

Manuel de Référence
Fascicule R5.03 : Mécanique non linéaire
Document : R5.03.80

Méthodes de pilotage du chargement

Résumé :

Ce document décrit les méthodes de pilotage du chargement disponibles dans le *Code_Aster* (par un degré de liberté, par longueur d'arc, par incrément de déformation et par prédiction élastique). Elles introduisent une inconnue supplémentaire, l'intensité de la part pilotable du chargement, et une équation supplémentaire, la contrainte de pilotage. Ces méthodes permettent en particulier de calculer la réponse d'une structure qui présenterait des instabilités, aussi bien d'origines géométrique (flambement) que matériau (adoucissement).

1 Principe des méthodes de pilotage du chargement

De manière générale, les fonctionnalités de pilotage disponibles dans le *Code_Aster* permettent de déterminer l'intensité d'une partie du chargement pour satisfaire une contrainte portant sur les déplacements. Leur emploi est limité à des simulations pour lesquelles le temps ne joue pas de rôle physique, ce qui exclut *a priori* les problèmes dynamiques ou visqueux. On peut distinguer trois gammes d'utilisation auxquelles répondent autant de méthodes de pilotage (mot clé facteur *PILOTAGE*) :

- contrôle physique des forces par le déplacement d'un point de la structure (par exemple pour ajuster l'intensité de la force exercée sur un câble de sorte que sa flèche atteigne une valeur donnée) : pilotage par degré de liberté imposé (*TYPE* : 'DDL_IMPO') ;
- suivi d'instabilités géométriques (flambement), la réponse de la structure pouvant exhiber des snap-back « doux » : pilotage par longueur d'arc (*TYPE* : 'LONG_ARC') ;
- suivi d'instabilités matériau (en présence de lois de comportement adoucissantes), la réponse de la structure pouvant exhiber des snap-back « brutaux » : pilotage par la prédiction élastique ou plus généralement par l'incrément de déformation (*TYPE* : 'PRED_ELAS').

Plus précisément, les méthodes de pilotage disponibles dans le *Code_Aster* reposent sur les deux idées suivantes. D'une part, on considère que le chargement (forces extérieures et déplacements imposés) se décompose additivement en deux termes, l'un connu et l'autre dont seule la direction est connue, son intensité η devenant une nouvelle inconnue du problème :

$$\begin{cases} \mathbf{F}_{\text{ext}} = \mathbf{F}_{\text{ext}}^{\text{cst}} + \eta \mathbf{F}_{\text{ext}}^{\text{pilo}} \\ \mathbf{U}_{\text{imp}} = \mathbf{U}_{\text{imp}}^{\text{cst}} + \eta \mathbf{U}_{\text{imp}}^{\text{pilo}} \end{cases} \quad \text{éq 1-1}$$

D'autre part, afin de pouvoir résoudre le problème, on lui associe une nouvelle équation qui porte sur les déplacements et qui dépend de l'incrément de temps : c'est la contrainte de pilotage, qui s'exprime par :

$$P(\Delta \mathbf{U}) = \Delta \tau \quad \text{avec} \quad P(\mathbf{0}) = 0 \quad \text{éq 1-2}$$

où $\Delta \tau$ est indirectement une donnée utilisateur qui s'exprime via le pas de temps courant Δt et un coefficient de pilotage (*COEF_MULT*) par $\Delta \tau = \Delta t / \text{COEF_MULT}$. La condition $P(\mathbf{0}) = 0$ est requise afin d'obtenir un incrément de déplacement d'autant plus petit que le pas de temps est petit.

Finalement, les inconnues du problème deviennent l'incrément des déplacements $\Delta \mathbf{U} = \mathbf{U} - \mathbf{U}^-$, les multiplicateurs de Lagrange $\boldsymbol{\lambda}$ associés aux conditions aux limites et l'intensité du chargement piloté η , baptisée *ETA_PILOTAGE*. Le système non linéaire à résoudre s'écrit dorénavant :

$$\begin{cases} \mathbf{F}_{\text{int}}(\Delta \mathbf{U}; \dots) + \mathbf{B}^T \boldsymbol{\lambda} = \mathbf{F}_{\text{ext}}^{\text{cst}} + \eta \mathbf{F}_{\text{ext}}^{\text{pilo}} \\ \mathbf{B} \mathbf{U} = \mathbf{U}_{\text{imp}}^{\text{cst}} + \eta \mathbf{U}_{\text{imp}}^{\text{pilo}} \\ P(\Delta \mathbf{U}) = \Delta \tau \end{cases} \quad \text{éq 1-3}$$

Remarque 1 :

A l'heure actuelle, les chargements suiveurs (i.e. qui dépendent des déplacements) et les conditions de Dirichlet de type 'DIDI' ne sont pas pilotables. Rien ne s'oppose à ce qu'ils le deviennent dans une version ultérieure du code.

