Méthode IMPLEX

Résumé :

Ce document présente une méthode de résolution du problème non linéaire, due à *Oliver et al.* [1], se substituant à la méthode de Newton pour certaines lois de comportement d'endommagement et de plasticité (ENDO_FRAGILE [3], ENDO_ISOT_BETON [4] et VMIS_ISOT_LINE [5] à l'heure actuelle). Elle se base sur une extrapolation explicite des variables internes pour déterminer les degrés de libertés (déplacement) à partir desquels le comportement est intégré implicitement. La nullité du résidu n'est pas vérifiée. Elle introduit de ce fait une approximation de la résolution mais permet de garantir la robustesse du calcul.

Il appartient donc à l'utilisateur d'avoir un regard critique sur les résultats obtenus, ceux-ci étant non convergés au sens classique du terme, et pouvant potentiellement être éloignés de la solution exacte ; on conseille par conséquent de réaliser plusieurs calculs avec des incréments de charge différents pour s'assurer que l'écart de résultats obtenus est faible.

Dans des cas d'expansion brutale de la zone endommagée se traduisant par un fort snap-back de la réponse globale force/déplacement, la méthode, si elle permet de franchir l'instabilité, ne saurait être fiable en terme de résultat.

Copyright 2015 EDF R&D - Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

1 Introduction

On présente ici une méthode de résolution, robuste mais approchée, du problème incrémental de mécanique quasi-statique non linéaire, pour certains comportements d'endommagement et de plasticité. Elle est activée par le mot clé METHODE = 'IMPLEX' de l'opérateur STAT_NON_LINE, et se substitue à la méthode classique de NEWTON (cf. [2]).

2 Position du problème

On se place dans le cadre général de la résolution d'un problème type de mécanique non-linéaire discrétisé en espace $(K \in \Omega)$ et en temps $(t \in [0,T])$, et écrit en déplacement. Sa résolution, à l'instant t_{n+1}, consiste à déterminer les déplacements $\boldsymbol{U}_{n+1}$ (donc les déformations $\boldsymbol{\varepsilon}(\boldsymbol{U}_{n+1})$), les variables internes α_{n+1} et les contraintes $\boldsymbol{\sigma}_{n+1}$ vérifiant :
(i) Au niveau global, l'équilibre des forces :

$$\boldsymbol{F}_{\text{ext}}(t_{n+1}) - \boldsymbol{F}_{\text{int}}(\boldsymbol{\sigma}_{n+1}(\boldsymbol{U}_{n+1}), t_{n+1}) \qquad (1)$$

(ii) Au niveau local, les équations constitutives de la loi de comportement considérée pour le matériau :
Loi d'état : $\boldsymbol{\sigma}_{n+1} = \sum(\boldsymbol{\varepsilon}(\boldsymbol{U}_{n+1}), \alpha_{n+1})$

$$\text{Loi d'évolution : } \begin{cases} f(\alpha_{n+1}, \boldsymbol{\sigma}_{n+1}) \leqslant 0 & \text{Convexe de réversibilité} \\ \dot{\alpha}_{n+1} = \dot{\lambda}_{n+1} \geqslant 0 & \text{Evolution des variables internes} \\ \dot{\lambda}_{n+1} f(\alpha_{n+1}, \boldsymbol{\sigma}_{n+1}) = 0 & \text{Condition de Kuhn-Tucker} \end{cases} \qquad (2)$$

avec $\boldsymbol{F}_{\text{ext}}$ et $\boldsymbol{F}_{\text{int}}$ les forces externes et internes, et $\dot{\lambda}_{n+1}$ le multiplicateur (d'endommagement ou de plasticité). On considère dans la suite un comportement indépendant du temps physique (pas de viscosité ni d'effet dynamique) ; le « pseudo-instant » t_{n+1} représente par conséquent le facteur de charge appliqué.

Dans la grande majorité des cas, chacune des équations (1) et (2) est de résolution relativement aisée.

