1}$

Et, suivant le comportement considéré,

- $\Delta\gamma_s=\Delta p_s(\tau_s,\Delta\beta_s)\xi_s$ et $\xi_s=\frac{\tau_s}{|\tau_s|}$ *ou* $\frac{\tau_s-f(\alpha)}{|\tau_s-f(\alpha)|}$ correspond au signe de l'écoulement

- $\Delta\beta_s$ représente soit l'incrément de glissement plastique $\Delta\gamma_s$, pour les lois `MONO_VISC*` , soit la variation de densité de dislocations $\Delta\omega_s$ pour les lois `MONO_DD_` *

Les dérivées de ces équations pour le calcul de la matrice jacobienne peuvent être écrites de façon générale :

HPP		GDEF	
$J_{11}=\frac{\partial\boldsymbol{R_1}(\sigma,\Delta\beta)_i}{\partial\sigma_j}$	$J_{12}=\frac{\partial\boldsymbol{R_1}(\sigma,\Delta\beta)_i}{\partial\beta_s}$	$J_{11}=\frac{\partial\boldsymbol{R_1}(\boldsymbol{S},\Delta\beta)_i}{\partial\boldsymbol{S}_j}$	$J_{12}=\frac{\partial\boldsymbol{R_1}(\boldsymbol{S},\Delta\beta)_i}{\partial\beta_s}$
$J_{21}=\frac{\partial\boldsymbol{R_2}(\sigma,\Delta\beta)_s}{\partial\sigma_i}$	$J_{22}=\frac{\partial\boldsymbol{R_2}(\sigma,\Delta\beta)_s}{\partial\beta_r}$	$J_{21}=\frac{\partial\boldsymbol{R_2}(\boldsymbol{S},\Delta\beta)_s}{\partial\boldsymbol{S}_i}$	$J_{22}=\frac{\partial\boldsymbol{R_2}(\boldsymbol{S},\Delta\beta)_s}{\partial\beta_r}$

- J_{11}

- En HPP : $\frac{\partial R_1^i}{\partial\sigma_j}=\left(\Lambda^{-1}\right)_{ij}+\sum_s(\mu_s)_i\frac{\partial\Delta\gamma_s}{\partial\tau_s}\frac{\partial\tau_s}{\partial\sigma_j}=\left(\Lambda^{-1}\right)_{ij}+\sum_s(\mu_s)_i(\mu_s)_j\frac{\partial\Delta\gamma_s}{\partial\tau_s}$

- et en grandes déformations $\frac{\partial\boldsymbol{R_1}(\boldsymbol{S},\Delta\omega)}{\partial\boldsymbol{S}}=\Lambda^{-1}-\frac{1}{2}\left(\frac{\partial\boldsymbol{F}^{eT}}{\partial\boldsymbol{S}}\boldsymbol{F}^{e}+\boldsymbol{F}^{eT}\frac{\partial\boldsymbol{F}^{e}}{\partial\boldsymbol{S}}\right)$ et

$\frac{\partial\boldsymbol{F}^{e}}{\partial\boldsymbol{S}}=\Delta\boldsymbol{F}\,\boldsymbol{F}^{e}_{n}\sum_s\left[\frac{\partial(\Delta\boldsymbol{F}^{p})^{-1}}{\partial\Delta\gamma_s}\frac{\partial\Delta\gamma_s}{\partial\tau_s}\frac{\partial\tau_s}{\partial\boldsymbol{S}}\right]$ où $\frac{\partial\tau_s}{\partial\boldsymbol{S}}$ et $\frac{\partial(\Delta\boldsymbol{F}^{p})^{-1}}{\partial\Delta\gamma_s}$ sont calculés en 49

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

• J_{12}

•En HPP : $\boxed{\frac{\partial R_i^{(1)}}{\partial \Delta \beta_s} = \sum_r (\mu_r)_i \frac{\partial \Delta \gamma_r}{\partial \Delta \beta_s}}$

•et en grandes déformations $\frac{\partial \boldsymbol{R_1}(\boldsymbol{S}, \Delta\beta)}{\partial \Delta \beta_s} = \frac{-1}{2}\left(\frac{\partial \boldsymbol{F}^{\mathbf{eT}}}{\partial \Delta \beta_s} \boldsymbol{F}^{\mathbf{e}} + \boldsymbol{F}^{\mathbf{eT}} \frac{\partial \boldsymbol{F}^{\mathbf{e}}}{\partial \Delta \beta_s} \right)$ et