Remarque 2 :

Le chargement ne dépend plus directement du temps mais résulte de la résolution de tout le système non linéaire [éq 1.3]. Cela implique que la part pilotée du chargement ne doit pas dépendre du temps physique, contrairement à un chargement thermique, par exemple, mais correspond à un effort que l'on ajuste pour satisfaire une contrainte cinématique supplémentaire.

2 Résolution du système global

L'introduction d'une nouvelle équation ne perturbe pas outre mesure la méthode de résolution du système non linéaire. En effet, on procède comme en [R5.03.01] par une linéarisation des équations de [éq 1.3] portant sur les forces intérieures et les conditions de Dirichlet :

$$\underbrace{\begin{bmatrix} \frac{\partial \mathbf{F}_{\text{int}}}{\partial \mathbf{U}}(\Delta \mathbf{U}^n) & \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{bmatrix}}_{\mathbf{K}_T} \begin{bmatrix} \delta \mathbf{U} \\ \delta \boldsymbol{\lambda} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{F}_{\text{ext}}^{\text{cst}} - \mathbf{F}_{\text{int}}(\Delta \mathbf{U}^n) - \mathbf{B}^T \boldsymbol{\lambda}^n \\ \mathbf{U}_{\text{imp}}^{\text{cst}} - \mathbf{B} \mathbf{U}^n \end{bmatrix}}_{\mathbf{R}^{\text{cst}}} + \eta \underbrace{\begin{bmatrix} \mathbf{F}_{\text{ext}}^{\text{pilo}} \\ \mathbf{U}_{\text{imp}}^{\text{pilo}} \end{bmatrix}}_{\mathbf{R}^{\text{pilo}}} \quad \text{éq 2-1}$$

On peut maintenant exprimer les corrections de déplacements $\delta \mathbf{U}$ et de multiplicateurs de Lagrange $\delta \boldsymbol{\lambda}$ en fonction de η moyennant la résolution du système linéaire [éq 2-1] par rapport à chacun des deux seconds membres :

$$\begin{bmatrix} \delta \mathbf{U} \\ \delta \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \delta \mathbf{U}^{\text{cst}} \\ \delta \boldsymbol{\lambda}^{\text{cst}} \end{bmatrix} + \eta \begin{bmatrix} \delta \mathbf{U}^{\text{pilo}} \\ \delta \boldsymbol{\lambda}^{\text{pilo}} \end{bmatrix} \quad \text{où} \quad \begin{bmatrix} \delta \mathbf{U}^{\text{cst}} \\ \delta \boldsymbol{\lambda}^{\text{cst}} \end{bmatrix} = \mathbf{K}_T^{-1} \mathbf{R}^{\text{cst}} \quad \text{et} \quad \begin{bmatrix} \delta \mathbf{U}^{\text{pilo}} \\ \delta \boldsymbol{\lambda}^{\text{pilo}} \end{bmatrix} = \mathbf{K}_T^{-1} \mathbf{R}^{\text{pilo}} \quad \text{éq 2-2}$$

On peut maintenant substituer la correction de déplacement $\delta \mathbf{U}$ en fonction de son expression [éq 2-2] dans l'équation de contrôle du pilotage du système [éq 1.3] ; il en résulte une équation scalaire en η :

$$\tilde{P}(\eta) \stackrel{\text{déf.}}{=} P(\Delta \mathbf{U}^n + \delta \mathbf{U}^{\text{cst}} + \eta \delta \mathbf{U}^{\text{pilo}}) = \Delta \tau \quad \text{éq 2-3}$$

La méthode de résolution de cette équation dépend de la nature du contrôle de pilotage adopté cf. [§3]. Finalement, il ne reste plus qu'à réactualiser les inconnues déplacements et multiplicateurs de Lagrange :

$$\begin{cases} \Delta \mathbf{U}^{n+1} = \Delta \mathbf{U}^n + \delta \mathbf{U}^{\text{cst}} + \eta \delta \mathbf{U}^{\text{pilo}} \\ \lambda^{n+1} = \lambda^n + \delta \lambda^{\text{cst}} + \eta \delta \lambda^{\text{pilo}} \end{cases} \quad \text{éq 2-4}$$

Remarque 3 :

Au cours des itérations de Newton, il peut arriver que l'équation [éq 2-3] n'admette pas de solution, sans pour autant qu'il y ait une erreur d'utilisation. Dans ce cas, on décide alors de déterminer η comme la valeur qui minimise \tilde{P} , à condition que l'on vérifie $\tilde{P}(\eta) > \Delta\tau$; comme l'incrément de déplacement est d'autant plus petit que $\Delta\tau$ est petit, une telle condition impose grossièrement que l'incrément de déplacement est au moins aussi grand (sans entrer dans des définitions mathématiques précises) que celui prescrit par l'équation de pilotage $\tilde{P}(\eta) = \Delta\tau$. De plus, on impose a minima une itération de Newton supplémentaire, afin de vérifier, à convergence, non seulement les équations d'équilibre et les conditions de Dirichlet, mais aussi l'équation de pilotage.

