POLYPUBLIE

Méthode numérique d'analyse des contraintes pour les matériaux anisotropes et non linéaires

Jean-Pierre Basque

Document en libre accès dans PolyPublie
Open Access document in PolyPublie

Date	2000
Type	Mémoire ou thèse / Dissertation or Thesis Mémoire de maîtrise/Master's thesis
URL de PolyPublie PolyPublie URL	https://publications.polymtl.ca/8678/
Conditions d'utilisation Terms of use	© Tous droits réservés / All rights reserved
Référence Citation	Basque, J.-P. (2000). Méthode numérique d'analyse des contraintes pour les matériaux anisotropes et non linéaires [Mémoire de maîtrise, École Polytechnique de Montréal]. PolyPublie. https://publications.polymtl.ca/8678/

Détails de la thèse
Thesis details

Directeurs de recherche Advisors	Robert Corthésy, & Denis Gill
Programme Program	Non spécifié

Ce fichier a été téléchargé à partir de PolyPublie, le dépôt institutionnel de Polytechnique Montréal
This file has been downloaded from PolyPublie, the institutional repository of Polytechnique Montréal
https://publications.polymtl.ca

UNIVERSITÉ DE MONTRÉAL

MÉTHODE NUMÉRIQUE D'ANALYSE DES CONTRAINTES POUR LES MATÉRIAUX ANISOTROPES ET NON LINÉAIRES

JEAN-PIERRE BASQUE
DÉPARTEMENT DES GÉNIES CIVIL, GÉOLOGIQUE ET DES MINES
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLOME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE MINÉRAL)
AVRIL 2000

© Jean-Pierre Basque, 2000.

National Library of Canada
Acquisitions and Bibliographic Services
395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale du Canada
Acquisitions et services bibliographiques
395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-53552-5

Canada

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé:

MÉTHODE NUMÉRIQUE D'ANALYSE DES CONTRAINTES POUR LES MATÉRIAUX ANISOTROPES ET NON LINÉAIRES

présenté par: BASQUE Jean-Pierre

en vue de l'obtention du diplôme de: Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de:

M. GAMACHE Michel, Ph.D., Président

M. CORTHÉSY Robert, Ph.D., membre et directeur de recherche

M. GILL Denis E., Ph.D. membre et codirecteur de recherche

M. SIMON Richard, Ph.D., membre

DÉDICACE

L'auteur tient à rendre hommage à Marie-France, son épouse, pour les efforts soutenus qu'elle a déployés afin que ce mémoire ne finisse pas ses jours dans une filière avec un tas de documents oubliés.

REMERCIEMENTS

L'auteur tient à remercier le département des génies civil, géologique et des mines de l'École Polytechnique et tout particulièrement les docteurs Robert Corthésy et Denis E. Gill pour leur support et leurs conseils.

Il tient également à remercier le Conseil de recherches en sciences naturelles et en génie du Canada et Noranda, mine Brunswick, pour les subventions ayant permis de procéder aux travaux de recherches.

RÉSUMÉ

Le présent travail porte sur le développement d'une méthode d'analyse numérique pour les matériaux montrant une isotropie transversale et une non linéarité et ce dans le domaine élastique, ce qui signifie que les déformations obtenues lors de sollicitations sont non permanentes.

Une revue de l'état de la recherche a permis de constater qu'il n'existe pas de modèle permettant de traiter ce phénomène de façon analytique. Nous avons tout de même remarqué quelques lois de comportement qui permettent de traiter les matériaux montrant une isotropie transversale et une non linéarité. Le but du présent mémoire est donc le développement d'un logiciel de modélisation numérique en utilisant ces lois de comportement.

Lors du développement du modèle, certaines anomalies lui étant associées ont été notées et des correctifs y ont été apportés.

Une étude sommaire sur l'influence de l'isotropie transversale accompagnée de non linéarité dans le domaine élastique sur la distribution des contraintes autour d'une excavation circulaire a été faite.

ABSTRACT

The subject of this thesis is the development of a numeric method of stress analysis for bodies showing transverse isotropy and non-linearity in the elastic domain, meaning that the deformations obtained under a given stress state are non permanent.

A literature review showed that there are no analytical models that allow to treat this phenomenon in a proper way. However, the presence of a few constitutive laws allowing us to determine stress-strain relations for a body showing non-linearity and transverse isotropy was noticed. The goal of this research is to develop a finite element model that will incorporate this behaviour.

During the development of this model, a few anomalies linked to it were noted and corrections were brought.

Once the finite element model was validated, a brief study to evaluate the effect of non-linearity and transverse isotropy on the stress and strain distribution around a circular hole in a plate was performed.

TABLE DES MATIÈRES

DÉDICACE iv

REMERCIEMENTS v

RÉSUMÉ vi

ABSTRACT vii

TABLE DES MATIÈRES viii

LISTE DES TABLEAUX xi

LISTE DES FIGURES xii

LISTE DES ANNEXES xiv

LISTE DES SYMBOLES xv

CHAPITRE I : INTRODUCTION 1

1.1 - Généralités 1

1.2 - Objectifs de la recherche 2

CHAPITRE II : REVUE DES MODÈLES EXISTANTS 3

2.1 - Généralités 3

2.2 - Concept de variation de volume et de forme 4

2.3 - Les matériaux élastiques linéaires 10

2.3.1 - Les matériaux isotropes 12

2.3.2 - Les matériaux anisotropes avec un plan d'isotropie 12

2.3.3 - Les matériaux orthotropes 17

2.4 - Les matériaux élastiques non linéaires 18

2.5 - La méthode des éléments finis 23

CHAPITRE III : DÉVELOPPEMENT DU MODÈLE TRANSVERSALEMENT ISOTROPE ET NON LINÉAIRE 25

3.1 - Généralités 25

3.2 - Mise au point sur le modèle de base 25

3.3 - Modifications nécessaires au modèle 27

3.3 - Méthodologie de calcul 36

CHAPITRE IV : APPLICATION INFORMATIQUE 43

4.1 - Généralités 43

4.2 - Algorithme du programme d'éléments finis 43

4.3 - Intégration informatique 46

CHAPITRE V : VÉRIFICATIONS ET VALIDATION DU MODÈLE 50

5.1 - Généralités 50

5.2 - Description du modèle analysé 51

5.3 - Analyse d'un matériau élastique linéaire et isotrope 52

5.4 - Analyse d'un matériau élastique linéaire et anisotrope 55

5.5 - Analyse d'un matériau élastique non linéaire et isotrope 59

5.6 - Analyse d'un matériau élastique non linéaire et transversalement isotrope 62

CHAPITRE VI : DISCUSSIONS 67

CHAPITRE VII : CONCLUSIONS ET RECOMMANDATIONS 70

RÉFÉRENCES 74

ANNEXE 78

LISTE DES TABLEAUX

Tableau 2.1 Relations empiriques du module d'élasticité: 20

Tableau 3.1 Récapitulation des méthodes de calcul des déformations pour un matériau isotrope et non linéaire: 34

Tableau 6.1 Compilations des simulations et des résultats: 69

LISTE DES FIGURES

Figure 2.1 Effet d'une compression hydrostatique sur des matériaux isotropes et anisotropes: 8

Figure 2.2 Effet d'une sollicitation à contrainte moyenne nulle (σ_m =0) sur des matériaux isotropes et anisotropes.: 9

Figure 2.3 Constantes élastiques et coefficients de Poisson pour un matériau transversalement isotrope. (Wittke 1991): 14

Figure 2.3 Constantes élastiques et coefficients de Poisson pour un matériau orthotrope. (Wittke 1991): 15

Figure 3.1 Compression biaxiale isotrope: 27

Figure 3.2 Fonctions de variation de forme de premier ordre: 34

Figure 3.3 Déformations dues à une compression uniaxiale: 40

Figure 4.1 Algorithme des modèles d'analyse de contraintes par éléments finis : 44

Figure 4.2 Algorithme utilisé dans notre modèle d'analyse de contraintes par éléments finis : 45

Figure 4.3 Lissage de courbe pour les fonctions proposées: 48

Figure 5.1 Maillage typique utilisé pour les essais de validation: 52

Figure 5.2 Modélisation d'un matériau isotrope et linéaire: 54

Figure 5.3 Matériau transversalement isotrope et linéaire sollicité perpendiculairement au plan d'isotropie: 57

Figure 5.4 Matériau transversalement isotrope et linéaire sollicité parallèlement au plan d'isotropie: 58

Figure 5.5 Matériau transversalement isotrope et linéaire sollicité à un angle de 45^0 du plan d'isotropie: 59

Figure 5.6 Modélisation d'un matériau isotrope et non linéaire: 60

Figure 5.7 Modules d'élasticité mesurés lors d'une compression biaxiale isotrope: 63

Figure 5.8 Niveau de contraintes tangentielles pour une compression parallèle au plan d'isotropie: 64

Figure 5.9 Niveau de contraintes tangentielles pour une compression perpendiculaire au plan d'isotropie: 65

LISTE DES ANNEXES

Annexe I Listing de programme: 79

LISTE DE SYMBOLES

Notations et symboles principaux:

K:	Module d'élasticité volumique ou compressibilité (Bulk Modulus)
G:	Module de cisaillement ou de rigidité isotrope
G_{ij}:	Module de cisaillement ou rigidité anisotrope
σ :	Contrainte normale
$\sigma_{1,2,3}$:	Contraintes principales
σ_m :	Contrainte moyenne
σ_r :	Contrainte radiale
σ_θ :	Contrainte tangentielle
s_{ij} :	Déviateur de contrainte
s_{ijmax} :	Déviateur de contrainte maximal
ε :	Déformation normale
$\varepsilon_{x, y, z}$:	Déformations selon l'axe x, y, z
e_{ij} :	Déviateur de déformation
e_{ii1} :	Déviateur de déformation de premier ordre selon l'axe I
e_{ii2} :	Déviateur de déformation de second ordre selon l'axe I
ε_m :	Variation de volume
ε_{m1} :	Variation de volume de premier ordre
ε_{m2} :	Variation de volume de second ordre

δ_{ij} : Delta de Kronecker (δ_{ij} = 1 si i = j, 0 dans les autres cas)

$\sigma_{x, y, z}$: Contrainte normale selon l'axe x, y, z

σ_{ij} : Contrainte normale

E: Module de Young isotrope linéaire élastique

$E_{x, y, z}$: Module de Young dans la direction axiale x, y, z

$E_{1, 2, 3}$: Module de Young selon les axes principaux

E_{θ} : Module de Young à une angle θ

ν: Coefficient de Poisson isotrope

P : Contrainte quelconque

P_o : Contrainte hydrostatique

fn : Fonctions quelconques

f_{ij} : Fonctions quelconques

a_{ij}: Composants de la matrice de déformabilité de la loi généralisée de Hooke

[A]: Matrice de déformation de la loi généralisée de Hooke

I_1 : Premier invariant des containtes

J_2 : Deuxième invariant de déviateur des contraintes

P_a : Pression atmosphérique

U_o : Energie de déformation élastique

$I_{1\varepsilon, 2\varepsilon, 3\varepsilon}$: Invariants de déformation

τ_{ij} : Contrainte de cisaillement

γ_{ij} : Déformations de cisaillement

déf:	Déformation
μdéf :	Micro-déformation
a,b,c,m	paramètres quelconques
r	Rayon de coordonnées polaires
θ :	Angle de coordonnées polaires ou angle selon lequel des calculs sont effectués
ϕ :	Angle entre la direction de l'application d'une contrainte et l'axe *x*
u_r :	Déplacement radial
u_θ :	Déplacement tangentiel
μ_1, μ_2 :	Paramètres complexes de Lekhnitskii

CHAPITRE I

INTRODUCTION

1.1 - GÉNÉRALITÉS

La venue de l'informatique a changé énormément la façon par laquelle nous statuons sur la stabilité des différents ouvrages d'ingénierie en rocher. Auparavant, nous devions nous fier aux méthodes empiriques ou encore analytiques pour évaluer si une galerie ou un chantier sera stable ou non. Maintenant nous possédons un outil supplémentaire qui est la simulation ou modélisation numérique. Grâce à l'informatique, il arrive maintenant très fréquemment que l'on fasse appel à des logiciels permettant d'estimer les contraintes et les déplacements des massifs rocheux autour des excavations avant de procéder aux travaux afin d'optimiser la géométrie de celles-ci.

Le réalisme avec lequel de tels programmes peuvent simuler le comportement des massifs rocheux dépend en bonne partie, mais non exclusivement, des lois de comportement que l'on peut leur associer. Il est reconnu que les roches dures présentent, à divers degrés, un comportement anisotrope et non linéaire associé à la présence de micro-fissures, de diaclases et autre défauts, Ces dernières pouvant avoir une orientation préférentielle. Il serait alors intéressant de vérifier l'influence d'un tel comportement sur la distribution des

contraintes et des déformations autour des excavations. Cette vérification n'a jamais été faite jusqu'ici faute de modèles, analytiques ou numériques, permettant de reproduire simultanément un tel comportement.

1.2 - OBJECTIFS DE LA RECHERCHE

L'objectif de la présente recherche est de créer un logiciel de modélisation numérique faisant appel à la méthode des éléments finis, permettant de modéliser en deux dimensions le comportement de matériaux élastiques non linéaires et anisotropes. L'anisotropie en cause est l'isotropie transversale et la non linéarité considérée suppose la réversibilité des relations contrainte - déformation (comportement élastique).

Le présent mémoire est divisé de la façon suivante. Suite à l'introduction, le chapitre II présente une brève revue des différents modèles simples de comportement des matériaux pour s'attarder par la suite à ceux des matériaux élastiques non linéaires et transversalement isotropes. Nous continuons en décrivant les traitements numériques inhérents aux éléments finis, ainsi que les modifications permettant d'introduire les caractéristiques des matériaux. À l'étape suivante, nous présentons des résultats permettant de valider notre modèle. Enfin, nous analysons les résultats afin de tirer quelques conclusions suivies de recommandations.

CHAPITRE II

REVUE DES MODÈLES EXISTANTS

2.1 - GÉNÉRALITÉS

L'Homme a tenté depuis plusieurs siècles de définir le comportement des matériaux. La loi de comportement des matériaux la mieux connue est sans aucun doute celle de Hooke, énoncée au XVII siècle et qui dit que l'extension est proportionnelle à la force.

Cette loi est encore aujourd'hui l'un des piliers de la mécanique appliquée et elle représente généralement bien les matériaux tels l'acier ou l'aluminium, en deçà de leur limite d'écoulement plastique ou de rupture, mais elle n'est pas très adéquate pour d'autres matériaux tels les roches dures. En effet, pour ces derniers, même en demeurant dans le domaine élastique, la présence de micro-fissures induit usuellement un comportement non linéaire. De plus, suivant l'orientation aléatoire équiprobable ou préférentielle des micro-fissures, le matériau peut montrer un comportement de déformation anisotrope.

L'étude du comportement mécanique des roches a fait l'objet de plusieurs recherches particulièrement depuis les années soixante. Encore aujourd'hui, le comportement mécanique de certains types de roches semble nous échapper.

2.2 - CONCEPT DE VARIATION DE VOLUME ET DE FORME

Tout matériau soumis à des efforts externes se déforme en fonction de sa rigidité et du niveau de contraintes appliquées sur ce matériau. Afin de décrire le comportement des matériaux, il est possible d'utiliser le concept de variation de forme et de variation de volume.

Dans le domaine des mesures de contraintes, Leeman et Denkhaus (1969) font appel à deux paramètres, K et G, pour modéliser un matériau isotrope linéaire. Le premier paramètre, K, est décrit par la relation suivante

$$K = \sigma_m / (\varepsilon_x + \varepsilon_y + \varepsilon_z) \qquad (2.1)$$

qui reflète les variations de volume. Le deuxième paramètre, G, s'écrit

$$G_{ij} = s_{ij} / (2\, e_{ij}) \qquad (2.2)$$

ce qui traduit la variation de forme. Dans ces équations, ε_x, ε_y, et ε_z sont les déformations selon les axes d'un référentiel *x*, *y* et *z*. Par ailleurs, σ_m, la contrainte moyenne, s_{ij}, le déviateur de contrainte et e_{ij} le déviateur de déformation sont définis comme suit

$$\sigma_m = (\sigma_x + \sigma_y + \sigma_z)/3$$

$$s_{ij} = \sigma_{ij} - \delta_{ij}\,\sigma_m$$

$$e_{ij} = \varepsilon_{ij} - \delta_{ij}\,\varepsilon_m \qquad (2.3)$$

où

$$\varepsilon_m = (\varepsilon_x + \varepsilon_y + \varepsilon_z) / 3.$$

Dans ces équations, δ_{ij} est le delta de Kronecker ($\delta_{ij} = 1$ si $i = j$ et $\delta_{ij} = 0$ pour les autres cas) et σ_x, σ_y et σ_z sont les contraintes normales selon les axes du référentiel.

Il est possible de convertir le module de Young, E, et le coefficient de Poisson, ν, en paramètres K et G, respectivement les modules de compressibilité et de cisaillement. Les relations sont

$$K = E / [3 (1 - 2 \nu)] \qquad (2.4)$$

et

$$G = E / [2 (1 + \nu)]. \qquad (2.5)$$

Il est également possible d'obtenir E et ν en ayant K et G à l'aide de

$$E = 9KG/(3K+G) \qquad (2.6)$$

et

$$\nu = (3K-2G)/(6K+2G). \qquad (2.7)$$

Pour déterminer les déformations à partir des contraintes, nous devons nous baser sur le principe de superposition des effets de forme aux effets de volume. Ainsi, la contrainte moyenne (σ_m) permet de déterminer la variation de volume (ε_m), et les déviateurs de contrainte (s_{ij}) donne les variations de forme (e_{ij}).

Lorsqu'un matériau est anisotrope, la variation de volume n'est plus uniquement due à la contrainte moyenne mais aussi aux déviateurs de contrainte. Il en va de même pour la variation de forme.

Corthésy et al. (1993) démontrent bien les relations entre les variations de forme et de volume en fonction de la contrainte moyenne et des déviateurs de contraintes pour un matériau transversalement isotrope.

Supposons un état de contrainte quelconque. Nous pouvons calculer une contrainte moyenne σ_m et des déviateurs de contrainte s_{ij}. Il est important de noter que la somme des déviateurs est toujours nulle peu importe la sollicitation. Ceci est conséquent à la façon de les calculer.

Il est également possible d'identifier les déformations ε_x, ε_y et ε_z à partir de la variation de volume (ε_m) et les déviateurs de déformation (e_{ij}) tels que définis plus haut. Nous obtenons

$$\varepsilon_{ij} = \delta_{ij}\, \varepsilon_m + e_{ij}. \qquad (2.8)$$

Notons que ces relations sont indépendantes de la loi de comportement choisie.