$\frac{\partial \boldsymbol{F}^{\mathbf{e}}}{\partial \Delta \beta_s} = \Delta \boldsymbol{F}\, \boldsymbol{F}_{\boldsymbol{n}}^{\mathbf{e}} \sum_r \left[\frac{\partial (\Delta \boldsymbol{F}^{\boldsymbol{p}})^{-1}}{\partial \Delta \gamma_r} \frac{\partial \Delta \gamma_r}{\partial \Delta \beta_s} \right]$, où $\frac{\partial (\Delta \boldsymbol{F}^{\boldsymbol{p}})^{-1}}{\partial \Delta \gamma_r}$ est calculé en Annexe 3

•

• J_{21}

•En HPP : $\frac{\partial R_s^2}{\partial \sigma_i} = -(\mu_s)_i \frac{\partial k_s}{\partial \tau_s}$

•et en grandes déformations $\frac{\partial \boldsymbol{R_2}(\boldsymbol{S}, \Delta\beta)}{\partial \boldsymbol{S}} = -\frac{\partial (k_s)}{\partial \boldsymbol{S}} = -\sum_r \frac{\partial k_s}{\partial \tau_r} \frac{\partial \tau_r}{\partial \boldsymbol{S}} = -\frac{\partial k_s}{\partial \tau_s} \frac{\partial \tau_s}{\partial \boldsymbol{S}}$.

$\frac{\partial \tau_s}{\partial \boldsymbol{S}}$ est calculé en Annexe 3 .

• J_{22}

•terme identique en HPP et en grandes déformations : $\frac{\partial R_2^s}{\partial \Delta \beta_r} = \delta_{sr} - \frac{\partial k_s}{\partial \Delta \beta_r}$ ou encore (suivant l'ordre de remplissage de la matrice jacobienne) :

$\frac{\partial R_2^r}{\partial \Delta \beta_s} = \delta_{rs} - \frac{\partial k_r}{\partial \Delta \beta_s}$

•L es dérivées précédentes ont en commun des termes spécifiques à chaque comportement, qui sont détaillés pour chacun dans ce document :

- $\frac{\partial \Delta \gamma_s}{\partial \tau_s} = \frac{\partial \Delta p_s}{\partial \tau_s} \xi_s$,
- $\frac{\partial \Delta \gamma_r}{\partial \Delta \beta_s} = \frac{\partial \Delta p_r}{\partial \Delta \beta_s} \xi_r$,
- $\frac{\partial k_s}{\partial \tau_s}$
- $\frac{\partial k_r}{\partial \Delta \beta_s}$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Annexe 5 Exemples de matrices d'interaction

Cas général :

Il s'agit de définir une matrice carrée symétrique d'interaction entre systèmes de glissement.

La forme la plus simple (utilisable pour tous les comportements monocristallins) consiste à utiliser un seul coefficient H permettant de distinguer les termes diagonaux des autres : la matrice h_{sr} est alors définie par :

$h_{rs}=1$ si $r=s$ et $h_{rs}=H$ si $r\neq s$, avec par défaut $H=0$.

Pour les familles à 12 systèmes de glissement uniquement, et les comportements MONO_VISC*, la matrice d'interaction peut être construite à partir de 6 coefficients.

h1	h2	h2	h4	h5	h5	h5	h6	h3	h5	h3	h6
h2	h1	h2	h5	h3	h6	h4	h5	h5	h5	h6	h3
h2	h2	h1	h5	h6	h3	h5	h3	h6	h4	h5	h5
h4	h5	h5	h1	h2	h2	h6	h5	h3	h6	h3	h5
h5	h3	h6	h2	h1	h2	h3	h5	h6	h5	h5	h4
h5	h6	h3	h2	h2	h1	h5	h4	h5	h3	h6	h5
h5	h4	h5	h6	h3	h5	h1	h2	h2	h6	h5	h3
h6	h5	h3	h5	h5	h4	h2	h1	h2	h3	h5	h6
h3	h5	h6	h3	h6	h5	h2	h2	h1	h5	h4	h5
h5	h5	h4	h6	h5	h3	h6	h3	h5	h1	h2	h2
h3	h6	h5	h3	h5	h6	h5	h5	h4	h2	h1	h2
h6	h3	h5	h5	h4	h5	h3	h6	h5	h2	h2	h1