Remarque 4 :

Il n'y a pas de linéarisation par rapport à la variable de pilotage η . De la sorte, on préserve toute la méthodologie de réactualisation de l'opérateur tangent déjà mise en œuvre pour les calculs sans pilotage. De plus, la structure « bande » de la matrice tangente est conservée.

Remarque 5 :

Ce mécanisme de résolution est incompatible avec l'emploi de la recherche linéaire. En fait, ce serait possible en présence d'une fonction de pilotage P linéaire, voir Shi and Crisfield [bib4], mais ce n'est plus vrai dans le cas général. C'est pourquoi l'emploi simultané de la recherche linéaire et du pilotage est interdit.

3 Equation de contrôle du pilotage

3.1 Pilotage par contrôle d'un degré de liberté des déplacements : DDL_IMPO

Pour ce premier type de pilotage, la fonction P se borne à extraire un degré de liberté de l'incrément de déplacement. En particulier, il s'agit donc d'une fonction linéaire :

$$P(\Delta \mathbf{U}) = \mathbf{L} \cdot \Delta \mathbf{U} \quad \text{avec} \quad \mathbf{L} = \begin{bmatrix} 0 & \cdots & 0 & \underset{\substack{\uparrow \\ \text{noeud } n, \text{ ddl } i}}{1} & 0 & \cdots & 0 \end{bmatrix} \quad \text{éq 3.1-1}$$

où le vecteur nodal \mathbf{L} est nul partout excepté pour le degré de liberté à extraire où il vaut 1. L'équation [éq 2-3] se réduit alors également à une équation linéaire qui conduit à :

$$\eta = \frac{\Delta\tau - \mathbf{L} \cdot \Delta\mathbf{U}^n - \mathbf{L} \cdot \delta\mathbf{U}^{\text{cst}}}{\mathbf{L} \cdot \delta\mathbf{U}^{\text{pilo}}} \quad \text{éq 3.1-2}$$

On notera qu'il n'y a pas de solution lorsque la correction de déplacement piloté $\delta\mathbf{U}^{\text{pilo}}$ ne permet pas d'ajuster le degré de liberté requis, ce qui peut arriver si, par erreur, on bloque le degré de liberté en question.

3.2 Pilotage par longueur d'arc : LONG_ARC

Une autre forme de pilotage très largement utilisée consiste à contrôler la norme de l'incrément de déplacement (par rapport à certains noeuds et certaines composantes) : on parle alors de pilotage par longueur d'arc, voir Bonnet and Wood [bib1]. Plus précisément, la fonction P s'exprime par :

$$P(\Delta\mathbf{U}) = \|\Delta\mathbf{U}\|_{\mathbf{L}} = \sqrt{\Delta\mathbf{U} \cdot \mathbf{L} \cdot \Delta\mathbf{U}} \quad \text{éq 3.2-1}$$

où, à nouveau, le vecteur nodal \mathbf{L} permet de sélectionner les degrés de liberté employés pour le calcul de la norme (il vaut 1 pour les ddl sélectionnés, 0 ailleurs). Dans ce cas, l'équation de pilotage se réduit à une équation du second degré :

$$\begin{aligned} & [\delta\mathbf{U}^{\text{pilo}} \cdot \mathbf{L} \cdot \delta\mathbf{U}^{\text{pilo}}] \eta^2 + 2[(\Delta\mathbf{U}^n + \delta\mathbf{U}^{\text{cst}}) \cdot \mathbf{L} \cdot \delta\mathbf{U}^{\text{pilo}}] \eta \\ & + [(\Delta\mathbf{U}^n + \delta\mathbf{U}^{\text{cst}}) \cdot \mathbf{L} \cdot (\Delta\mathbf{U}^n + \delta\mathbf{U}^{\text{cst}}) - \Delta\tau^2] = 0 \end{aligned} \quad \text{éq 3.2-2}$$