La figure 2.1 illustre les déformations obtenues pour des matériaux ayant des comportements différents allant de l'isotropie à l'anisotropie en passant par l'isotropie transversale définie par les modules E_x, E_y, E_z et par un coefficient de Poisson (ν) unique pour les besoins de cette mise en situation. Ces matériaux sont soumis à une compression hydrostatique, c'est-à-dire à un niveau de contrainte égal selon les trois axes ($\sigma_x = \sigma_y = \sigma_z = Po = 10$ MPa). Les équations ayant mené à ces résultats sont présentées à la section 2.2.

Pour tous les cas illustrés, nous notons que $\sigma_m = 10$ MPa et $s_{xx} = s_{yy} = s_{zz} = 0$. Sur cette figure, nous constatons que le matériau isotrope ne montre qu'une variation de volume. Cette variation de volume est dite primaire ou de premier ordre et est représentée par ε_{m1}. Par contre, les matériaux transversalement isotropes et orthotropes ont des variations de volume et de forme qui sont associées à la contrainte moyenne. Cette variation de forme est dite secondaire ou de deuxième ordre et est identifiée par e_{ij2}.

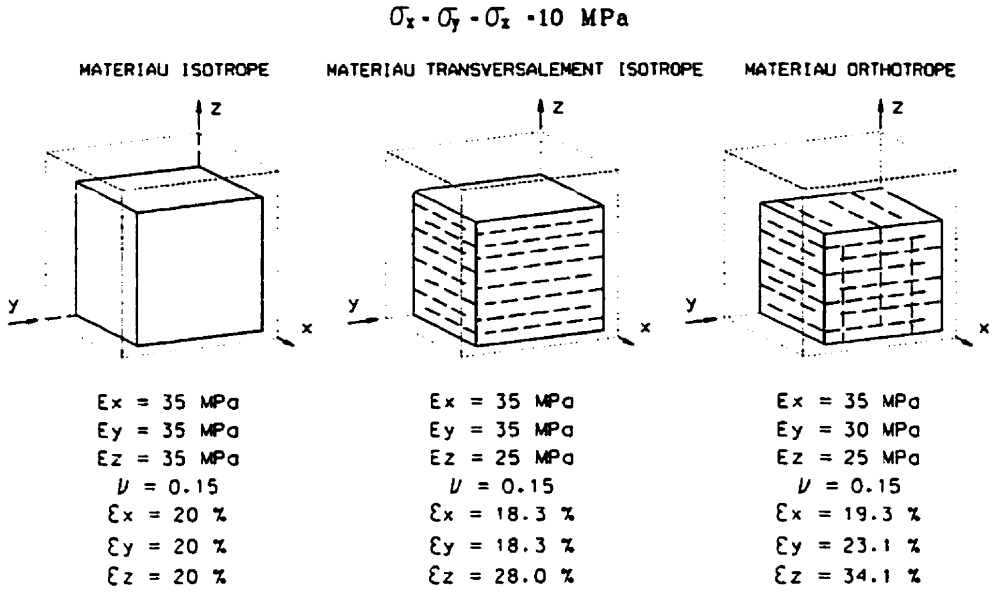


Figure 2.1: Effet d'une compression hydrostatique sur des matériaux isotropes et anisotropes

La figure 2.2, pour sa part, montre les déformations obtenues pour un corps soumis aux niveaux de contraintes suivants: σ_x = -5 MPa; σ_y = -5 MPa et σ_z = 10 MPa, de façon à ce que σ_m = 0 et que, par conséquent, $s_{xx} = \sigma_x$; $s_{yy} = \sigma_y$; et $s_{zz} = \sigma_z$. Nous pouvons voir que le matériau isotrope montre uniquement les variations de forme dites de premier ordre appelées e_{ij1}. Pour les matériaux transversalement isotropes et orthotropes, il y a des variations de forme et de volume. Nous allons donc définir la variation de volume de deuxième ordre, identifiée par ε_{m2}, comme étant associée aux déviateurs de contrainte.

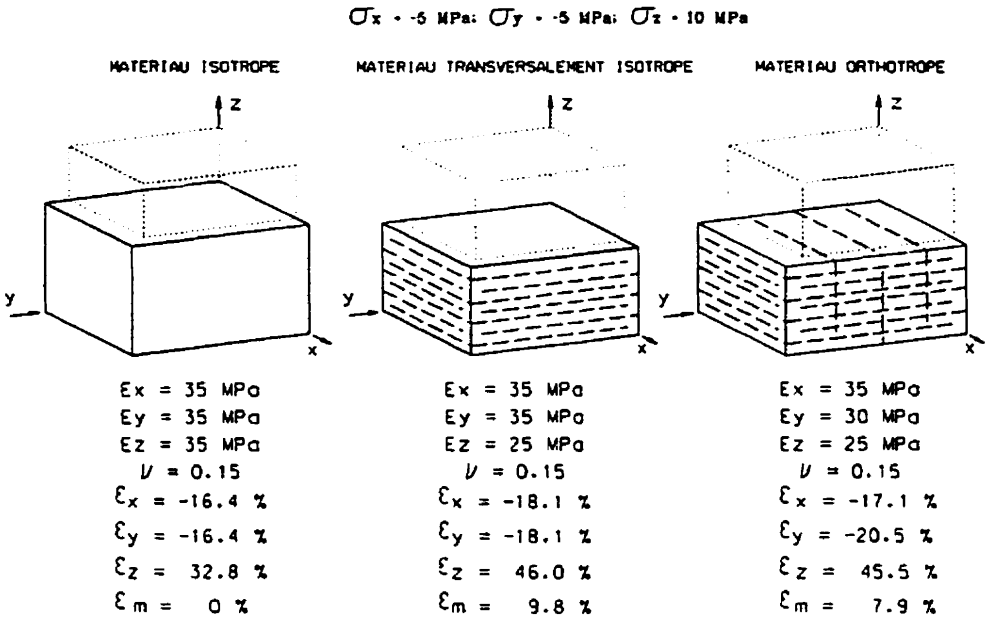


Figure 2.2: Effet d'une sollicitation à contrainte moyenne est nulle (σ_m =0) sur des matériaux isotropes et anisotropes.

Ainsi nous pouvons définir les relations suivantes,

$$\varepsilon_{m1} = fn1\ (\sigma_m)$$

$$e_{ij1} = fn2\ (s_{xx}, s_{yy}, s_{zz})$$

$$\varepsilon_{m2} = fn3\ (s_{xx}, s_{yy}, s_{zz}) \qquad (2.9)$$

et

$$e_{ij2} = fn4\ (\sigma_m).$$

Pour obtenir la variation de volume totale, nous pouvons superposer les variations de volume de premier et de deuxième ordre. Ainsi,

$$\varepsilon_m = \varepsilon_{m1} + \varepsilon_{m2}. \tag{2.10}$$

Il en est de même pour les variations de forme. Avec,

$$e_{ij} = e_{ij1} + e_{ij2}. \tag{2.11}$$

Par la suite, pour obtenir les déformations, il suffit de superposer à nouveau les variations de forme à la variation de volume. C'est-à-dire

$$\varepsilon_{ij} = \delta_{ij}\, \varepsilon_m + e_{ij}\ . \tag{2.12}$$

Cette façon de caractériser les matériaux est celle qui sera utilisée plus tard dans le programme. Elle permet de traiter de façon plus rigoureuse les matériaux qui présentent un comportement élastique non linéaire.

2.3 - LES MATÉRIAUX ÉLASTIQUES LINÉAIRES

Tel que mentionné plus haut, la loi de Hooke est adéquate pour représenter les matériaux dits élastiques linéaires. Cette loi s'écrit comme suit pour une sollicitation uniaxiale,

$$\varepsilon = \frac{\sigma}{E}, \tag{2.12}$$

où, ε est la déformation linéaire, σ est la contrainte normale appliquée et E, le module de Young. Ce modèle est de loin le plus utilisé et est valide pour beaucoup d'applications.

Il est possible de généraliser ce modèle à des sollicitations en trois dimensions (3D), en considérant les contraintes normales et de cisaillement appliquées sur les faces d'un élément. Ici, la loi de Hooke généralisée peut s'écrire sous forme matricielle comme suit pour un matériau quelconque,

$$\begin{vmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{vmatrix} = \begin{vmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} & a_{26} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} & a_{36} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} & a_{46} \\ a_{51} & a_{52} & a_{53} & a_{54} & a_{55} & a_{56} \\ a_{61} & a_{62} & a_{63} & a_{64} & a_{65} & a_{66} \end{vmatrix} \times \begin{vmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{vmatrix} \tag{2.13}$$

où les ε_i sont les déformations linéaires, les γ_{ij} sont les déformations angulaires, les σ_i sont les contraintes normales, les τ_{ij} sont les contraintes de cisaillement et les paramètres a_{11} à a_{66} définissent le type et le degré d'anisotropie du matériau. Il est possible d'inverser cette matrice afin d'obtenir les contraintes à partir de déformations.

Ces équations sont les bases fondamentales de la mécanique appliquée en élasticité linéaire. Lekhnitskii (1963) fait une revue de tous les types d'élasticité linéaire, dont les principaux sont l'isotropie, l'anisotropie avec un plan d'isotropie ou encore l'orthotropie.

2.3.1 LES MATÉRIAUX ISOTROPES

Le cas le plus simple de la loi de comportement élastique linéaire est sans aucun doute lorsque le matériau est isotrope, ou encore lorsque les propriétés du matériau ne varient pas selon l'orientation considérée.

Pour ce cas, tous les auteurs s'entendent pour écrire que

$$a_{11} = a_{22} = a_{33} = 1/E$$

$$a_{12} = a_{13} = a_{23} = a_{21} = a_{31} = a_{32} = -\nu/E \qquad (2.14)$$

$$a_{44} = a_{55} = a_{66} = 1/G = 2(1+\nu)/E.$$

Tous les autres paramètres a_{ij} sont nuls.

Ces équations sont simples et tous les logiciels d'éléments finis permettent leur traitement. Lors du traitement en deux dimensions, il est possible de traiter ce problème en état planes de contraintes (plane stress state) ou en état planes de déformations (plane strain state).

2.3.2 - LES MATÉRIAUX ANISOTROPES AVEC UN PLAN D'ISOTROPIE

Pour définir la loi de comportement d'un matériau élastique linéaire anisotrope possédant une isotropie transversale, Lekhnitskii (1963) présente un modèle où il généralise la loi de Hooke afin d'inclure cette anisotropie. Pour ce cas, les valeurs des paramètres a_{ij} sont les suivantes.

$$a_{11} = a_{22} = 1/E_1$$

$$a_{33} = 1/ E_2$$

$$a_{12} = a_{21} = -\nu_1/E_1 \qquad (2.15)$$

$$a_{13} = a_{31} = a_{23} = a_{32} = -\nu_2/E_2$$

$$a_{44} = 1/G_1 = 2(1+\nu_1)/E_1$$

$$a_{55} = a_{66} = 1/G_2$$

ici, E_1 et ν_1 sont respectivement le module de Young et le coefficient de Poisson dans le plan d'isotropie, E_2 et ν_2 sont respectivement le module de Young et le coefficient de Poisson mesuré perpendiculairement au plan d'isotropie et G_2 est le module de rigidité mesuré perpendiculairement au plan d'isotropie. Tous les autres paramètres a_{ij} sont nuls.

Les figure 2.3 et 2.4, tirée de Wittke (1991), montre les différents modules de rigidité et les coefficients de Poisson.

Brady et Brown (1993) pour leur part offrent une version différente pour les paramètres. Ils proposent que $a_{13} = a_{23} = -\nu_2/E_1$. Il est intéressant de noter cette différence. Dans ce mémoire, nous allons considérer uniquement l'approche de Lekhnitskii (1963).

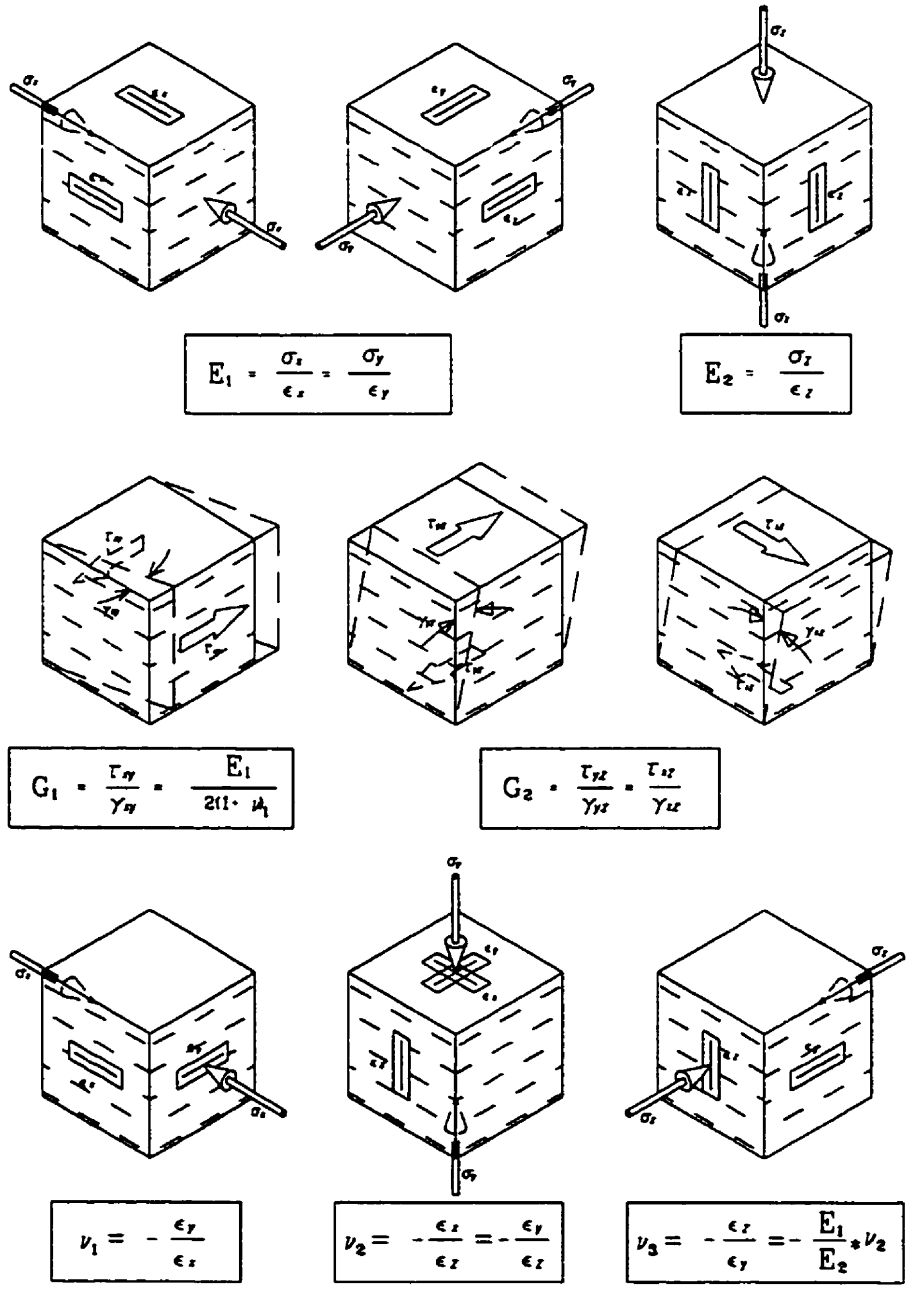


Figure 2.3 : Constantes élastiques et coefficients de Poisson pour matériau transversalement isotrope. (Wittke 1991)

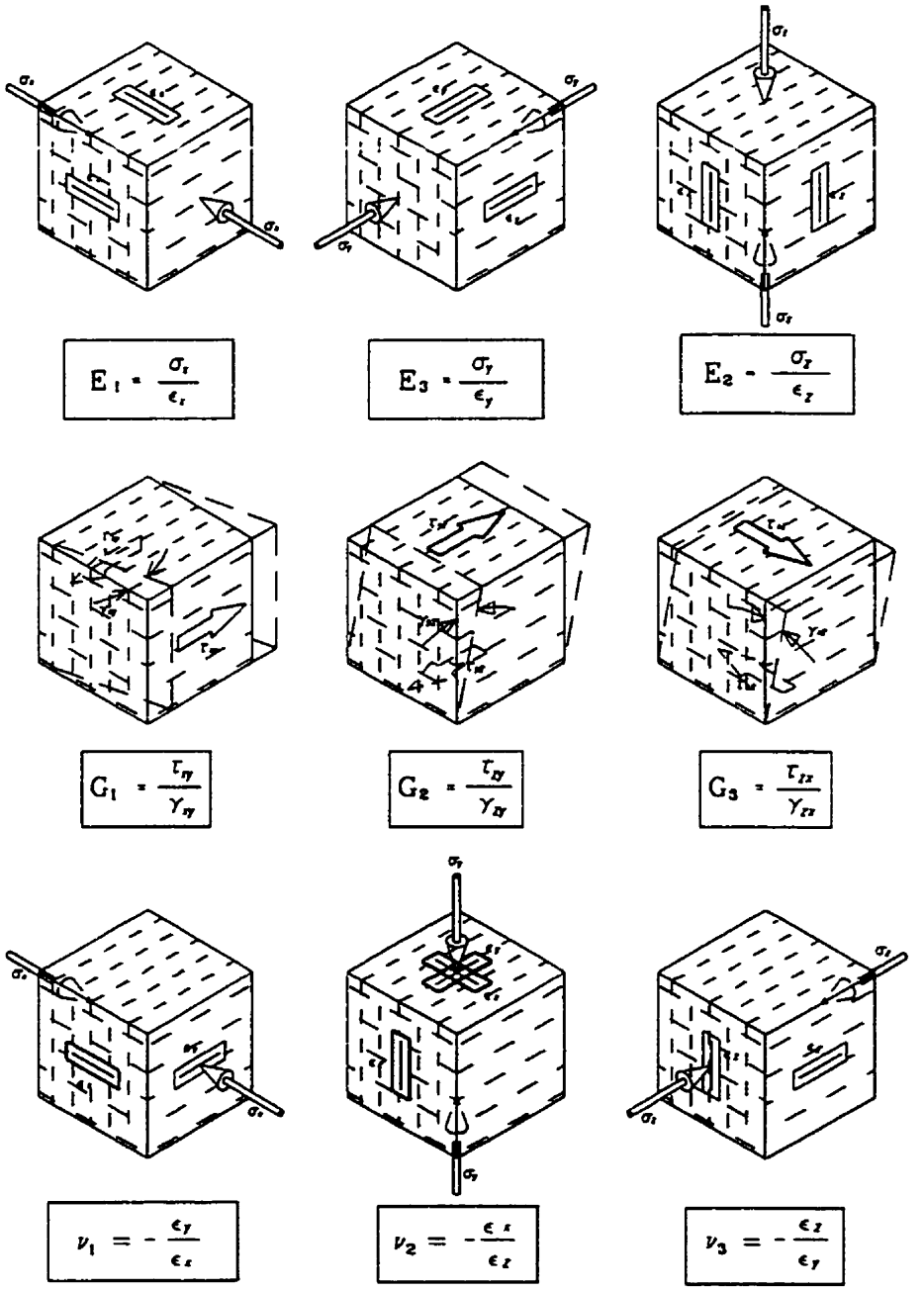


Figure 2.4 : Constantes élastiques et coefficients de Poisson pour matériau orthotrope. (Wittke 1991)

Pour la détermination de G_2, lorsqu'il est impossible de procéder à des essais en laboratoire, il est possible d'utiliser une estimation. Dans ce mémoire nous utiliserons celle de Batugin et Nirenburg (1972). Ils proposent que:

$$G_2 = E_1 E_2 / (E_1 + E_2 + 2 \nu E_2) \tag{2.16}$$

Ceci ne demeure qu'une estimation, mais elle est généralement bonne pour de nombreuses roches.

