

# Remerciements

Ma gratitude s'adresse particulièrement à Monsieur Adnan IBRAHIMBEGOVIĆ, Directeur du DEA M.a.i.S.E., dont la compétence, l'ouverture d'esprit et l'enthousiasme m'ont permis de réaliser mes études et ce stage à l'ENS Cachan. Je le remercie pour son soutien amical, sa patience illimitée, ses conseils et ses inspirations.

Je remercie Monsieur Arnaud DELAPLACE pour son soutien permanent et ses conseils utiles surtout dans le cadre de la mécanique des structures.

Je remercie Madame Catherine KNOPF-LENOIR-VAYSSADE et Monsieur Pierre VILLON, Professeurs à l'Université de Technologie de Compiègne, pour leurs conseils dans le traitement des exemples numériques.

Je remercie également Monsieur Matěj LEPŠ, Ingénieur de la Faculté de Génie Civil à Université Technique de Prague, pour sa patience illimitée et ses conseils utiles surtout dans le cadre des algorithmes génétiques.

Je remercie aussi Monsieur Zdeněk BITTNAR, Directeur de la Faculté de Génie Civil à Université Technique de Prague, pour me donner la possibilité de réaliser mes études et ce stage à l'ENS Cachan et pour son soutien permanent.

Enfin, ma reconnaissance et mes remerciements vont aussi à ma famille, mes amis et collègues qui m'ont aidé et soutenu durant les quatre mois passés à préparer et à rédiger ce mémoire.

# Table des matières

<b>L'introduction</b>	<b>1</b>
<b>1 La description du problème mécanique non-linéaire</b>	<b>3</b>
1.1 Le modèle 2D de la poutre géométriquement exacte de Reissner en grands déplacements . . . . .	3
1.2 Approximation par les éléments finis pour la poutre de Reissner . . . . .	5
<b>2 Le problème d'optimisation couplé</b>	<b>8</b>
2.1 L'optimisation de forme . . . . .	8
2.2 Le contrôle de chargement . . . . .	9
<b>3 La procédure de résolution - méthode basée sur l'algorithme génétique</b>	<b>11</b>
3.1 L'algorithme SADE . . . . .	12
3.2 L'algorithme GRADE . . . . .	13
<b>4 Les exemples numériques</b>	<b>15</b>
4.1 Les exemples de contrôle de chargement . . . . .	15
4.1.1 L'approche non-simultanée . . . . .	15
4.1.2 L'approche simultanée . . . . .	21
4.2 L'exemple d'optimisation de forme . . . . .	23
4.2.1 L'approche non-simultanée . . . . .	25
4.2.2 L'approche simultanée . . . . .	25
<b>Conclusions</b>	<b>28</b>
<b>Bibliographie</b>	<b>30</b>
<b>A La procédure de solution pour le problème mécanique non-linéaire</b>	<b>32</b>
<b>B La formulation discrétisée du problème d'optimisation couplé</b>	<b>34</b>
B.1 La discrétisations du problème mécanique non-linéaire . . . . .	34
B.2 La discrétisation pour le problème du contrôle de chargement . . . . .	36
B.3 La discrétisation pour le problème d'optimisation de forme . . . . .	36

<b>C</b>	<b>Les limites de la fonction maximisée pour les trois problèmes de contrôle</b>	<b>38</b>
<b>D</b>	<b>La solution du problème de la 'lettre B' par l'algorithme SADE</b>	<b>41</b>
<b>E</b>	<b>Le contrôle de chargement pour le problème de 'lettre T' par l'approximation diffuse</b>	<b>43</b>

# L'introduction

Les structures modernes doivent être souvent dimensionnées pour résister aux grands déplacements et rotations et rester entièrement fonctionnelles. De plus, la phase de construction doit aussi être contrôlée lors de l'assemblage des éléments de la structure. Enfin, les critères économiques sont de plus en plus importants et sont la raison pour décrire ce problème sur des bases plus théorique. Notamment, les méthodes d'optimisation peuvent être utilisées pour aider la phase de conception et pour trouver les réductions de forme désirées. Analogiquement, les méthodes de contrôle sont un moyen utile pour estimer le chargement minimal pour obtenir la déformation désirée d'une construction. Les deux problèmes, l'optimisation de forme ou le contrôle de chargement, peuvent être présentés comme la minimisation de valeur de fonction coût, qui précise notre objectif. La plus grande différence entre ces deux problèmes est le choix de variables qui définit la fonction de coût : les variables de forme de construction sont typiquement liées avec les propriétés mécaniques ( ex. module d'Young ) ou géométriques de la structure ( ex. coordonnées particulière de la configuration initial ), tandis que les variables de contrôle sont liées avec les forces appliquées à la construction. Au lieu de s'appesantir sur cette différence et dénouer le problème d'optimisation et de contrôle de façon différente, comme c'est habituellement effectué, on se repère aux propriétés, qui sont les mêmes pour les deux problèmes, ce qui nous donne la possibilité de les présenter de façon plus homogène et puis développer une nouvelle procédure de résolution.

Dans la première partie du papier, on présente les approches de la formulation du problème d'optimisation ou de contrôle comme un problème couplé avec le problème mécanique de structure non-linéaire. L'approche plus traditionnelle (ex. voir [7] pour un rappel assez récent) résout le problème d'optimisation ou de contrôle d'un côté et le problème mécanique non-linéaire de l'autre côté separement. Typiquement on utilise deux différents programmes, un pour la mécanique et l'autre pour l'optimisation ou contrôle. Dans ce cas la communication entre chaque procédure de résolution est réduite au minimum (ex. [15] ou [11]).

La deuxième approche pour la formulation du problème couplé utilise la méthode classique de multiplicateurs de Lagrange ([9] ou [14]) pour élever le problème mécanique non-linéaire et d'optimisation ou de contrôle au même niveau pour les résoudre simultanément. Dans ce cas, les équations d'équilibre de mécanique non-linéaire sont d'abord réduit à la limitation du problème seulement avec le respect d'admissibilité d'état de structure imposée, ses déplacements et rotations. En utilisant les multiplicateurs de Lagrange les équations d'équilibre sont changées dans une équation gouvernant toutes les équations à résoudre dans le problème couplé et les relations intrinsèque des variables d'état ( déplacements et rotations ) avec le respect des variables d'opti-

misation ou de contrôle peuvent être éliminer par changement toutes les variables aux variables indépendentes. Cette idée est aussi développée dans le cadre d'approximation discrète basée sur la méthode des éléments finis, ce qui nous offre le modèle des élément finis avec les degrés de liberté ne concernant pas seulement les déplacement et rotation mais aussi les variables d'optimisation ou de contrôle. Le développement détaillé est présenté sur le modèle 2D de la poutre géométriquement exacte [6].

Dans la deuxième partie du papier on présente la méthode numérique qui peut résoudre les problèmes susmentionnés. Dans ce stage on utilise les algorithmes génétiques [1], [10]. On a utilisé l'algorithme SADE [4], qui était déjà testé pour résoudre différents problèmes dans le cadre du génie civil [5]. Puis on présente une modification de cet algorithme qui a élevé la vitesse de la convergence d'algorithme et augmenté son efficacité.

La charpente du papier est suivant. Dans la chapitre prochaine on présente brièvement le modèle 2D choisi de la poutre géométriquement exacte, capable de représenter les grands déplacements et rotations. Les formulations théorétiques des problèmes d'optimisation et de contrôle sont présenté, dans la chapitre 2. La procédure de résolution est présentée dans la chapitre 3. Plusieurs exemples sont présentés dans la chapitre 4.

# Chapitre 1

## La description du problème mécanique non-linéaire

Dans ce chapitre on donne la formulation détaillée pour le modèle 2D de la poutre incurvée géométriquement exacte avec une cinématique non-linéaire (voir [6]). Puis on introduit l'approximation de ce modèle pour une modélisation éléments finis. Comme pour rechercher une solution traditionnelle pour un problème de mécanique non-linéaire, on utilise l'analyse incrémentale, qui est brièvement expliquée dans l'annexe A.

### 1.1 Le modèle 2D de la poutre géométriquement exacte de Reissner en grands déplacements

Dans ce modèle, on utilise l'hypothèse de Marguerre pour la poutre incurvée et l'hypothèse de Timoshenko pour tenir compte des déformations de cisaillement. La formulation proposée est obtenue par linéarisation de la théorie de la poutre de Reissner. D'abord on suit Ibrahimbegović et al. en supposant que la déformation initiale de la poutre mince peut être décrite comme la dérivation de la poutre droite par une transformation isométrique. Si  $(\mathbf{g}_1, \mathbf{g}_2)$  sont les vecteurs de base du système orthogonal de coordonnées de référence, les vecteurs de base pour la poutre incurvée  $(\hat{\mathbf{g}}_1, \hat{\mathbf{g}}_2)$  sont calculés par :

$$\begin{bmatrix} \hat{\mathbf{g}}_1^T \\ \hat{\mathbf{g}}_2^T \end{bmatrix} = \begin{bmatrix} \cos \alpha & \sin \alpha \\ -\sin \alpha & \cos \alpha \end{bmatrix} = \begin{bmatrix} \mathbf{g}_1^T \\ \mathbf{g}_2^T \end{bmatrix} \quad (1.1)$$

La mesure des déformations généralisées sont décrites suivant Reissner. Pour cette raison, on se place dans un système de coordonnées mobile. Ainsi on peut obtenir par la rotation de système  $(\hat{\mathbf{g}}_1, \hat{\mathbf{g}}_2)$ , un axe (dit  $\mathbf{n}$ ) orthogonal à la section et l'autre (dit  $\mathbf{t}$ ) dans le plan de section. Donc (voir Image 1.1)

$$\begin{bmatrix} \mathbf{n}^T \\ \mathbf{t}^T \end{bmatrix} = \begin{bmatrix} \cos \psi & \sin \psi \\ -\sin \psi & \cos \psi \end{bmatrix} \begin{bmatrix} \hat{\mathbf{g}}_1^T \\ \hat{\mathbf{g}}_2^T \end{bmatrix} = \begin{bmatrix} \cos(\alpha + \psi) & \sin(\alpha + \psi) \\ -\sin(\alpha + \psi) & \cos(\alpha + \psi) \end{bmatrix} \begin{bmatrix} \mathbf{g}_1^T \\ \mathbf{g}_2^T \end{bmatrix} \quad (1.2)$$

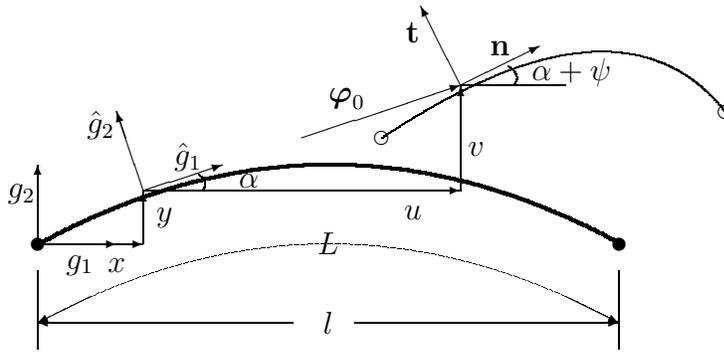


FIG. 1.1 – Le système des coordonnées mobile.

En considérant les grands déplacements, le vecteur position dans la configuration déformée peut être écrit comme

$$\varphi = \varphi_0 + \zeta \mathbf{t} = \begin{pmatrix} x + u \\ y + v \end{pmatrix} + \zeta \begin{pmatrix} -\sin(\alpha + \psi) \\ \cos(\alpha + \psi) \end{pmatrix} \quad (1.3)$$

où  $x$  et  $y$  sont les coordonnées de la position initiale de la poutre,  $u$  et  $v$  sont les composantes du déplacement dans le système des coordonnées globales et  $\zeta$  est la coordonnée le long de la normale de l'axe de la poutre dans la configuration déformée.

Le gradient de déformation est exprimé comme

$$\mathbf{F} = \begin{bmatrix} \frac{dx}{ds} + \frac{du}{ds} - \zeta \frac{d\psi}{ds} \cos(\alpha + \psi) & -\sin(\alpha + \psi) \\ \frac{dy}{ds} + \frac{dv}{ds} - \zeta \frac{d\psi}{ds} \sin(\alpha + \psi) & \cos(\alpha + \psi) \end{bmatrix} \quad (1.4)$$

Puis le gradient de déformation peut être décomposé en

$$\mathbf{F} = \mathbf{R}\mathbf{U}; \quad \mathbf{R} = \begin{bmatrix} \cos(\alpha + \psi) & -\sin(\alpha + \psi) \\ \sin(\alpha + \psi) & \cos(\alpha + \psi) \end{bmatrix} \quad (1.5)$$

et en utilisant la mesure de déformation  $\mathbf{H} = \mathbf{U} - \mathbf{I}$ , ou  $\mathbf{U} = \mathbf{R}^T \mathbf{F}$ , on obtient pour les composantes non-nulles

$$H_{11} = \Sigma - \zeta K; \quad H_{21} = \Gamma \quad (1.6)$$

où  $\Sigma, K, \Gamma$  sont les mesures des déformations généralisées proposées par Reissner dans la forme

$$\begin{aligned} \Sigma &= \cos(\alpha + \psi) \left( \frac{dx}{ds} + \frac{du}{ds} \right) + \sin(\alpha + \psi) \left( \frac{dy}{ds} + \frac{dv}{ds} \right) - 1 \\ \Gamma &= -\sin(\alpha + \psi) \left( \frac{dx}{ds} + \frac{du}{ds} \right) + \cos(\alpha + \psi) \left( \frac{dy}{ds} + \frac{dv}{ds} \right) \\ K &= \frac{d\psi}{ds} \end{aligned} \quad (1.7)$$

L'équation (1.7) donne en notation matricielle

$$\Sigma = \Lambda^T (\mathbf{h}(\mathbf{u}) - \mathbf{n}) = \Lambda^T \mathbf{h}(\mathbf{u}) - \mathbf{e}_1 \quad (1.8)$$

où

$$\boldsymbol{\Sigma} = \begin{pmatrix} \Sigma \\ \Gamma \\ K \end{pmatrix}, \quad \boldsymbol{\Lambda} = \begin{bmatrix} \cos(\alpha + \psi) & -\sin(\alpha + \psi) & 0 \\ \sin(\alpha + \psi) & \cos(\alpha + \psi) & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{h}(\mathbf{u}) = \begin{pmatrix} \frac{dx}{ds} + \frac{du}{ds} \\ \frac{dy}{ds} + \frac{dv}{ds} \\ \frac{d\psi}{ds} \end{pmatrix}, \quad \mathbf{n} = \boldsymbol{\Lambda} \mathbf{e}_1, \quad \mathbf{e}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Avec la loi de comportement

$$N = (EA)\Sigma, \quad V = (GA)\Gamma, \quad M = (EI)K \quad (1.9)$$

on peut écrire le vecteur de contrainte  $\mathbf{N}$  en notation matricielle comme

$$\mathbf{N} = \mathbf{C}\boldsymbol{\Sigma} = \mathbf{C}\boldsymbol{\Lambda}^T(\mathbf{h}(\mathbf{u}) - \mathbf{n}) \quad (1.10)$$

où

$$\mathbf{N}^T = (N, V, M)^T, \quad \mathbf{C} = \text{diag}(EA, GA, EI)$$

Pour définir la formulation faible des équations d'équilibre, on a encore besoin de présenter les expressions pour les déformations virtuelles, qui sont

$$\begin{aligned} \delta\boldsymbol{\Sigma} &= \delta[\boldsymbol{\Lambda}^T \mathbf{h}(\mathbf{u}) - \mathbf{e}_1] \\ &= \delta\boldsymbol{\Lambda}^T \mathbf{h}(\mathbf{u}) + \boldsymbol{\Lambda}^T \delta\mathbf{h}(\mathbf{u}) \\ &= \boldsymbol{\Lambda}^T (\mathbf{W}\mathbf{h}(\mathbf{u})\delta\psi + \mathbf{d}(\delta\mathbf{u})) \end{aligned} \quad (1.11)$$

où

$$\mathbf{W} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{d}(\delta\mathbf{u}) = \begin{pmatrix} \frac{d\delta u}{ds} \\ \frac{d\delta v}{ds} \\ \frac{d\delta\psi}{ds} \end{pmatrix}, \quad \delta\mathbf{u} = \begin{pmatrix} \delta u \\ \delta v \\ \delta\psi \end{pmatrix}$$