Cette équation peut ne pas admettre de solution. Dans ce cas, on choisit la valeur η qui minimise le polynôme [éq 3.2-2]. On vérifie alors bien $\tilde{P}(\eta) > \Delta\tau$. Dans le cas contraire, elle admet deux racines (ou une racine double). On choisit celle des deux qui minimise l'angle formé par $\Delta\mathbf{U}^{\text{avant}}$ et $\Delta\mathbf{U}^{n+1}$ (où $\Delta\mathbf{U}^{\text{avant}}$ est l'incrément de déplacement solution du pas de temps précédant), c'est-à-dire celle qui maximise le cosinus de cet angle dont l'expression est :

$$\cos(\Delta\mathbf{U}^{\text{avant}}, \Delta\mathbf{U}^{n+1}) = \frac{(\Delta\mathbf{U}^n + \delta\mathbf{U}^{\text{cst}} + \eta\delta\mathbf{U}^{\text{pilo}}) \cdot \Delta\mathbf{U}^{\text{avant}}}{\|\Delta\mathbf{U}^n + \delta\mathbf{U}^{\text{cst}} + \eta\delta\mathbf{U}^{\text{pilo}}\| \|\Delta\mathbf{U}^{\text{avant}}\|} \quad \text{éq 3.2-3}$$

3.3 Pilotage par l'incrément de déformation : PRED_ELAS

Les deux derniers modes de pilotage, contrôle par incrément de déformation et contrôle par prédiction élastique, cf. Lorentz et Badel [bib3], sont activés par le même mot clé PRED_ELAS. En fait, le second dépend explicitement de la loi de comportement et n'est implanté que pour certaines lois (ENDO_LOCAL et BETON_ENDO_LOCAL) ; lorsqu'il est disponible, il est employé. Pour les autres lois, c'est le pilotage par incrément de déformation qui est activé. Il est probable que ce mécanisme évolue dans le cadre de la version 6 du Code_Aster et que ces deux modes de pilotage soient activables indépendamment l'un de l'autre.

Le pilotage par incrément de déformation consiste à exiger que l'incrément de déformation du pas courant reste proche en direction de la déformation au début du pas de temps, et ce pour au moins un point de Gauss de la structure. On requiert ainsi qualitativement qu'**a minima** un point de la structure conserve le mode de déformation qu'il avait au préalable (par exemple, traction dans une direction donnée). Mathématiquement, on peut rendre compte de cette exigence moyennant le choix de la fonction de pilotage suivante :

$$P(\Delta U) = \max_g \frac{\boldsymbol{\varepsilon}_g^-}{\|\boldsymbol{\varepsilon}_g^-\|} \cdot \Delta \boldsymbol{\varepsilon}_g \quad \text{avec} \quad \boldsymbol{\varepsilon}_g^- = \mathbf{B}_g \mathbf{U}^- \quad \text{et} \quad \Delta \boldsymbol{\varepsilon}_g = \mathbf{B}_g \Delta U \quad \text{éq 3.3-1}$$

où l'indice g balaie les points de Gauss de la structure et où la déformation en un point de Gauss se déduit du vecteur nodal des déplacements via l'emploi des matrices « partie symétrique du gradient des fonctions de forme » \mathbf{B}_g (à ne pas confondre avec la matrice des conditions de Dirichlet). Le contrôle du pilotage en fonction de η s'écrit alors :

$$\tilde{P}(\eta) = \max_g \underbrace{\left(A_g^{(0)} + \eta A_g^{(1)} \right)}_{L_g(\eta)} = \Delta \tau \quad \text{avec} \quad \begin{cases} A_g^{(0)} = \frac{\boldsymbol{\varepsilon}_g^-}{\|\boldsymbol{\varepsilon}_g^-\|} \cdot \mathbf{B}_g (\Delta \mathbf{U}^n + \delta \mathbf{U}^{\text{cst}}) \\ A_g^{(1)} = \frac{\boldsymbol{\varepsilon}_g^-}{\|\boldsymbol{\varepsilon}_g^-\|} \cdot \mathbf{B}_g \delta \mathbf{U}^{\text{pilo}} \end{cases} \quad \text{éq 3.3-2}$$

Une telle fonction est convexe et linéaire par morceaux. Elle admet généralement aucune, une ou deux solutions, cf. [Figure 3.3-a]. Lorsqu'elle n'admet pas de solutions, on choisit comme précédemment la valeur η qui minimise $\tilde{P}(\eta)$; elle remplit la condition $\min \tilde{P}(\eta) > \Delta \tau$. Lorsqu'elle admet deux solutions, on choisit celle qui conduit au $\tilde{\mathbf{U}}(\eta)$ le plus proche de \mathbf{U}^- .