Corthésy et al. (1993) présentent les conversions pour passer des modules E_i, G_i et ν_i aux équations liant les variations de forme et de volume aux contraintes. Ces conversions sont les suivantes:

$$\begin{aligned}
\varepsilon_{m1} &= \sigma_m (2\, a_{11} + 2\, a_{12} + 4\, a_{13} + a_{33}) / 3 \\
e_{xx1} &= s_{xx}\, a_{11} + s_{yy}\, a_{12} + s_{zz} (2\, a_{13} + a_{11} + a_{12} - a_{33}) / 3 \\
e_{yy1} &= s_{yy}\, a_{11} + s_{xx}\, a_{12} + s_{zz} (2\, a_{13} + a_{11} + a_{12} - a_{33}) / 3 \\
e_{zz1} &= s_{zz} (a_{11} + a_{12} - 4\, a_{13} + 2\, a_{33}) / 3 \\
\varepsilon_{m2} &= s_{zz} (- a_{11} - a_{12} + a_{13} + a_{33}) / 3 \\
e_{xx2} &= e_{yy2} = \sigma_m (a_{11} + a_{12} - a_{13} - a_{33}) / 3 \\
e_{zz2} &= 2\, \sigma_m (- a_{11} - a_{12} + a_{13} + a_{33}) / 3
\end{aligned} \tag{2.17}$$

Pour ce cas, les paramètres associés à ces équations sont des constantes, ce qui indique bien la linéarité des équations. Dans ces équations nous remarquons que $e_{zz1} = fn(s_{zz})$ alors que e_{xx1} et $e_{yy1} = fn(s_{zz}, s_{xx}, s_{yy})$. Les variations de forme de premier ordre en x et y sont donc plus complexes à représenter puisqu'elles dépendent de trois composantes de contraintes déviatoriques.

Encore ici, le traitement de ce type de matériau est assez simple en éléments finis et la plupart des logiciels existants permettent de le considérer.

2.3.3 LES MATÉRIAUX ORTHOTROPES

Pour ce qui est des matériaux orthotropes, les paramètres a_{ij} sont les suivants

$$
\begin{aligned}
a_{11} &= 1/E_1 \\
a_{22} &= 1/E_2 \\
a_{33} &= 1/E_3 \\
a_{12} = a_{21} &= -\nu_1/E_1 \\
a_{13} = a31 &= -\nu_2/E_2 \\
a_{23} = a32 &= -\nu_3/E_3 \\
a_{44} &= 1/G_1 \\
a_{55} &= 1/G_2
\end{aligned}
\qquad (2.18)
$$

$$a_{66} = 1/G_3.$$

Nous ne traiterons toutefois pas des matériaux orthotropes dans ce mémoire, même si certaines roches présentent un tel comportement (Voight (1968)). La raison principale est que le banc d'essai généralement utilisé afin d'obtenir les paramètres ne permet pas de considérer l'orthotropie, sans compliquer à outrance la procédure expérimentale.

2.4 LES MATÉRIAUX ÉLASTIQUES NON LINÉAIRES

Le terme non linéaire implique que la relation entre la contrainte et la déformation n'en n'est pas une de premier degré. Il est généralement accepté que l'élasticité non linéaire présentant des relations contrainte - déformation concaves est due à la présence de micro-fissures dans le matériau qui se referment sous un état de contraintes donné pour produire un accroissement de la raideur du matériau. Lorsque les contraintes sont relâchées, les micro-fissures s'ouvrent à nouveau pour revenir à leur état de départ en ne laissant aucune déformation permanente.

La relation entre la déformation et la contrainte peut être définie comme une fonction de la contrainte. Ainsi, en compression uniaxiale;

$$\varepsilon = \sigma / fn(\sigma) \qquad (2.19)$$

Il est donc possible de définir un module de Young sécant pour n'importe quelle contrainte σ_i. Pour ce cas de chargement, le module de Young sécant est

$$E_t = \sigma_t / \varepsilon_t = fn(\sigma_t). \tag{2.20}$$

Il existe plusieurs modèles qui permettent de traiter de l'élasticité non linéaire. Nunes (1997) fait état de plusieurs relations empiriques des modèles élastiques et elles sont présentées au tableau 2.1

Dans la plupart des modèles mentionnés dans ce tableau, on fait appel à un moduie de Young de référence E_o. On fait ici allusion au module de Young en compression uniaxiale, mais on sait très bien que le module de Young augmente avec la contrainte, donc, à quel niveau de contrainte devrait-on définir E_o ?

Tableau 2.1 - Relations empiriques du module d'élasticité.

Modules de Young non linéaires, isotropes et anisotropes	
Référence	Relation empirique
Kulhawy (1975)	$E = E_o(\sigma_3)$
Santarelli (1986, 1987)	$E = E_o\,(1 + A\sigma_3)^2$
Lade & Nelson (1987)	$E = m\,P_a\{(I_1/P_a)^2 + 6\,(1+\nu)\,J_2/(1-2\nu)\,P_a\}$
Brown et al. (1989)	$E = B - C\,\exp(-A\,\sigma_3)$ $B = E_{\infty}\,(\sigma_3 \rightarrow \infty)$, $C = E_{\infty} - E_o$
Wang et Dusseault (1991) (Modèle anisotrope)	$E_i = E_o + A\,(\sigma_i)^{\eta}$ i = axial, radial
Wang et al. (1993) (Modèle anisotrope)	$E_a = E_o\,(1 + A\,\sigma_i)^{\eta}$ i = axial, radial
Shao et al. (1994)	$E = E_o\,\{\,1 + A\,(\sigma_3/P_a)^m\}$
a, A, B, C, η, m = constantes empiriques E_o = Module de Young en compression uniaxiale P_a = Pression atmosphérique ν = Coefficient de Poisson I_1 = Premier invariant des contraintes J_2 = Deuxième invariant de déviateur des contraintes	

Reste encore le dilemme de l'anisotropie apparente ou induite. Selon certains auteurs, tels Wu & Hudson (1991), un matériau élastique non linéaire et isotrope soumis à une compression uniaxiale va montrer un module de Young suivant l'axe de sollicitation plus grand que le module de Young perpendiculaire à l'axe de chargement. Ceci serait dû à la fermeture des micro-fissures perpendiculaire au chargement, tandis que les micro-fissures parallèles au chargement ne seraient que peu affectées et pourraient même s'ouvrir.

Desai & Siriwardane (1984) présentent des modèles élastiques de deuxième ordre (non linéaires) à partir desquels les termes associés à l'anisotropie induite apparaissent. Il s'agit des modèles élastiques de Cauchy et de Green tel que présenté par Eringen (1962). Pour le premier, on suppose une relation unique entre la contrainte et la déformation

$$\sigma_{ij} = f_{ij} (\varepsilon_{kl}), \tag{2.21}$$

où f_{ij} sont les fonctions de réponse (response functions). Pour le modèle de Green, les relations de contrainte-déformation sont déduites en tenant compte de l'énergie interne de déformation

$$\sigma_{ij} = \partial U_0 / \partial \varepsilon_{ij}, \tag{2.22}$$

où U_0 est l'énergie de déformation élastique qui est l'aire sous la relation contrainte - déformation. Pour les matériaux linéaires, on peut démontrer que les modèles de Cauchy

et de Green sont équivalents. Les différences apparaissent lorsque la non linéarité est introduite. Le modèle de Cauchy de deuxième ordre s'écrit:

$$\sigma_{ij} = (a_1 I_{1\varepsilon} + a_2 I_{1\varepsilon}^2 + a_3 I_{2\varepsilon})\delta_{ij} + (a_4 + a_5 I_{1\varepsilon})\varepsilon_{ij} + a_6 \varepsilon_{im}\varepsilon_{mj} \qquad (2.23)$$

Où $I_{1\varepsilon}$, $I_{2\varepsilon}$ et $I_{3\varepsilon}$ sont les invariants de déformation et a_1 à a_6 sont des constantes qui doivent être déterminées expérimentalement. Pour sa part, le modèle de Green de deuxième ordre s'écrit

$$U_0 = b_2 I_{1\varepsilon}^2 + b_3 I_{1\varepsilon}^3 + b_4 I_{1\varepsilon} I_{2\varepsilon} + b_5 I_{2\varepsilon} \cdot b_6 I_{3\varepsilon}, \qquad (2.24)$$

où b_2 à b_6 sont des constantes devant être déterminées expérimentalement. Nous remarquons que le modèle de Green requiert cinq constantes alors que le modèle de Cauchy en demande six.

Nous constatons donc que sans même introduire l'anisotropie intrinsèque (par opposition à l'anisotropie induite) nous avons déjà besoin de déterminer cinq ou six constantes de façon expérimentale pour caractériser le comportement non linéaire des matériaux intrinsèquement isotropes. Les modèles sont alors peu attrayants d'un point de vue pratique. Ceci explique la raison pour laquelle nous énonçons dans l'élaboration du modèle présenté au chapitre III une hypothèse de travail voulant qu'un matériau isotrope non linéaire demeure isotrope, peu importe les solliciations auxquelles il est soumis.

Comprenons bien qu'un matériau intrinsèquement anisotrope et montrant de la non linéarité peut superposer l'anisotopie induite à l'anisotropie intrinsèque. Nous pouvons supposer qu'il devient très ardu d'identifier la portion d'anisotropie à attribuer à chaque phénomène. Aussi, pour le dévelopment du modèle du chapitre III, nous allons prendre comme hypothèse qu'un matériau intrinsèquement anisotrope et non linéaire montre seulement les effets de l'anisotropie intrinsèque, sachant qu'elle peut inclure la portion de l'anisotropie induite par le type de sollicitation utilisée pour déterminer les relations contraintes-déformation.

Pour ce qui est de ce type de matériau, il existe à notre connaissance très peu de publications qui traitent de ce sujet. Corthésy et al. (1993) proposent, dans le cadre d'une mise au point d'une méthodologie de calcul des contraintes in situ, un modèle basé sur le principe de la superposition des variations de forme et de volume de premier ordre et de deuxième ordre afin de calculer les contraintes à partir de déformations connues. Les relations contrainte-déformation s'obtiennent, moyennant certaintes hypothèses, à partir d'essais en laboratoire qui seront décrits plus loin. Ce modèle est toutefois imparfait et nous allons en discuter plus longuement au cours du chapitre suivant.

2.5 LA MÉTHODE DES ÉLÉMENTS FINIS

La méthode d'analyse des contraintes et des déformations par éléments finis est aujourd'hui très répandue et nous ne croyons pas nécessaire d'en faire une description complète.

Zienkiewicz (1971) est l'un des nombreux auteurs ayant produit des ouvrages très complets en ce qui a trait à cette méthode numérique.

Le logiciel utilisé dans le présent mémoire est grandement inspiré des sous-routines présentés dans les ouvrages de Zienkiewicz. Ces sous-routines permettent de résoudre le cas d'éléments triangulaires à trois noeuds avec un état planes de déformations et ce pour des matériaux isotropes linéaires seulement. Toutefois, Zienkiewicz (1971) propose des modifications qui permettent le traitement en état planes de contraintes et aussi l'introduction de l'anisotropie des éléments.

CHAPITRE III

DÉVELOPPEMENT DU MODÈLE TRANSVERSALEMENT ISOTROPE ET NON LINÉAIRE

3.1 GÉNÉRALITÉS

Dans le chapitre précédent, nous avons fait une revue des différentes lois de comportement des matériaux élastiques. Dans le présent chapitre, nous discuterons du modèle que nous avons décidé d'adopter et des raisons qui nous ont poussé à le faire.

Même si nous voulons modéliser des matériaux transversalement isotropes et non linéaires, il demeure essentiel de pouvoir traiter les autres types de lois de comportement, soit élastique linéaire / isotrope, élastique linéaire / anisotrope et élastique non linéaire / isotrope.

3.2 MISE AU POINT SUR LE MODÈLE DE BASE

Le modèle choisi comme base pour notre programme est celui de Corthésy et al. (1993). Ce modèle a été développé pour le calcul des contraintes à partir de mesures de contraintes in situ faites à l'aide du *doorstopper*. Le modèle est basé sur la superposition

des variations de volume de premier ordre et de deuxième ordre ainsi que des variations de forme de premier et de deuxième ordre.

Ce modèle fait appel à quelques hypothèses. L'une d'entre elles, énoncée au chapitre II est qu'il n'y a pas d'anisotropie induite. En fait, la procédure expérimentale permettant la détermination des relations contrainte-déformation génère une anisotropie induite qui est reprise dans les termes exprimant l'anisotropie. La deuxième est que $\nu_1 = \nu_2$, et ces paramètres sont constants pour tout niveau de contrainte lors d'une compression biaxiale isotrope. La troisième hypothèse est que les variations de forme et de volume sont indépendantes les unes des autres. La dernière hypothèse est que l'image miroir des courbes contrainte-déformation est valide lorsque la valeur des contraintes est négative. En d'autre mots, on peut dire que $f_n(\sigma) = - f_n(-\sigma)$. Il est également convenu que toutes les sommes des invariants sont nulles, qu'il s'agisse de phénomène de 1er ou de 2ème ordre, donc

$$\begin{aligned} s_{xx} + s_{yy} + s_{zz} &= 0 \\ e_{xx1} + e_{yy1} + e_{zz1} &= 0 \\ e_{xx2} + e_{yy2} + e_{zz2} &= 0. \end{aligned} \tag{3.1}$$

À ce point, il peut être utile de présenter le concept de compression biaxiale isotrope. Pour cet essai, une carotte de roche comportant déjà une cellule *doorstopper*, qui comporte une rosette à 4 jauges, est comprimée radialement tout en étant non sollicitée

axialement. Les déformations sont mesurées suivant les axes x et z et calculée suivant y, la direction suivant l'axe de la carotte, en postulant un état de contraintes planes. C'est de cet essai dont on tire les relations contrainte déformation requises dans le modèle. La figure 3.1 montre une telle compression biaxiale isotrope.

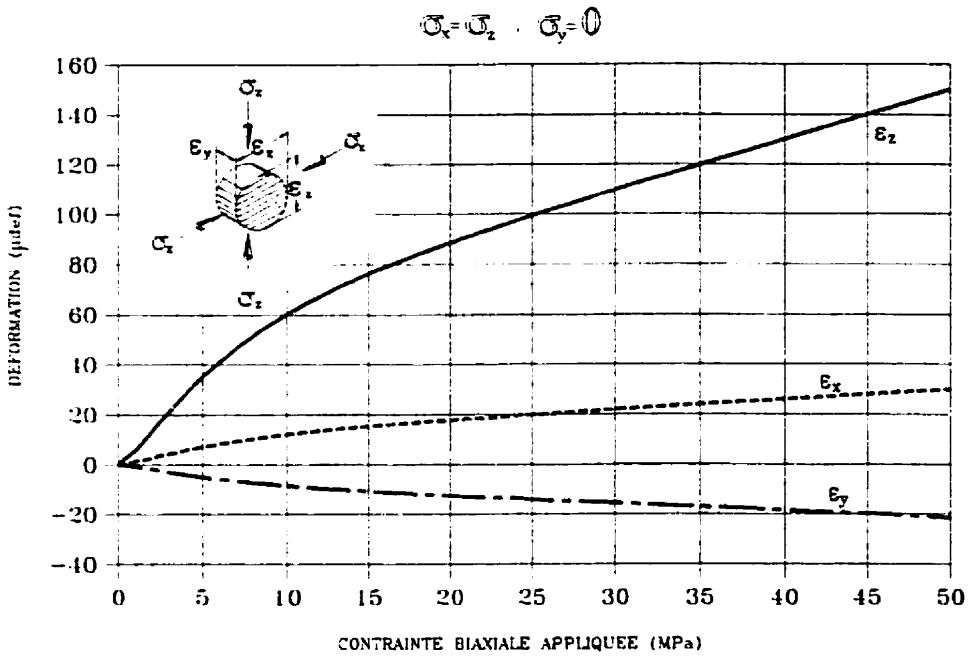


Figure 3.1 Compression biaxiale isotrope

3.3 MODIFICATIONS NÉCESSAIRES AU MODÈLE

Tout en développant le programme d'éléments finis, nous nous sommes rendu compte que le modèle de Corthésy et al. (1993) possédait des lacunes. Lors de simulations, un matériau isotrope non linéaire se comportait de façon anisotrope. Voici le développement.