Au niveau global, la méthode de Newton conduit à une succession de problèmes linéaires du type :

$$\boldsymbol{K}^{i}_{n+1} \delta \boldsymbol{U}^{i+1}_{n+1} = \boldsymbol{F}_{\text{ext}}(t_{n+1}) - \boldsymbol{F}_{\text{int}}(\boldsymbol{\sigma}^{i}_{n+1}(\boldsymbol{U}^{i}_{n+1})) \qquad (3)$$

où l'indice haut représente l'itération de Newton considérée, $\delta \boldsymbol{U}^{i+1}_{n+1}$ est l'incrément de déplacement entre deux itérations successives de Newton et $\boldsymbol{K}^{i}_{n+1}$ est la matrice tangente globale. Notons que cette matrice tangente globale s'écrit (cf. [2])

$$\begin{aligned} \boldsymbol{K}_{n+1} &= \frac{\partial \boldsymbol{F}_{\text{int}}(\boldsymbol{\sigma}_{n+1}(\boldsymbol{U}_{n+1}))}{\partial \boldsymbol{U}_{n+1}} \\ &= \mathbf{A}_{e=1}^{p} \left(\int_{\Omega_e} \nabla N_e . \boldsymbol{C}_{e_{n+1}} . \nabla N_e \, d\Omega \right) \end{aligned} \qquad (4)$$

où $\mathbf{A}$ est l'opérateur d'assemblage, p le nombre d'éléments du maillage, $\boldsymbol{C}_{e_{n+1}}$ l'opérateur tangent local (issu du comportement) et N_e les fonctions de forme (pour l'élément considéré, noté e). La quantité $\boldsymbol{F}_{\text{ext}}(t_{n+1}) - \boldsymbol{F}_{\text{int}}(\boldsymbol{\sigma}^{i}_{n+1}(\boldsymbol{U}^{i}_{n+1}))$ représente le résidu d'équilibre, qu'il faut en toute rigueur annuler. Déterminer $\delta \boldsymbol{U}^{i+1}_{n+1}$ se limite donc à l'inversion de la matrice $\boldsymbol{K}^{i}_{n+1}$. Si donc les éléments locaux issus du comportement (contraintes, matrice tangente) sont connus et fixés, et engendrent une matrice $\boldsymbol{K}$ bien conditionnée, la résolution est aisée.

Copyright 2015 EDF R&D - Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

De plus localement, si les déplacements U_{n+1}, et par conséquent les déformations locales $\varepsilon(U_{n+1})$, sont connus, la résolution de la majorité des lois de comportement, c'est-à-dire la détermination des contraintes σ_{n+1}, des variables internes α_{n+1} et de la matrice tangente locale $C_{e_{n+1}}$, est relativement aisée.

C'est donc dans la simultanéité de ces deux lois que réside la difficulté de résolution implicite. La méthode la plus classique de résolution est l'algorithme itératif de Newton (cf. [2]). Ce schéma itératif ne converge pas toujours. Bien que cela reste sujet à discussion (il existe d'autres facteurs possibles de perte de convergence, tels que la sortie du bassin d'attraction de Newton ou la présence de bifurcations de la solution), les auteurs attribuent principalement cette perte de robustesse au caractère singulier des matrices, notamment pour les lois d'endommagement, dont les variables internes impactent les matrices de rigidité du comportement : dans les cas d'endommagement important, proche ou atteignant la rupture, l'adoucissement conduit à des matrices tangentes localement très « faibles », qui peuvent à l'extrême ne plus être définies positives. A travers le processus d'assemblage (4), lorsque l'endommagement progresse dans la structure, la matrice globale K est détériorée : elle devient trop « souple », et sa valeur propre minimale tend vers zéro. Elle devient alors singulière et l'algorithme diverge. La robustesse du calcul n'est alors plus assurée.