Remarque 6 :

En présence de grandes déformations, on peut généraliser la fonction de pilotage [éq 3.3-1] en employant des déformations de Green - Lagrange (mesure lagrangienne des déformations dans la configuration initiale) :

$$P(\Delta \mathbf{U}) = \text{Max}_g \frac{\mathbf{e}_g^-}{\|\mathbf{e}_g^-\|} \cdot \Delta \mathbf{e}_g \quad \text{avec} \quad \mathbf{e} = \frac{1}{2}(\mathbf{F}^T \mathbf{F} - \mathbf{Id}) \quad \mathbf{F} = \mathbf{Id} + \nabla \mathbf{u} \quad \Delta \mathbf{e} = \mathbf{e} - \mathbf{e}^-$$

éq 3.3-3

Toutefois, on n'aboutirait plus comme précédemment à une fonction affine par morceaux. Pour y remédier, on décide de procéder à une linéarisation de $\Delta \mathbf{e}_g$ par rapport à $\Delta \mathbf{U}$. P a alors une expression similaire à [éq 3.3-2] avec :

$$\begin{cases} A_g^{(0)} = \frac{\mathbf{e}_g^-}{\|\mathbf{e}_g^-\|} \cdot \text{sym} \left[\mathbf{F}_g^T \nabla_g (\Delta \mathbf{U}^n + \delta \mathbf{U}^{\text{cst}}) \right] \\ A_g^{(1)} = \frac{\mathbf{e}_g^-}{\|\mathbf{e}_g^-\|} \cdot \text{sym} \left[\mathbf{F}_g^T \delta \mathbf{U}^{\text{pilo}} \right] \end{cases}$$

éq 3.3-4

où $\nabla_g \mathbf{U}$ désigne le gradient (non symétrisé) des déplacements \mathbf{U} évalué au point de Gauss d'indice g .

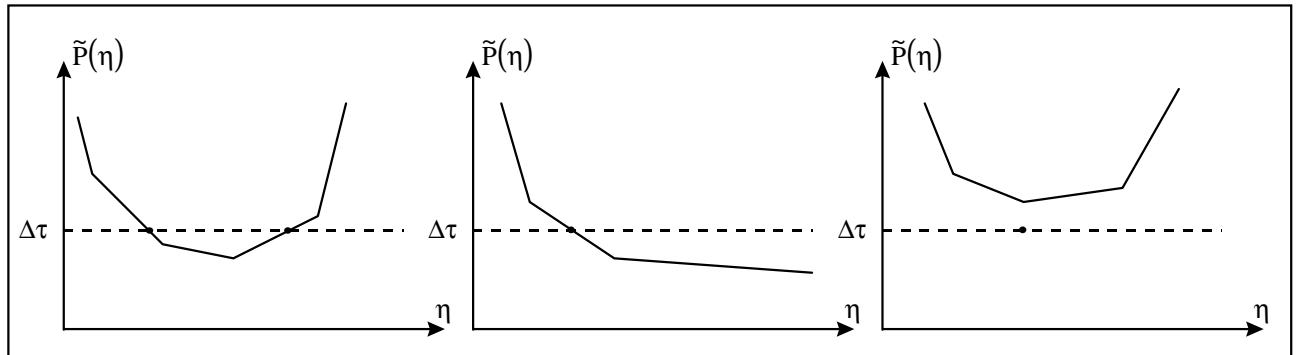


Figure 3.3-a : Les différents cas de figure pour l'équation [éq 3.3-2] : deux, une ou aucune solutions

Pour résoudre l'équation [éq 3.3-2], on propose l'algorithme présenté dans la table [An1-1], [§5]. Il est basé sur la construction d'intervalles emboîtés : les bornes du dernier d'entre eux sont les solutions de l'équation et, comme annoncé précédemment, on choisit celle qui conduit au $\tilde{\mathbf{U}}(\eta)$ le plus proche de

\mathbf{U}^- . Cet algorithme, rapide, s'appuie sur la résolution de G équations scalaires linéaire, où G désigne le nombre total de points de Gauss. L'algorithme peut prendre fin prématurément lorsque l'un des intervalles est vide, ce qui signifie que l'équation [éq 3.3-2] n'admet pas de solutions.

Lorsqu'il n'existe pas de solutions à l'équation [éq 3.3-2], il s'agit alors, conformément à la remarque 3, de minimiser $\tilde{P}(\eta)$, problème qui peut encore s'exprimer comme la minimisation d'une fonction linéaire à deux variables sous G contraintes linéaires inégalités :

$$\min_{\eta} \max_g L_g(\eta) \quad \Leftrightarrow \quad \min_{\substack{\eta, y \\ \forall g \quad l_g(\eta, y) \leq 0}} y \quad \text{avec} \quad l_g(\eta, y) = L_g(\eta) - y \quad \text{éq 3.3-5}$$

On emploie ici l'algorithme du simplexe, cf. Bonnans et al. [bib2], dont l'application à notre problème est présentée dans la table [An1-2], [§5].

3.4 Pilotage par la prédiction élastique : PRED_ELAS

Si le pilotage par l'incrément de déformation s'avère suffisant pour suivre des solutions dissipatives dans la plupart des instabilités matériaux, l'existence de solutions n'est néanmoins pas prouvée. On lui préfère alors une méthode de pilotage fondée sur la prédiction élastique pour laquelle l'existence de solutions est démontrée mais qui, en revanche, est spécifique à chaque loi de comportement (implantée uniquement pour les lois ENDO_LOCAL et BETON_ENDO_LOCAL). Plus précisément, lorsque la loi de comportement est gouvernée par un seuil, on définit P comme le maximum sur tous les points de Gauss de la valeur de la fonction seuil dans le cas d'un essai élastique (réponse incrémentale élastique du matériau).