L'équation d'équilibre est la suivante

$$\mathbf{G}(\mathbf{u}, \delta\mathbf{u}) = \underbrace{\int_L (\delta\boldsymbol{\Sigma}^T \mathbf{N}) ds}_{\mathbf{G}_{int}} - \underbrace{\int_L \delta\mathbf{u}^T \mathbf{f}^{ext} ds}_{\mathbf{G}_{ext}} = 0 \quad (1.12)$$

où  $\mathbf{f}^{ext}$  est le vecteur des forces externes appliquées sur la construction et le terme des travaux virtuel internes est

$$\mathbf{G}_{int}(\mathbf{u}, \delta\mathbf{u}) = \int_L ((\mathbf{d}(\delta\mathbf{u}) + \mathbf{W}\mathbf{h}(\mathbf{u})\delta\psi)^T \boldsymbol{\Lambda} \mathbf{C} \boldsymbol{\Lambda}^T (\mathbf{h}(\mathbf{u}) - \mathbf{n})) ds \quad (1.13)$$

## 1.2 Approximation par les éléments finis pour la poutre de Reissner

On suppose l'élément avec deux noeuds. Les fonctions de forme linéaire et leurs dérivées sont les suivantes :

$$\begin{aligned}
N_1(\xi) &= \frac{1}{2}(1 - \xi), & N_1' &= -\frac{1}{2} \\
N_2(\xi) &= \frac{1}{2}(1 + \xi), & N_2' &= \frac{1}{2}
\end{aligned} \tag{1.14}$$

$$\begin{aligned}
N_1(\xi) &= \frac{1}{2}(1 - \xi), & N_1' &= -\frac{1}{2} \\
N_2(\xi) &= \frac{1}{2}(1 + \xi), & N_2' &= \frac{1}{2}
\end{aligned} \tag{1.15}$$

L'approximation de la forme de la déformée initiale de la poutre est

$$\begin{aligned}
x^e &= N_1(\xi)x_1 + N_2(\xi)x_2 \\
y^e &= N_1(\xi)y_1 + N_2(\xi)y_2
\end{aligned} \tag{1.16}$$

et le jacobien est

$$\frac{ds}{d\xi} = \frac{L^e}{2}, \quad L^e = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{1.17}$$

L'orientation de la section est définée par

$$\alpha^e = \arctan \frac{y_2 - y_1}{x_2 - x_1} \tag{1.18}$$

Les approximations des déplacements et des rotations sont

$$\begin{aligned}
u^e &= N_1(\xi)u_1 + N_2(\xi)u_2 \\
v^e &= N_1(\xi)v_1 + N_2(\xi)v_2 \\
\psi^e &= N_1(\xi)\psi_1 + N_2(\xi)\psi_2
\end{aligned} \tag{1.19}$$

Les approximations discrètes des déformations sont

$$\begin{aligned}
\Sigma^e &= \frac{1}{L^e} [(x_2 - x_1) + (u_2 - u_1)] \cos \left[ \alpha^e + \frac{1}{2}(\psi_1 + \psi_2) + \xi \frac{1}{2}(\psi_2 - \psi_1) \right] \\
&+ \frac{1}{L^e} [(y_2 - y_1) + (v_2 - v_1)] \sin \left[ \alpha^e + \frac{1}{2}(\psi_1 + \psi_2) + \xi \frac{1}{2}(\psi_2 - \psi_1) \right] - 1
\end{aligned} \tag{1.20}$$

$$\begin{aligned}
\Gamma^e &= -\frac{1}{L^e} [(x_2 - x_1) + (u_2 - u_1)] \sin \left[ \alpha^e + \frac{1}{2}(\psi_1 + \psi_2) + \xi \frac{1}{2}(\psi_2 - \psi_1) \right] \\
&+ \frac{1}{L^e} [(y_2 - y_1) + (v_2 - v_1)] \cos \left[ \alpha^e + \frac{1}{2}(\psi_1 + \psi_2) + \xi \frac{1}{2}(\psi_2 - \psi_1) \right]
\end{aligned} \tag{1.21}$$

$$K^e = \frac{d\psi^e}{ds} = \frac{1}{L^e}(\psi_2 - \psi_1) \tag{1.22}$$

Le cas de flexion pure (la contrainte de Kirchhoff) définie par

$$\Sigma^e(\xi) = 0, \quad \forall \xi \quad \text{et} \quad \Gamma^e(\xi) = 0, \quad \forall \xi \tag{1.23}$$

est obtenue seulement si les expressions (1.21) et (1.21) sont indépendentes de  $\xi$ . La contrainte imposée devient

$$\frac{1}{2}(\psi_2 - \psi_1) = 0 \Rightarrow K^e = \frac{1}{L^e}(\psi_2 - \psi_1) = 0 \quad (1.24)$$

qui, avec l'équation (1.22), annule la courbure. La solution le plus simple est l'utilisation d'un seul point d'intégration de Gauss, qui élimine le terme dépendant de  $\xi$  et permet d'obtenir une solution sans problème de blocage.

En supposant un seul point d'intégration de Gauss en  $\xi = 0$  et avec les définitions suivantes

$$\beta^e = \alpha^e + \frac{1}{2}(\psi_1 + \psi_2) \quad (1.25)$$

$$\Delta(\cdot) = (\cdot)_2 - (\cdot)_1 \quad (1.26)$$

on obtient les approximations des déformations simplifiées

$$\begin{aligned} \Sigma^e &= \frac{1}{L^e}(\Delta x + \Delta u) \cos \beta^e + \frac{1}{L^e}(\Delta y + \Delta v) \sin \beta^e - 1 \\ \Gamma^e &= -\frac{1}{L^e}(\Delta x + \Delta u) \sin \beta^e + \frac{1}{L^e}(\Delta y + \Delta v) \cos \beta^e \\ K^e &= \frac{\Delta \psi}{L^e} \end{aligned} \quad (1.27)$$

L'équation d'équilibre discrétisée a la forme

$$\mathbf{G}(\mathbf{u}, \delta \mathbf{u}) = (\mathbf{d}(\delta \mathbf{u}) + \mathbf{W}\mathbf{h}(\mathbf{a})\delta \psi)^T \mathbf{\Lambda} \mathbf{C} \mathbf{\Lambda}^T (\mathbf{h}(\mathbf{u}) - \mathbf{n}) - \delta \mathbf{u}^T \mathbf{f}^{ext} = 0 \quad (1.28)$$

où

$$\mathbf{d}(\delta \mathbf{u}) = \frac{1}{L^e} \begin{pmatrix} \delta u_2 - \delta u_1 \\ \delta v_2 - \delta v_1 \\ \delta \psi_2 - \delta \psi_1 \end{pmatrix}, \quad \mathbf{h}(\mathbf{u}) = \frac{1}{L^e} \begin{pmatrix} \Delta x + \Delta u \\ \Delta y + \Delta v \\ \Delta \psi \end{pmatrix}, \quad \delta \psi = \frac{1}{2}(\delta \psi_1 + \delta \psi_2),$$

$$\mathbf{\Lambda} = \begin{bmatrix} \cos \beta^e & -\sin \beta^e & 0 \\ \sin \beta^e & \cos \beta^e & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ et } \delta \mathbf{u} = \frac{1}{2} \begin{pmatrix} \delta u_1 + \delta u_2 \\ \delta v_1 + \delta v_2 \\ \delta \psi_1 + \delta \psi_2 \end{pmatrix}$$

Enfin, on trouve le système d'équations non-linéaire

$$\mathbf{f}^{int} - \mathbf{f}^{ext} = \mathbf{0} \quad (1.29)$$

si on regroupe les termes multipliés par le même déplacement virtuel. Puis on peut dire, que l'équation d'équilibre est valable si chaque terme multiplié par un déplacement virtuel est égal à zéro. Puis les termes des déplacements virtuels on peut éliminer. Les expressions discretisées de chaque composante de vecteur  $\mathbf{f}^{int}$  est présentée dans l'annexe B.1 avec la formulation discrétisée de la matrice de raideur tangente  $\mathbf{K}$  correspondante.

# Chapitre 2

## Le problème d'optimisation couplé

Le modèle de la poutre présenté nous donne une base excellente pour décrire le problème d'optimisation ou de contrôle des structures élastiques géométriquement non-linéaires. L'optimisation des propriétés géométriques ou le contrôle du chargement externe peuvent être formulés de façon équivalente, comme on va le montrer dans les chapitres suivants.

### 2.1 L'optimisation de forme

Le problème d'optimisation de forme consiste à choisir les propriétés du modèle mécanique, et à laisser libre le choix des propriétés géométrique du modèle de la poutre, par exemple l'épaisseur ou le choix de la géométrie initiale. Les paramètres du chargement sont ici considérés comme les données. D'un point de vue mathématique, l'optimisation de forme peut être formulée comme la minimisation de la fonction coût  $J(\cdot)$ , qui définit les propriétés voulues de la construction. Puis cette fonction n'est pas dépendant seulement de variables de forme  $\mathbf{d}$  (ex. l'épaisseur ou la géométrie) mais aussi de la déformation de la construction  $\mathbf{u}$ .

Avec l'approche traditionnelle, on suppose que pour chaque variation des variables de forme les équation d'équilibre définissent la déformée de la construction correspondante. La procédure d'optimisation  $\hat{J}(\cdot)$  devient

$$\hat{J}(\mathbf{d}) = \min J(\mathbf{u}(\mathbf{d}), \mathbf{d}) ; \quad \mathbf{u}(\mathbf{d}) : G(\mathbf{u}(\mathbf{d}), \mathbf{d}; \delta\mathbf{u}) = 0 \quad (2.1)$$

Les propriétés les plus importantes de cette approche sont la réduction des variables du problème d'optimisation d'une part grâce à l'élimination des paramètres de déformation comme variables du problème d'optimisation ; d'autre part, le calcul de l'évolution des variables est beaucoup plus cher, car pour chaque variation on doit calculer la déformation par une méthode itérative (ex. par l'analyse incrémentale), laquelle résout la formulation faible des équations d'équilibre, puis évaluer la fonction coût.

L'autre approche est la formulation du problème d'optimisation en utilisant la méthode des multiplicateurs de Lagrange avec

$$\max_{\forall \lambda} \min_{\forall (\mathbf{u}, \mathbf{d})} L(\mathbf{u}, \mathbf{d}; \lambda)$$

$$L(\mathbf{u}, \mathbf{d}; \boldsymbol{\lambda}) = J(\mathbf{u}, \mathbf{d}) + G(\mathbf{u}, \mathbf{d}; \boldsymbol{\lambda}) \quad (2.2)$$

Dans l'équation (2.1)  $\boldsymbol{\lambda}$  sont les multiplicateurs de Lagrange qui remplacent les déplacements et les rotations virtuels  $\delta \mathbf{u}$  dans la formulation faible d'équation d'équilibre ( voir l'équation (1.12 et 1.13) de la façon suivante

$$\mathbf{G}(\mathbf{u}, \boldsymbol{\lambda}) = \int_L ((\mathbf{d}(\boldsymbol{\lambda}) + \mathbf{W}\mathbf{h}(\mathbf{u})\lambda_\psi)^T \boldsymbol{\Lambda} \mathbf{C} \boldsymbol{\Lambda}^T (\mathbf{h}(\mathbf{u}) - \mathbf{n})) ds \quad (2.3)$$

avec  $\boldsymbol{\lambda} = (\lambda_u, \lambda_v, \lambda_\psi)^T$ .

La différence la plus importante par rapport la formulation en (2.1) concerne le fait que les paramètres de la déformée  $\mathbf{u}$  sont considérés comme les variables du problème d'optimisation indépendantes des variables de forme  $\mathbf{d}$ , et le nombre des multiplicateurs de Lagrange est largement plus élevé que le nombre des variables (soit la dimension) du problème d'optimisation. Il faut aussi remarquer que par rapport au problème (2.1), une évaluation de la fonction coût est dans ce cas beaucoup moins cher car il n'est pas nécessaire de calculer le déplacement pour chaque variation des variables de contrôle.

La condition d'optimalité de Kuhn-Tucker (ex. [9]) liée avec le problème de minimisation en (2.2) peut être écrit comme

$$\mathbf{0} = \mathbf{r}_u \cdot \delta \mathbf{u} = \frac{\partial L(\cdot)}{\partial \mathbf{u}} \cdot \delta \mathbf{u} = \frac{\partial J(\mathbf{u}, \mathbf{d})}{\partial \mathbf{u}} \cdot \delta \mathbf{u} + \boldsymbol{\lambda} \mathbf{K} \cdot \delta \mathbf{u} \quad (2.4)$$

où  $\mathbf{K}$  est la matrice de raideur tangente.

De plus, on a

$$0 = \mathbf{r}_d \cdot \delta \mathbf{d} = \frac{\partial L(\cdot)}{\partial \mathbf{d}} \cdot \delta \mathbf{d} = \frac{\partial J(\mathbf{u}, \mathbf{d})}{\partial \mathbf{d}} \cdot \delta \mathbf{d} + \boldsymbol{\lambda} \frac{\partial \mathbf{f}^{int}(\mathbf{u}, \mathbf{d})}{\partial \mathbf{d}} \cdot \delta \mathbf{d} \quad (2.5)$$

Et finalement, on a aussi

$$0 = \mathbf{r}_\lambda \cdot \delta \boldsymbol{\lambda} = \frac{\partial L(\cdot)}{\partial \boldsymbol{\lambda}} \cdot \delta \boldsymbol{\lambda} = [\mathbf{f}^{int}(\mathbf{u}, \mathbf{d}) - \mathbf{f}^{ext}] \cdot \delta \boldsymbol{\lambda} \quad (2.6)$$

Dans les équations (2.4), (2.5) et (2.6) les vecteurs  $\mathbf{r}_u, \mathbf{r}_d, \mathbf{r}_\lambda$  sont les résidus du problème d'optimisation. Puis, on peut définir la procédure de résolution de la façon suivante

$$\min_{\forall(\mathbf{u}, \mathbf{d}, \boldsymbol{\lambda})} \mathbf{r}^T \mathbf{r}; \quad \mathbf{r}(\mathbf{u}, \mathbf{d}; \boldsymbol{\lambda}) = (\mathbf{r}_u, \mathbf{r}_d, \mathbf{r}_\lambda) \quad (2.7)$$

## 2.2 Le contrôle de chargement

Le contrôle de chargement étudié ici concerne le chargement externe quasi-statique, qui est choisi pour mettre la structure directement dans un état optimal ou désirée ou pour aboutir à l'autre objectif, qui est définie par la fonction coût  $J(\cdot)$ . Les paramètres de la forme initiale de la construction, sa géométrie, sont dans ce cas considérés comme les données. Enfin, en supposant le chargement indépendant de la déformation de la construction (ex. qui ne suit pas la déformation), les variables de contrôle sont

exactement les paramètres de chargement, qui définissent le travail des forces externes comme

$$G^{ext}(\mathbf{c}; \delta \mathbf{u}) := \int_I \{\delta \mathbf{u} \cdot \mathbf{F}_0 \mathbf{c}\} ds \quad (2.8)$$

où  $\mathbf{F}_0$  est la matrice qui contient la distribution choisie du chargement externe. Les valeurs de chargement sont données par le vecteur  $\mathbf{c}$ .

Comme c'est habituellement réalisé, pour chaque évolution des variables de contrôle  $\mathbf{c}$ , on peut résoudre les équations d'équilibre et trouver la déformation correspondante. Avec cette hypothèse on peut décrire la procédure de contrôle de chargement  $\hat{J}(\cdot)$  comme

$$\hat{J}(\mathbf{c}) = \min J(\mathbf{u}(\mathbf{c}), \mathbf{c}) ; \quad \mathbf{u}(\mathbf{c}) : G(\mathbf{u}(\mathbf{c}), \mathbf{c}; \delta \mathbf{c}) = 0 \quad (2.9)$$

ce qui est une formulation bien équivalente à la formulation du problème d'optimisation (2.1).