Pour fin de démonstration, supposons un matériau quelconque pour lequel on défini dans le plan x,z quatre fonctions de variations définies au chapitre II soit

$$\varepsilon_{m1} = 70\ \sigma_m^{0.75}$$

$$e_{zz1} = 100\ s_{zz}^{0.75} \qquad (3.2)$$

$$e_{zz2} = 0$$

$$\varepsilon_{m2} = 0.$$

Ce matériau est isotrope, car il ne possède pas d'effet de deuxième ordre. Pour la première partie de cette simulation, considérons une compression uniaxiale de 15 MPa selon l'axe z. La contrainte moyenne et les déviateurs de contrainte sont donc

$$\sigma_m = 5 \text{ MPa}$$

$$s_{xx} = -5 \text{ MPa}$$

$$s_{yy} = -5 \text{ MPa}$$

$$s_{zz} = 10 \text{ MPa}.$$

Nous obtenons pour les variations de forme et volume

$$\varepsilon_{m1} = 234\ \mu \text{ déf.}$$

$$e_{zz1} = 562\ \mu \text{ déf.}.$$

Étant donné que la somme des déviateurs de déformation doit être nulle, nous obtenons

$$e_{xx1} = e_{yy1} = -\ 281\ \mu \text{ déf.},$$

et donc,

$$\varepsilon_x = \varepsilon_y = -\ 46.9\ \mu \text{ déf.}$$

$$\varepsilon_z = 796\ \mu \text{ déf.}.$$

Maintenant, pour la seconde partie de cette simulation, considérons une compression uniaxiale de 15 MPa suivant l'axe x. Le même exercice montre que

$$\sigma_m = 5 \text{ MPa}$$

$$s_{xx} = 10 \text{ MPa}$$

$$s_{yy} = -5 \text{ MPa}$$

$$s_{zz} = -5 \text{ MPa}.$$

Nous obtenons

$$\varepsilon_{m1} = 234\ \mu \text{ déf.}$$

$$e_{zz1} = -\ 334\ \mu \text{ déf.}.$$

Le matériau étant isotrope, nous obtenons $e_{yy1} = e_{zz1} = -334$ µdéf. car le niveau de contrainte est le même et $e_{xx1} = -(e_{yy1} + e_{zz1})$. Ce qui donne

$$\varepsilon_x = 902 \text{ µdéf.}$$

$$\varepsilon_y = \varepsilon_z = -100 \text{ µdéf..}$$

Donc, pour un matériau isotrope, le modèle original conduit à une déformation de 796 µdéf dans la direction de l'application de la contrainte uniaxiale lorsque le chargement se fait suivant l'axe *z* et une déformation de 902 µdéf dans la direction de l'application de la contrainte uniaxiale lorsque le chargement se fait suivant l'axe *x*. La déformation transversale est de -46.9 µdéf dans le premier cas et de -100 µdéf dans le second cas. Nous pouvons en conclure que ce modèle doit être modifié car les déformations pour un matériau isotrope sont dépendantes de l'orientation de l'application du chargement.

D'autre part, si pour remédier à ce problème nous supposons qu'étant donné le fait que le matériau est isotrope, la même relation définit les trois déviateurs de déformation. Donc, pour la compression uniaxiale selon l'axe x proposé plus haut, nous avons

$$s_{xx} = 10 \text{ MPa}$$

$$s_{yy} = -5 \text{ MPa}$$

$$s_{zz} = -5 \text{ MPa.}$$

Ce qui donne les déviateurs de déformations suivants

$$e_{xx} = 562 \ \mu \text{ déf.}$$

$$e_{yy} = -334 \ \mu \text{ déf.}$$

$$e_{zz} = -334 \ \mu \text{ déf..}$$

Si on fait la somme des déviateurs de déformation, nous notons

$$e_{xx} + e_{yy} + e_{zz} = -106 \ \mu \text{ déf..}$$

Nous avons vu plus tôt que la somme des trois déviateurs de déformation doit être par définition nulle en tout temps. Ceci pourrait être le cas s'il y avait des effets de variation de volume de deuxième ordre. Or nous avons postulé que lorsque le matériau est isotrope, il n'y a pas d'effet de deuxième ordre. Si l'on suppose qu'il y a une anisotropie apparente ou induite, il faudrait considérer la variation de volume de deuxième ordre. Toutefois, nous n'avons trouvé aucun moyen de modéliser cette variation de volume de deuxième ordre qui permettrait un état de sollicitation quelconque.

La solution proposée pour pouvoir répondre à ces exigences est la suivante. Nous considérons que le comportement du matériau est uniquement influencé par le déviateur de contrainte le plus grand en valeur absolue. Ainsi on obtient

$$s_{ijmax} = \max (| s_{11} |, | s_{22} |, | s_{33} |), \tag{3.3}$$

où s_{11}, s_{22} et s_{33} sont les déviateurs des contraintes principales, c'est-à-dire dans les directions où il n'y a pas de cisaillement.

Ainsi les déviateurs de déformation seront établis de façon suivante

$$\begin{aligned} e_{xx1} &= fn'(s_{xx}) = fn(s_{ijmax}) * s_{xx} / s_{ijmax} \\ e_{yy1} &= fn'(s_{yy}) = fn(s_{ijmax}) * s_{yy} / s_{ijmax} \\ e_{zz1} &= fn'(s_{zz}) = fn(s_{ijmax}) * s_{zz} / s_{ijmax}. \end{aligned} \tag{3.4}$$

De cette façon, pour la compression uniaxiale selon l'axe x, nous avons $\sigma_x = \sigma_1 = 15$ MPa, $\sigma_y = \sigma_2 = 0$ et $\sigma_z = \sigma_3 = 0$, donc

$$\sigma_m = 5 \text{ MPa}$$

$$s_{11} = 10 \text{ MPa}$$

$$s_{22} = -5 \text{ MPa}$$

$$s_{33} = -5 \text{ Mpa}.$$

À la figure 3.2, on montre s_{11} = 10 MPa; s_{22} = -3.33 MPa et s_{33} = -6.67 MPa.

Donc,

$$s_{ijmax} = 10 \text{ MPa.}$$

Ainsi on obtient

$$e_{xx} = 562 \text{ µdéf.}$$

$$e_{yy} = -281 \text{ µdéf.}$$

$$e_{zz} = -281 \text{ µdéf..}$$

Ainsi nous avons ε_1 = 796 µdéf. et $\varepsilon_2 = \varepsilon_3$ = -47 µdéf., peu importe l'axe selon lequel on effectue la compression uniaxiale.

Nous pouvons voir une récapitulation des différents calculs effectués dans les pages précédantes dans le tableau 3.1 qui suit.

Tableau 3.1 - Récapitulations des méthodes de calcul des déformations pour un matériau isotrope et non linéaire

	Cas 1a	Cas 1b	Cas 2	Cas 3a	Cas 3b
σ_x	0 MPa	15 MPa	15 MPa	15 MPa	0 MPa
σ_z	15 MPa	0 MPa	0 MPa	0 MPa	15 MPa
σ_m	5 Mpa	5 Mpa	5 Mpa	5 Mpa	5 Mpa
σ_{xx}	-5 Mpa	10 Mpa	10 Mpa	10 Mpa	-5 Mpa
σ_{yy}	-5 Mpa	-5 Mpa	-5 Mpa	-5 Mpa	-5 Mpa
σ_{zz}	10 Mpa	-5 Mpa	-5 Mpa	-5 Mpa	10 Mpa
ε_m	234 µdéf.	234 µdéf.	234 µdéf.	234 µdéf.	234 µdéf.
ε_{xx}	-281 µdéf.	668 µdéf.	562 µdéf.	562 µdéf.	-281 µdéf.
ε_{yy}	-281 µdéf.	-334 µdéf.	-334 µdéf.	-281 µdéf.	-281 µdéf.
ε_{zz}	582 µdéf.	-334 µdéf.	-334 µdéf.	-281 µdéf.	562 µdéf.
Inv_1	0 µdéf.	0 µdéf.	-106 µdéf.	0 µdéf.	0 µdéf.
ε_x	-47.9 µdéf.	902 µdéf.	796 µdéf.	796 µdéf.	-47 µdéf.
ε_y	-47.9 µdéf.	-100 µdéf.	-100 µdéf.	-47 µdéf.	-47 µdéf.
ε_z	796 µdéf.	-100 µdéf.	-100 µdéf.	-47 µdéf.	796 µdéf.

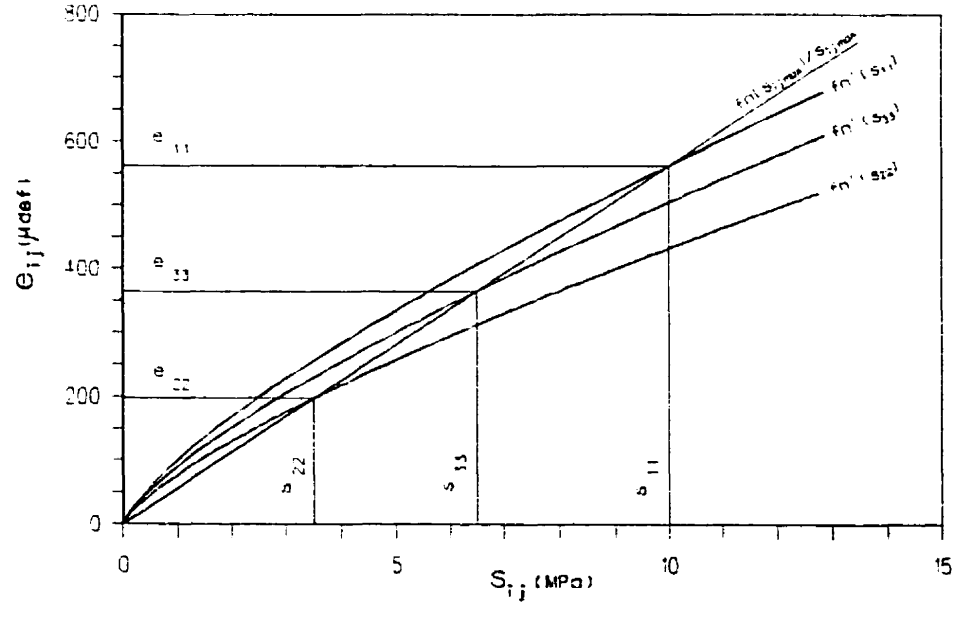


Figure 3.2 - Fonctions de variation de forme de premier ordre.

Si l'on fait un bref rappel, les fonctions de variation de formes ont été obtenues à partir d'un essai de compression biaxiale isotrope, où $\sigma_1 = P$, $\sigma_2 = P$ et $\sigma_3 = 0$. Donc, $s_{11} = s_{22} = P/3$ et $s_{33} = -2P/3$. Nous pouvons donc constater que le déviateur de contrainte le plus grand est s_{33}. C'est donc celui-ci que nous allons devoir utiliser afin de déterminer la variation de forme de premier ordre.

Un autre élément qui a retenu notre attention est la façon de déterminer e_{xx1} et e_{yy1} pour un niveau de contrainte quelconque lorsque le matériau est anisotrope. Les paramètres obtenus par Corthésy et al. (1993) sont pour un cas bien particulier soit une compression biaxiale isotrope où les contraintes principales dans le plan sont égales et la contrainte perpendiculaire au plan est nulle.

Dans l'annexe de Corthésy et al. (1993), les auteurs déterminent une façon de calculer e_{xx1} et e_{yy1} en fonction de s_{xx}, s_{yy} et s_{zz}. Il suffit de soustraire une équation à l'autre pour obtenir

$$e_{xx1} - e_{yy1} = (s_{xx} - s_{yy})\,(a_{11} - a_{12}), \qquad (3.5)$$

où $a_{11} = 1/E_1$ et $a_{12} = -\nu/E_1$, et E_1 est le module sécant pour ce niveau de contrainte.

Donc, nous savons que e_{zz1} est fonction de s_{zz}, nous savons également que $e_{xx1} - e_{yy1}$ est fonction de $s_{xx} - s_{yy}$, et finalement, nous savons que la somme des déviateurs de

déformation doit être nulle. Nous avons donc tout ce qui nous est nécessaire pour déterminer les trois paramètres. Nous allons donc devoir utiliser une fonction supplémentaire à celles utilisées par Corthésy et al. (1993).

Pour ce qui est des effets des phénomènes de deuxième ordre, nous ne voyons pas d'inconvénients à utiliser le modèle qu'ils proposent.

3.3 MÉTHODOLOGIE DE CALCUL

Afin de mieux expliquer la façon dont les calculs sont faits à l'intérieur du programme d'éléments finis, nous allons procéder à l'aide d'un exemple.

Supposons un matériau quelconque dont les fonctions définies au chapitre II sont des polynômes de deuxième degré définis comme suit

$$\begin{aligned} \varepsilon_{m1} &= 61.7\,\sigma_m - 1.85\,\sigma_m^2 \\ e_{zz1} &= (143.3\,s_{ijmax} - 4.3\,s_{ijmax}^2)\,(s_{zz}/s_{ijmax}) \\ e_{zz2} &= 13.3\,\sigma_m - 0.4\,\sigma_m^2 \\ \varepsilon_{m2} &= 6.7\,s_{zz} - 0.4\,s_{zz}^2. \end{aligned} \tag{3.7}$$

La fonction supplémentaire définie plus haut est

$$e_{xx1} - e_{yy1} = 120\,(s_{xx}-s_{yy}) - 0.4\,(s_{xx}-s_{yy})^2 \quad .$$

Supposons un état de compression uniaxiale selon l'axe z ou perpendiculaire au plan d'isotropie où la contrainte appliquée est de 10 MPa. Nous avons donc

$$\sigma_m = 3.3 \text{ MPa}$$

$$s_{xx} = -3.3 \text{ MPa}$$

$$s_{yy} = -3.3 \text{ MPa}$$

$$s_{zz} = 6.7 \text{ MPa}$$

$$s_{ijmax} = 6.7 \text{ MPa}$$

$$s_{xx}-s_{yy} = 0.$$

Nous pouvons donc calculer

$$\varepsilon_{m1} = 185.0 \text{ µdéf.}$$

$$e_{zz1} = 764.4 \text{ µdéf.}$$

$$e_{zz2} = 40 \text{ µdéf.}$$

$$\varepsilon_{m2} = 26.7 \text{ µdéf.}$$

$$e_{xx1} - e_{yy1} = 0 \text{ µdéf..}$$

Donc

$$e_{xx1} = e_{yy1} = -382.2 \text{ µdéf.}$$

$$e_{xx2} = e_{yy2} = -20 \text{ µdéf..}$$

Ce qui nous donne

$$\varepsilon_x = \varepsilon_y = -190.6 \ \mu\text{déf.}$$

$$\varepsilon_z = 1016.1 \ \mu\text{déf..}$$

Si l'on suppose une compression uniaxiale de 10 MPa selon l'axe x ou parallèle au plan d'isotropie cette fois, nous avons

$$\sigma_m = 3.3 \text{ MPa}$$

$$s_{xx} = 6.7 \text{ MPa}$$

$$s_{yy} = -3.3 \text{ MPa}$$

$$s_{zz} = -3.3 \text{ MPa}$$

$$s_{ijmax} = 6.7 \text{ MPa}$$

$$s_{xx}-s_{yy} = 10 \text{ MPa.}$$

Nous pouvons donc calculer

$$\varepsilon_{m1} = 185.0 \ \mu\text{déf.}$$

$$e_{zz1} = -382.2 \ \mu\text{déf.}$$

$$e_{zz2} = 40 \ \mu\text{déf.}$$

$$\varepsilon_{m2} = -17.8 \ \mu\text{déf.}$$

$$e_{xx1} - e_{yy1} = 960 \ \mu\text{déf..}$$

Donc

$$e_{xx1} = 671.1 \text{ µdéf.}$$

$$e_{yy1} = -288.9 \text{ µdéf.}$$

$$e_{xx2} = -20 \text{ µdéf.}$$

$$e_{yy2} = -20 \text{ µdéf..}$$

Ce qui nous donne

$$\varepsilon_x = 818.3 \text{ µdéf.}$$

$$\varepsilon_y = -141.7 \text{ µdéf.}$$

$$\varepsilon_z = -175.0 \text{ µdéf..}$$

Comme on peut le constater pour ce cas, le matériau est plus déformable perpendiculairement au plan d'isotropie que dans ce plan. La figure 3.3 montre le comportement du matériau utilisé en exemple pour un état de compression uniaxiale selon le plan d'isotropie et perpendiculairement à ce dernier.

Afin de pouvoir utiliser ce modèle dans le programme d'éléments finis, il faut fournir les modules d'élasticité que génèrent ces calculs. Les modules d'élasticité sont alors obtenus par les équations suivantes qui sont dérivées des équations du chapitre II.

$$E_2 = [\sigma_z - \nu(\sigma_x + \sigma_y)]/\varepsilon_z \qquad (3.8)$$

$$E_1 = (\sigma_x - \nu\sigma_y)/[\varepsilon_x + (\nu\sigma_z/E_2)]$$

Il s'agit de modules sécants qui sont définis pour un état de contrainte polyaxial.

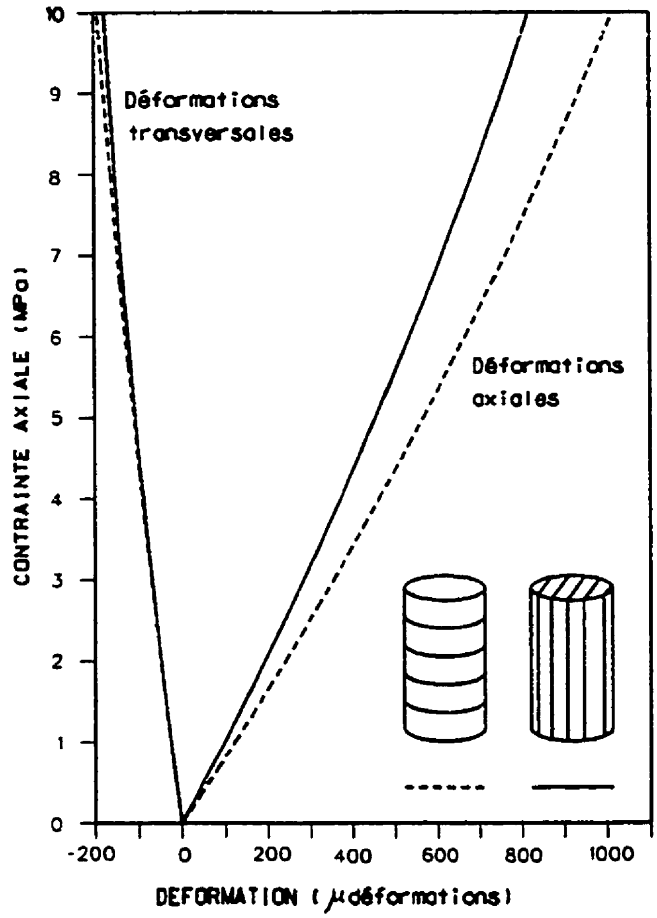


Figure 3.3 : Déformations dues à une compression uniaxiale

Ainsi pour le premier cas étudié plus haut, on a σ_z = 10 MPa; $\sigma_x = \sigma_y = 0$; ε_z = 1016.1 μdéf. et $\varepsilon_x = \varepsilon_y$ = -190.6 μdéf. Donc pour ce cas

$$\nu = 0.188$$

$$E_2 = 9.84 \text{ GPa}$$

et

$$E_1 \text{ est indéterminé.}$$

Pour ce premier cas, il est impossible d'évaluer E_1. Ceci peut être expliqué par le fait que pendant une compression uniaxiale suivant un axe perpendiculaire au plan d'isotropie, E_2 et ν_2 sont seuls à contrôler la déformabilité de l'éprouvette. Même s'il y a une indétermination pour la valeur de E_1, nous devons tout de même fournir une valeur pour ce module. Nous allons donc utiliser $E_1 = E_2$ sachant que les modules sont similaires et que E_1 n'intervient pas dans les calculs de déformation, et sa valeur n'est pas vraiment importante pour ce cas particulier.