Ainsi, considérons que l'état du matériau est décrit par la déformation $\boldsymbol{\varepsilon}$ et un ensemble de variables internes a . Appelons respectivement $\boldsymbol{\sigma}(\boldsymbol{\varepsilon}, a)$ et $A(\boldsymbol{\varepsilon}, a)$ les contraintes et les forces thermodynamiques associées à a . Supposons en outre que les lois d'évolution de a soient gouvernées par un seuil $f(A, \boldsymbol{\varepsilon}, a)$ et une fonction d'écoulement $G(A, \boldsymbol{\varepsilon}, a)$ de la manière suivante :

$$\dot{a} = \lambda G(A, \boldsymbol{\varepsilon}, a) \quad \text{avec} \quad \lambda \geq 0 \quad f(A, \boldsymbol{\varepsilon}, a) \leq 0 \quad \lambda f(A, \boldsymbol{\varepsilon}, a) = 0 \quad \text{éq 3.4-1}$$

Une telle formulation englobe la plupart des modèles de comportement dissipatifs et indépendants de la vitesse de chargement. La fonction seuil vaut alors pour un essai élastique :

$$f^{\text{el}}(\boldsymbol{\varepsilon}) = f(A(\boldsymbol{\varepsilon}, a^-), \boldsymbol{\varepsilon}, a^-) \quad \text{éq 3.4-2}$$

On simplifie le problème en linéarisant f^{el} par rapport à $\boldsymbol{\varepsilon}$ au voisinage d'un point $\boldsymbol{\varepsilon}^*$ qu'on définira plus tard :

$$f_L^{\text{el}}(\boldsymbol{\varepsilon}) \stackrel{\text{def}}{=} f^{\text{el}}(\boldsymbol{\varepsilon}^*) + \left(\frac{\partial f}{\partial A} \frac{\partial A}{\partial \boldsymbol{\varepsilon}} + \frac{\partial f}{\partial \boldsymbol{\varepsilon}} \right) \bigg|_{\boldsymbol{\varepsilon}^*} \cdot (\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^*) \quad \text{éq 3.4-3}$$

Finalement, la fonction de contrôle du pilotage est définie comme le maximum de f_L^{el} par rapport à tous les points de Gauss g , fonction qui ne dépend que de $\Delta \mathbf{U}$:

$$P(\Delta \mathbf{U}) = \text{Max}_g f_L^{\text{el}}(\boldsymbol{\varepsilon}^- + \mathbf{B}_g \Delta \mathbf{U}) \quad \text{avec} \quad \Delta \boldsymbol{\varepsilon}_g = \mathbf{B}_g \Delta \mathbf{U} \quad \text{éq 3.4-4}$$

Il reste à définir $\boldsymbol{\varepsilon}^*$, point au voisinage duquel on effectue la linéarisation. Il est choisi comme le point le plus proche de $\boldsymbol{\varepsilon}^-$ qui réalise :

$$\boldsymbol{\varepsilon}^* = \tilde{\boldsymbol{\varepsilon}}(\boldsymbol{\eta}^*) \quad \text{avec} \quad \tilde{\boldsymbol{\varepsilon}}(\boldsymbol{\eta}) = \mathbf{B}_g (\mathbf{U}^- + \Delta \mathbf{U}^n + \delta \mathbf{U}^{\text{cst}} + \boldsymbol{\eta} \delta \mathbf{U}^{\text{pilo}}) \quad \text{et} \quad \boldsymbol{\eta}^* = \arg \min_{\boldsymbol{\eta}} |\tilde{f}_L^{\text{el}}(\boldsymbol{\eta})| \quad \text{éq 3.4-5}$$

Une interprétation graphique de ce choix est donnée sur la [Figure 3.4-a]. Finalement, l'équation de contrôle du pilotage s'écrit :

$$\max_g \left[\underbrace{A_g^{(1)} \eta + A_g^{(0)}}_{f_L^{el}(\eta)} \right] = \Delta \tau \quad \begin{cases} A^{(0)} = f^{el}(\boldsymbol{\varepsilon}^*) + \left(\frac{\partial f}{\partial A} \frac{\partial A}{\partial \boldsymbol{\varepsilon}} + \frac{\partial f}{\partial \boldsymbol{\varepsilon}} \right) \Big|_{\boldsymbol{\varepsilon}^*} \cdot (\mathbf{B}_g (\mathbf{U}^- + \Delta \mathbf{U}^n + \delta \mathbf{U}^{est}) - \boldsymbol{\varepsilon}^*) \\ A^{(1)} = \left(\frac{\partial f}{\partial A} \frac{\partial A}{\partial \boldsymbol{\varepsilon}} + \frac{\partial f}{\partial \boldsymbol{\varepsilon}} \right) \Big|_{\boldsymbol{\varepsilon}^*} \cdot (\mathbf{B}_g \delta \mathbf{U}^{pilo}) \end{cases} \quad \text{éq 3.4-6}$$

On est ainsi ramené à un problème identique à celui du pilotage par l'incrément de la déformation. On emploie bien entendu les mêmes algorithmes de résolution que ceux présentés au [§5].