Aussi l'introduction des multiplicateurs de Lagrange dans le problème de contrôle nous donne la formulation équivalente à celle du problème d'optimisation, ce qu'on peut voir en comparant l'équation (2.2) et la formulation suivante

$$\begin{aligned} \max_{\forall \lambda} \min_{\forall (\mathbf{u}, \mathbf{c})} L(\mathbf{u}, \mathbf{c}; \boldsymbol{\lambda}) \\ L(\mathbf{u}, \mathbf{c}; \boldsymbol{\lambda}) = J(\mathbf{u}, \mathbf{c}) + G(\mathbf{u}, \mathbf{c}; \boldsymbol{\lambda}) \end{aligned} \quad (2.10)$$

Des différences peuvent être trouvées dans la formulation des conditions d'optimalité de Kuhn-Tucker, qui sont pour le problème de contrôle :

$$0 = \mathbf{r}_\lambda \cdot \delta \boldsymbol{\lambda} = \frac{\partial L(\cdot)}{\partial \boldsymbol{\lambda}} \cdot \delta \boldsymbol{\lambda} := [\mathbf{f}^{int}(\mathbf{u}, \mathbf{c}) - \mathbf{F}_0 \mathbf{c}] \cdot \delta \boldsymbol{\lambda} \quad (2.11)$$

$$0 = \mathbf{r}_u \cdot \delta \mathbf{u} = \frac{\partial L(\cdot)}{\partial \mathbf{u}} \cdot \delta \mathbf{u} := \frac{\partial J(\mathbf{u}, \mathbf{c})}{\partial \mathbf{u}} \cdot \delta \mathbf{u} + \boldsymbol{\lambda} \mathbf{K} \cdot \delta \mathbf{u} \quad (2.12)$$

$$0 = \mathbf{r}_c \cdot \delta \mathbf{c} = \frac{\partial L(\cdot)}{\partial \mathbf{c}} \cdot \delta \mathbf{c} := \frac{\partial J(\mathbf{u}, \mathbf{c})}{\partial \mathbf{c}} \cdot \delta \mathbf{c} + \boldsymbol{\lambda} \mathbf{F}_0 \cdot \delta \mathbf{c} \quad (2.13)$$

Enfin, la procédure de résolution finale peut être écrite de la même façon que pour le problème d'optimisation comme

$$\min_{\forall (\mathbf{u}, \mathbf{c}, \boldsymbol{\lambda})} \mathbf{r}^T \mathbf{r} ; \quad \mathbf{r}(\mathbf{u}, \mathbf{c}; \boldsymbol{\lambda}) = (\mathbf{r}_u, \mathbf{r}_d, \mathbf{r}_\lambda) \quad (2.14)$$

# Chapitre 3

## La procédure de résolution - méthode basée sur l'algorithme génétique

Les algorithmes génétiques font partie des méthodes d'optimisation les plus modernes et populaires. Ils se basent sur une analogie avec les processus observés dans la nature concernant l'évolution des organismes vivants sur de longues périodes. Contrairement aux méthodes à gradients classiques, les algorithmes génétiques travaillent avec la soi-disant population, qui est un groupe de solutions admissibles. Puis, sur la population sont appliqués les opérateurs génétiques, comme le croisement, la mutation et la sélection. Les principes des algorithmes génétiques ont été proposés initialement par J. H. Holland [2]. Les algorithmes génétiques étaient appliqués avec succès sur l'ensemble des problèmes envisagés (ex. voir les livres de D. E. Goldberg [1] et de Z. Michalewicz [10]).

Les algorithmes génétiques dans leur forme originale travaillent avec une population de soi-disant chromosomes. Ce sont des chaînes binaires qui représentent la solution admissible d'une certaine façon. Pour des problèmes d'ingénieurs on travaille typiquement avec des variables réels, qui sont dans notre application précédemment présentée les valeurs du chargement ou les propriétés de la construction (ex. le module d'Young ou l'épaisseur de la poutre sont les quantités réelles). L'adaptation de l'idée des algorithmes génétiques sur ce problème était réalisée par R. Storn [12, 13], qui considère les chromosomes comme des vecteurs réels au lieu de chaînes binaires. Puis, il a développé les opérateurs capables de calculer la distance entre les chromosomes.

Dans ce stage, on a utilisé une version de ce groupe d'algorithmes désignés comme SADE<sup>1</sup> ([8]). [4] a montré que l'algorithme est utile pour résoudre des problèmes avec un nombre de variables élevé, et qu'il était construit pour explorer tout les minima possibles et trouver le minimum global, même si la fonction coût a des gradients très élevés et des valeurs de pic isolées. Une description brève est donnée dans les paragraphes suivants (voir [4] pour une description plus détaillée). On a amélioré cet algorithme par plusieurs modifications décrites dans la deuxième partie de ce chapitre et on a créé une nouvelle version que l'on appelle algorithme GRADE<sup>2</sup>.

---

<sup>1</sup>Simplified Atavistic Differential Evolution

<sup>2</sup>GRadient Atavistic Differential Evolution

### 3.1 L'algorithme SADE

Dans les méthodes d'évolution traditionnelles, le première pas est une création de la génération initiale des chromosomes, par un choix de valeurs aléatoires de toutes les variables. Puis on répète le cycle suivant

- création de la nouvelle génération des chromosomes par les opérateurs : mutation, mutation locale et croisement
- évaluation et sélection, qui réduit le nombre des chromosomes actuels au nombre initial

jusqu'à la convergence.

Dans les calculs suivants, on travaille avec la population de ' $PR \times n$ ' chromosomes, où  $n$  est le nombre total des inconnues du problème et  $PR$  le paramètre d'algorithme égal à 10. La population peut évoluer grâce aux opérations suivantes :

Mutation : on suppose que  $CH_i(t)$  est le  $i$ -ième chromosome de la génération  $t$ ,

$$CH_i(t) = (ch_{i1}(t), ch_{i2}(t), \dots, ch_{in}(t)), \quad (3.1)$$

où  $n$  est le nombre de variables de la fonction coût. Si un chromosome  $CH_i(t)$  est choisi pour une mutation, un chromosome aléatoire  $RP$  est créé dans la domaine de la fonction coût et le nouveau chromosome  $CH_k(t + 1)$  est calculé par la relation suivante

$$CH_k(t + 1) = CH_i(t) + MR(RP - CH_i(t)). \quad (3.2)$$

Le paramètre  $MR$  est la constante de l'algorithme égal à 0.5. Le nombre de chromosomes créés par l'opérateur 'mutation' est définie par la 'radioactivité', qui est un autre paramètre de l'algorithme et sa valeur constante est égale à 0.1 pour tous nos calculs.

Mutation locale : si un certain chromosome est choisi pour la mutation locale, toutes ses composantes sont changées par une valeur aléatoire choisi dans un intervalle typiquement très petit. L'objectif de cet opérateur est d'utiliser les chromosome voisins existants, et d'essayer d'améliorer la solution plus vite. Cela est utile surtout pour les fonctions optimisées, où proches de la valeur optimale, un petit changement des valeurs des variables présente un grand changement de la valeur de la fonction coût (ex. voir [3]). Le nombre des chromosomes créés par l'opérateur 'mutation locale' est définie par la 'radioactivité locale', qui est un autre paramètre de l'algorithme et sa valeur constante est égale à 0.1 pour nos calculs.

Croisement : l'objectif de cet opérateur est la création de nouveaux chromosomes pour obtenir le même nombre de nouveaux chromosomes (avec les nouveaux chromosomes créés par les opérateurs 'mutation' et 'mutation locale') que dans la génération parentale. L'opérateur crée le nouveau chromosome  $CH_i(t + 1)$  par rapport au schéma suivant : choix aléatoire des deux chromosomes de la population  $CH_q(t)$  et  $CH_r(t)$ , calcul du vecteur de leur différence, multiplication de ce vecteur par le coefficient  $CR$  et addition à un troisième chromosome aléatoirement choisi dans la population  $CH_p(t)$ ,

$$CH_i(t + 1) = CH_p(t) + CR(CH_q(t) - CH_r(t)). \quad (3.3)$$

Le schéma est présenté aussi sur la figure 3.1.

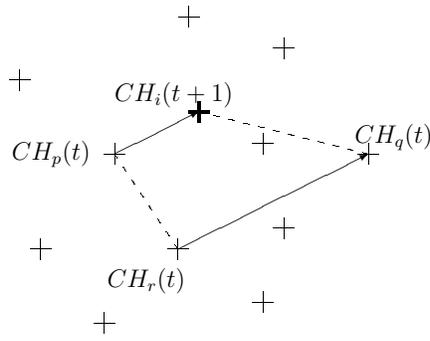


FIG. 3.1 – Le schéma de l’opérateur ‘croisement’ pour algorithme SADE

Chaque composante dépassant l’intervalle de sa valeur fixée est changée en la valeur limite correspondante. Le paramètre  $CR$  a probablement la plus grande influence sur le comportement de l’algorithme ; cela implique que si on a besoin d’accélérer la convergence, ce paramètre doit être fixé à une valeur petite (typiquement de 0.05 à 0.1) Au contraire, les valeurs plus grandes de ce paramètre peuvent augmenter le pouvoir de l’algorithme et dépasser les problèmes aux minimas locaux. Dans les calculs sa valeur est fixée à 0.3.

La sélection représente le cœur de l’algorithme génétique. L’objectif est de garantir l’amélioration progressive de toute la population par la réduction du nombre des chromosomes vivants et la conservation des meilleurs d’entre eux. On utilise la stratégie de tournoi modifié avec le schéma suivant : choix des deux chromosomes aléatoires de la population, comparaison des leurs valeurs et élimination du plus mauvais chromosome. La diversité de la population est conservée grâce à la possibilité pour les pires chromosomes de rester vivants.

## 3.2 L’algorithme GRADE

La modification de l’algorithme SADE avait deux motivations principales :

- l’augmentation de la vitesse de résolution de l’algorithme pour les fonctions lisses continues avec un seul optimum ;
- la réduction du nombre des paramètres externes de l’algorithme SADE et leur influence sur le comportement de l’algorithme, car jusqu’à maintenant on peut estimer leurs valeurs seulement par la méthode ‘essai et erreur’, ce qui présente plus ou moins de problèmes.

Pour rappel, les paramètres de l’algorithme SADE sont :  $PR$ , qui définit le nombre des chromosomes dans la population, ‘radioactivité’ et  $MR$  dans l’opérateur ‘mutation’, ‘radioactivité locale’ dans l’opérateur ‘mutation locale’ et enfin  $CR$  dans l’opérateur ‘croisement’. Puis, dans l’opérateur ‘mutation locale’ il faut définir un petit intervalle pour les valeurs qui sont utilisées pour la création du nouveau chromosome.

Pour ces raisons on a développé un nouvel algorithme GRADE, qui est plutôt une nouvelle version de l’algorithme SADE avec le même schéma. On présente ici seulement les modifications, qui sont les suivantes :

- l’élimination de l’opérateur ‘mutation locale’ ;
- le paramètre  $MR$  n’est plus constant, mais pour chaque création de nouveau

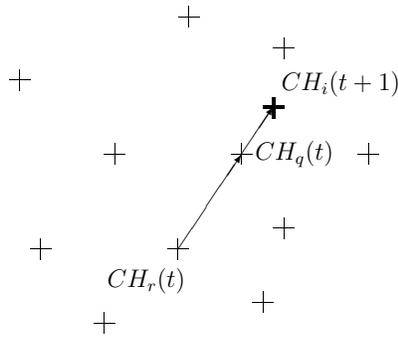


FIG. 3.2 – Le schéma de l'opérateur 'croisement' pour algorithme GRADE

chromosome par cet opérateur, sa valeur est choisie aléatoirement dans l'intervalle  $\langle 0, 1 \rangle$  ;

- le schéma de l'opérateur 'croisement' est changé dans le schéma de la façon suivante :

$$CH_i(t+1) = \max(CH_q(t); CH_r(t)) + SG.CR(CH_q(t) - CH_r(t)). \quad (3.4)$$

ce qui est présenté aussi sur la figure 3.2. Seulement deux chromosomes  $CH_q(t)$  et  $CH_r(t)$  sont aléatoirement choisis dans la population. Le vecteur de leur différence est calculé, multiplié par le paramètre  $CR$  et la direction est changé par la multiplication avec le paramètre  $SG$ , qui est égal à  $-1$  si le vecteur  $CH_r(t)$  est meilleur que  $CH_q(t)$  et égal à  $1$  dans tous les autres cas. Le vecteur obtenu est additionné au meilleur vecteur du couple  $CH_q(t)$  et  $CH_r(t)$ . Le paramètre  $CR$  n'est plus constant mais choisi dans l'intervalle  $\langle 0, CL \rangle$ , où  $CL$  est un nouveau paramètre de cet opérateur, mais son influence sur le comportement de l'algorithme par rapport celle du paramètre  $CR$  est réduite. Si ce n'est pas précisé, sa valeur est fixée à  $1$ .

Enfin, les paramètres externes de l'algorithme GRADE sont les suivants :  $PR$ , 'radioactivité' et  $CL$ . La valeur du paramètre  $PR$  est aussi fixée à  $10$  comme pour l'algorithme SADE, la valeur du paramètre 'radioactivité' est pour les calculs fixée à la valeur  $0.2$ .

# Chapitre 4

## Les exemples numériques

### 4.1 Les exemples de contrôle de chargement

L'objectif le plus fréquent pour le problème de contrôle de chargement est de trouver le chargement correspondant à une déformation désirée  $\mathbf{u}^d$ , ce qui nous définit la fonction coût  $J(\cdot)$  comme

$$J(\mathbf{u}) = \|\mathbf{u} - \mathbf{u}^d\| \quad (4.1)$$

où le vecteur  $\mathbf{u}^d$  définit les paramètres de la déformation désirée. Dans les deux chapitres prochains, on montre les résultats pour l'approche traditionnelle utilisant l'analyse incrémentale pour chaque évaluation de fonction coût (l'approche non-simultanée), puis l'autre approche basée sur la méthode des multiplicateurs de Lagrange (l'approche simultanée).

#### 4.1.1 L'approche non-simultanée

Avec l'approche traditionnelle et en utilisant l'algorithme génétique comme une méthode de maximisation, on peut décrire la procédure de résolution  $\hat{J}(\cdot)$  de la façon suivante :

$$\hat{J}(\mathbf{c}) = \max [-J(\mathbf{u}(\mathbf{c}))] \quad (4.2)$$

avec

$$-J(\mathbf{u}(\mathbf{c})) = -\|\mathbf{u}(\mathbf{c}) - \mathbf{u}^d\| ; \quad \mathbf{u}(\mathbf{c}) : G(\mathbf{u}(\mathbf{c}), \mathbf{c}; \delta \mathbf{c}) = 0 \quad (4.3)$$

où  $\mathbf{u}(\mathbf{c})$  est calculée par l'analyse incrémentale en utilisant 100 pas de pseudo-temps 't', qui résout les équations d'équilibre  $G(\cdot)$ .

Par l'approximation discrétisée basée sur la méthode des éléments finis, la fonction coût est imposée comme

$$J = \frac{1}{4} \sum_{i=0}^{n_{el}} \sum_{j=1}^2 [(u_{ij} - u_{ij}^d)^2 + (v_{ij} - v_{ij}^d)^2] * L_i \quad (4.4)$$

$n_{el}$  nombre des éléments finis

$l_i$  longueur initiale de  $i$ -ième élément

$u_{ij}, v_{ij}$  composantes de déformation calculées par l'analyse incrémentale

$u_{ij}^d, v_{ij}^d$  composantes correspondantes de la déformation désirée

On peut remarquer que dans un cas réel, on ne connaît pas toutes les composantes du vecteur de la déformation désirée et pour cette raison dans nos exemples on considère les rotations comme les composantes inconnues.