MONO_DD_KR

La forme de la matrice d'interaction utilisée avec le modèle MONO_DD_KR et adaptés aux 24 systèmes de glissement associés aux deux familles cubiques centrées (BCC24) est la suivante :

$$h_{rs}=\begin{matrix}
h_2 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 \\
h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 \\
h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 \\
h_2 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 \\
h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 \\
h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 \\
h_1 & h_2 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 \\
h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 \\
h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 \\
h_1 & h_1 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 \\
h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 \\
h_1 & h_1 & h_1 & h_1 & h_2 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 \\
h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_4 & h_4 & h_4 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 \\
h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_4 & h_4 & h_4 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 \\
h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_4 & h_4 & h_4 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 \\
h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_3 & h_3 & h_3 & h_4 & h_4 & h_4 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 \\
h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_3 & h_3 & h_3 & h_4 & h_4 & h_4 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 \\
h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_3 & h_3 & h_3 & h_4 & h_4 & h_4 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 \\
h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_4 & h_4 & h_4 & h_3 & h_3 & h_3 \\
h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_4 & h_4 & h_4 & h_3 & h_3 & h_3 \\
h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_1 & h_1 & h_1 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_4 & h_4 & h_4 & h_3 & h_3 & h_3 \\
h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_4 & h_4 & h_4 \\
h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_4 & h_4 & h_4 \\
h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_1 & h_2 & h_2 & h_2 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_3 & h_4 & h_4 & h_4
\end{matrix}$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

MONO_DD_CFC et MONO_DD_FAT

Pour la loi de comportement MONO_DD_CFC, la matrice d'interaction adaptée aux 12 systèmes de glissement est obtenue par la dynamique des dislocations discrètes (DDD). Ces valeurs sont fonctions du type de dislocations qui interagissent, le tableau suivant récapitule ces informations. Le paramétrage des systèmes CFC utilisés est celui de Schmid et Boas. Pour la loi MONO_DD_FAT, on utilise actuellement la même matrice d'interaction que celle de la loi MONO_DD_CFC.

Normale	$\bar{1}11$			111			$\bar{1}\bar{1}1$			$1\bar{1}1$		
Direction	$0\bar{1}1$	101	110	$0\bar{1}1$	$\bar{1}01$	$\bar{1}10$	011	101	$\bar{1}10$	011	$\bar{1}01$	110
Désignation	A2	A3	A6	B2	B4	B5	C1	C3	C5	D1	D4	D6

La correspondance entre les numéros des systèmes de glissement de la famille octaédrique et le paramétrage de Schmid et Boas est la suivante :

Normale	$\bar{1}11$			111			$\bar{1}\bar{1}1$			$1\bar{1}1$		
Direction	$0\bar{1}1$	101	110	$0\bar{1}1$	$\bar{1}01$	$\bar{1}10$	011	101	$\bar{1}10$	011	$\bar{1}01$	110
Numéro (SB)	1	2	3	4	5	6	7	8	9	10	11	12
Désignation	A2	A3	A6	B2	B4	B5	C1	C3	C5	D1	D4	D6
Numéro Aster	7	9	8	2	1	3	12	11	10	5	4	6

La matrice d'interaction est formée de 4×4 bloc, chaque bloc étant une matrice 3×3. Nous précisons ici cette matrice dans l'ordre des systèmes de glissement de Schmid et Boas :

Schmid&Boas / **Aster**	**A2 A3 A6** **7 9 8**	**B2 B4 B5** **2 1 3**	**C1 C3 C5** **12 11 10**	**D1 D4 D6** **5 4 6**
A2 A3 A6 **7 9 8**	A_0	A_1	A_2	A_3
B2 B4 B5 **2 1 3**		A_0	A_3	A_2
C1 C3 C5 **12 11 10**			A_0	A_1
D1 D4 D6 **5 4 6**				A_0

Les matrices 3×3 sont définies de la façon suivante :

$$A_0=\begin{pmatrix} a^* & a^* & a^* \\ a^* & a^* & a^* \\ a^* & a^* & a^* \end{pmatrix} \qquad A_1=\begin{pmatrix} a_{colinéaire} & a_{glissile} & a_{glissile} \\ a_{glissile} & a_{Hirth} & a_{Lomer} \\ a_{glissile} & a_{Lomer} & a_{Hirth} \end{pmatrix}$$

$$A_2=\begin{pmatrix} a_{Hirth} & a_{glissile} & a_{Lomer} \\ a_{glissile} & a_{colinéaire} & a_{glissile} \\ a_{Lomer} & a_{glissile} & a_{Hirth} \end{pmatrix} \qquad A_3=\begin{pmatrix} a_{Hirth} & a_{Lomer} & a_{glissile} \\ a_{Lomer} & a_{Hirth} & a_{glissile} \\ a_{glissile} & a_{glissile} & a_{colinéaire} \end{pmatrix}$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