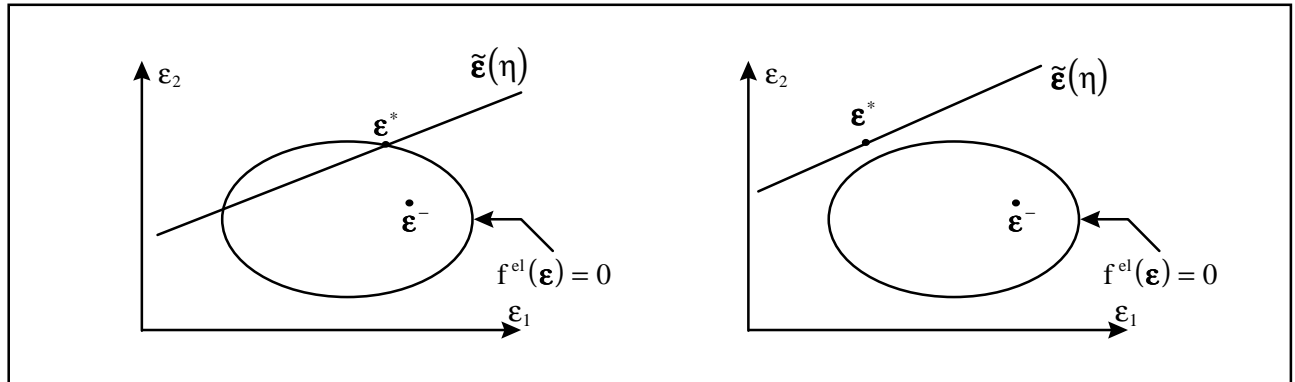


Figure 3.4-a : Définition de $\boldsymbol{\varepsilon}^*$ en fonction des positions relatives de la surface seuil f^{el} et des déformations $\tilde{\boldsymbol{\varepsilon}}$

4 Bibliographie

- [1] Bonnet J. and Wood R. D. [1997]. Nonlinear continuum mechanics for finite element analysis. Cambridge university press.
- [2] Bonnans J. F., Gilbert J.-C., Lemaréchal C. et Sagastizabal C. [1997]. Optimisation numérique : aspects théoriques et pratiques. Mathématiques et Applications, 27, ed. Springer.
- [3] Lorentz E. and Badel P. [2001]. A load control method for damage finite element simulations. Note interne EDF R&D, à paraître.
- [4] Shi J. and Crisfield M. A. [1995]. Combining arc-length control and line searches in path following. Comm. Num. Meth. Eng. 11, pp. 793-803.

Annexe 1 : Algorithmes de résolution

(1)	Initialisation de l'intervalle	$I_0 =]-\infty \quad +\infty[$
(2)	Boucle sur les points de Gauss g	
(2.1)	Racine de la fonction linéaire active	$\eta_g \quad \text{tq} \quad L_g(\eta_g) = \Delta\tau$
(2.2)	Construction de l'intervalle suivant	
(2.2.1)	Si la fonction linéaire active est croissante	$A_g^{(1)} > 0 \Rightarrow I_g = I_{g-1} \cap]-\infty \quad \eta_g]$
(2.2.2)	Si la fonction linéaire active est décroissante	$A_g^{(1)} < 0 \Rightarrow I_g = I_{g-1} \cap [\eta_g \quad +\infty[$
(2.3)	Arrêter si l'intervalle est vide	
(3)	Les solutions sont les bornes de l'intervalle	$\eta \in \text{Fr}(I_G) \Rightarrow \max_g L_g(\eta) = \Delta\tau$