Pour augmenter la robustesse du calcul dans ces situations, Oliver *et al.* [1] proposent une méthode particulière de résolution, baptisée IMPLEX. L'équilibre global est alors vérifié approximativement à l'aide d'une matrice tangente (sécante dans le cas des lois d'endomagement) locale extrapolée explicitement et assemblée, et les lois de comportements sont résolues implicitement à partir du champ de déplacement approché. La méthode est présentée ci-après.

3 La méthode IMPLEX

Les éléments généraux de la méthode IMPLEX sont présentés ici. Pour plus de détails, on peut se référer notamment à [1].

La méthode proposée est basée sur deux étapes successives réalisées pour déterminer l'ensemble des inconnues au « pseudo-instant » (charge) t_{n+1}.

La première étape consiste en une *extrapolation explicite* des variables internes, puis des contraintes, en fonction des quantités calculées précédemment (à la charge t_n) et du pas de charge Δt_{n+1}. Grâce à cette extrapolation, la matrice tangente locale est évaluée et figée ; la résolution de l'équation d'équilibre (3) permet de déterminer les déplacements, que l'on considère justes et qui sont donc figés à leur tour.

La seconde étape consiste à réaliser *l'intégration implicite* du comportement, en fonction des degrés de liberté évalués à l'étape précédente.

La première étape est de type explicite, alors que la seconde est de type implicite, d'où le nom d'IMPLEX. A l'issue de ces deux étapes, l'équation d'équilibre n'est pas vérifiée (elle l'est en terme des champs extrapolés à l'étape (1) uniquement) ; cependant, plus l'incrément de charge est petit, plus l'erreur commise doit être faible ; on conseille de ce fait de réaliser les calculs avec plusieurs pas de charges différents afin de vérifier que la différence entre les réponses obtenues est faible (réponse convergée en terme de pas de charge). Dans la suite, les deux étapes sont décrites. Un bilan des points clés est dressé.

3.1 Une méthode en 2 étapes

Nous allons ici présenter successivement les 2 étapes fondamentales de la méthode IMPLEX, synthétisées dans le Tableau 1.

3.1.1 Extrapolation explicite et détermination des degrés de liberté

Copyright 2015 EDF R&D - Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Cette première étape concerne en particulier la variable interne α , dont l'évolution est régie par une loi de type (2).
Au début du pas de charge t_{n+1} , on dispose de toutes les informations issues du pas de charge t_n et des pas de charge antérieurs. Il est alors possible d'écrire les développements de Taylor suivants (l'évolution de la variable interne est considérée suffisamment régulière pour pouvoir le faire) :

$$\left\{\begin{array}{lcl}\alpha_{n+1} & = & \alpha_n+\Delta t_{n+1}\dot{\alpha}_n+O(\Delta t_{n+1}^2)\\ \alpha_n & = & \alpha_{n\text{-}1}+\Delta t_n\dot{\alpha}_{n\text{-}1}+O(\Delta t_n^2)\\ \Delta t_n\dot{\alpha}_n & = & \Delta t_n\dot{\alpha}_{n\text{-}1}+\Delta t_n(\Delta t_n\ddot{\alpha}_{n\text{-}1})+\Delta t_n O(\Delta t_n^2)\quad \Rightarrow \Delta t_n\dot{\alpha}_n=\Delta t_n\dot{\alpha}_{n\text{-}1}+O(\Delta t_n^2)\end{array}\right. \qquad (5)$$

$$\Rightarrow\left\{\begin{array}{lcl}\dot{\alpha}_n & = & \dfrac{\Delta\alpha_n}{\Delta t_n}\text{-}O(\Delta t_n^2)\\ \alpha_{n+1} & = & \alpha_n+\dfrac{\Delta t_{n+1}}{\Delta t_n}\Delta\alpha_n-\Delta t_{n+1}O(\Delta t_n^2)+O(\Delta t_{n+1}^2)\end{array}\right.$$

avec $\Delta X_i=X_i-X_{i\text{-}1}$. Par troncature, en négligeant les termes d'ordre deux, on obtient la prédiction suivante $\tilde{\alpha}_{n+1}$ pour la variable α_{n+1} :

$$\tilde{\alpha}_{n+1}=\alpha_n+\frac{\Delta t_{n+1}}{\Delta t_n}\Delta\alpha_n \qquad (6)$$

A travers (6), on note que $\tilde{\alpha}_{n+1}$ est bien obtenu explicitement, à la charge t_{n+1} , en fonction des grandeurs obtenues implicitement à la charge t_n . La Figure 1 schématise ce procédé d'extrapolation.