Pour le deuxième cas étudié plus haut, nous avons $\sigma_x = 10$ MPa; $\sigma_y = \sigma_z = 0$; $\varepsilon_x = 818.3$ µdéf., $\varepsilon_y = -141.7$ µdéf. et $\varepsilon_z = -175.0$ µdéf. Donc pour cette analyse

$$\nu = 0.214$$

$$E_2 = 12.2 \text{ GPa}$$

et

$$E_1 = 12.2 \text{ GPa.}$$

Donc nous voyons que pour ce type de sollicitation, le matériau est plus rigide que pour le cas antérieur.

Pour la détermination de G_2 , nous utilisons l'approximation de Batugin et Nirenburg (1972) telle que discutée précédemment à la section 2.3.2.

CHAPITRE IV

APPLICATION INFORMATIQUE

4.1 GÉNÉRALITÉS

Comme mentionné dans l'introduction, le but de cette recherche est de développer un programme d'analyse de contraintes et de déplacements en utilisant la méthode des éléments finis qui permet de simuler un matériau possédant une loi de comportement élastique non linéaire et transversalement isotrope.

Dans les chapitres précédents, nous avons discuté de différents modèles qu'il était possible d'utiliser, pour ensuite s'attarder sur le modèle élastique non linéaire et transversalement isotrope. Dans le présent chapitre, nous allons décrire la méthode utilisée afin de permettre l'introduction dans un code d'éléments finis de cette loi de comportement .

4.2 ALGORITHME DU PROGRAMME D'ÉLÉMENTS FINIS

La plupart des programmes d'analyse de contraintes par éléments finis utilisent cet algorithme:

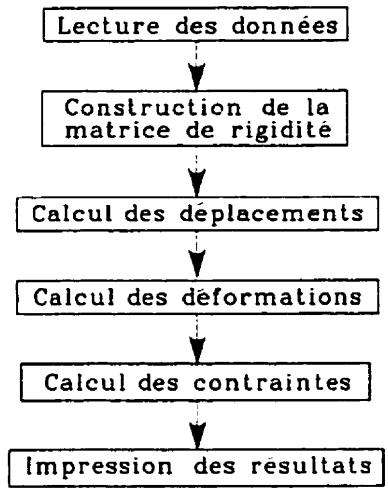


Figure 4.1 - Algorithme des modèles d'analyse de contraintes par éléments finis

Cet algorithme est valable pour les matériaux montrant un comportement élastique linéaire isotrope ou anisotrope, car le module de Young et le coefficient de Poisson sont constants pour tout niveau de contrainte. Pour ce qui est des matériaux élastiques non linéaires, étant donné que le module de Young varie en fonction des contraintes appliquées, il est nécessaire d'ajouter une boucle à cet algorithme. À l'intérieur de cette boucle, les modules de Young seront déterminés élément par élément en utilisant les contraintes calculées à la boucle précédente. Il est bon de noter que le modèle proposé suppose un coefficient de Poisson unique et constant. Ainsi, à chaque itération une nouvelle matrice de rigidité globale va être calculée pour déterminer de nouveaux déplacements et de nouvelles contraintes pour chaque éléments. Le test de convergence est effectué sur le niveau de contraintes entre deux itérations successives.

Nous avons donc modifié l'algorithme ci-haut de façon suivante :

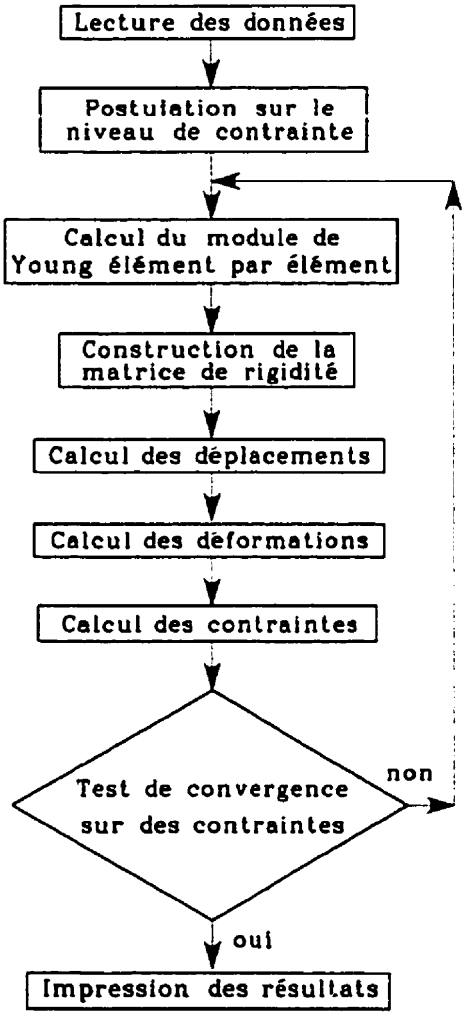


Figure 4.2 - Algorithme utilisé dans notre modèle d'analyse de contraintes par éléments finis

La loi de comportement du matériau est introduite au calcul du module de Young élément par élément et c'est cette sous-routine dans le logiciel développé dans ce présent mémoire qu'il faut modifier afin d'introduire d'autres lois de comportement.

4.3 L'INTÉGRATION INFORMATIQUE

Afin d'intégrer ces notions à l'intérieur d'un programme informatique, nous avons utilisé le programme *Microsoft Extended Quick Basic*, qui fonctionne sur ordinateur personnel

Gill & Gill (1988) ont écrit un programme dans lequel ils ont traduit les sous-routines de Zienkiewicz (1971) en Basica fonctionnant sur *PC XT*. Nous avons utilisé ce programme comme base pour notre programmation. Nous sommes conscients que le langage *Basic* n'est certainement pas le plus efficace, mais sa simplicité d'utilisation et sa connaissance par un grand nombre de personnes ont incité son utilisation pour ce projet.

La méthodologie de calcul présentée au chapitre III et utilisée dans la sous-routine *detmodule* est le coeur du logiciel. C'est cette dernière qui permet d'introduire les lois de comportement désirées. Les paramètres d'entrée de cette sous-routine sont les contraintes dans le plan d'analyse, le type de traitement, soit état plan de contraintes ou état plan de déformations et les propriétés du matériau.

C'est à cette étape que les fonctions du matériau telles que définies aux chapitres II et III sont utilisées afin de caractériser le matériau. Les fonctions utilisées afin de caractériser chaque courbe sont du type suivant :

$$y = mx + b(1 - e^{-ax})^c \tag{4.1}$$

Où m, b, a et c sont les paramètres de la fonction. Cette fonction permet de représenter des lois de comportement tendant vers une asymptote d'équation y = mx + b. Les deux autres paramètres permettent d'atténuer la fonction, c pour de faibles valeurs de x et a pour de plus grandes valeurs de x. La figure 4.1 qui suit montre l'effet des différents paramètres sur l'allure des courbes dans l'espace x-y.

Nous recommandons l'utilisation de l'option solveur du logiciel Microsoft Excel afin de déterminer ces différents paramètres.

Les paramètres déterminés ici sont ceux qui permettent de caractériser le matériau à l'intérieur du logiciel d'éléments finis. Pour les matériaux élastiques linéaires, le logiciel permet d'entrer directement les valeurs des modules et du coefficient de Poisson et de calculer directement les paramètres définis plus haut.

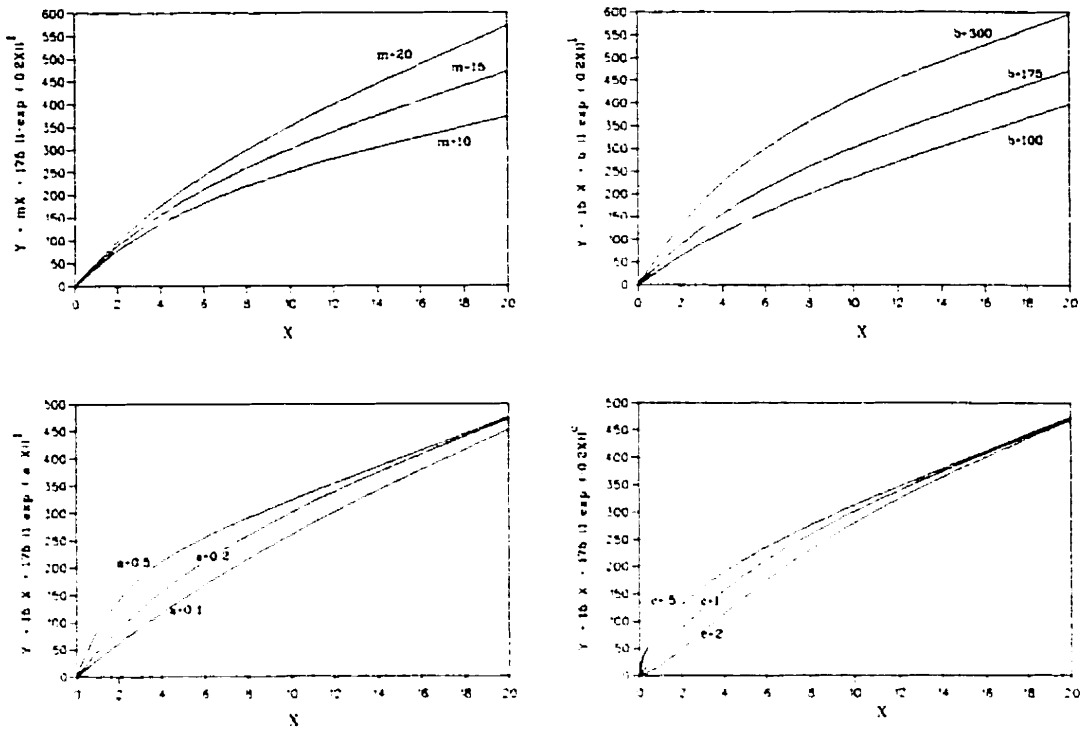


Figure 4.3 - Lissage de courbe pour les fonctions proposées

Donc, pour résumer, la sous-routine *detmodule* utilise les fonctions définies ci-haut afin de déterminer les déformations d'un élément pour un niveau de contrainte désiré. Considérant ces déformations et le niveau de contrainte, la sous-routine évalue des modules de Young sécants ou équivalents pour cet élément. Pour la suite de cette itération, le logiciel considère que chaque élément est constitué d'un matériau possédant différentes propriétés élastiques si les contraintes qui prévalent dans chacun de ces éléments sont différentes.

Le format du fichier d'entrée des données est celui défini par Gill et Gill (1988) pour le logiciel *ELEFIN*. Toutefois, l'utilisation d'un préprocesseur est très avantageuse. Ainsi nous avons écrit un programme de transfert de format. Il est possible alors d'utiliser les fichiers *NEUTRAL FILE* générés par le logiciel de génération de maillage *MTOOL* développé par l'Université Catholique de Rio de Janiero (1992).

Comme postprocesseur nous utilisons soit un chiffrier électronique du genre EXCEL ou encore un logiciel de dessin assisté par ordinateur du genre AUTOCAD par l'entremise de fichier de format DXF.

CHAPITRE V

VÉRIFICATIONS ET VALIDATION DU MODÈLE

5.1 GÉNÉRALITÉS

Aux chapitres précédents, nous avons développé une loi de comportement pour un matériau élastique non linéaire et transversalement isotrope. Nous avons aussi expliqué la méthodologie utilisée afin d'intégrer ce modèle à l'intérieur d'un programme d'éléments finis.

Dans le présent chapitre, nous allons procéder à une vérification des résultats obtenus à l'aide de notre logiciel d'éléments finis. Nous allons faire des analyses en utilisant un matériau isotrope et linéaire, puis un matériau anisotrope et linéaire, ensuite un matériau isotrope et non linéaire et finalement un matériau anisotrope et non linéaire.

Il est important de mentionner que les résultats générés par le logiciel d'éléments finis ne sont que des approximations. La qualité de ces estimations est influencée par la quantité d'éléments utilisés pour définir le modèle et également le type d'élément utilisé. Ainsi, il est possible d'augmenter la précision des résultats en utilisant plus d'éléments. Il est aussi possible d'augmenter la précision des résultats en utilisant des éléments quadratiques du

genre triangles à 6 noeuds plutôt que des éléments linéaires du genre triangle à 3 noeuds. Le logiciel d'éléments finis utilisé dans le présent mémoire ne permet que le traitement des triangles à 3 noeuds. Il est possible d'utiliser jusqu'à 750 éléments pour définir une maille.

5.2 DESCRIPTION DU MODÈLE ANALYSÉ

Pour analyser le comportement d'un matériau élastique non linéaire, il est nécessaire d'utiliser un modèle dans lequel il y a un gradient de contrainte, c'est-à-dire dans lequel le niveau de contrainte varie d'un élément à l'autre. Ceci va permettre à la non linéarité de se manifester par une variation des modules de Young des différents éléments soumis à différents niveaux de contraintes.

Nous allons simuler une plaque mince dans laquelle un trou circulaire est percé. Il existe des solutions analytiques pour ce type de modèle ce qui nous permet de vérifier la solution obtenue à l'aide du logiciel que nous avons développé. La figure 5.1 montre le maillage typique utilisé pour les analyses.

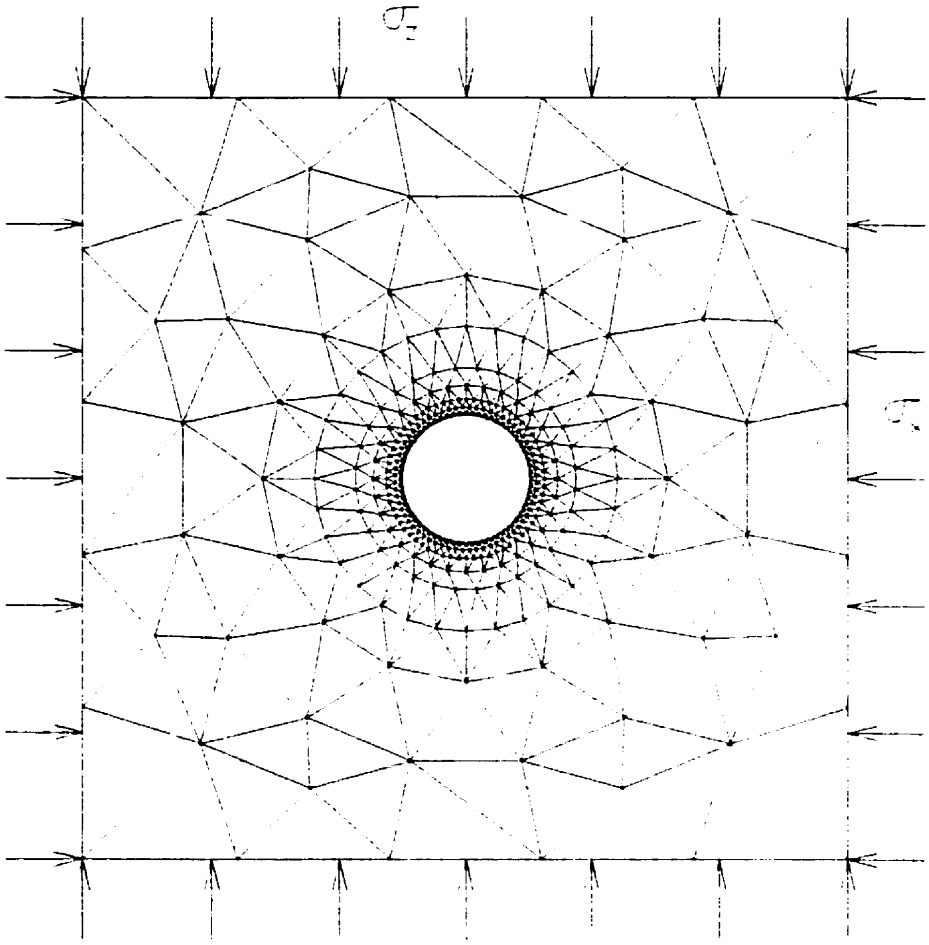


Figure 5.1 : Maillage typique utilisé pour les essais de validations

5.3 ANALYSE D'UN MATÉRIAU ÉLASTIQUE LINÉAIRE ET ISOTROPE

Le premier cas que nous allons considérer est celui du matériau élastique linéaire et isotrope. La solution analytique pour ce matériau nous vient de Kirsch (1898). Nous pouvons donc calculer les contraintes à l'aide de

$$\sigma_r = \frac{\sigma_z}{2}\left[(1+K)\left(1-\frac{a^2}{r^2}\right)-(1-K)\left(1-4\frac{a^2}{r^2}+3\frac{a^4}{r^4}\right)\cos(2\theta)\right]$$

(5.1)

$$\sigma_\theta = \frac{\sigma_z}{2}\left[(1+K)\left(1+\frac{a^2}{r^2}\right)+(1-K)\left(1+3\frac{a^4}{r^4}\right)\cos(2\theta)\right]$$

où σ_r est la contrainte radiale, σ_θ est la contrainte tangentielle, σ_z est la contrainte verticale sur les bordures de la plaque, a est le rayon de l'orifice, r et θ sont les coordonnées polaires du point étudié et $K = \sigma_x/\sigma_z$, avec σ_x représentant la contrainte horizontale sur la bordure vertiale de la plaque. Il est également possible de calculer les déplacements en utilisant

$$u_r = -\frac{\sigma_z a^2}{4Gr}\left[(1+K)-(1-K)\left\{4(1-\nu)-\frac{a^2}{r^2}\right\}\cos(2\theta)\right]$$

(5.2)

$$u_\theta = \frac{\sigma_z a^2}{4Gr}\left[(1-K)\left\{2(1-2\nu)+\frac{a^2}{r^2}\right\}\sin(2\theta)\right]$$

où u_r est le déplacement radial, u_θ est le déplacement tangentiel, ν le coefficient de Poisson, G le module de rigidité et les autres paramètres tels que définis plus haut. Pour fin de vérification, nous supposons l'état de contrainte suivant : $\sigma_x = 0$ MPa et $\sigma_z = 40$ MPa. Pour a = r = 0.5; E = 10 GPa et $\nu = 0.15$, nous avons sur la figure 5.2, un

graphique des contraintes radiales et tangentielles ainsi que les déplacements radiaux et tangentiels. Les traits pleins montrent les valeurs calculées à l'aide des équations de Kirsch (1898) et les points sont les valeurs obtenues à l'aide du programme d'éléments finis. Les zones hachurées montrent une valeur négative. Il est à noter que les contraintes calculées avec le logiciel sont pour le centre de l'élément en question. Ceci explique les valeurs de contraintes radiales non nulles.