Table An1-1 : Algorithme de résolution de l'équation affine par morceaux

(1)	Initialisation avec η_0 donné	
(1.1)	Gradient de la fonction à minimiser	$\mathbf{G} = (0, -1)$
(1.2)	Sommet initial	$\mathbf{S}_0 = (\eta_0, y_0) \quad \text{avec} \quad y_0 = \max_g L_g(\eta_0)$
(1.3)	Contrainte activée	$g_0 \quad \text{tq} \quad y_0 = L_{g_0}(\eta_0)$
(2)	Exploration des sommets successifs : boucle sur s	
(2.1)	Définition d'une direction de descente	$\mathbf{D}_s = \mathbf{G} - \frac{\mathbf{G} \cdot \mathbf{N}_s}{\mathbf{N}_s \cdot \mathbf{N}_s} \mathbf{N}_s \quad \text{avec} \quad \mathbf{N}_s = (-A_{g_s}^{(1)}, 1)$
(2.2)	Minimum trouvé si $\mathbf{D}_s \cdot \mathbf{G} = 0$ (fond plat)	
(2.3)	Avancée admissible ρ_g pour chaque contrainte g	$\forall g \quad \rho_g = \max_{1_g(\mathbf{S}_s + \rho \mathbf{D}_s) \leq 0} \rho$
(2.4)	Avancée effective $\bar{\rho}_s$	$\bar{\rho}_s = \min_g \rho_g$
(2.5)	Minimum trouvé si $\bar{\rho}_s = 0$	
(2.5)	Sommet suivant	$\mathbf{S}_{s+1} = \mathbf{S}_s + \bar{\rho}_s \mathbf{D}_s$
(2.6)	Contrainte activée suivante g_{s+1}	$g_{s+1} \quad \text{tq} \quad \bar{\rho}_s = \rho_{g_{s+1}}$

Table An1-2 : Algorithme de minimisation de la fonction affine par morceaux

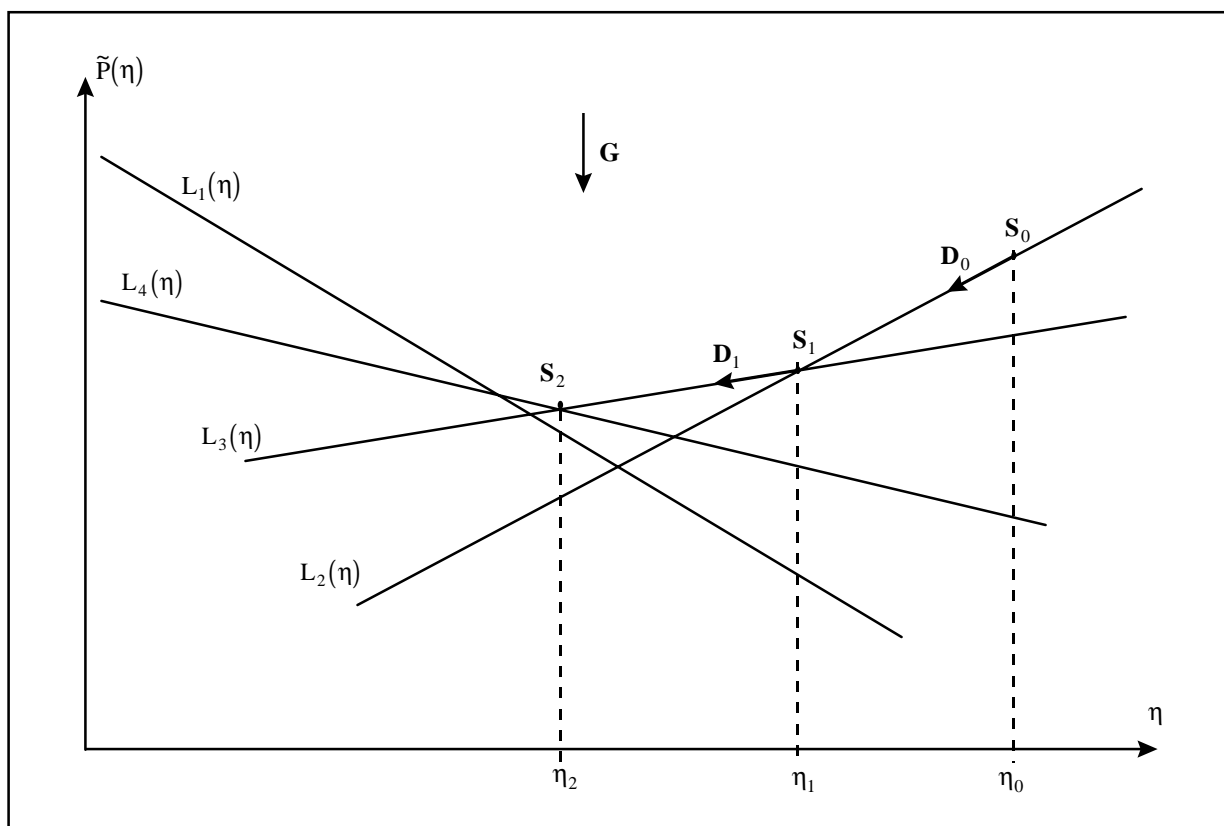


Figure An1-a : Illustration graphique de l'algorithme de minimisation

Page laissée intentionnellement blanche.