MONO_DD_CC

Pour la loi de comportement MONO_DD_CC, la matrice d'interaction adaptée aux 12 systèmes de glissement de type CUBIQUE1 est obtenue par la dynamique des dislocations discrètes (DDD). Ces valeurs sont fonctions du type de dislocations qui interagissent, le tableau suivant récapitule ces informations. Le paramétrage des systèmes CC utilisés est :

Direction	$[1\bar{1}1]$			$[111]$		
Normale	(110)	$(\bar{1}01)$	$(0\bar{1}\bar{1})$	$(0\bar{1}1)$	$(10\bar{1})$	$(\bar{1}10)$
Numéro	1	2	3	4	5	6

Direction	$[\bar{1}11]$			$[11\bar{1}]$		
Normale	$(\bar{1}\bar{1}0)$	(101)	$(01\bar{1})$	$(1\bar{1}0)$	$(\bar{1}0\bar{1})$	(011)
Numéro	7	8	9	10	11	12

La matrice d'interaction est formée de 4×4 bloc, chaque bloc étant une matrice 3×3 . Nous précisons ici cette matrice dans l'ordre des systèmes de glissement défini dans Aster (famille CUBIQUE1) :

Systèmes de glissement	1 , 2, 3	4 , 5, 6	7 , 8, 9	10 , 11, 12
1 , 2, 3	A_0	A_4	A_3	A_2
4 , 5, 6	A_4	A_0	A_5	A_5^T
7 , 8, 9	A_3	A_5^T	A_0	A_1
10 , 11, 12	A_2	A_5	A_1	A_0

Les matrices 3×3 sont définies de la façon suivante :

$$A_0=\begin{pmatrix} h_0 & h_1 & h_1 \\ h_1 & h_0 & h_1 \\ h_1 & h_1 & h_0 \end{pmatrix} \qquad A_1=\begin{pmatrix} h_4 & h_3 & h_2 \\ h_3 & h_5 & h_3 \\ h_2 & h_3 & h_4 \end{pmatrix} \qquad A_2=\begin{pmatrix} h_4 & h_2 & h_3 \\ h_2 & h_4 & h_3 \\ h_3 & h_3 & h_5 \end{pmatrix}$$

$$A_3=\begin{pmatrix} h_5 & h_3 & h_3 \\ h_3 & h_4 & h_2 \\ h_3 & h_2 & h_4 \end{pmatrix} \qquad A_4=\begin{pmatrix} h_3 & h_5 & h_3 \\ h_4 & h_3 & h_2 \\ h_4 & h_3 & h_2 \end{pmatrix} \qquad A_4=\begin{pmatrix} h_3 & h_3 & h_5 \\ h_2 & h_4 & h_3 \\ h_4 & h_2 & h_3 \end{pmatrix}$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Annexe 6 Procédure de création de maillage

La création de maillages repose sur une répartition aléatoire de germes (centres des cellules de Voronoï). On construit les grains sous forme de cellules de Voronoï (à l'aide de la bibliothèque qhull, confer http://www.qhull.org/) Mais lorsque l'on part d'une répartition aléatoire de germes, on constate que l'on obtient, sans retraitement, des cellules de Voronoï comportant des arêtes parfois très petites, ce qui conduit à un maillage soit de mauvaise qualité, soit impossible à réaliser étant donnée la dispersion des densités locales.

Pour l'élimination des petites arêtes on utilise la propriété suivante : la présence d'une petite arête est liée au fait qu'un germe est proche de la sphère circonscrite au tétraèdre ayant pour sommets 4 autres germes, sans être exactement sur cette sphère. Dans ce cas, il suffit de projeter ce germe sur la sphère pour éliminer la petite arête. L'algorithme construit donc une répartition aléatoire de germes de façon itérative, en veillant à ce que la position de chaque nouveau germe par rapport aux sphères circonscrites existantes soit correcte. (avec projection éventuelle).