Dans la suite, on montre trois exemples du contrôle de chargement illustrant la procédure de résolution (4.2) en utilisant les algorithmes génétiques du chapitre précédent. Les paramètres des algorithmes génétiques utilisés sont marqués dans le tableau 4.1. Les bornes de la fonction maximisée  $-J(\cdot)$  pour les trois exemples sont montrées dans l'annexe C.

Algorithme SADE	Algorithme GRADE
$PR = 10$	$PR = 10$
'radioactivité' = 0.1	'radioactivité' = 0.1
$MR = 0.5$	$CL = 1.0$
'radioactivité locale' = 0.1	
$CR = 0.3$	

TAB. 4.1 – Les paramètres des algorithmes SADE et GRADE.

Le première exemple concerne une structure simple représentant la lettre 'T' avec un chargement à deux composantes. En d'autres termes c'est un problème de maximisation avec deux inconnues. Les propriétés mécaniques de la construction, sa forme initiale, sa déformation désirée et la distribution du chargement est montrée sur la figure 4.1.

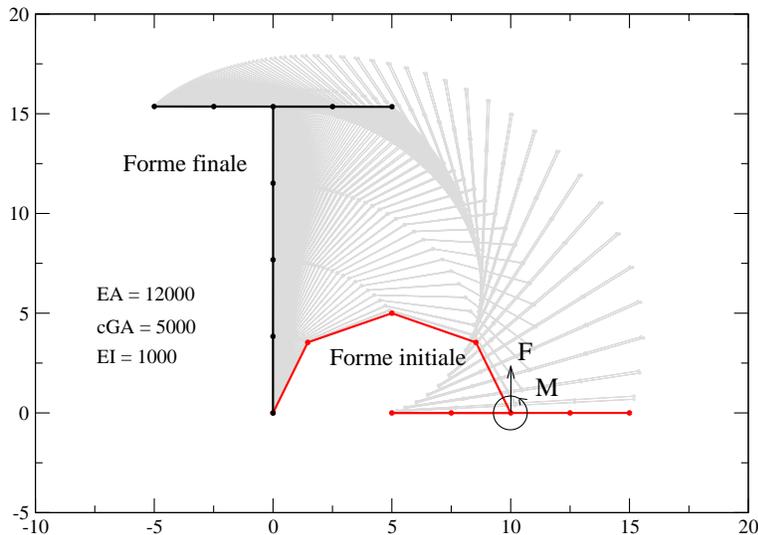


FIG. 4.1 – Lettre T - La déformation initiale et désirée de la construction avec la distribution du chargement.

Parce qu'on a choisi les exemples pour tester les procédures de résolution et montrer leur efficacité, le chargement optimal est dans ce cas connu. Pour le problème de 'lettre T', les valeurs optimales des composantes du vecteur  $\mathbf{c} = (F, M)$  et leur valeurs admissibles minimales et maximales sont données dans le tableau 4.2.

Composante	Optimale	Minimale	Maximale
$F$	40	10	60
$M$	205	175	225

TAB. 4.2 – Les valeurs du chargement de la lettre T.

Parce que l’algorithme génétique est une méthode stochastique, il est nécessaire de lancer le calcul par exemple cent fois pour minimiser l’influence de l’aléa. Comme on ne peut jamais trouver la valeur optimale avec une précision absolue, on a arrêté les calculs si l’algorithme a trouvé une valeur de la fonction de contrôle maximisée  $-J(\cdot)$  supérieure à  $-10^{-7}$ , ou en d’autres termes s’il a trouvé une valeur optimale égale à zéro avec la précision choisie égale à  $-10^{-7}$ . Avec ces conditions, on compte le nombre d’évaluations de la fonction  $-J(\cdot)$  nécessaire et l’erreur, pour laquelle étaient trouvées les valeurs du chargement. Tout ces résultats sont marqués dans les tableaux 4.3 et 4.4<sup>1</sup>.

Algorithme	Minimal	Maximal	Moyen
SADE	240	2860	648.8
GRADE	280	1180	512.4

TAB. 4.3 – Lettre T - Nombre d’évaluation de la fonction  $-J(\cdot)$  nécessaire.

Composante	Minimale	Maximale	Moyenne	Ecart type
$F$	39.912	40.084	40.002	0.0474
$M$	205.00	205.00	205.00	0.001

TAB. 4.4 – Lettre T - La statistique sur les valeurs du chargement trouvé.

Le deuxième exemple concerne une structure un peu plus compliquée représentant la lettre 'B' avec un chargement à cinq composantes. En d’autres termes, c’est un problème de maximisation avec cinq inconnues. Les propriétés mécaniques de la construction, sa forme initiale, sa déformation désirée et la distribution du chargement sont montrées sur la figure 4.2.

Pour le problème de la 'lettre B', les valeurs optimales des composantes du vecteur  $\mathbf{c} = (H, V, M_1, M_2, M_3)$  et leur valeurs admissibles minimales et maximales sont données dans le tableau 4.5.

Les calculs sont arrêtés si l’algorithme a trouvé la valeur de la fonction de contrôle maximisée  $-J(\cdot)$  supérieure à  $-10^{-6}$ . Les résultats sont donnés dans les tableaux 4.6 et 4.7 de la même façon que pour le problème de la 'lettre T'.

L’annexe D présente la convergence de l’algorithme SADE pour résoudre ce problème, avec une remarque concernant les difficultés probables de l’algorithme SADE. Comme précisé plus haut, on considère l’algorithme GRADE comme la méthode la plus efficace et dans les calculs suivants on n’a utilisé que lui.

<sup>1</sup>La statistique présentée dans ce tableau était fait par rapport aux résultats de l’algorithme GRADE. Les calculs obtenus par les deux algorithmes génétiques étaient limités par la même précision de la fonction coût  $-J(\cdot)$ , les erreurs dans les valeurs du chargement trouvé sont aussi plus ou moins équivalents.

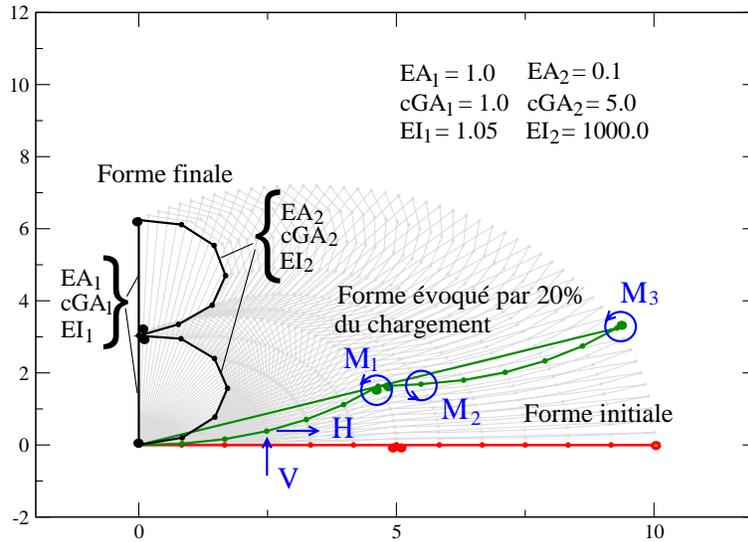


FIG. 4.2 – Lettre B - La déformation initiale et désirée de la construction avec la distribution du chargement.

Composante	Optimale	Minimale	Maximale
$H$	0.04	0.025	0.050
$V$	-0.05	-0.06	-0.035
$M_1$	0.782	0.6	0.9
$M_2$	-0.792	-0.9	-0.65
$M_3$	0.792	0.6	0.85

TAB. 4.5 – Les valeurs du chargement de la lettre B

Algorithme	Minimal	Maximal	Moyen
SADE	2600	165800	46887.5
GRADE	1900	117850	20632.0

TAB. 4.6 – Lettre B - Nombre d'évaluations de la fonction  $-J(\cdot)$  nécessaires.

Composante	Minimale	Maximale	Moyenne	Ecart type
$H$	0.039638	0.040353	0.039977	0.0002218
$V$	-0.050265	-0.49733	-0.049998	0.0001590
$M_1$	0.78178	0.78221	0.78199	0.000121
$M_2$	-0.79237	-0.79163	-0.79202	0.000214
$M_3$	0.79180	0.79224	0.79200	0.000092

TAB. 4.7 – Lettre B - La statistique sur les valeurs du chargement trouvé.

Le troisième exemple concerne une construction simple représentant la lettre 'I', le vecteur du chargement a cette fois de nouveau seulement deux composantes. Mais ces deux composantes sont choisies pour un cas de contrôle spécial du chargement, où la

formulation de la fonction coût 4.1<sup>2</sup> n'est plus suffisante pour définir un seul optimum, car on a une infinité de solutions optimales. Toutefois, on ne peut pas dire que chaque variation du chargement est une solution optimale, mais il y a une relation entre les composante du chargement qui nous définit l'espace des solutions optimales. Dans le calcul suivant on montre la performance de l'algorithme GRADE pour obtenir cette relation.

Les propriétés mécaniques de la construction, sa forme initiale, sa déformation désirée et la distribution du chargement sont montrées sur la figure 4.3.

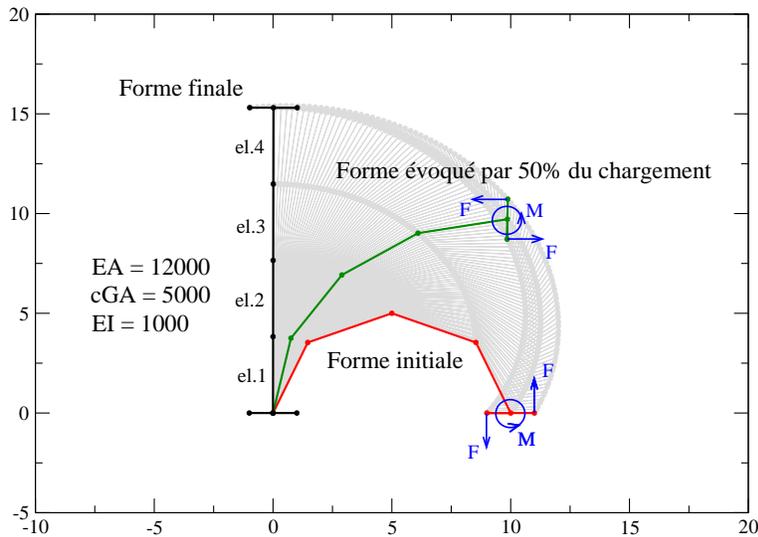


FIG. 4.3 – Lettre I - La déformation initiale et désirée de la construction avec la distribution du chargement.

Pour obtenir une infinité de solutions différentes, on a choisi le chargement comme le moment et un couple de forces suiveuses avec plus ou moins d'influence sur la déformation de la construction. Le chargement, qui suit la déformation de la construction (voir le figure 4.3), est modelisé par la formulation d'efforts externes suivante :

$$\mathbf{f}^{ext} = \lambda \mathbf{F}_0 \mathbf{c} \quad (4.5)$$

Les valeurs optimale connues des composantes du vecteur  $\mathbf{c} = (F, M)$  et leur valeurs admissibles minimales et maximales sont pour le problème de la 'lettre I' données dans le tableau 4.8.

Pour résoudre ce problème, l'algorithme GRADE est aussi lancé cent fois. Les calcul sont arrêtés si l'algorithme trouve une valeur de la fonction de contrôle maximisée  $-J(\cdot)$  supérieure à  $-10^{-7}$ . Les nombres d'évaluations minimaux, maximaux et moyens sur tous les calcul sont donnés dans le tableau 4.9.

<sup>2</sup>Dans cet exemple, on a fait une petite modification dans la formulation discrétisée de la fonction coût, où  $n_{el}$  ne signifie plus le nombre d'éléments finis total, la valeur de la fonction coût étant évaluée seulement sur les élément verticaux de la forme finale.

Composante	Optimale	Minimale	Maximale
$F$	0.0	0.0	120.0
$M$	205.4	0.0	230.0

TAB. 4.8 – Les valeurs du chargement de la lettre I

Algorithme	Minimal	Maximal	Moyen
GRADE	180	640	359.8

TAB. 4.9 – Lettre I - Nombre d'évaluation de la fonction  $-J(\cdot)$  nécessaire.

Comme chaque solution trouvée est différente, on les a distinguée dans le graphe de la figure 4.4, où on a trouvé facilement la relation entre les composantes de chargement, qui définit l'espace des solutions optimales, par une régression linéaire.

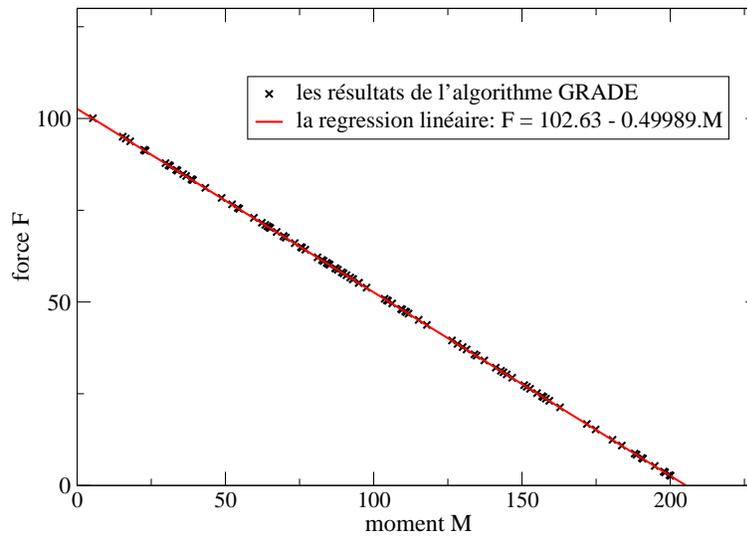


FIG. 4.4 – Cent résultats différents du problème de 'lettre I'

La relation exacte des espaces des solutions optimales est

$$F = 102.5 - 0.5M \quad (4.6)$$

Les poutres horizontales dans la déformation finale sont supposées infinitivement rigides. Ce n'est pas exactement notre cas, ce qui peut expliquer la petite différence entre la solution exacte et la solution trouvée par l'algorithme GRADE.

Dans l'exemple suivant on veut montrer comment on peut changer le problème précédent de la 'lettre I' avec une infinité de solutions dans un problème avec une seule solution. Il suffit d'augmenter la formulation de la fonction coût par un terme qui minimise l'énergie du chargement externe par minimisation des valeurs de ses composantes. La formulation de la fonction coût  $J(\cdot)$  devient

$$J(\mathbf{u}, \mathbf{c}) = \|\mathbf{u} - \mathbf{u}^d\| + \alpha\|\mathbf{c}\| \quad (4.7)$$

où  $\alpha$  est un multiplicateur constant, qui définit le poids du deuxième terme sur la valeur de la fonction coût. Pour les calculs sa valeur est fixée à  $10^{-9}$ , parcequ'on suppose que trouver la forme désirée est toujours l'objectif premier.

Avec l'approche traditionnelle on trouve la formulation de la fonction coût discrétisée comme

$$J = \frac{1}{4} \sum_{i=0}^{n_{el}} \sum_{j=1}^2 [(u_{ij} - u_{ij}^d)^2 + (v_{ij} - v_{ij}^d)^2] * L_i - \alpha \sum_{i=0}^{n_c} c_i^2 \quad (4.8)$$

où le nouveau paramètre  $n_c$  est le nombre des composantes du chargement.

Pour résoudre ce nouveau problème, l'algorithme GRADE est lancé aussi centfois, mais il est chaque fois arrêté après 20000 évaluations de la fonction coût, car dans ce cas on ne connaît pas la valeur optimale de la fonction coût. Pour chaque calcul a été trouvée la même solution suivante :

$$\begin{aligned} \mathbf{c} &= (F, M) = (68.466, 68.526) \\ -J &= -1.407863054.10^{-5} \end{aligned} \quad (4.9)$$

## 4.1.2 L'approche simultanée

L'approche simultanée est illustrée sur la construction représentant la lettre T présentée dans la chapitre précédent. La fonction coût minimisée est imposée comme

$$J(\mathbf{u}, \mathbf{c}) = \alpha_1 \|\mathbf{u} - \mathbf{u}^d\| + \alpha_2 \|\mathbf{c}\| \quad (4.10)$$

avec les paramètres  $\alpha_1 = 10^3$  et  $\alpha_2 = 10^{-4}$ .