L'erreur d'extrapolation peut être définie comme la différence entre la valeur de la variable interne extrapolée et sa valeur finale réelle ; sous réserve d'une évolution suffisamment régulière de la variable interne pour en réaliser les développements de Taylor à l'ordre suffisant, on peut évaluer cette erreur à :

$$e_{\alpha_{n+1}}=\left|\tilde{\alpha}_{n+1}-\alpha_{n+1}\right|\approx\left|\ddot{\alpha}_n\right|\Delta t_{n+1}^2 \qquad (7)$$

Ceci montre que dans le cas d'évolution suffisamment régulière de la variable interne, l'erreur diminue de façon quadratique en fonction du pas de charge.

Une fois la variable interne extrapolée, on peut déterminer la contrainte $\tilde{\boldsymbol{\sigma}}_{n+1}(\boldsymbol{\varepsilon}_{n+1},\tilde{\alpha}_{n+1})$ et l'opérateur tangent local $\tilde{\boldsymbol{C}}_{e_{n+1}}$.

Dans le cas particulier des lois d'endommagement isotropes, c'est l'opérateur sécant local qui est utilisé ; les contraintes et cet opérateur extrapolés s'écrivent alors :

$$\left\{\begin{array}{lcl}\tilde{\boldsymbol{\sigma}}_{n+1}(\boldsymbol{\varepsilon}_{n+1},\tilde{\alpha}_{n+1}) & = & (1-\alpha_{n+1})\boldsymbol{C}^{\text{elas}}:\boldsymbol{\varepsilon}_{n+1}\\ \tilde{\boldsymbol{C}}_{e_{n+1}}=\dfrac{\partial\tilde{\boldsymbol{\sigma}}_{n+1}}{\partial\tilde{\boldsymbol{\varepsilon}}_{n+1}} & = & (1-\tilde{\alpha}_{n+1})\boldsymbol{C}^{\text{elas}}\end{array}\right. \qquad (8)$$

Ces quantités locales extrapolées servent à l'assemblage de la matrice tangente globale $\tilde{\boldsymbol{K}}_{n+1}$ *via* (4), et à la détermination de la force interne.

L'équation d'équilibre en terme de champ extrapolé $\boldsymbol{F}_{\text{ext}}(t_{n+1})-\boldsymbol{F}_{\text{int}}(\boldsymbol{U}_{n+1},\tilde{\boldsymbol{\sigma}}_{n+1},t_{n+1})=\boldsymbol{0}$ est alors résolue avec cette matrice extrapolée et le champ de déplacement $\tilde{\boldsymbol{U}}_{n+1}$ obtenu. Elle est résolue par une méthode de Newton-Raphson classique ; elle est donc linéarisée par $\tilde{\boldsymbol{K}}_{n+1}\delta\tilde{\boldsymbol{U}}_{n+1}=\boldsymbol{F}_{\text{ext}}(t_{n+1})-\boldsymbol{F}_{\text{int}}\left(\tilde{\sigma}_{n+1},t_{n+1}\right)$. Dans le cas des lois de comportement traitées ici (endommagement isotrope avec opérateur sécant et plasticité isotrope linéaire), l'opérateur de comportement est constant durant un pas de temps (c'est-à-dire indépendant

Copyright 2015 EDF R&D - Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

de l'état de déformation courant $\boldsymbol{\varepsilon}_{n+1}$) ; le processus de résolution de Newton-Raphson devient linéaire par pas de temps et converge donc en une itération. Cette étape est cependant itérative dans certains cas (plasticité non linéaire par exemple), du fait du caractère non constant de l'opérateur tangent. A la première itération, on prendrait alors $\boldsymbol{\varepsilon}^{0}_{n+1}=\boldsymbol{\varepsilon}_{n}$.