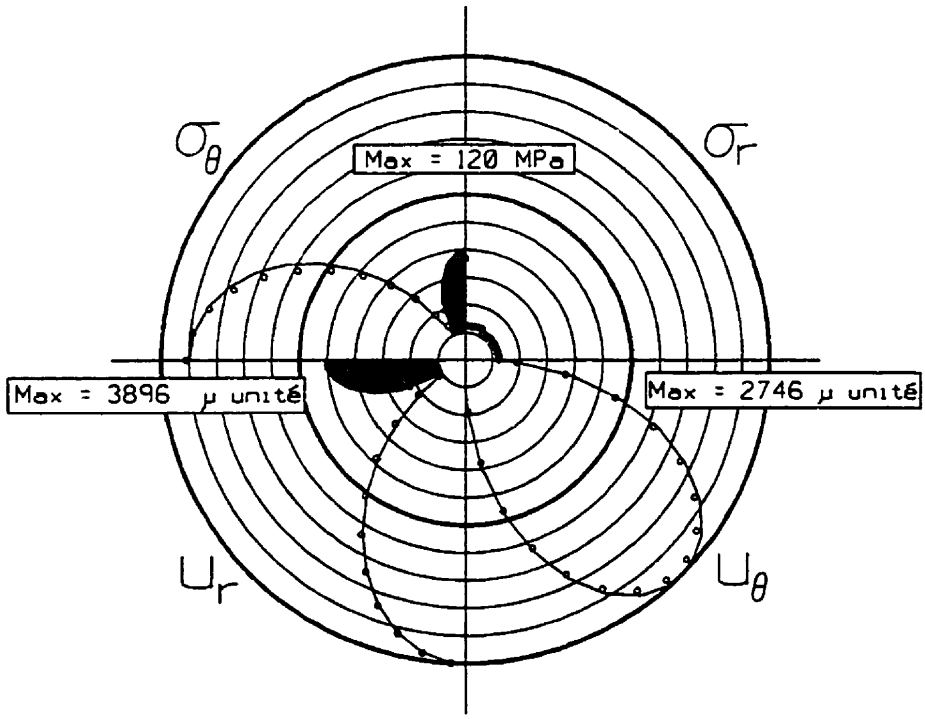


Figure 5.2 : Modélisation d'un matériau isotrope et linéaire

Comme nous pouvons le voir sur cette figure, les résultats obtenus avec le logiciel se rapprochent beaucoup des valeurs obtenues analytiquement. Comme mentionné plus haut,

les différences sont probablement dues au fait que le logiciel ne donne qu'une approximation des contraintes et des déformations.

5.4 ANALYSE D'UN MATÉRIAU ÉLASTIQUE LINÉAIRE ET ANISOTROPE

La prochaine étape de vérification est l'analyse d'un matériau élastique linéaire et transversalement isotrope. Pour ce faire, nous allons procéder de façon similaire à l'analyse précédente. Cette fois, nous allons utiliser pour de fins de comparaison l'approche analytique présentée par Leknitskii (1963). Cette approche analytique nous permet de déterminer la distribution de contraintes à l'intérieur d'une plaque anisotrope, comportant une cavité circulaire, chargée selon une direction seulement. Elle ne donne toutefois pas de façon de déterminer les déplacements autour de la cavité.

Soit une plaque (voir figure 5.1) comportant une anisotropie telle que le module de Young E_1 suivant l'axe x est plus faible que le module de Young, E_2 suivant l'axe y.

Nous allons donc supposer un chargement suivant l'axe x, un chargement suivant l'axe y, et un chargement appliqué suivant un angle $\phi = 45^0$ (mesuré à partir de x).

Le modèle de Leknitskii utilise les modules E_1 et E_2 et le coefficient de Poisson ν. On peut calculer G_2 à l'aide de Batugin et Nirenburg (1972). Leknitskii présente l'équation suivante

$$\frac{\mu^4}{E_1}+\left(\frac{1}{G_2}-\frac{2\nu}{E_1}\right)\mu^2+\frac{1}{E_2}=0. \tag{5.3}$$

À partir de cette équation on peut tirer les racines complexes μ_1 et μ_2. On peut calculer

$$n = -i\,(\mu_1+\mu_2). \tag{5.4}$$

Il est possible de calculer le module d'élasticité angulaire par

$$E_\theta=\frac{\sin^4\theta}{E_1}+\left(\frac{1}{G_2}-\frac{2\nu}{E_1}\right)\sin^2\theta\cos^2\theta+\frac{\cos^4\theta}{E_2} \tag{5.5}$$

où θ est l'angle de la direction suivant laquelle le module d'élasticité est calculé. À partir de ces paramètres, il est possible de calculer le niveau de contrainte tangentielle au pourtour de la cavité à l'aide de

$$\sigma_\theta = P\frac{E_\theta}{E_1}\begin{Bmatrix}\left[\cos^2\phi+(\mu_1\mu_2-n)\sin^2\phi\right]\mu_1\mu_2\cos^2\theta\\ +\left[(1+n)\cos^2\phi+\mu_1\mu_2\sin^2\phi\right]\sin^2\theta\\ -n(1+n-\mu_1\mu^2)\sin\phi\cos\phi\sin\theta\cos\theta\end{Bmatrix} \qquad (5.6)$$

où σ_θ est la contrainte tangentielle calculée au point faisant un angle θ par rapport à axe x et P est la contrainte appliquée à un angle ϕ de l'axe x.

Pour la vérification, nous allons utiliser les paramètres suivants: E_1 = 50 GPa, E_2 = 10 GPa, ν = 0.1 et G_2 = 8.065 GPa (établis par la relation de Batugin et Nirenburg (1972)). Les figures 5.3, 5.4 et 5.5 montrent le niveau de contrainte autour de la cavité circulaire. Sur ces figures, C indique compression et T indique tension.

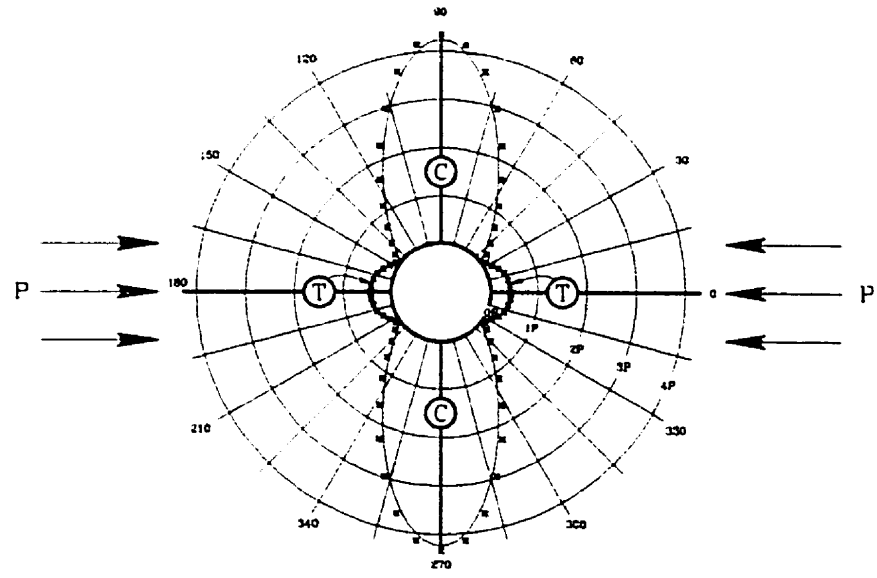


Figure 5.3: Matériau transversalement isotrope et linéaire sollicité perpendiculairement au plan d'isotropie

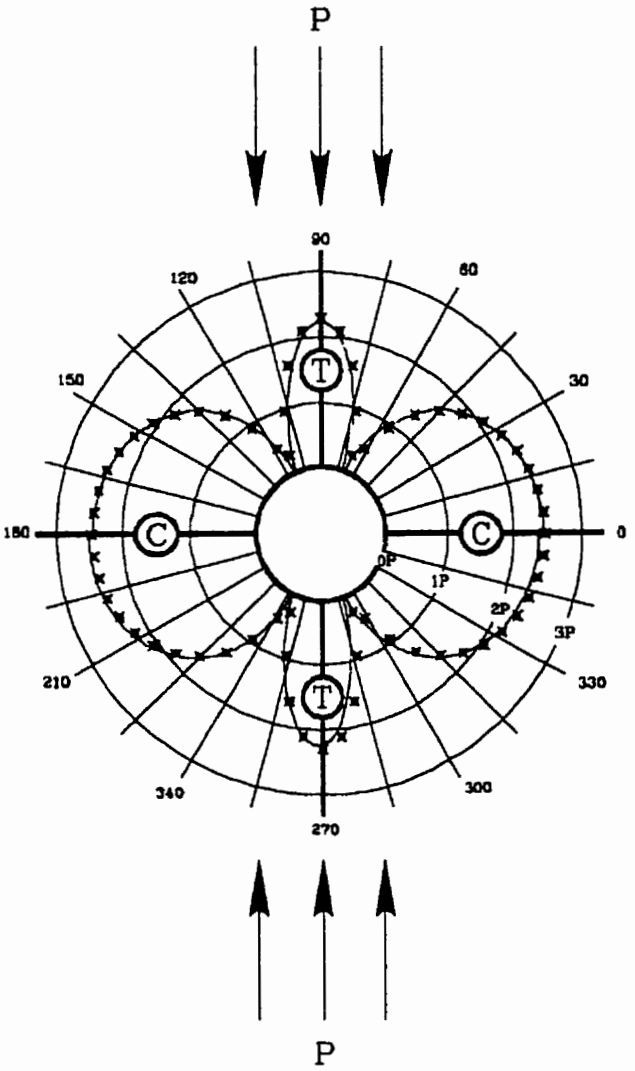


Figure 5.4: Matériau transversalement isotrope et linéaire sollicité parallèlement au plan d'isotropie

Sur ces figures le trait plein indique les valeurs calculées à l'aide des équations de Leknitskii (1963) telles que présentées plus haut et les points représentent les valeurs

générées à l'aide du logiciel d'éléments finis. La corrélation entre les deux résultats est excellente si on tient compte des limitations des éléments finis.

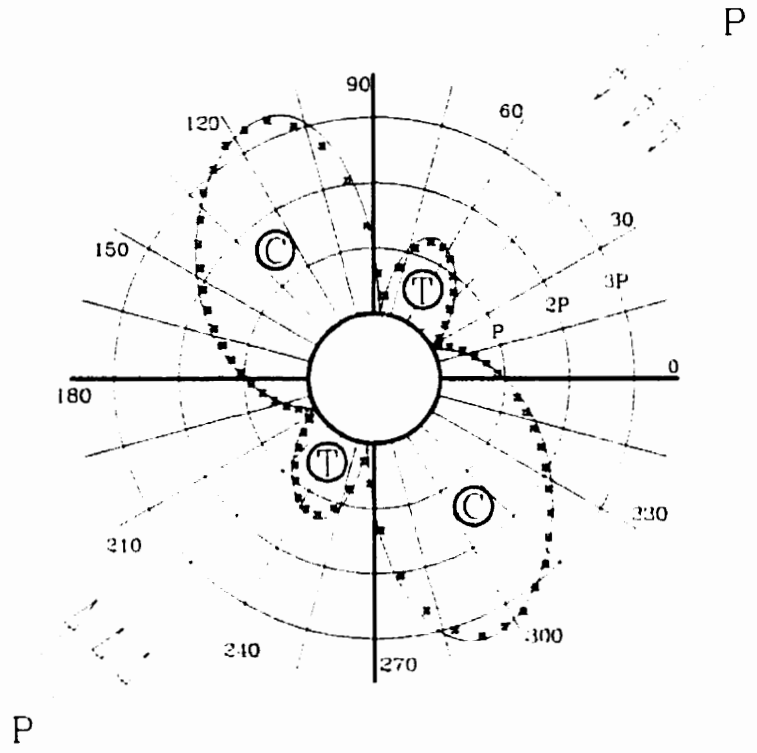


Figure 5.5: Matériau transversalement isotrope et linéaire sollicité à un angle de 45^0 à E_1

5.5 ANALYSE D'UN MATÉRIAU ÉLASTIQUE NON LINÉAIRE ET ISOTROPE

Pendant cette étape de vérification, nous allons démontrer que le modèle est valide pour modéliser un matériau possédant une loi de comportement isotrope et élastique non linéaire. Pour ce faire, nous allons utiliser le logiciel d'élément finis COSMOS/M. Ce

logiciel est reconnu dans l'industrie. Il permet de considérer la non linéarité du matériau en spécifiant le module de Young associé à un niveau de contrainte pour une compression uniaxiale.

Donc, en utilisant notre logiciel il nous est possible de simuler une compression uniaxiale à des niveaux de contrainte différents afin de générer la *courbe de rigidité* utilisé par COSMOS/M. Nous allons utiliser le modèle de la figure 5.1 en acceptant que les contraintes verticales et horizontales sont identiques, c'est-à-dire que le chargement est hydrostatique. Appelons ces contraintes P. Le paramètre mesuré suite à ces analyses est le rapport entre la contrainte tangentielle et la contrainte appliquée (σ_θ/P) en fonction du rapport de la distance radiale et du rayon du trou (a/r) . Les résultats obtenus sont présenté à la figure 5.6.

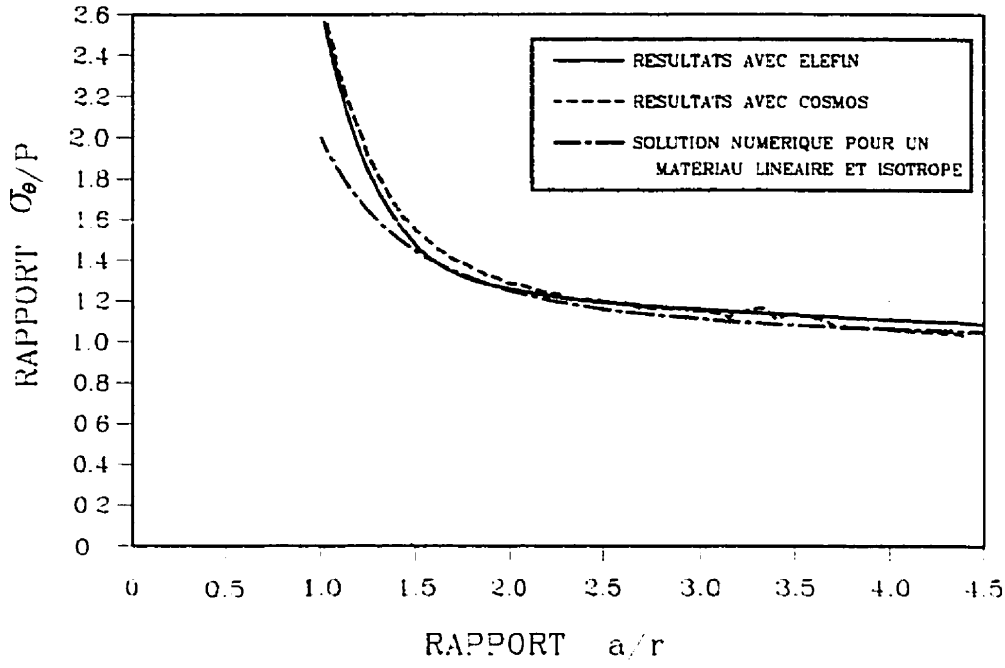


Figure 5.6: Modélisation d'un matériau isotrope et non linéaire

Comme on peut voir à la figure 5.6, les résultats obtenus par ELEFIN et COSMOS/M sont très semblables. Le niveau d'erreur peut être aux erreurs intrinsèques des éléments finis. Pour fin de comparaison, nous avons inclus la solution numérique pour un matériau linéaire et isotrope. On peut remarquer que le niveau de contrainte est significativement plus élevé au pourtour du trou pour ce qui est du matériau ayant un comportement non linéaire. Ceci s'explique par le fait que les contraintes plus élevées en périphérie du trou génèrent une zone plus rigide à cet endroit, ce qui a pour effet de créer une concentration de contrainte comme le ferait un revêtement rigide.

5.6 - ANALYSE D'UN MATÉRIAU ÉLASTIQUE NON LINÉAIRE ET TRANSVERSALEMENT ISOTROPE

La capacité de modéliser numériquement des matériaux montrant un comportement élastique non linéaire et transversalement isotrope est le but ultime de ce mémoire de maîtrise. Malheureusement pour nous, il n'existe pas de façon de vérifier nos résultats car, à notre connaissance, aucun autre modèle, que ce soit analytique ou numérique, ne permet de traiter ce type de comportement, d'où l'intérêt de développer un tel modèle. Il est toutefois intéressant de pouvoir observer les résultats des analyses avec ce matériau.

Nous croyons bon de mentionner ici qu'étant donné que le problème montre deux plans de symétrie nous avons décidé d'utiliser un quart de modèle afin de réduire le temps de calcul.

Pour illustrer les effets associés aux matériaux élastiques non linéaires et transversalement isotrope, nous allons supposer un matériau possédant la loi de comportement suivante:

$$\varepsilon_{m1} = 0.8178\,\sigma_m + 13.63\,(1-e^{-0.3\,\sigma_m})^{1.5}$$

$$e_{zz1} = \{1.9644\,s_{ijmax} + 32.74\,(1-e^{-0.3\,s_{ijmax}})^{1.5}\}\,s_{zz}/s_{ijmax}$$

$$\varepsilon_{m2} = 0.48\,s_{zz} + 4\,(1-e^{-0.6\,s_{zz}})^{1.5} \qquad (5.7)$$

$$e_{zz2} = 0.96\,\sigma_m + 16\,(1-e^{-0.3\,\sigma_m})^{1.5}$$

$$e_{xx1} - e_{yy1} = 0.6844(s_{xx}-s_{yy}) + 17.11111(1-e^{-0.2\,(s_{xx}-s_{yy})})^{1.5}$$

Ces équations sont issues d'essais réalisés sur du granite du lac DuBonnet au laboratoire de mécanique des roches de l'École Polytechnique.

À la figure 5.7 nous pouvons voir les modules de Young (E_1 et E_2) générés par le modèle lorsque soumis à une compression biaxiale isotrope. L'essai de compression biaxiale isotrope est décrite au paragraphe 3.2 et à la figure 3.1 Tel que nous pouvons le constater le matériau est bien anisotrope et non linéaire. Nous voyons aussi que le rapport entre les modules pour ce cas est de 3.5 et demeure constant tout au long de l'essai de compression biaxiale isotrope.