La procédure de résolution  $\hat{J}(\cdot)$  est définie comme

$$\hat{J}(\mathbf{u}, \mathbf{c}, \boldsymbol{\lambda}) = \max_{\forall(\mathbf{u}, \mathbf{c}, \boldsymbol{\lambda})} [-\mathbf{r}^T \mathbf{r}] \quad (4.11)$$

avec  $\mathbf{r}$  définie par les équations (2.11 - 2.14). La formulation discrétisée pour la fonction  $J(\cdot)$  précédente est présentée dans l'annexe B.2.

Les inconnues du problème de contrôle sont les paramètres du chargement (2 inconnues), les composantes de la déformation (21 inconnues) et les multiplicateurs de Lagrange (21 inconnues), ce qui fait 44 inconnues au total. Les valeurs limites pour le chargement sont les mêmes que dans le tableau 4.2. Puis on calcule la déformation désirée  $\mathbf{u}_d$  par le chargement optimal. Les valeurs limites pour la déformation sont calculées comme

$$\mathbf{u}_{min} = (1 - EP)\mathbf{u}_d \quad (4.12)$$

$$\mathbf{u}_{max} = (1 + EP)\mathbf{u}_d \quad (4.13)$$

où  $EP$  est considéré comme l'erreur de prédiction de la déformation finale. Les calculs sont faits pour plusieurs valeurs différentes de ce paramètre. Les valeurs limites des multiplicateurs  $\boldsymbol{\lambda}$  sont  $\pm 0.000225$ .

Comme pour la méthode de maximisation, on a utilisé l'algorithme GRADE avec les valeurs de ses paramètres  $PR = 30$ ,  $CL = 2.0$  et 'radioactivité' = 0.2.

Dans la suite on va montrer cinq différents types de calcul pour obtenir les propriétés de l'approche simultanée. Le premier groupe de calculs montre l'influence de l'erreur de la prédiction de la déformation finale, définie par le paramètre  $EP$ . Le calcul est lancé cent fois, et arrêté après deux millions d'évaluations de la fonction coût. La statistique des résultats trouvés est donnée dans le tableau 4.10.

	$EP$	Minimal	Maximal	Moyen	Ecart type
$F$	0.05	36.982	40.648	39.761	0.4787
	0.02	39.219	40.420	39.894	0.2386
	0.005	39.536	40.327	39.986	0.1359
	0.001	39.880	40.089	39.998	0.0390
$M$	0.05	199.01	207.80	204.27	1.170
	0.02	201.86	205.82	204.54	0.595
	0.005	204.27	205.48	204.85	0.232
	0.001	204.73	205.16	204.98	0.065
$-\mathbf{r}^T \mathbf{r}$	0.05	-120.44	-3.10	-20.94	21.874
	0.02	-35.908	-1.698	-11.427	6.9532
	0.005	-16.754	-0.565	-3.699	2.3862
	0.001	-1.1357	-0.0686	-0.4233	0.24133

TAB. 4.10 – Les résultats du contrôle trouvés - calcul 1.

Dans le deuxième type de calcul, on change la condition pour arrêter l'algorithme de résolution, afin d'obtenir des résultats comparables. Le calcul est fait pour une valeur du paramètre  $EP$  égale à 0.0001, mais l'algorithme est arrêté quand il trouve la solution correspondant à la valeur de  $-\mathbf{r}^T \mathbf{r}$  supérieure à  $-0.01$ . La statistique des valeurs du chargement et le nombre d'évaluations de la fonction coût nécessaires sont présentés dans le tableau 4.11.

	Minimal	Maximal	Moyen	Ecart type
composante $F$	39.952	40.046	39.999	0.0201
composante $M$	204.92	205.09	204.99	0.0384
nombre d'évaluations de $J(\cdot)$	232320	966240	571705	————

TAB. 4.11 – Lettre T - Les résultats du contrôle trouvés - calcul 2.

Dans ce calcul on veut montrer que si on connaît la déformation finale avec une précision "suffisante", on peut simplifier le problème du contrôle par une élimination des multiplicateurs de Lagrange comme variables du problème. Cela est possible grâce à la formulation des équations (2.12), d'où on peut pour chaque variation des vecteurs  $\mathbf{u}$  et  $\mathbf{c}$  calculer les multiplicateurs de Lagrange comme

$$\boldsymbol{\lambda} = -\frac{\partial J(\mathbf{u}, \mathbf{c})}{\partial \mathbf{u}} \mathbf{K}^{-1} \quad (4.14)$$

Puis, le vecteur des résidus  $\mathbf{r}$  devient  $\mathbf{r} = (\mathbf{r}_c, \mathbf{r}_\lambda)$  et le nombre des variables diminue à 23 inconnues. Tous les autres paramètres de calcul sont gardés comme dans le calcul précédant. La statistique des valeurs du chargement et le nombre d'évaluations de la fonction coût nécessaires sont présentés dans le tableau 4.12. On peut remarquer que

	Minimal	Maximal	Moyen	Ecart type
composante $F$	39.900	40.094	39.999	0.0329
composante $M$	204.80	205.21	204.98	0.0852
nombre d'évaluations de $J(\cdot)$	162840	1165410	381570	————

TAB. 4.12 – Lettre T - Les résultats du contrôle trouvés - calcul 3.

grâce à ce changement la précision des valeurs du chargement trouvé diminue un peu, mais le nombre d'évaluations diminue beaucoup plus, ce qui monte son efficacité.

Pour le quatrième calcul, on utilise l'exemple du troisième calcul, en changeant cette fois la formulation de la fonction  $J(\cdot)$ . Comme on a effacé le deuxième terme de la fonction, on obtient la même formulation que dans (4.1), utilisée dans le chapitre précédent. Les résultats sont donnés de la même façon dans le tableau 4.13. En comparant ces résultats avec ceux du calcul 3, on peut voir que la simplification de la fonction coût précédente donne des résultats plus précis et diminue aussi le nombre d'évaluations nécessaires.

	Minimal	Maximal	Moyen	Ecart type
composante $F$	39.957	40.048	40.000	0.0199
composante $M$	204.86	205.15	205.00	0.0548
nombre d'évaluations de $J(\cdot)$	100740	298080	172176	————

TAB. 4.13 – Lettre T - Les résultats du contrôle trouvés - calcul 4.

Enfin, pour le dernier calcul on a pris l'exemple du calcul 4, mais avec une prédiction de la déformation encore plus précise (donnée par la valeur du paramètre  $EP = 0.00001$ ) et on a aussi amélioré les valeurs des paramètres de l'algorithme GRADE de façon suivant :  $PR = 20$ ,  $CL = 1.0$  et 'radioactivité' = 0.2. Dans ce cas, on peut dire, que les résultats donnés dans le tableau 4.14 sont comparable avec les résultats obtenus par l'approche non-simultanée présentée dans la chapitre 4.1.1. Il faut seulement rappeler qu'une évaluation de la fonction coût par l'approche non-simultanée comprenant cent pas de calcul par l'analyse incrémentale est à peu près aussi "chère" que cent évaluations de la fonction coût par l'approche simultanée.

	Minimal	Maximal	Moyen	Ecart type
composante $F$	39.973	40.034	40.000	0.0135
composante $M$	204.96	205.05	205.00	0.0192
nombre d'évaluations de $J(\cdot)$	14720	201480	37701	————

TAB. 4.14 – Lettre T - Les résultats du contrôle trouvés - calcul 5.

## 4.2 L'exemple d'optimisation de forme

Les problèmes d'optimisation de forme constituent un groupe de problèmes plus variés que les problèmes de contrôle. Typiquement, on connaît le chargement et on connaît souvent aussi par exemple la déformation désirée. Pour montrer le principe

de résolution d'un problème d'optimisation, on a choisi une construction simple avec une fonction coût de sens plus évident.

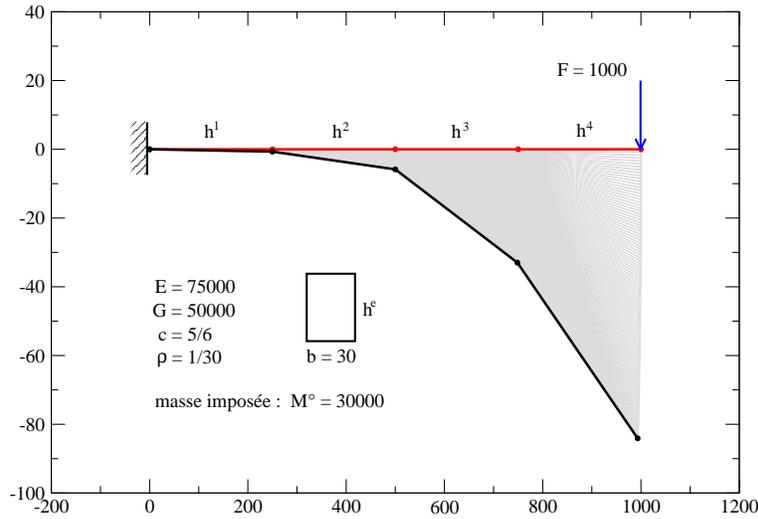


FIG. 4.5 – La déformation initiale de la construction, son chargement et propriétés du matériau.

Les variables d'optimisation  $\mathbf{d}$  choisies sont les paramètres définissant la hauteur de la poutre. L'objectif de cet exemple est la minimisation de la déformation créée par le chargement (voir la figure 4.5) et la limitation de la masse de la poutre à  $M_0 = 30000$ . La formulation utilisée pour calculer la masse de la poutre est

$$M = \int_L \rho b h ds \quad (4.15)$$

avec  $\rho = 1/30$  et  $b = 30$ .

Par l'approximation des éléments finis, on utilise quatre éléments et la hauteur  $h$  de la poutre est supposée constante sur chaque élément. Le vecteur des variables d'optimisation devient  $\mathbf{d} = \mathbf{h} = (h_1, h_2, h_3, h_4)$ , on a donc un problème d'optimisation à quatre inconnues. Dans ce cas, on ne connaît pas l'optimum du problème à l'avance. Seules les valeurs minimales et maximales admissibles pour les variables d'optimisation sont connues et données dans le tableau 4.15.

	$h_1$	$h_2$	$h_3$	$h_4$
Minimale	30	30	15	5
Maximale	60	60	35	25

TAB. 4.15 – Les valeurs limites des variables d'optimisation.

## 4.2.1 L'approche non-simultanée

Pour l'approche non-simultanée, on formule la fonction de coût  $J(\cdot)$  à maximiser comme

$$J(\mathbf{u}, \mathbf{d}) = -\alpha \|\mathbf{u}\| - (M - M_0)^2 \quad (4.16)$$

où  $\alpha$  est une constante ayant pour valeur  $\alpha = 1$  avec la formulation discrétisée suivante

$$J = -\alpha \sum_{i=0}^{n_{el}} \sum_{j=0}^6 u_{ij}^2 L_i - \left( \rho b \sum_{i=0}^{n_{el}} (h_i L_i) - M_0 \right)^2 \quad (4.17)$$

avec le vecteur de la déformation sur un élément  $\mathbf{u}_i = (u_{i1}, u_{i2}, u_{i3}, u_{i4}, u_{i5}, u_{i6})$ , et où  $n_{el}$  signifie le nombre d'éléments.

En utilisant l'algorithme GRADE (avec les paramètres  $PR = 10$ ,  $CL = 1.0$  et 'radioactivité' = 0.2) on a calculé pour chaque variation de hauteurs la déformation correspondante par l'analyse incrémentale avec cent pas du pseudo-temps 't'. La valeur optimale de la fonction  $J(\cdot)$  est

$$J_{opt}(\cdot) = -627646$$

Puis on a lancé le calcul avec l'algorithme GRADE cent fois et chaque fois le calcul est arrêté si l'algorithme GRADE a trouvé la valeur de la fonction  $J(\cdot)$  optimale avec une précision égale à 1 (ou si la valeur de  $J(\cdot)$  a dépassé la valeur  $-627647$ ). La statistique sur les valeurs des hauteurs trouvées dans les calculs est présentée dans le tableau 4.16 et la statistique sur les nombres d'évaluations de la fonction coût  $J(\cdot)$  est donnée dans le tableau 4.17.

$h_e$	Minimal	Maximal	Moyen	Ecart type
$h_1$	43.772	43.807	43.790	0.0094
$h_2$	35.914	35.949	35.932	0.0088
$h_3$	26.313	26.346	26.328	0.0082
$h_4$	14.184	14.210	14.197	0.0064

TAB. 4.16 – Les hauteurs trouvées par l'approche non-simultanée.

Algorithme	Minimal	Maximal	Moyen
GRADE	1440	9960	3497

TAB. 4.17 – Nombre d'évaluations de la fonction  $-J(\cdot)$  nécessaires par l'approche non-simultanée.

## 4.2.2 L'approche simultanée

Pour l'approche simultanée on a définie la fonction coût  $J(\cdot)$  seulement pour minimiser la déformation comme

$$J(\mathbf{u}) = -\alpha_1 \|\mathbf{u}\| \quad (4.18)$$

avec le paramètre  $\alpha_1 = 10^{-3}$ .

Les multiplicateurs de Lagrange ne sont pas considérés comme les variables du problème à optimiser, et pour chaque variation de la déformation et des hauteurs on les calcule avec la relation (2.4) comme

$$\boldsymbol{\lambda} = -\frac{\partial J(\mathbf{u}, \mathbf{d})}{\partial \mathbf{u}} \mathbf{K}^{-1} \quad (4.19)$$

La limitation de la masse par  $M_0$  est contenue dans le terme  $r_{lim}$  qui est définie comme

$$r_{lim} = \alpha_2 (M - M_0) \quad (4.20)$$

et qui est ajoutée dans le vecteur  $\mathbf{r}$  comme sa dernière composante. La valeur du paramètre  $\alpha_2$  est  $\alpha_2 = 1$ .

La procédure de solution  $\hat{J}(\cdot)$  est définie comme

$$\hat{J}(\mathbf{u}, \mathbf{d}) = \underset{\forall(\mathbf{u}, \mathbf{c})}{max} [-\mathbf{r}^T \mathbf{r}] ; \quad \mathbf{r} = (\mathbf{r}_d, \mathbf{r}_\lambda, r_{lim}) \quad (4.21)$$

où  $\mathbf{r}_d, \mathbf{r}_\lambda$  sont définis dans les équations (2.5) et (2.6) et dans la forme discrétisée le vecteur  $\mathbf{r}$  est présenté dans la chapitre B.3.

Enfin, les variables de la procédure de résolution sont les paramètres de la déformation (15 inconnues) et les valeurs de la hauteur discrétisée (4 inconnues), ce qui fait 19 inconnues au total.

Les valeurs limites pour les hauteurs était les même que dans le tableau 4.15. Puis, pour obtenir les valeurs limites de la déformation, on a fait d'abord le calcul suivant. Avec les hauteur trouvés par l'approche non-simultanée

$$\mathbf{h}^{init} = (43.79, 35.93, 26.32, 14.20); \quad M^{init} = 30060 \quad (4.22)$$

on a calculé la déformation correspondante  $\mathbf{u}^{init}$  pour le chargement imposé (voir le figure 4.5). Les valeurs limites  $\mathbf{u}_{min}$  et  $\mathbf{u}_{max}$  pour la déformation sont définies comme

$$\mathbf{u}_{min} = (1 - EP) \mathbf{u}^{init} \quad (4.23)$$

$$\mathbf{u}_{max} = (1 + EP) \mathbf{u}^{init} \quad (4.24)$$

où la valeur du paramètre  $EP$  est égal à 0.0001.

Comme pour la méthode de maximisation on a utilisé l'algorithme GRADE avec les valeurs de ses paramètres  $PR = 20$ ,  $CL = 2.0$  et 'radioactivité' = 0.1. Le calcul est lancé cent fois et arrêté chaque fois que l'algorithme GRADE trouve la valeur de  $-\mathbf{r}^T \mathbf{r}$  supérieure à  $-10000.0$ . La statistique sur les valeurs des hauteur trouvées dans les calcul sont présentée dans le tableau 4.18 et la statistique sur le nombre d'évaluations de la fonction coût  $J(\cdot)$  est donnée dans le tableau 4.19.