A la fin de cette première étape, le champ de déplacement $\boldsymbol{U}_{n+1}$ est considéré comme égal au champ obtenu par la résolution de l'équation d'équilibre écrite en terme des champs extrapolés : $\boldsymbol{U}_{n+1}=\tilde{\boldsymbol{U}}_{n+1}$. Ce champ de déplacement ne sera plus modifié par la suite.

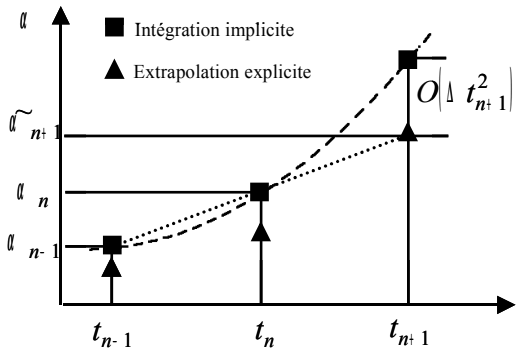

Figure 1 : Schématisation de la méthode d'extrapolation

3.1.2 Détermination implicite des éléments de la loi de comportement

Suite à la première phase, le champ de déplacement est connu. Il est figé pour le pas de charge et ne sera donc pas modifié.

On détermine alors la déformation $\boldsymbol{\varepsilon}_{n+1}(\boldsymbol{U}_{n+1})$, puis les équations (2) de la loi de comportement sont résolues implicitement afin d'obtenir les champs de contrainte $\boldsymbol{\sigma}_{n+1}$ et de variable interne α_{n+1} .

Cette étape est identique à la résolution standard des équations des lois de comportement. La seule différence est la nécessité de stocker $\dfrac{\Delta\alpha_{n+1}}{\Delta t_{n+1}}$ pour réaliser l'extrapolation de la variable interne au pas suivant.

A la fin de cette étape, le champ de déplacement $\boldsymbol{U}_{n+1}$ et les champs de contrainte $\boldsymbol{\sigma}_{n+1}$ et de variable interne α_{n+1} sont donc connus. Une différence majeure par rapport à un calcul classique implicite est le fait que l'équation d'équilibre réelle n'est pas vérifiée ; elle ne l'est qu'en terme des champs extrapolés.

Étape 1 : Extrapolation Explicite	**Étape 2 : Intégration Implicite**
Extrapolation des variables internes : $\tilde{\alpha}_{n+1}=\alpha_n+\dfrac{\Delta t_{n+1}}{\Delta t_n}\Delta\alpha_n$	Déplacements figés : $\dot{\boldsymbol{U}}_{n+1}=\boldsymbol{0}$
Calcul de la contrainte extrapolée : $\tilde{\boldsymbol{\sigma}}_{n+1}(\boldsymbol{\varepsilon}_{n+1},\tilde{\alpha}_{n+1})$	Résolution implicite des lois de comportement :

Copyright 2015 EDF R&D - Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Résolution de l'équation d'équilibre en champs extrapolés : $\boldsymbol{F}_{\text{ext}}(t_{n+1}) - \boldsymbol{F}_{\text{int}}(\boldsymbol{U}_{n+1}, \tilde{\boldsymbol{\sigma}}_{n+1}, t_{n+1}) = \boldsymbol{0}$	Loi d'état : $\boldsymbol{\sigma}_{n+1} = \sum(\boldsymbol{\varepsilon}(\boldsymbol{U}_{n+1}), \alpha_{n+1})$ Loi d'évolution : $\begin{cases} f(\alpha_{n+1}, \boldsymbol{\sigma}_{n+1}) \leqslant 0 \\ \dot{\alpha}_{n+1} = \dot{\lambda}_{n+1} \geqslant 0 \\ \dot{\lambda}_{n+1} f(\alpha_{n+1}, \boldsymbol{\sigma}_{n+1}) = 0 \end{cases}$
Sortie de l'étape 1 : $\boldsymbol{U}_{n+1}$	**Sortie de l'étape 2 :** $\boldsymbol{U}_{n+1}$, $\boldsymbol{\sigma}_{n+1}$, α_{n+1}