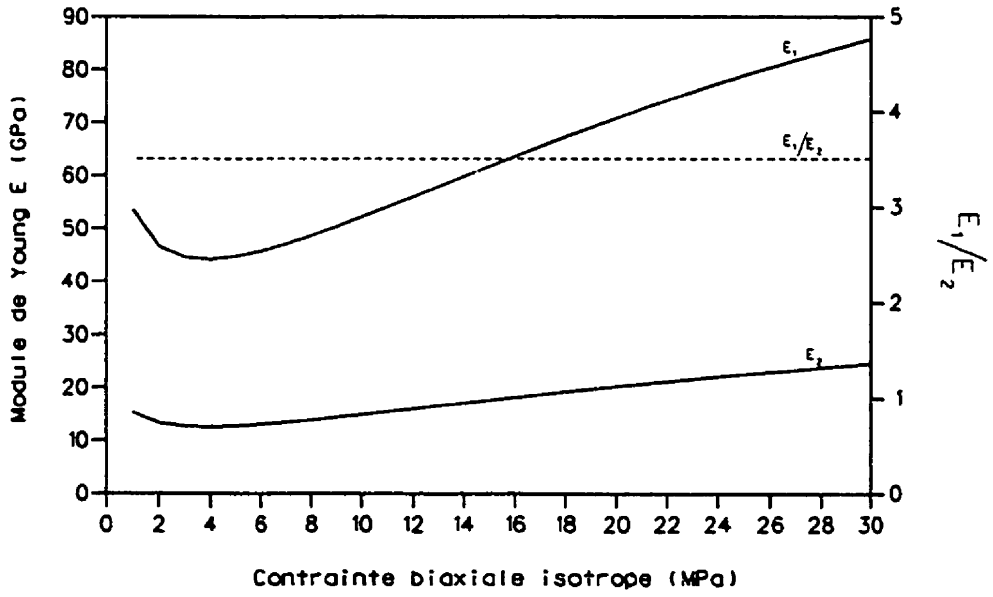


Figure 5.7 - Modules d'élasticité mesurés lors d'une compression biaxiale isotrope.

En utilisant ces lois de comportement, nous avons fait deux simulations, soit la première en appliquant une contrainte P parallèlement à E_1 et la seconde en appliquant P perpendiculairement à E_1. Les figures 5.8 et 5.9 nous montrent les résultats.

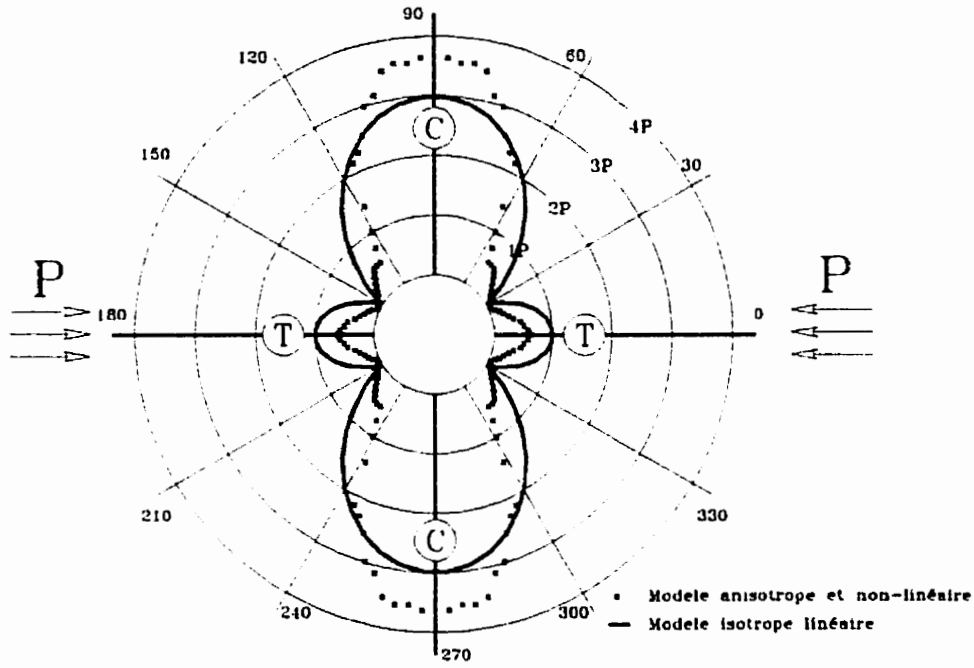


Figure 5.8 Niveau de contraintes tangentielles pour une compression parallèle au plan de symétrie

À la figure 5.8, nous pouvons voir que le niveau de contrainte en compression maximale est supérieur d'environ 20% au niveau de contrainte avec le modèle isotrope linéaire. Nous pouvons donc conclure que dans un cas où la contrainte est dans l'axe de E_1, c'est-à-dire au plan d'isotropie, une analyse utilisant un modèle élastique linéaire et isotrope génère une marge d'erreur de 20% pour les données de la simulation

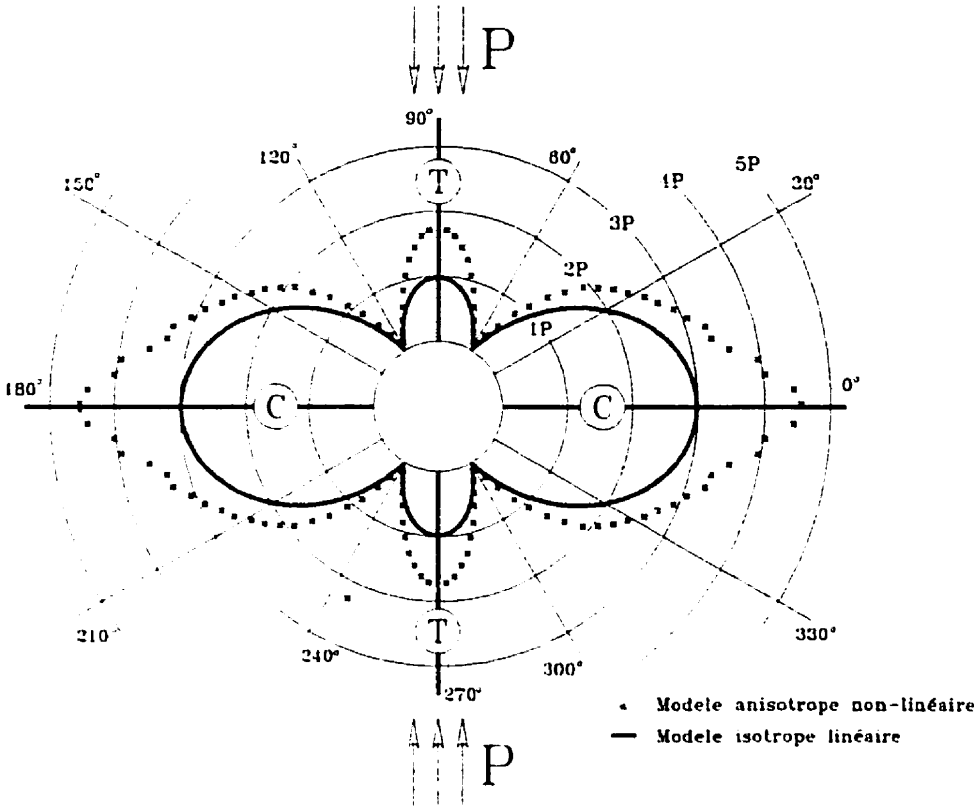


Figure 5.9 : Niveau de contraintes tangentielles pour une compression perpendiculaire au plan de symétrie

La figure 5.9 pour sa part montre un niveau de contrainte sensiblement plus élevé pour le modèle transversalement isotrope non linéaire que le modèle isotrope linéaire. Nous pouvons donc conclure ici que l'utilisation d'un modèle élastique linéaire et isotrope générait ici un erreur de 60 %, autant en tension qu'en compression, pour les données de la simulation. Dans le cas d'une anayse de stabilité, le modèle proposé nous place du côté plus sécuritaire, toujours pour les données de cette simulation.

Nous pouvons tenter d'expliquer ceci en considérant que les effets de non linéarité sont une fonction de la déformation d'un matériau. En conséquent, lorsque un matériau est de plus en plus déformé, ses caractéristiques non linéaires deviennent plus importantes. Pour le premier cas que nous avons analysé (Fig. 5.8), le matériau est sollicité selon son axe le plus rigide, donc les effets de la non linéarité sont moins prononcés que le second cas où la sollicitation est selon l'axe le plus déformable, et, si on regarde la figure 5.7, ce rapport est d'environ de 3.5. Par conséquent la déformation est beaucoup plus importante, et les effets de la non linéarité plus prononcés.

CHAPITRE VI

DISCUSSIONS

Lors de la rédaction de ce document, il a été jugé préférable de présenter des points particuliers de discussion dans les sections pertinentes. Les autres point de discussion font l'objet du présent chapitre.

Le plus gros problème auquel nous avons eu à faire face est la capacité limitée du modèle de comportement. Pour certaines conditions de sollicitation le modèle de Corthésy et al. (1993) donne des résultats quelque peu étranges. Voici ce que nous voulons dire par cet énoncé. Supposons un état de contrainte tel que $\sigma_z = \sigma_x / 10$, et un coefficient de Poisson $\nu = 0.1$. Nous savons que le module de Young E_2 est donné par

$$E_2 = (\sigma_z - \nu\sigma_x) / \varepsilon_z = 0/\varepsilon_z . \qquad (6.1)$$

Ceci indique que le module E_2 est nul. Nous savons que ceci n'est pas le cas. Nous savons toutefois que si le matériau se comporte de façon linéaire, nous allons obtenir $\varepsilon_z = 0$, ce qui va nous donner une forme indéterminée, mais ceci n'est pas le cas pour les matériaux non linéaires.

Pour le cas des matériaux non linéaires, une des hypothèses de base du modèle est l'existence d'un seul coefficient de Poisson et celui-ci invariable pour l'essai de compression biaxiale isotrope. (fig. 3.1). Or, lors de l'étude en compression uniaxiale nous avons déterminé que le coefficient de Poisson peut être variable selon le niveau de contrainte et également différent selon l'orientation de l'axe étudié. Si on se réfère à la figure 3.2, nous pouvons voir que les coefficients de Poisson sont différents selon des axes. Ceci révèle que les lois de comportements possèdent intrinsèquement des coefficients de Poisson. Il est possible de déterminer ces coefficients de Poisson lorsque il s'agit d'une compression uniaxiale, mais comment déterminer ces coefficients lorsque l'état de sollicitation est quelconque ? Nous avons pensé utiliser les fonctions K et G_{ij} pour déterminer les coefficients de Poisson et module de Young, mais encore là, lorsque le déviateur de contrainte est nul, Gij est également nul, et cause encore ici des problèmes.

Il en résulte qu'il n'y a pas de façon élégante pour déduire les modules de Young avec la loi de comportement simplifiée que nous avons utilisée. Nous avons déjà indiqué au chapitre II que l'utilisation d'une loi de comportement rigoureuse se ferait au prix de la nécessité de déterminer un grand nombre de paramètres à partir de plusieurs essais différents en laboratoire.

Pour ce qui est de la nécessité d'utiliser un programme utilisant les effets de la non linéarité et de l'anisotropie, le tableau suivant indique les variations de contrainte calculées

au pourtour de l'orifice pour le même état de contrainte, mais pour différents types de traitement.

Tableau 6.1 : Compilations des simulations et des résultats

Sollicitation	Matériau isotrope et linéaire	Matériau anisotrope et linéaire	Matériau isotrope et non linéaire	Matériau transversalement isotrope et non linéaire
$\sigma_x = \sigma_z = P$	$\sigma_{max} = 2\ P$ $\sigma_{min} = 2\ P$	non évaluées	$\sigma_{max} = 2.6\ P$ $\sigma_{min} = 2.6\ P$	non évaluées
$\sigma_x = P$ $\sigma_z = 0$	$\sigma_{max} = 3\ P$ $\sigma_{min} = -1\ P$	$\sigma_{max} = 4.3\ P$ $\sigma_{min} = -0.45\ P$	non évaluées	$\sigma_{max} = 3.7\ P$ $\sigma_{min} = -0.6\ P$
$\sigma_x = 0$ $\sigma_z = P$	$\sigma_{max} = 3\ P$ $\sigma_{min} = -1\ P$	$\sigma_{max} = 2.5\ P$ $\sigma_{min} = -2.3\ P$	non évaluées	$\sigma_{max} = 4.6\ P$ $\sigma_{min} = -1.7\ P$

Dans ce tableau nous pouvons remarquer que la variation du niveau de contrainte peut être très importante, pouvant aller jusqu'à 130 %. Nous croyons que ce tableau justifie de traiter les matériaux selon leur véritable loi de comportement.

Nous croyons toutefois important de mentionner que la nécessité d'utiliser un modèle approprié augmente avec le degré de non linéarité ou l'anisotropie.

CHAPITRE VII

CONCLUSIONS ET RECOMMANDATIONS

Dans le présent document, nous avons tenté de démontrer la validité de notre recherche. Nous croyons avoir atteint ce but.

Nous avons mis au point un logiciel d'éléments finis qui permet de procéder à des analyses de contraintes pour les lois de comportement élastique des types suivants:

i) linéaire et isotrope;

ii) linéaire et transversalement isotrope

iii) non linéaire et isotrope;

iv) non linéaire et transversalement isotrope.

La non linéarité est prise en ligne de compte par une approche itérative considérant que chaque élément du modèle simulé possède des paramètres de déformabilité qui lui sont propres.

Pour les cas de non linéarité et d'isotropie, nous avons réussi à définir les relations contrainte-déformation en supposant qu'il n'y avait pas d'effet de deuxième ordre. L'objectif de cette simplification découle du fait que nous voulons obtenir une loi de

comportement dont les paramètres puissent être déterminés à partir de deux essais simples à réaliser en laboratoire.

Les fonctions des paramètres de déformabilité trouvées comme étant les plus représentatives du comportement des roches montrant une élasticité non linéaire sont de la forme:

$$\varepsilon = m\sigma + b\{1-\exp(-a\sigma)\}^c$$

où ε est le paramètre de déformabilité, σ est la contrainte appliquée, a,b,c et m étant des paramètres.

Dans le but de valider le logiciel d'analyses de contraintes, le cas d'une plaque mince trouée a été modélisé et analysé, les solutions ayant été comparées à des solutions existantes. Ainsi, dans le cas d'un matériau élastique linéaire et isotrope, les résultats ont été comparés à ceux produits par le modèle analytique de Kirsch (1898) pour ce qui à trait au champ de contraintes et au champ de déplacement. Dans le cas d'une plaque formée d'un matériau élastique linéaire et transversalement isotrope, les champs de contraintes obtenus ont été comparés à ceux générés à l'aide des équations proposées par Leknitskii (1963). Les champs de contraintes dans la plaque obtenus avec le logiciel développé dans le présent mémoire ont été comparés à ceux produits avec le logiciel COSMOS/M pour le cas d'un matériau élastique non linéaire et isotrope. Ces trois comparaisons montrent que

le logiciel mis au point par l'auteur de ce document conduit à d'excellents résultats. Il en conclut que le logiciel est validé pour les cas analysés.

Le logiciel proposé et validé a permis d'analyser des contraintes dans le même cas de plaque trouée que précédemment, mais avec un matériau de la plaque élastique non linéaire et transversalement isotrope. Puisqu'il n'existe pas d'autre modèles analytiques ou numériques permettant de traiter un tel cas, il a été impossible de procéder à une comparaison. Néanmoins, en comparant les résultats à ceux des trois autres cas analysés, les effets combinés de l'élasticité non linéaire et de l'isotropie transversale peuvent y être observés et appréciés.

L'ensemble des analyses effectuées dans ce mémoire démontre que l'approche itérative mise à profit par le logiciel développé est valable pour traiter le cas des matériaux élastiques non linéaire et transversalement isotrope.

Nous complétons ce chapitre en faisant les recommandations suivantes, dans l'optique de recherches futures:

- Développement d'un modèle de comportement de matériau anisotrope et non linéaire plus stable que le modèle de Corthésy.
- Reprogrammation du code éléments finis en un langage plus performant que le Quick Basic

- Analyses de contraintes et déformations sur des matériaux anisotropes et non linéaires afin de comparer les résultats expérimentaux à ceux obtenu par le modèle d'éléments finis afin de vérifier si la loi de comportement proposée représente adéquatement le comportement des roches transversalement isotropes et non linéaires malgré les hypothèses simplificatrices auxquelles elle est associée.

RÉFÉRENCES

BATUGIN, S.A.; NIRENBURG, R.K, (1972). Approximate relation between the elastic constants of anisotropic rocks and the anisotropy parameters. Soviet Mining Science, Vol.8, No.1, pp. 5-8

BRADY, B.H.G.; BROWN, E.T. (1993). Rock Mechanics for underground mining. Chapman & Hall, London. 571 p.

BROWN, E.T.; BRAY, J.W.; SANTARELLI, F.J. (1989). Influence of stress-dependent elastic moduli on stress and strains around axisymetric boreholes. Rock mechanics and rock engineering. Vol.22, pp. 184-203

CORTHÉSY, R. (1982). Dévelopment d'une méthode analytique d'interprétation des mesures de pressions de terrains par la technique du "doorstopper" pour les roches anisotropes et non-linéaires. Mémoire de maîtrise, Départment de génie minéral, École Polytechnique de Montréal, 66 p.

CORTHÉSY, R.; GILL, D.E.; LEITE, M.H. (1993). An integrated-approach to rock stress measurement in anisotropic non linear elastic rock. Int. Journal of Rock Mechanics mineral science and geomechanics, Abstr., Vol. 30, No. 4, pp. 395-411.

DESAI, C. S., SIRIWARDANE, H. J. (1984), Constitutive laws for engineering materials, with emphasis on geologic materials. Prentice-Hall, Inc. Englewood Cliffs, New Jersey, 07632.

ERINGEN, J. C. (1962). Nonlinear Theory of Continuous Media. McGraw-Hill Book Company, New York.

Kirsch G., (1898), Die theorie der elastizitat und die bedurfnisse der festigkeislehre. Veit. Deit. Ing. 42 (28), 797-807

KULHAWY, F.H. (1975). Stress deformation properties of rock and rock discontinuities. Engineering Geology, Vol.9, pp. 327-350.

LADE, P.V.; NELSON, R.B. (1987). Modeling the elastic behavior of granular materials. International Journal of Numerical and Analytical Methods in Geomechanics, Vol. 11, No.5, pp. 521-542.

LEEMAN, E.R.; DENKHAUS, H.G. (1969). Determination of stress in rock with linear or non-linear elastic characteristics Rock Mechanics. Vol. 1, pp. 198-206.

LEKHNITSKII, S.G. (1963). Theory of elasticity of an anisotropic elastic body. Holden-Day Series in Mathematics Physics, ed. J.J.Brandstatter, Holden Day Inc., San-Francisco, 404 p.

NUNES, A.L. (1997). Nouvelle méthode de détermination de la déformabilité des roches transversalement isotropes avec la cellule triaxiale CSIR. Thèse de doctorat. Départment de génie minéral, École Polytechnique de Montréal, 355 p.