Dans ce problème d'optimisation, il est plus difficile de comparer les résultats par l'approche simultanée et non-simultanée car avec la formulation différente du problème à maximiser sont changés les poids entre la fonction de coût  $J(\cdot)$  et la limitation de la masse, ce qui fait converger l'algorithme dans les solutions différentes. Quand même on a essayé de formuler les problèmes pour les faire plus ou moins comparables et on peut dire que l'approche simultanée semble plus efficace que celle non-simultanée.

$h_e$	Minimal	Maximal	Moyen	Ecart type
$h_1$	43.782	43.794	43.789	0.0026
$h_2$	35.925	35.935	35.930	0.0021
$h_3$	26.315	26.324	26.319	0.0019
$h_4$	14.197	14.202	14.200	0.0010

TAB. 4.18 – Les hauteurs trouvées par l’approche simultanée.

Algorithme	Minimal	Maximal	Moyen
GRADE	111340	968240	313006

TAB. 4.19 – Nombre d’évaluations de la fonction coût  $-J(\cdot)$  nécessaire par l’approche simultanée.

# Conclusions

Dans ce stage on a d'abord utilisée l'approche traditionnelle pour résoudre les problèmes de contrôle de chargement, et on a formulé ce problème pour utiliser les méthodes modernes basées sur les algorithmes génétiques. Une analyse incrémentale est utilisée pour résoudre le problème de mécanique non-linéaire<sup>3</sup>. Pour les calculs on a utilisé d'abord l'algorithme SADE présenté dans [4] puis on a développé une version nouvelle de cet algorithme, appelé algorithme GRADE utilisant un opérateur basé sur la méthode de gradient simplifiée. Dans la chapitre 4.1.1 on a présenté la comparaison de ces deux algorithmes sur les deux exemples du problème de contrôle de chargement avec deux et cinq inconnues. Pour les deux exemples on a montré que l'algorithme GRADE est la version la plus efficace des deux. Plusieurs remarques concernant les difficultés rencontrées pour les algorithmes utilisés pour résoudre les problèmes de contrôle (et probablement d'autres problèmes) sont données dans l'annexe D. Sur le troisième exemple, le problème de contrôle, on a montré la capacité de l'algorithme GRADE à résoudre le problème ayant une infinité de solutions optimales.

Dans la deuxième partie du stage on a résolu la formulation du problème de contrôle de chargement en utilisant des multiplicateurs de Lagrange, ce qui nous donne la possibilité de résoudre le problème de contrôle et le problème de mécanique non-linéaire simultanément. Dans l'exemple présenté dans la chapitre 4.1.2 on a montré que par cette approche l'algorithme GRADE est capable de résoudre le problème, mais il n'est pas très efficace. Toutefois, on a montré que son efficacité augmente vite avec la diminution de l'espace des solutions admissibles, notamment pour les composantes de déplacement. En d'autres termes, si on prédit la déformation finale avec une précision suffisante, l'approche simultanée devient plus efficace que l'approche non-simultanée comme cela est présenté dans le tableau 4.20 qui compare les résultats obtenus pour les deux approches sur l'exemple du problème de la 'lettre T'. De la même façon on compare la précision des composantes du chargement  $F$  et  $M$  trouvées, caractérisées par l'écart type de leur valeur obtenu pour cent calculs effectués, puis on compare le nombre d'évaluations nécessaire. (Pour cela, on multiplie le nombre d'évaluations par l'approche non-simultanée par 100, car il faut tenir compte que cette évaluation est effectuée pour les cent pas de l'analyse incrémentale.)

Dans la troisième et dernière partie du stage on a formulé le problème d'optimisation de forme par les deux approches susmentionnées. Avec l'expérience des calculs précédents, on a choisi un exemple simple pour montrer le principe de résolution du problème d'optimisation de forme. Pour l'exemple choisi on peut aussi constater que

---

<sup>3</sup>Tout les calculs présentés dans ce stage étaient fait avec un logiciel développé pendant le stage en utilisant le langage de programmation C/C++.

	l'approche non-simultanée	l'approche simultanée
ecart type pour F	0.0474	0.0135
ecart type pour M	0.001	0.019
nombre d'évaluations	$512.4 \times 100 = 52140$	37701

TAB. 4.20 – Le comparaison des résultats obtenus par l'approche simultanée et non-simultanée pour le problème de contrôle sur l'exemple du problème de la 'lettre T'.

l'approche simultanée était plus efficace que celle non-simultanée. Le comparaison des résultat, comme dans le cas du contrôle, est présentée dans le tableau 4.21.

	l'approche non-simultanée	l'approche simultanée
ecart type pour $h_1$	0.0094	0.0026
ecart type pour $h_2$	0.0088	0.0021
ecart type pour $h_3$	0.0082	0.0019
ecart type pour $h_4$	0.0064	0.0010
nombre d'évaluations	$3497 \times 100 = 349700$	313006

TAB. 4.21 – Le comparaison des résultats obtenus par l'approche simultanée et non-simultanée pour le problème d'optimisation de forme.

Par rapport au problème de contrôle, on ne connaît pas la déformation finale en début de calcul (car la déformation désirée n'est pas définie) et pour prédire les valeurs admissibles limites pour les composantes de déformation, on a besoin d'utiliser un résultat approché donné par l'autre méthode de résolution du problème d'optimisation. Il faut préciser que dans le cas présenté, grâce à l'utilisation des déplacements optimaux obtenus par l'approche non-simultanée, on a la possibilité de limiter l'espace admissible pour la déformation proche de la solution optimale.

En conclusion, on a présenté deux approches différentes capable de résoudre le problème de contrôle ou d'optimisation et on a montré leurs avantages et leurs inconvénients. Le but est d'essayer de trouver la méthode la plus efficace. L'enjeu pour la suite est de trouver la méthode de résolution d'optimisation ou de contrôle qui nous donne une solution du problème approché par des calculs pas 'chers'. Cette méthode pourrait être combinée avec succes avec l'approche simultanée en utilisant l'algorithme génétique pour trouver la solution exacte. Concrètement on peut essayer d'optimiser la surface de réponse, ce qui est une méthode avec les mêmes propriétés que précédemment et un petit exemple d'utilisation cette méthode est présenté dans l'annexe E.

# Bibliographie

- [1] D.E. Goldberg. *Genetic algorithms in search, optimization and machine learning*. Addison-Wesley, 1989.
- [2] J. H. Holland. *Adaptation in natural and artificial systems*. University of Michigan, Ann Arbor, MI, Internal report, 1975.
- [3] O. Hrstka and A. Kučerová. Search for optimization method on multidimensional real domains. *CTU Report*, 4 :87–104, 2000.
- [4] O. Hrstka and A. Kučerová. Improvements of the different types of binary and real coded genetic algorithms preventing the premature convergence. *Accepted to Advances in Engineering Software*, 2003.
- [5] O. Hrstka, A. Kučerová, M. Lepš, and J. Zeman. A competitive comparison of different types of evolutionary algorithms. In B.V.H. Topping and B. Kumar, editors, *The Sixth International Conference on the Applications of Artificial Intelligence to Civil and Structural Engineering*, pages 87–88. Civil-Comp Press, 2001.
- [6] A. Ibrahimbegović and F. Frey. Finite element analysis of linear and non-linear planar deformations of elastic initially curved beams. *Int. J. Numer. Methods Eng.*, 36 :3239–3258, 1993.
- [7] M. Kleiber, H. Antunez, T.D. Hein, and P. Kowalczyk. Parameter sensitivity in nonlinear mechanics ; theory and finite element computations. 1997.
- [8] A. Kučerová and O. Hrstka. Homepage of SADE.  
[http ://klobouk.fsv.cvut.cz/~ondra/sade/sade.html](http://klobouk.fsv.cvut.cz/~ondra/sade/sade.html).
- [9] D.G. Luenberger. Linear and nonlinear programming. *Addison-Wesley Publ.*, 1984.
- [10] Z. Michalewicz. *Genetic Algorithms+Data Structures=Evolution Programs*. Springer-Verlag, 1992.
- [11] B. Rousselet. A finite strain rod model and its design sensitivity. *Mech. Struct. Mach.*, 20 :415–432, 1992.
- [12] R. Storn. Homepage of Differential Evolution.  
[http ://www.icsi.berkeley.edu/~storn/code.html](http://www.icsi.berkeley.edu/~storn/code.html).
- [13] R. Storn. On the usage of differential evolution for function optimization. In *Biennial Conference of the North American Fuzzy Information Processing Society*, pages 519–523, 1996.
- [14] G. Strang. Introduction to applied mathematics. *Wellesley-Cambridge Press*, 1986.

- [15] D.A. Tortorelli and P. Michaleris. Design sensitivity analysis : overview and review. *Inverse Prob. Engng.*, 1 :71–105, 1994.

# Annexe A

## La procédure de solution pour le problème mécanique non-linéaire

En général, dans un problème de mécanique on suppose comme données les propriétés mécaniques de la construction (qui sont représentées par la matrice de raideur) et l'effort appliqué à la construction. Puis, on utilise une méthode numérique pour calculer les composantes du déplacement, qui sont inconnues. Le problème à résoudre est le système des équations d'équilibre, qui sont dans le cas précédent non-linéaires par rapport au déplacement inconnu :

$$\mathbf{f}^{int}(\mathbf{u}) = \mathbf{f}^{ext} \quad (\text{A.1})$$

Une méthode typique qui est utilisée pour résoudre le système des équations d'équilibre est l'analyse incrémentale. Dans cette analyse on introduit le paramètre de pseudo-temps 't', ou le paramètre de chargement

$$\mathbf{f}^{int}(\mathbf{u}(t)) = \mathbf{f}^{ext}(t); \quad t \in [0, T] \quad (\text{A.2})$$

et on choisit les 'pas de temps'

$$[0, T] = \bigcup_{n=1}^{n_{inc}} [t_n, t_{n+1}] \quad (\text{A.3})$$

avec  $\mathbf{f}^{ext}(T)$  comme la charge totale à imposer.

Le chargement proportionnel

$$\mathbf{f}^{ext}(t) = \mathbf{f}_0^{ext} g(t); \quad g(T) = \frac{\|\mathbf{f}^{ext}(T)\|}{\|\mathbf{f}_0^{ext}\|} \quad (\text{A.4})$$

où  $\mathbf{f}_0^{ext}$  est le vecteur fixé et  $g(t)$  est une fonction positive, croissante, e.g.,  $g(t) = t$ .

L'incrément de la charge extérieure est défini comme

$$\Delta \mathbf{f}_{n+1}^{ext} = \mathbf{f}_{n+1}^{ext} - \mathbf{f}_n^{ext}; \quad \mathbf{f}_{n+1}^{ext} = \mathbf{f}_0^{ext} g(t_{n+1}); \quad \mathbf{f}_n^{ext} = \mathbf{f}_0^{ext} g(t_n) \quad (\text{A.5})$$

L'incrément de déplacement

$$\Delta \mathbf{u}_{n+1} = \mathbf{u}_{n+1} - \mathbf{u}_n; \quad \mathbf{u}_{n+1} = \mathbf{u}(t_{n+1}); \quad \mathbf{u}_n = \mathbf{u}(t_n) \quad (\text{A.6})$$

est calculé par

$$\Delta \mathbf{u}_{n+1} = K^{-1}(\mathbf{u}_n) [\mathbf{f}_n^{ext} - \mathbf{f}^{int}(\mathbf{u}_n) + \Delta \mathbf{f}_{n+1}^{ext}] \quad (\text{A.7})$$

où  $K(\mathbf{u}_n)$  est la matrice de raideur tangente calculé par la dérivation de  $f^{int}(\mathbf{u}_n)$  par rapport au déplacement  $\mathbf{u}$ .

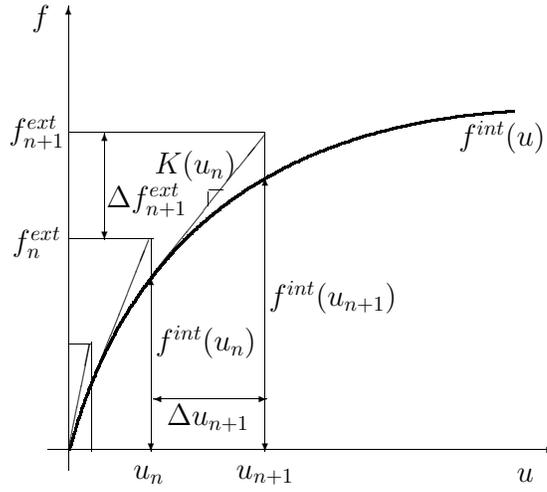


FIG. A.1 – Le schéma de l'analyse incrémentale

# Annexe B

## La formulation discrétisée du problème d'optimisation couplé

### B.1 La discrétisations du problème mécanique non-linéaire

L'écriture suivante est choisie pour faciliter la programmation (ex. en C/C++).

Pour la formulation du système des équations d'équilibre (1.29), on peut définir d'abord le vecteur  $\lambda N^e$  pour un élément comme

$$\lambda N^e = \Lambda^e C^e \Lambda^{eT} (\mathbf{h}^e(\mathbf{u}) - \mathbf{n}) = \begin{pmatrix} \Lambda N_1 \\ \Lambda N_2 \\ \Lambda N_3 \end{pmatrix} = \begin{pmatrix} \cos \beta^e \cdot EA^e \cdot \Sigma^e - \sin \beta^e \cdot GA^e \cdot \Gamma^e \\ \sin \beta^e \cdot EA^e \cdot \Sigma^e + \cos \beta^e \cdot GA^e \cdot \Gamma^e \\ EI^e \cdot K^e \end{pmatrix} \quad (\text{B.1})$$

avec les formulation discrètes de  $\beta^e$ ,  $\Sigma^e$  et  $\Gamma^e$  présentées dans les équations (1.25) et (1.27).