Tableau 1 : Récapitulatif de la méthode IMPLEX

3.2 Gestion automatique du pas de temps

La méthode IMPLEX introduit, comme toutes les méthodes explicites, une erreur intrinsèque qui doit diminuer de façon quasi-quadratique avec le pas de charge. La solution peut donc dépendre légèrement du pas de charge sélectionné par l'utilisateur.
Celui-ci peut, s'il le souhaite, utiliser une gestion automatique du pas de pas de charge *via* la commande DEFI_LIST_INST, en précisant METHODE= 'AUTO' et MODE_CALCUL_TPLUS='IMPLEX' (cf. [9]). Cette méthode permet de contrôler l'erreur tout en optimisant la durée de calcul sous réserve que l'utilisateur ait choisi un premier pas de temps bien calibré.
Le but est donc de minimiser l'erreur définie par l'équation (7). Pour cela on va maximiser l'incrément de variable extrapolée par une grandeur notée Tol :

$$\Delta\tilde{\alpha}_{n+1} = \frac{\Delta t_{n+1}}{\Delta t_n}\Delta\alpha_n \leqslant Tol \tag{9}$$

D'où :

$$\Delta t_{n+1} \leqslant \frac{Tol\,.\,\Delta t_n}{\Delta\alpha_n} \tag{10}$$

Comme les incréments de variable interne dépendent du point considéré, le pas de temps va être choisi comme la valeur minimale de l'expression (10) sur l'ensemble de la structure, soit finalement :

$$\Delta t_{n+1} = Tol\,.\,\Delta t_n MIN_{x\in\Omega}\left(\frac{1}{\Delta\alpha_n(x)}\right) \tag{11}$$

On ajoute alors des conditions limitant l'accélération et la décélération, ainsi que des bornes minimale et maximale pour l'incrément de temps. Les conditions d'accélération (μ) et décélération ne sont pas modifiables et ont été calibrées sur des cas concrets, alors que les bornes minimale et maximale peuvent être modifiées par l'utilisateur (on donne ici les valeurs par défaut).

$$\begin{aligned} \Delta t_{n+1} &\leqslant \mu\Delta t_n = 1{,}2\,\Delta t_n \\ \Delta t_{n+1} &\geqslant 0.5\,\Delta t_n \\ \Delta t_{n+1} &\leqslant 10\,\Delta t_0 \\ \Delta t_{n+1} &\geqslant 0.001\,\Delta t_0 \end{aligned} \tag{12}$$

Ainsi, le premier pas de temps Δt_0, fourni par l'utilisateur, définit les bornes de l'incrément de temps. Pour le choisir, on conseille à l'utilisateur d'avoir réalisé des calculs préalables avec la méthode de Newton et de connaître la limite élastique de la structure ; il semblerait que le choix d'un premier pas de temps égal à la moitié de la limite élastique permette une bonne efficacité de la méthode.

Copyright 2015 EDF R&D - Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Montrons alors que l'erreur commise est bien maîtrisée. Pour cela on fait l'hypothèse que l'équation 9 est vérifiée. On a donc $\frac{\Delta t_{n+1}}{\Delta t_n}\Delta\alpha_n \leqslant Tol$, et si on considère que les facteurs limite d'accélération et décélération permettent d'écrire $\frac{\Delta t_{n+1}}{\Delta t_n}\approx 1$, on a $\Delta\alpha_n \leqslant Tol$; et de proche en proche :

$$\forall\, n\in\mathbb{N}, \Delta\alpha_n \leqslant Tol \tag{13}$$

Ce qui permet finalement d'écrire :

$$e_{\alpha_{n+1}} \approx \left|\ddot{\alpha}_n\right| \Delta t_{n+1}^2 \leqslant Tol\,\mu^2\left(1+\mu\right) \tag{14}$$

L'erreur est donc maîtrisée.