Ensuite il faut restreindre la géométrie obtenue au cube représentant le V.E.R. Pour cela on utilise la bibliothèque PyXL (http://www.tc.cornell.edu/~myers/PyXL) qui propose des outils (écrits en python et en C) pour tronquer des cellules de Voronoï sur une géométrie donnée. Après cette opération, il est nécessaire de vérifier à nouveau que l'on n'a pas créé de trop petites arêtes. Dans ce cas, on ajoute localement des germes sur les cellules comportant les petites arêtes, puis on recalcule avec ces nouveaux germes les cellules de Voronoï et leurs enveloppes convexes. A la fin de la génération du nombre de germes requis, on calcule le nombre de petites arêtes dont la longueur est inférieure à un certain seuil ainsi que la longueur de la plus petite arête, et on élimine les sommets des cellules quasi-confondus (suite à la projection d'un germe sur une sphère circonscrite).

Il suffit ensuite de générer les données nécessaires à un logiciel de maillage. Les géométries des agrégats sont écrites au format SALOME. Le maillage est effectué avec le module MESH.

En résumé la création d'un maillage consiste donc en l'appel à une unique procédure python qui enchaîne les actions suivantes :

- Création de la géométrie : La procédure enchaîne des appels à des modules écrits en fortran pour gagner en temps calcul (projections sur les sphères) et les bibliothèques Qhull et PyXL ainsi que des procédures python pour créer la géométrie avec le nombre de germes souhaités. Ceci crée un fichier de géométrie au format SALOME, ainsi que des directives de maillage.
- Lecture du fichier python et création du maillage dans SALOME, avec calcul du volume de chaque grain et la position des centres de gravité des grains. Le maillage est écrit au format MED pour le calcul avec Code_Aster.
- Une description complète de cette procédure est fournie dans [10]. Elle fonctionne en 2D et en 3D :Quelques exemples de maillage sont fournis ci-dessous :

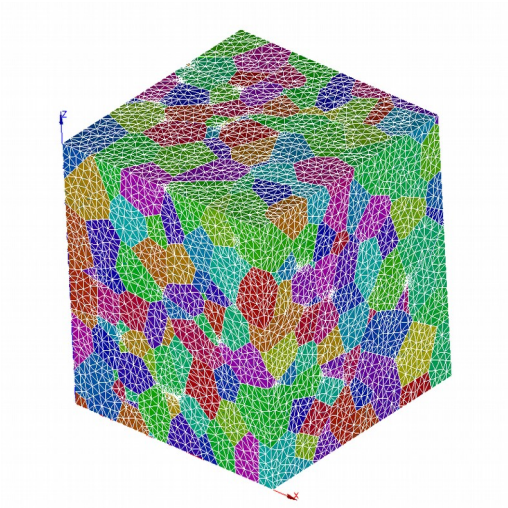


Figure 8-a. maillage d'un agrégat à 988 grains comportant 311510 tétraèdres Tetra4

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

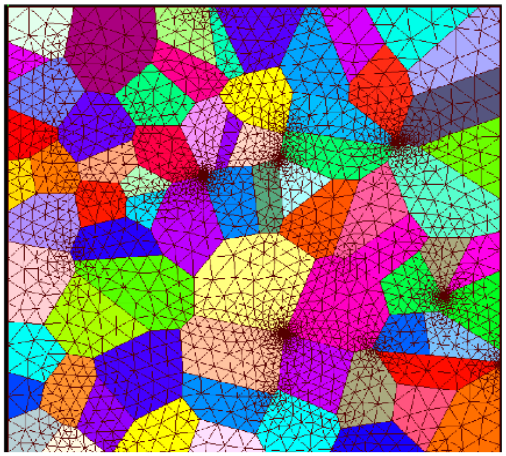


Figure 8-b. maillage d'un agrégat 2D comportant 70 cellules

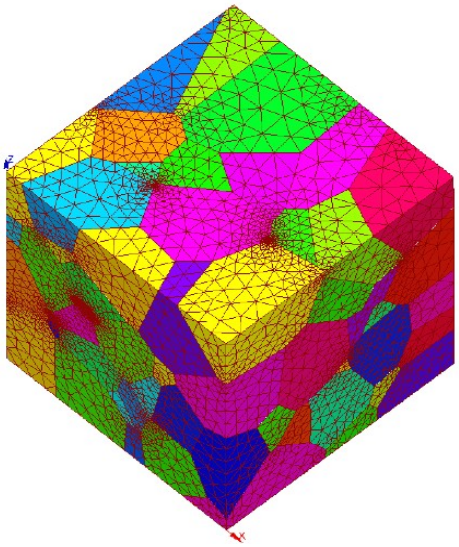


Figure 8-c. maillage 3D de 135 cellules

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)