Puis, le vecteur des forces internes peut simplement s'écrire de la façon suivante

$$\mathbf{f}_e^{int} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \end{pmatrix} = \begin{pmatrix} -\Lambda N_1 \\ -\Lambda N_2 \\ \frac{1}{2}(\Delta x + \Delta u)(-\Lambda N_2) + \frac{1}{2}(\Delta y + \Delta v)(\Lambda N_1) - \Lambda N_3 \\ \Lambda N_1 \\ \Lambda N_2 \\ \frac{1}{2}(\Delta x + \Delta u)(-\Lambda N_2) + \frac{1}{2}(\Delta y + \Delta v)(\Lambda N_1) + \Lambda N_3 \end{pmatrix} \quad (\text{B.2})$$

La matrice de raideur tangent est définie par

$$\mathbf{K}_{[6 \times 6]}^e = \frac{\partial \mathbf{f}_{(6 \times 1)}^{int}}{\partial \mathbf{u}_{(6 \times 1)}}; \quad \mathbf{u}^e = (u_1, v_1, \psi_1, u_2, v_2, \psi_2)^T \quad (\text{B.3})$$

comme la dérivée du vecteur  $\mathbf{f}^{int}$  par rapport à chaque composante du vecteur déplacement  $\mathbf{u}$ . En utilisant plusieurs analogies dans le vecteur  $\mathbf{f}^{int}$  et dans la matrice  $\mathbf{K}$ , on

peut écrire la forme de la matrice  $\mathbf{K}^e$  comme

$$\mathbf{K}^e = \begin{bmatrix} K_{11}^e & K_{12}^e & K_{13}^e & -K_{11}^e & -K_{12}^e & K_{13}^e \\ K_{12}^e & K_{22}^e & K_{23}^e & -K_{12}^e & -K_{22}^e & K_{23}^e \\ K_{13}^e & K_{23}^e & K_{33}^e & -K_{13}^e & -K_{23}^e & K_{36}^e \\ -K_{11}^e & -K_{12}^e & -K_{13}^e & K_{11}^e & K_{12}^e & -K_{13}^e \\ -K_{12}^e & -K_{22}^e & -K_{23}^e & K_{12}^e & K_{22}^e & -K_{23}^e \\ K_{13}^e & K_{23}^e & K_{36}^e & -K_{13}^e & -K_{23}^e & K_{33}^e \end{bmatrix} \quad (\text{B.4})$$

Pour obtenir la matrice de raideur tangente, on a besoin d'exprimer les termes suivants

$$\begin{aligned} K_{11}^e &= \frac{\partial f_1}{\partial u_1} = \frac{1}{L^e} (EA^e \cdot \cos^2 \beta^e + GA^e \cdot \sin^2 \beta^e) \\ K_{12}^e &= \frac{\partial f_1}{\partial v_1} = \frac{1}{L^e} (\cos \beta^e \cdot EA^e \cdot \sin \beta^e - \sin \beta^e \cdot GA^e \cdot \cos \beta^e) \\ K_{13}^e &= \frac{\partial f_1}{\partial \psi_1} = \frac{1}{2} (EA^e - GA^e) (\sin \beta^e \Sigma^e - \cos \beta^e \Gamma^e) - \frac{1}{2} \sin \beta^e \cdot GA \\ K_{22}^e &= \frac{\partial f_2}{\partial v_1} = \frac{1}{L^e} (EA^e \cdot \sin^2 \beta^e + GA^e \cdot \cos^2 \beta^e) \\ K_{23}^e &= \frac{\partial f_2}{\partial \psi_1} = \frac{1}{2} (EA^e - GA^e) (-\cos \beta^e \Sigma^e - \sin \beta^e \Gamma^e) + \frac{1}{2} \cos \beta^e \cdot GA \\ K_{33}^e &= \frac{\partial f_3}{\partial \psi_1} = \frac{1}{4} (\Delta x + \Delta u) \left[ (EA^e - GA^e) (-\cos \beta^e \Sigma^e - \sin \beta^e \Gamma^e) + \frac{1}{2} \cos \beta^e \cdot GA \right] + \\ &\quad + \frac{1}{4} (\Delta y + \Delta v) \left[ EA^e - GA^e (-\sin \beta^e \Sigma^e + \cos \beta^e \Gamma^e) + \frac{1}{2} \sin \beta^e \cdot GA \right] + \\ &\quad + \frac{EI^e}{L^e} \\ K_{36}^e &= \frac{\partial f_3}{\partial \psi_2} = \frac{1}{4} (\Delta x + \Delta u) \left[ (EA^e - GA^e) (-\cos \beta^e \Sigma^e - \sin \beta^e \Gamma^e) + \frac{1}{2} \cos \beta^e \cdot GA \right] + \\ &\quad + \frac{1}{4} (\Delta y + \Delta v) \left[ EA^e - GA^e (-\sin \beta^e \Sigma^e + \cos \beta^e \Gamma^e) + \frac{1}{2} \sin \beta^e \cdot GA \right] - \\ &\quad - \frac{EI^e}{L^e} \end{aligned} \quad (\text{B.5})$$

avec  $L^e$  définie dans (1.18).

Le vecteur  $\mathbf{f}^{int}$  et la matrice  $\mathbf{K}$  dans la forme globale pour toute la construction sont

$$\mathbf{f}^{int}_{(n_{ddl} \times 1)} = \mathbf{A}_{e=1}^{n_{el}} \{ \mathbf{f}_e^{int} \}_{(6 \times 1)}; \quad \mathbf{K}_{(n_{ddl} \times n_{ddl})} = \mathbf{A}_{e=1}^{n_{el}} \{ \mathbf{K}^e \}_{(6 \times 6)} \quad (\text{B.6})$$

où  $n_{ddl}$  est le nombre total des degrés de liberté de toute la construction et  $n_{el}$  est le nombre d'éléments.

## B.2 La discrétisation pour le problème du contrôle de chargement

La formulation du problème de contrôle de chargement est donné pour l'exemple du chapitre 4.1.2. Avec l'approche simultanée on trouve les dérivées de la fonction coût  $J(\cdot)$  dans la forme discrétisée suivante

$$\frac{\partial J(\mathbf{u}, \mathbf{c})}{\partial \mathbf{u}} = \alpha_1(\mathbf{u} - \mathbf{u}^d); \quad \mathbf{u} = (u_1, u_2, \dots, u_{n_{ddl}})^T \quad \mathbf{u}^d = (u_1^d, u_2^d, \dots, u_{n_{ddl}}^d)^T \quad (\text{B.7})$$

$$\frac{\partial J(\mathbf{u}, \mathbf{c})}{\partial \mathbf{c}} = \alpha_2 \mathbf{c}; \quad \mathbf{c} = (F, M)^T \quad (\text{B.8})$$

et le vecteur résidu  $\mathbf{r} = (\mathbf{r}_\lambda, \mathbf{r}_u, \mathbf{r}_c)$  du problème de contrôle est écrit comme

$$\mathbf{r}^T = \left( \begin{array}{c} (\mathbf{f}^{int} - \mathbf{F}_0 \mathbf{c})^T \\ (1 \times n_{ddl}) \end{array}; \alpha_1(\mathbf{u} - \mathbf{u}^d)^T + \boldsymbol{\lambda}^T \mathbf{K}; \alpha_2 \mathbf{c}^T + \boldsymbol{\lambda}^T \mathbf{F}_0 \right) \quad (\text{B.9})$$

où  $n_{ddl}$  est le nombre des degrés de liberté de toute la construction et  $n_c$  est le nombre des composantes du chargement (dans notre cas  $n_c = 2$ ).

## B.3 La discrétisation pour le problème d'optimisation de forme

Pour le problème d'optimisation de forme on a besoin d'exprimer encore les trois termes pour calculer le vecteur des résidus  $\mathbf{r}$ . Deux termes sont les dérivés de la fonction coût  $J(\cdot)$  par rapport aux déplacements et aux variables d'optimisation et le troisième est la dérivée des équations d'équilibre par rapport aux variables d'optimisation. On va formuler ces trois expressions pour l'exemple de la fonction coût  $J(\cdot)$  présentée dans la chapitre 4.2.2.

Les variables d'optimisation choisies sont dans ce cas les valeurs discrètes qui définissent la hauteur de la poutre. Pour ça on va un peu changer la notation utilisée et le vecteur des variables d'optimisation  $\mathbf{d}$  est remplacé par le vecteur  $\mathbf{h}$

$$\mathbf{d} = \mathbf{h} = (h_1, h_2, h_3, h_4)^T \quad (\text{B.10})$$

On montre l'expression de la dérivée des équations d'équilibre pour un  $e$ -ième élément, ce qui nous fait travailler avec une seule variable  $h_e$ . Dans les équations d'équilibre c'est cette fois le vecteur  $\mathbf{f}^{int}$  qui est dépendant de l'hauteur de la poutre. La matrice de raideur du matériau est

$$\mathbf{C} = \text{diag}(EA, GA, EI) = \text{diag}(Ebh_e, Gbh_e, E \frac{1}{12}bh_e^3) \quad (\text{B.11})$$

On peut définir le vecteur  $\Lambda \mathbf{N}^{e'}$  comme

$$\Lambda \mathbf{N}' = \frac{\partial \Lambda \mathbf{N}^e}{\partial h} = \begin{pmatrix} \Lambda N'_1 \\ \Lambda N'_2 \\ \Lambda N'_2 \end{pmatrix} = \begin{pmatrix} \cos \beta \cdot Eb \cdot \Sigma - \sin \beta \cdot Gb \cdot \Gamma \\ \sin \beta \cdot Eb \cdot \Sigma + \cos \beta \cdot Gb \cdot \Gamma \\ \frac{1}{4} Ebh_e^2 \cdot K \end{pmatrix} \quad (\text{B.12})$$

et puis on obtient

$$\frac{\partial \mathbf{f}_e^{int}}{\partial \mathbf{h}} = \begin{pmatrix} -\Lambda N'_1 \\ -\Lambda N'_2 \\ \frac{1}{2}(\Delta x + \Delta u)(-\Lambda N'_2) + \frac{1}{2}(\Delta y + \Delta v)(\Lambda N'_1) - \Lambda N'_3 \\ \Lambda N'_1 \\ \Lambda N'_2 \\ \frac{1}{2}(\Delta x + \Delta u)(-\Lambda N'_2) + \frac{1}{2}(\Delta y + \Delta v)(\Lambda N'_1) + \Lambda N'_3 \end{pmatrix} \quad (\text{B.13})$$

La matrice  $\frac{\partial \mathbf{f}^{int}}{\partial \mathbf{h}}$  dans la forme globale pour toute la construction est

$$\frac{\partial \mathbf{f}^{int}}{\partial \mathbf{h}} = \underset{(n_{ddl} \times n_{el})}{\mathbf{A}} = \underset{e=1}{\overset{n_{el}}{\mathbf{A}}} \left\{ \underset{(6 \times 1)}{\frac{\partial \mathbf{f}_e^{int}}{\partial \mathbf{h}}} \right\}; \quad (\text{B.14})$$

où  $n_{ddl}$  est le nombre total des degrés de liberté de toute la construction et  $n_{el}$  est le nombre d'éléments.

Les dérivées de la fonction coût  $J(\cdot)$  définies dans (4.18) sont

$$\frac{\partial J(\mathbf{u})}{\partial \mathbf{u}} = \alpha_1 \mathbf{u}; \quad \mathbf{u} = (u_1, u_2, \dots, u_{n_{ddl}})^T \quad (\text{B.15})$$

et

$$\frac{\partial J(\mathbf{u})}{\partial \mathbf{h}} = 0 \quad (\text{B.16})$$

Enfin pour définir le vecteur des résidus du problème d'optimisation, on a besoin encore d'exprimer la formulation discrétisée pour la masse de la poutre  $M$  comme

$$M = \rho b \sum_{i=0}^4 h_i L_i \quad (\text{B.17})$$

La relation discrétisée pour calculer les multiplicateurs de Lagrange devient

$$\boldsymbol{\lambda} = -\alpha_1 \mathbf{u}^T \mathbf{K}^{-1} \quad (\text{B.18})$$

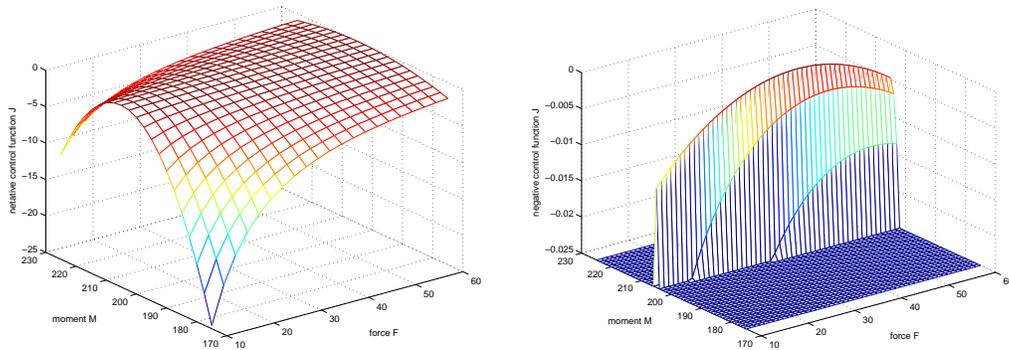
Enfin, le vecteur des résidus  $\mathbf{r} = (\mathbf{r}_d, \mathbf{r}_\lambda, r_{lim})^T$  devient

$$\mathbf{r}^T = \left( \underset{(1 \times n_{el})}{\boldsymbol{\lambda}^T \frac{\partial \mathbf{f}^{int}}{\partial \mathbf{h}}}; \underset{(1 \times n_{ddl})}{(\mathbf{f}^{int} - \mathbf{f}^{ext})^T}; \alpha_2 (M - M_0) \right) \quad (\text{B.19})$$

où  $n_{ddl}$  est le nombre de degrés de liberté et  $n_{el}$  le nombre d'éléments (dans notre cas  $n_{el} = 4$ ).

# Annexe C

## Les limites de la fonction maximisée pour les trois problèmes de contrôle



L'étendue complète des valeurs de  $-J(\cdot)$  Détaille de la valeur optimale de  $-J(\cdot)$ .

FIG. C.1 – Lettre T - Les limites de la fonction de contrôle maximisée  $-J(\cdot)$ .

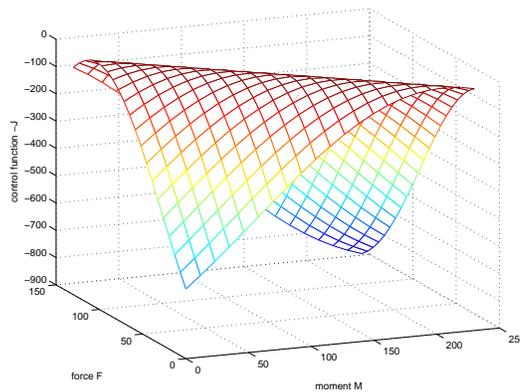
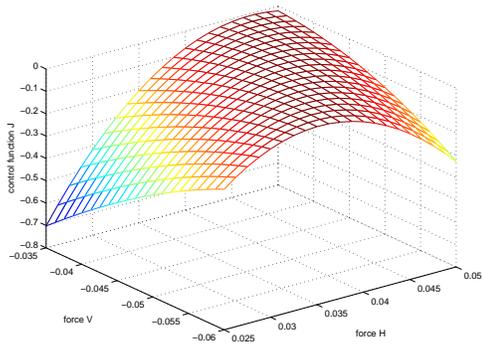
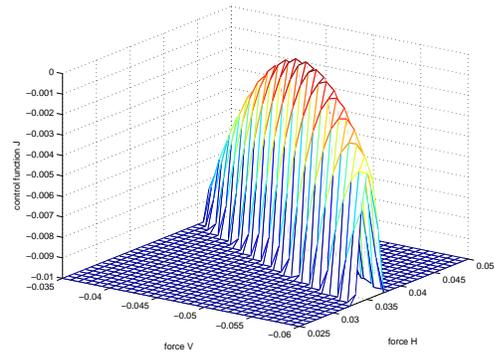


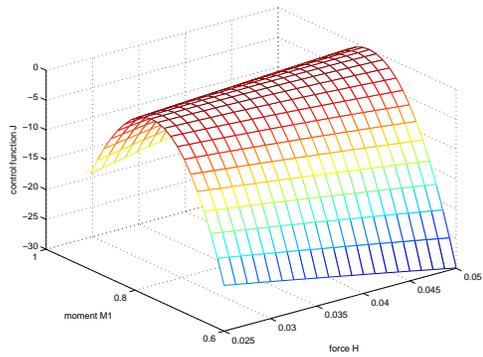
FIG. C.2 – Lettre I - Les limites de la fonction de contrôle maximisée  $-J(\cdot)$  dans la formulation non-augmentée