3.3 Points clés de la méthode

Après cette présentation sommaire, quelques remarques sont ici faites pour comprendre l'intérêt de la méthode et également ses limitations.

Tout d'abord, elle est basée sur une extrapolation des variables internes, déterminée à partir de développement de Taylor. Les développements de Taylor n'étant valables que pour des fonctions suffisamment régulières, la méthode le sera aussi. Ainsi, le franchissement de la limite élastique, ou le passage d'un état de charge à décharge, sont des points pour lesquels la méthode n'est en toute rigueur pas adaptée : la dérivée de l'endommagement est nulle du côté de la décharge ou de la limite élastique, et non nulle du côté de la charge. Toutefois, si les pas de charge sont suffisamment petits, l'approximation peut être faite, dans la mesure où la correction implicite a lieu et donc que les erreurs d'extrapolations sont en partie gommées. Dans les situations instables toutefois, par exemple lorsque la zone endommagée croît brutalement (ce qui se caractérise en général par un snap-back important de la réponse globale force/déplacement), la méthode, bien que robuste, ne peut garantir une réponse fiable, quel que soit l'incrément de charge utilisé : elle n'est pas adaptée à ce genre de situation.

D'après l'équation (8), dans le cas de lois d'endommagement isotropes, du fait de la limitation de α à 1 et du caractère symétrique et défini positif de la matrice élastique locale $\boldsymbol{C}^{\text{elas}}$, la matrice sécante locale $\tilde{\boldsymbol{C}}_{e_{n+1}}$ est toujours symétrique définie positive. De ce fait, par assemblage, la matrice tangente globale $\tilde{\boldsymbol{K}}_{\boldsymbol{n+1}}$ demeure bien conditionnée : les problèmes de robustesse évoqués en introduction, liés au caractère de plus en plus singulier de $\boldsymbol{K}$, doivent donc être éliminés.

De plus, pour un pas de charge donné et pour les lois de comportement développées ici (ENDO_FRAGILE confer [3], ENDO_ISOT_BETON confer [4] et VMIS_ISOT_LINE confer [5]), la matrice tangente locale est connue par extrapolation explicite et demeure constante durant tout le pas de charge. La linéarisation de l'équation d'équilibre (1) par une méthode de Newton-Raphson aboutit à une matrice tangente globale constante $\tilde{\boldsymbol{K}}_{\boldsymbol{n+1}}$; en d'autres termes, l'équation d'équilibre devient linéaire à chaque pas de charge, et sa résolution ne nécessite qu'une itération.

Ensuite, et au risque d'être redondant, la méthode conduit à des forces externes et internes non équilibrées à la fin de chaque pas de charge ; elles ne le sont qu'à la fin de la première phase, c'est-à-dire uniquement en terme de champs extrapolés :

$$\begin{cases} \boldsymbol{F}_{\text{ext}}(t_{n+1}) - \boldsymbol{F}_{\text{int}}(\boldsymbol{\sigma}_{\boldsymbol{n+1}}(\boldsymbol{U}_{\boldsymbol{n+1}}), t_{n+1}) & \neq & \boldsymbol{0} \\ \boldsymbol{F}_{\text{ext}}(t_{n+1}) - \boldsymbol{F}_{\text{int}}(\boldsymbol{U}_{\boldsymbol{n+1}}, \tilde{\boldsymbol{\sigma}}_{n+1}, t_{n+1}) & = & \boldsymbol{0} \end{cases} \tag{15}$$

Par conséquent, lors du calcul, il ne faut pas demander à l'algorithme de vérifier le résidu à une tolérance près, comme il est usuel de le faire (en implicite). De ce point de vue encore, la robustesse est garantie, puisque le critère classique d'arrêt est sans objet.

Pour finir, cette méthode n'a pour vocation que d'augmenter la robustesse du calcul, et pas la qualité de la réponse obtenue. Ainsi, elle introduit une erreur intrinsèque par le truchement de l'extrapolation. Au mieux, on ne peut obtenir que des résultats presque aussi bons que ceux obtenus par une méthode classique implicite de résolution locale. Cette erreur, doit diminuer de façon quasi-quadratique en fonction du pas de charge imposé, sous réserve d'une évolution suffisamment régulière de la variable interne (ce qui exclue de fait les propagations instables de zones endommagées).

Copyright 2015 EDF R&D - Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

L'utilisation de cette méthode nécessite donc un certain regard critique. Afin de se prémunir de fortes erreurs, on préconise de réaliser des calculs avec des incréments de charges différents et petits, afin de vérifier que la solution ne diffère pas trop d'un incrément à l'autre (en d'autres termes, que la solution est proche de la convergence en terme d'incréments de charge). De plus, dans le cas de franchissement de situations instables, les résultats obtenus ne doivent être considérés que pour leur aspect qualitatif.

4 Aspects pratiques d'utilisation

Cette méthode de résolution est activée en en précisant, sous le mot clé simple METHODE de l'opérateur STAT_NON_LINE, METHODE= 'IMPLEX' et en précisant la loi de comportement utilisée sous COMPORTEMENT.
Seules certaines lois de comportement sont actuellement disponibles avec la méthode IMPLEX. Le Tableau 2 récapitule les lois de comportement disponibles suivant le type d'éléments considéré.

Éléments surfaciques ou volumiques	**Éléments de barre**	**Éléments de membrane et grille_membrane**
ELAS	ELAS	ELAS
VMIS_ISOT_LINE	VMIS_ISOT_LINE	
ENDO_ISOT_BETON		
ENDO_FRAGILE		

TABLEAU 2 : lois de comportement disponibles avec la méthode IMPLEX

Pour chaque loi de comportement, la dernière variable interne est modifiée et correspondant dans ce cas au ratio $\frac{\Delta \alpha}{\Delta t}$.
La méthode impose une ré-actualisation de la matrice à chaque incrément (REAC_INCR = 1) et une seule itération. Le résidu d'équilibre est calculé, mais aucun critère ne lui est associé. On pourra se rendre compte d'une éventuelle importante erreur en voyant que le résidu relatif est élevé.

Pour plus de renseignements sur les aspects pratiques, on se référera à [6] et [7]. La méthode IMPLEX est illustrée à travers le cas test SSNP140 [8] et SSLS132.

5 Bibliographie

[1] J. Oliver, A.E. Huespe, J.C. Cante, An implicit/explicit integration scheme to increase computability of non-linear material and contact/friction problems, Computer Methods in Applied Mechanics and Engineering., 197, 1865-1889, 2008.

[2] Manuel de référence Aster R5.03.01, Algorithme non linéaire quasi-statique

[3] Manuel de référence Aster R5.03.18, Loi d'endommagement d'un matériau élastique fragile

[4] Manuel de référence Aster R7.01.04-C, Loi de comportement ENDO_ISOT_BETON

[5] Manuel de référence Aster R5.03.02, Intégration des relations de comportement élasto-plastique de Von Mises

[6] Manuel d'utilisation Aster U4.51.03, Opérateur STAT_NON_LINE

[7] Manuel d'utilisation Aster U4.51.11, Comportements non linéaires

Copyright 2015 EDF R&D - Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

[8] Manuel de validation Aster V06.03.140, Plaque trouée en traction avec la méthode IMPLEX

[9] Manuel d'utilisation Aster U4.34.03, Opérateur DEFI_LIST_INST

6 Description des versions du document

Version Aster	Auteur(s) Organisme(s)	Description des modifications
10.2	R.BARGELLINI-S.FAYOLLE R&D/AMA	Texte initial
10 4	R.BARGELLINI	Ajout de la gestion automatique du pas de temps

Copyright 2015 EDF R&D - Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)