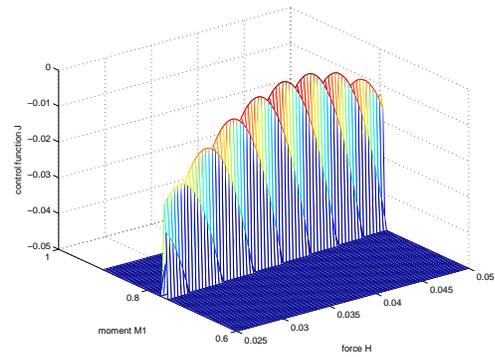
section  $H - V$



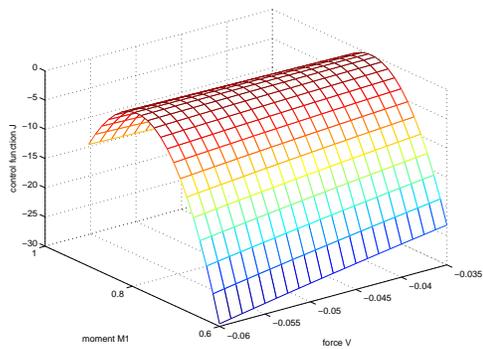
détail de la section  $H - V$



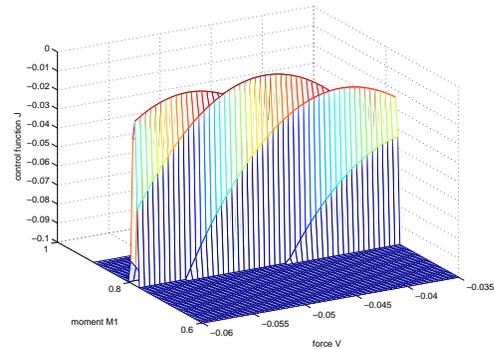
section  $H - M_1$



détail de la section  $H - M_1$

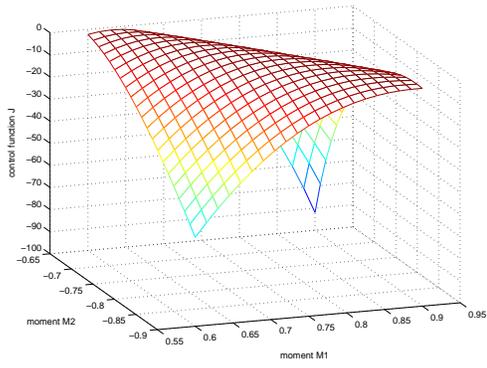


section  $V - M_1$

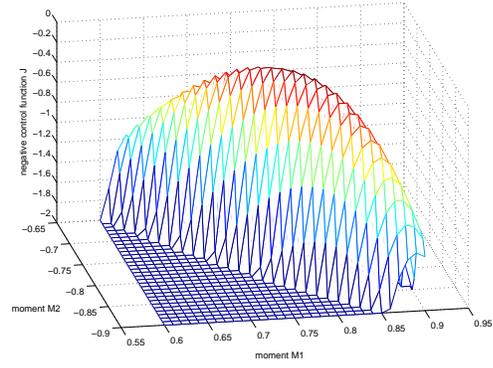


détail de la section  $V - M_1$

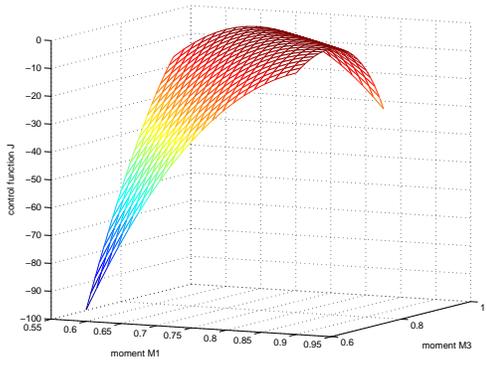
FIG. C.3 – Lettre B - Les limites de la fonction de contrôle maximisée  $-J(\cdot)$ .



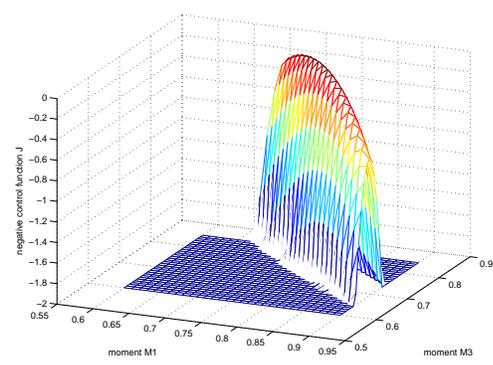
section  $M_1 - M_2$



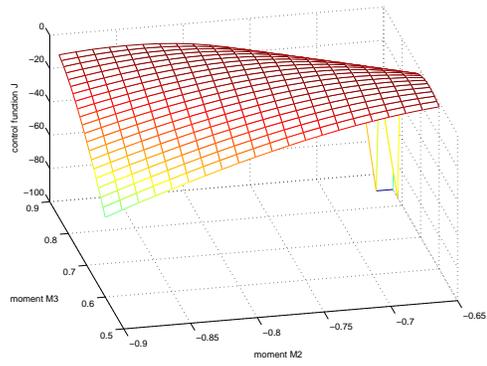
détail de la section  $M_1 - M_2$



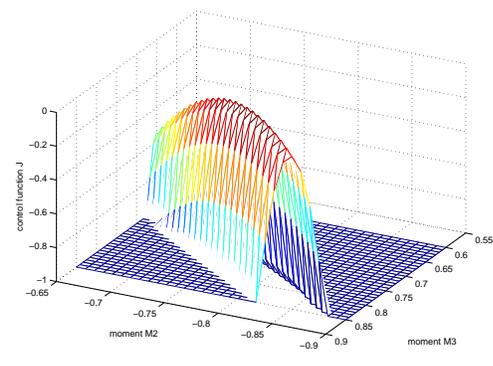
section  $M_1 - M_3$



détail de la section  $M_1 - M_3$



section  $M_2 - M_3$



détail de la section  $M_2 - M_3$

FIG. C.4 – Lettre B - Les limites de la fonction de contrôle maximisée  $-J(\cdot)$ .

# Annexe D

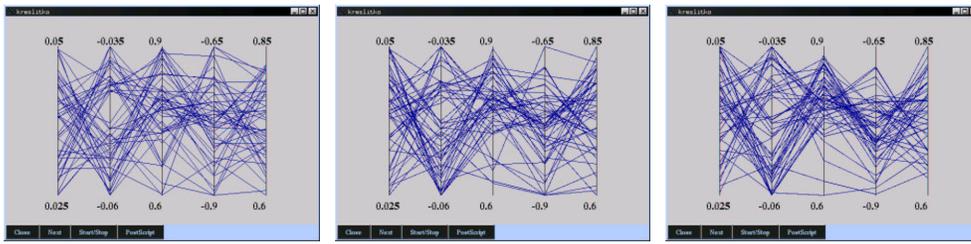
## La solution du problème de la 'lettre B' par l'algorithme SADE

La figure D.1 montre la convergence de l'algorithme SADE sur le problème de la 'lettre B'. Sur la figure, les lignes verticales représentent les échelles de chaque variable du problème optimisé. Chaque chromosome est marqué par une ligne continue, qui lie les valeurs de chaque composante sur les échelles correspondantes. Bien que on a fait cette figure pour un calcul avec une convergence rapide, on peut voir que les composante de chargement  $M_1$ ,  $M_2$  et  $M_3$  convergent plus vite que les composantes  $H$  et  $V$ . Le raison probable est que les moments  $M_1$ ,  $M_2$  et  $M_3$  ont beaucoup plus d'influence sur la valeur de la fonction coût  $J(\cdot)$  que les forces  $H$  et  $V$ . Cela est visible aussi sur les figures C.3 et C.4.

La grande différence de l'influence des variables du problème à optimiser sur la valeur de la fonction coût peut être significatif d'un problème pour l'algorithme SADE. Dès le début du calcul l'algorithme converge aux valeurs optimales des moments  $M_1$ ,  $M_2$  et  $M_3$ , mais il converge aussi (plus lentement) aux n'importe quelles valeurs des forces  $H$  et  $V$ . A la fin du calcul, les moments sont convergés aux valeurs optimales et les changements des valeurs des forces commencent à avoir plus d'influence sur la valeur de la fonction coût, mais tous les chromosomes sont déjà groupés dans un agrégat et l'algorithme est moins performant pour évoluer dans direction de l'optimum.

Ce phénomène est aussi un problème pour l'algorithme GRADE, mais grâce à l'utilisation du gradient simplifié dans l'opérateur croisement, ce n'est pas un phénomène aussi dangereux.

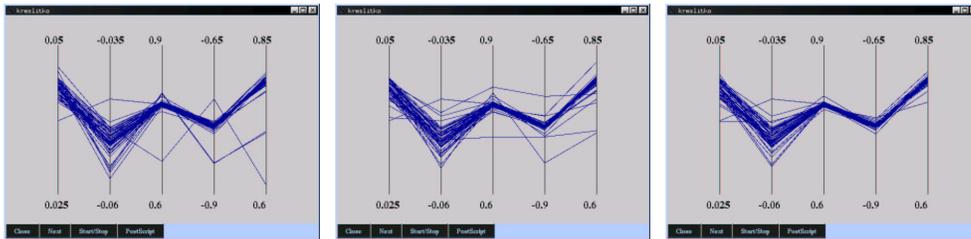
Ce phénomène est aussi rencontré dans les problèmes formulés par l'approche simultanée, et la difficulté est de bien choisir les paramètres (=les poids) qui définissent l'importance de chaque terme dans la fonction coût.



G2 :-J=-0.0071823

G3 :-J=-0.0071823

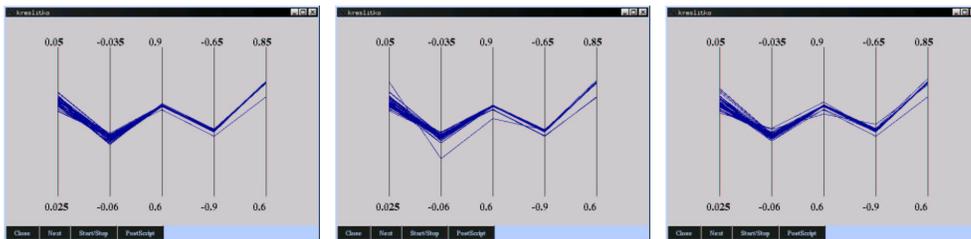
G4 :-J=-0.0071823



G26 :-J=-0.000131106

G27 :-J=-0.000131106

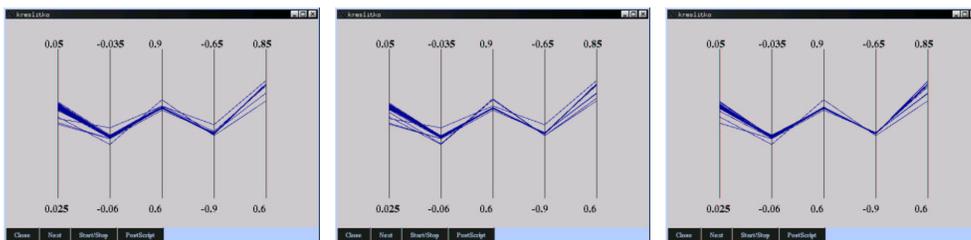
G28 :-J=-0.000131106



G41 :-J=-1.3302e-05

G42 :-J=-1.1839e-05

G43 :-J=-1.1839e-05



G67 :-J=-1.8423e-06

G68 :-J=-1.3831e-06

G69 :-J=-9.9515e-07

FIG. D.1 – La convergence de l’algorithme SADE sur le problème de la ‘lettre B’

# Annexe E

## Le contrôle de chargement pour le problème de 'lettre T' par l'approximation diffuse

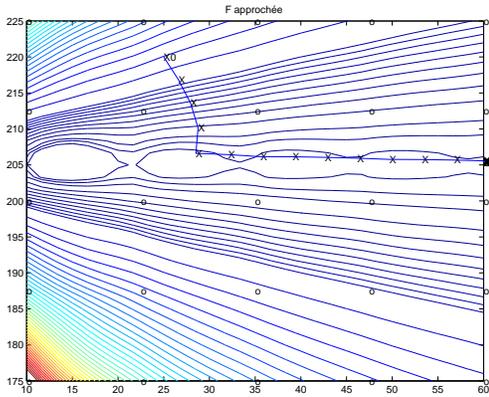
Pour comparaison avec la résolution du problème de contrôle par l'algorithme GRADE on présente encore les résultats obtenus par la méthode l'approximation diffuse. C'est une méthode de résolution, qui utilise la surface de réponse.

Pour construire ce surface on calcul d'abord les valeurs du grille de la fonction coût. Puis le surface de réponse on obtient par l'interpolation des valeurs de ce grille. Il faut marquer, que pour chaque évaluation de la fonction coût pour la construction du grille, on doit résoudre le problème mécanique non-linéaire par une méthode itérative (ex. l'analyse incrémentale). L'approximation diffuse est après utilisée pour trouve l'optimum de surface de réponse et une évaluation du coût dans ce calcul signifie évaluation du surface de réponse, ce qui n'est pas 'chère'.

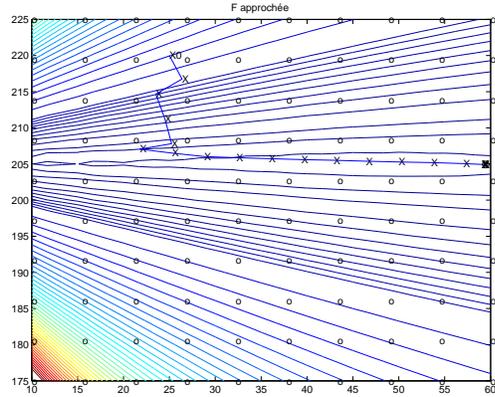
Les calculs suivants étaient fait par Catherine Knopf-Lenoir-Vayssade de l'Université de Technologie de Compiègne pour l'exemple du problème de 'lettre T' présenté dans la chapitre 4.1.1. Pour l'approximation diffuse était utilisé le point initial  $x_0 = (25, 220)$ .

En regardant les résultats présentés, on peut dire que l'approximation diffuse seul n'est pas une méthode efficace pour résoudre le problème de 'lettre T'. Dans le quatrième calcul était utilisé le grille  $(20 \times 20)$ , pour qui était nécessaire 400 évaluations de la fonction coût. Puis, la valeur trouvée de la force  $F = 47.444$  est assez différent de sa valeur optimale  $F = 40.000$ . La difficulté du problème de 'lettre T' est la grande différence entre l'influence des valeurs de force et de moment sur la fonction coût, ce qu'on peut voir sur le graphe présenté dans l'annexe C. On peut dire, que pour résoudre ce problème est l'algorithme GRADE plus efficace.

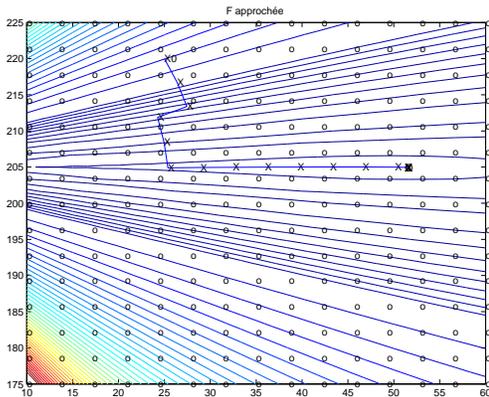
Quand même il faut aussi marquer, que déjà avec le grille  $(5 \times 5)$  était trouvée une bonne valeur du moment  $M = 205.26$ , qui est proche de la valeur optimale  $M = 205.00$ . Et dans ce cas était utilisé seulement 25 évaluations de la fonction coût. Ce fait nous donne la possibilité d'utiliser la méthode l'approximation diffuse comme une méthode, qui nous donne la solution approchée et on pourrait la combiner avec l'approche simultanée et par exemple l'algorithme GRADE pour créer une méthode de résolution plus efficace.



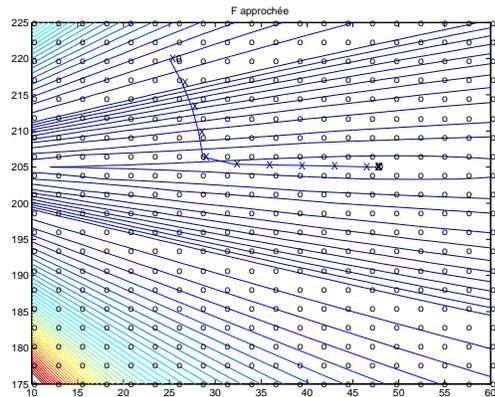
Grille ( $5 \times 5$ )  
 Solution :  $F = 60.000$   
 $M = 205.26$   
 38 évaluations du coût par AD



Grille ( $10 \times 10$ )  
 Solution :  $F = 59.073$   
 $M = 204.91$   
 38 évaluations du coût par AD



Grille ( $15 \times 15$ )  
 Solution :  $F = 51.218$   
 $M = 204.95$   
 19 évaluations du coût par AD



Grille ( $20 \times 20$ )  
 Solution :  $F = 47.444$   
 $M = 204.97$   
 15 évaluations du coût par AD

FIG. E.1 – La solution du problème de 'lettre T' par l'approximation diffuse