SANTARELLI, F.J.; BROWN, E.T. (1987). Performance of deep wellbores in rock with a confining pressure-dependant elastic modulus. Proc. 6th Int. Congr. ISRM, Montreal, pp 1217-1222.

SANTARELLI, F.J.; BROWN, E.T.; MAURY, V. (1996). Analysis of borehole stresses using pressure-dependant, linear elasticity. Int. Journal of Rock Mechanics mineral science and geomechanics, Abstr., Vol.23, No 6, pp 445-449.

SHOA, J.F.; KONDO, D.; IKOGOU, S. (1994). Stress- induced microcracking in rock and its influence on wellbore stability analysis. Int. Journal of Rock Mechanics mineral science and geomechanics, Abstr., Vol. 31, No 2, pp. 149-155.

VOIGHT, B. (1968). Determination of the virgin state of stress in the vicinity of a borehole from measurements of a partial anelastic strain tensor in drill holes. Felsmech. Inginieurgeologie, Vol. 6, pp 201-215.

WANG, Y; KESSLER, N.; SANTARELLI, J.F. (1993). Stress-induced anisotropy and its effects on borehole responses. Proc. 34th U.S. Symp. on Rock Mech., Pre-print, pp. 519-522.

WANG, Y; DUSSEAULT, M.B. (1991). Hydraulic fracture stress measurement in rocks with stress-dependent Young's moduli. Proc. 32th U.S. Symp. on Rock Mech., Norman, pp 3-14.

WITTKE, W. (1991). Rock mechanics - Theory and applications with case histories. Springer Verlag, Berlin.

WU, B.; HUDSON, J.A. (1991). Stress-induced anisotropy in rock and its influence on wellbore stability. Proc. 32th U.S. Symp. on Rock Mech., Norman, pp 941-950.

ZIENKIEVICZ, O.C. (1971). The finite element method in Engineering science. McGraw Hill, Maidenhead, Berkshire, England, 521 p.

ANNEXE I

Listing du programme

```
DECLARE SUB MAINTRANSfer ()
DECLARE SUB dxfout ()
DECLARE SUB MAIN ()
DECLARE SUB MODDON ()
DECLARE SUB AFFDON ()
DECLARE SUB CREDON ()
DECLARE SUB EXECUTE ()
DECLARE SUB MENPRINC (CHOIX!)
DECLARE SUB DIMVAR ()
DECLARE SUB ATTENTE ()
DECLARE SUB present ()
DECLARE SUB AJPDPROPFICH ()
DECLARE SUB IMPPDPROP ()
DECLARE SUB AFFPDPROP ()
DECLARE SUB CALCPDPROP ()
DECLARE SUB DESSGRIL ()
DECLARE SUB MODETEXTE ()
DECLARE SUB MODEGRAPH ()
DECLARE SUB MODELE ()
DECLARE SUB MODPTND ()
DECLARE SUB AFFFICHRES ()
DECLARE SUB AFFNOMTR ()
DECLARE SUB ENTNOMTRAV (TITRE$)
DECLARE SUB IMPRESULTAT ()
DECLARE SUB IMPDONNEES ()
DECLARE SUB MODCONDCHAR ()
DECLARE SUB MODCONDFRT ()
DECLARE SUB MODMAT ()
DECLARE SUB AFFCONDCHAR ()
DECLARE SUB AFFCONDFRT ()
DECLARE SUB AFFELEMENTS ()
DECLARE SUB AFFCORPTND ()
DECLARE SUB AFFCARMAT ()
DECLARE SUB AFFLIMITES ()
DECLARE SUB accfichier ()
DECLARE SUB crefichier ()
DECLARE SUB ENTDONNEES ()
DECLARE SUB ENTLIM ()
DECLARE SUB AFFLIM ()
DECLARE SUB CONFIGURATION ()
DECLARE SUB CONTRAINTE ()
'$DYNAMIC
OPTION BASE 1
REM  *****************************************************************
REM  *              Programme ACCES                  *
REM  *                                               *
REM  *              ELEFIN V. 1.1                    *
REM  *****************************************************************
COMMON nom$, lecteur$, imprimante$
'COMMON SHARED NP, NE, NB, NLD, NDF, NMAT, NOM$, LECTEUR$, IMPRIMANTE$, TITRE$
  Np = 10
  ne = 10
  NB = 10
  nld = 10
```

```
  NDF = 10
  NMAT = 10
  DIM SHARED ORT(NMAT, 5), CORD(Np, 2), nop(ne, 3), IMAT(ne), NBC(NB)
  DIM SHARED nfix(NB), nq(nld), R(nld, 2), POID(Np), poiun(NMAT)

REM ***************************************************************
REM Configuration
REM ***************************************************************

CALL CONFIGURATION

  DEF FNNT (X) = INT(X * 10 ^ 5) / 10 ^ 5
DO

CALL MENPRINC(CHOIX)

  IF CHOIX = 1 THEN CALL CREDON        ' Creer un fichier de donnees
  IF CHOIX = 2 THEN CALL DESSGRIL      ' Dessiner la grille
  IF CHOIX = 3 THEN CALL CALCPDPROP    ' Poids propre
  IF CHOIX = 4 THEN CALL MODDON        ' Modification du fichier de donnees
  IF CHOIX = 5 THEN CALL AFFDON        ' Afficher un fichier de donnees
  IF CHOIX = 6 THEN CALL IMPDONNEES    ' Imprimer un fichier de donnees
  IF CHOIX = 7 THEN CALL MAIN          ' Executer une analyse de contraintes
  IF CHOIX = 8 THEN CALL AFFFICHRES    ' Afficher un fichier de resultats
  IF CHOIX = 9 THEN CALL IMPRESULTAT   ' Imprimer un fichier de resultat
  IF CHOIX = 10 THEN CALL dxfout       ' DXFOUT AVEC FICHIER DE SORTIE
  IF CHOIX = 11 THEN CALL MAINTRANSfer ' TRANSFER UN FICHIER DE .NF A .ENT
LOOP UNTIL CHOIX = 12
REM OPTION 10: Quitter
  END        'Fin du programme

REM $STATIC
SUB accfichier
'***************************************************************
'  Acces a un fichier de donnees
'***************************************************************
SHARED Np, ne, NB, nld, NDF, NMAT, nom$, lecteur$, TITRE$

  CLS
  LOCATE 10, 10: PRINT "Inserer la disquette de donnees dans "; lecteur$
  NOMPTMP$ = nom$
  PRINT TAB(10); "Quelle est le nom du fichier <"; NOMPTMP$; ">";
  INPUT nom$
  IF nom$ = "" THEN nom$ = NOMPTMP$
  OPEN "i", #1, lecteur$ + nom$ + ".ent"
  INPUT #1, Np, ne, NB
  INPUT #1, nld, NDF, NMAT
  CALL DIMVAR
  FOR N = 1 TO NMAT
    INPUT #1, ORT(N, 1), ORT(N, 2), ORT(N, 3), ORT(N, 4), ORT(N, 5), poiun(N)
  NEXT N
  FOR N = 1 TO Np
    INPUT #1, CORD(N, 1), CORD(N, 2)
  NEXT N
```

```
  FOR N = 1 TO ne
    INPUT #1, nop(N, 1), nop(N, 2), nop(N, 3), IMAT(N)
  NEXT N
  FOR N = 1 TO NB
    INPUT #1, NBC(N), nfix(N)
  NEXT N
  FOR N = 1 TO nld
    INPUT #1, nq(N), R(N, 1), R(N, 2)
  NEXT N
  INPUT #1, TITRE$
  CLOSE #1

END SUB

SUB AFFCARMAT
REM ***************************************************************
REM Affichage des caracteristiques des materiaux
REM ***************************************************************
SHARED NMAT
  CLS
  PRINT "Caracteristiques des materiaux"
  PRINT "Materiau   Direction 1     Direction 2     Rigidité,       Poids/volume"
  PRINT "          E(1)   v(1)    E(2)    v(2)       G2  "
  FOR N = 1 TO NMAT
    PRINT TAB(2); N; TAB(12); ORT(N, 1); TAB(20); ORT(N, 2); TAB(28); ORT(N, 3); TAB(38); ORT(N,
          4); TAB(49); ORT(N, 5); TAB(67); poiun(N)
    IF (N / 20 <> INT(N / 20)) AND N <> NMAT THEN GOTO 2800
    CALL ATTENTE
    CLS
    IF N / 20 = INT(N / 20) THEN PRINT "Materiaux     Module d'Young          Coefficient de Poisson
          poids/volume"
2800
  NEXT N
END SUB

SUB AFFCONDCHAR
REM ***************************************************************
REM Affichage des conditions de charges
REM ***************************************************************
SHARED nld

  CLS
  PRINT "Conditions de charges"
  PRINT "  Pt de charge         Charge X      Charge Y   "
  FOR N = 1 TO nld
    PRINT N; ")"; TAB(6); nq(N); TAB(30); R(N, 1); TAB(42); R(N, 2)
    IF (N / 20 <> INT(N / 20)) AND N <> nld THEN GOTO 3200
    CALL ATTENTE: CLS
    IF N / 20 = INT(N / 20) THEN PRINT "  Pt de charge         Charge X      Charge Y   "
3200
  NEXT N
END SUB

SUB AFFCONDFRT
```

```
REM ************************************************************
REM Affichage des conditions frontieres
REM ************************************************************
 SHARED NB
  CLS
  PRINT "Conditions frontieres"
  PRINT "    Pt. frontiere          Condition "
  FOR N = 1 TO NB
    PRINT N; ")"; TAB(6); NBC(N); TAB(27); nfix(N)
    IF (N / 20 <> INT(N / 20)) AND N <> NB THEN GOTO 3100
    CALL ATTENTE: CLS
    IF N / 20 = INT(N / 20) THEN PRINT "    Pt. frontiere          Condition "
3100
  NEXT N
END SUB

SUB AFFCORPTND
REM ************************************************************
REM Affichage des coordonnees des points nodaux
REM ************************************************************
SHARED Np
  CLS
  PRINT "Coordonnees des points nodaux"
  PRINT "Point nodal      X              Y   "
  FOR N = 1 TO Np
    PRINT TAB(2); N; TAB(18); CORD(N, 1); TAB(36); CORD(N, 2)
    IF (N / 20 <> INT(N / 20)) AND N <> Np THEN GOTO 2900
    CALL ATTENTE: CLS
    IF N / 20 = INT(N / 20) AND N <> Np THEN PRINT "Point nodal      X              Y   "
2900
  NEXT N
  PRINT
END SUB

SUB AFFDON
 SHARED Np, ne, NB, nld, NDF, NMAT, nom$, lecteur$, imprimante$, TITRE$
  CALL accfichier              'Acces a un fichier de donnees
  CALL AFFNOMTR: CALL ATTENTE     'Afficher le nom du travail
  CALL AFFLIM                  'Affichage des limites (texte)
  CALL AFFLIMITES: CALL ATTENTE   'Affichage des limite
  CALL AFFCARMAT               'Affichage des caracteristiques des materiaux
  CALL AFFCORPTND               'Affichage des coordonnees des points nodaux
  CALL AFFELEMENTS              'Affichage des elements
  CALL AFFCONDFRT              'Affichage des conditions aux frontiere
  CALL AFFCONDCHAR              'Affichage des conditions de charge
END SUB

SUB AFFELEMENTS
REM ************************************************************
REM Affichage des elements
REM ************************************************************
SHARED ne
  CLS
  PRINT "Elements"
```

```
  PRINT "Element     i          j          m          Materiau "
  FOR N = 1 TO ne
    PRINT TAB(2); N; TAB(14); nop(N, 1); TAB(29); nop(N, 2); TAB(47); nop(N, 3); TAB(70); IMAT(N)
    IF (N / 20 <> INT(N / 20)) AND N <> ne THEN GOTO 3000
    CALL ATTENTE: CLS
    IF N / 20 = INT(N / 20) AND N <> ne THEN PRINT "Element     i          j          m
                  Materiau "
3000
  NEXT N
  PRINT
END SUB

SUB AFFFICHRES
REM ***************************************************************
REM Affichage d'un fichier de resultat
REM ***************************************************************
  SHARED Np, ne, NB, nld, NDF, NMAT, nom$, lecteur$, imprimante$, TITRE$
  CALL accfichier
  CALL AFFNOMTR: CALL ATTENTE  'Affichage du nom du travail
  CLS
  OPEN "i", #2, lecteur$ + nom$ + ".sor"
  PRINT "Deplacements"
  PRINT "Numero du noeud    Composante X    Composante Y"
  FOR N = 1 TO Np
    INPUT #2, disx, disy
    disx = FNNT(disx)
    disy = FNNT(disy)
    PRINT TAB(3); N; TAB(21);
    PRINT USING "###.#####"; disx; : PRINT TAB(39);
    PRINT USING "###.#####"; disy
    IF (N / 20 <> INT(N / 20)) AND N <> Np THEN GOTO 5250
    CALL ATTENTE: CLS
    IF N / 20 = INT(N / 20) AND N <> Np THEN PRINT "Numero du noeud    Composante X
                  Composante Y"
5250
  NEXT N
  CLS
  PRINT "Contraintes suivant X et Y et contraintes principales"
  PRINT "Element   SigmaX     SigmaY     TAUXY     Sigma1    Sigma2   Angle"
  FOR N = 1 TO ne
    INPUT #2, XCENT, YCENT, sigmax, sigmay, tauxy, SIGMA1, SIGMA2, angle
    sigmax = FNNT(sigmax)
    sigmay = FNNT(sigmay)
    tauxy = FNNT(tauxy)
    SIGMA1 = FNNT(SIGMA1)
    SIGMA2 = FNNT(SIGMA2)
    angle = FNNT(angle)
    PRINT TAB(3); N; TAB(8);
    PRINT USING "######.###"; sigmax; : PRINT TAB(20);
    PRINT USING "######.###"; sigmay; : PRINT TAB(32);
    PRINT USING "######.###"; tauxy; : PRINT TAB(44);
    PRINT USING "######.###"; SIGMA1; : PRINT TAB(56);
    PRINT USING "######.###"; SIGMA2; : PRINT TAB(68);
    PRINT USING "###.##"; angle
```

```
    IF (N / 20 <> INT(N / 20)) AND N <> ne THEN GOTO 5410
    CALL ATTENTE: CLS
    IF N / 20 = INT(N / 20) AND N <> ne THEN PRINT "Element   SigmaX     SigmaY     TAUXY
                  Sigma1    Sigma2   Angle"
5410
  NEXT N
  CLOSE #2

END SUB

SUB AFFLIM
REM
REM Affichage des limites (texte)
REM
  CLS : PRINT
  PRINT "Nombre de points nodaux"
  PRINT "Nombre d'elements"
  PRINT "Nombre de points avec restriction de deplacement"
  PRINT "Nombre de points avec chargements"
  PRINT "Nombre de degres de liberte"
  PRINT "Nombre de materiaux differents"
END SUB

SUB AFFLIMITES
REM
REM Affichage des limites
REM
  SHARED Np, ne, NB, nld, NDF, NMAT
  LOCATE 2, 50: PRINT Np
  LOCATE 3, 50: PRINT ne
  LOCATE 4, 50: PRINT NB
  LOCATE 5, 50: PRINT nld
  LOCATE 6, 50: PRINT NDF
  LOCATE 7, 50: PRINT NMAT
END SUB

SUB AFFNOMTR
REM ***************************************************************
REM Affichage du nom du travail
REM ***************************************************************
SHARED TITRE$
  CLS
  LOCATE 10, 10
  PRINT "Le nom du travail est"
  PRINT TAB(10); TITRE$
END SUB

SUB AJPDPROPFICH
REM ***************************************************************
REM Ajouter les poids propres au fichier de donnees
REM ***************************************************************
 SHARED Np, NB, nld

  IF nld < Np - NB THEN PRINT "ERREUR": RETURN
```

```
  FOR N = 1 TO nld
    FOR NN = 1 TO NB
      IF NBC(NN) = nq(N) AND nfix(NN) <> 10 THEN GOTO 6890
    NEXT NN
    R(N, 2) = R(N, 2) + POID(nq(N))
6890
  NEXT N
END SUB

SUB ATTENTE
  LOCATE 23, 1
  PRINT "Appuyer sur une touche"
  WHILE INKEY$ = ""
  WEND
END SUB

SUB CALCPDPROP
REM ************************************************************
REM Calcul du poids propre
REM ************************************************************
  SHARED Np, ne, NB, nld, NDF, NMAT, nom$, lecteur$, TITRE$
  CALL accfichier  'Acces a un fichier de donnees
  FOR X = 1 TO ne
    i = nop(X, 1): j = nop(X, 2): K = nop(X, 3)
    AJ = CORD(j, 1) - CORD(i, 1)
    AK = CORD(K, 1) - CORD(i, 1)
    BJ = CORD(j, 2) - CORD(i, 2)
    BK = CORD(K, 2) - CORD(i, 2)
    AIRE = (AJ * BK - AK * BJ) / 2
    poids = (ABS(AIRE) * poiun(IMAT(X))) / 3
    POID(i) = POID(i) + poids
    POID(j) = POID(j) + poids
    POID(K) = POID(K) + poids
  NEXT X
DO
  CLS : LOCATE 5, 10
  PRINT "----------------------------------------"
  PRINT TAB(10); "1. Affichage des poids propres"
  PRINT TAB(10); "2. Impression des poids propres"
  PRINT TAB(10); "3. Ajout des poids propres au fichier de donnees"
  PRINT TAB(10); "4. Retour au menu principal"
  PRINT TAB(10); "----------------------------------------"
1020
  LOCATE 11, 12: INPUT "Choix"; CHOI3
  IF CHOI3 > 4 OR CHOI3 < 1 OR CHOI3 <> INT(CHOI3) THEN BEEP: BEEP: GOTO 1020

  IF CHOI3 = 1 THEN        'Affichage des poids propres
    CLS
    PRINT "points nodaux              poids"
    FOR X = 1 TO Np
      PRINT X; TAB(30); POID(X)
      IF X / 20 = INT(X / 20) OR X = Np THEN PRINT : CALL ATTENTE
    NEXT X
  END IF
```

```
  IF CHOI3 = 2 THEN         ' Impression des poids propres
    FOR X = 1 TO NMAT
      LPRINT "poids propre du materiaux "; X; " : "; poiun(IMAT(X))
    NEXT X
    LPRINT "points nodaux                poids"
    FOR X = 1 TO Np
      LPRINT X; TAB(30); POID(X)
    NEXT X
  END IF

  IF CHOI3 = 3 THEN
    CALL AJPDPROPFICH
    CALL crefichier
  END IF
LOOP UNTIL CHOI3 = 4
END SUB

SUB CONFIGURATION
SHARED lecteur$, imprimante$
lecteur$ = "c:\ELEFIN\"
  imprimante$ = "parrallele"

 CALL present    'Page de presentation

25
  CLS
  LOCATE 4, 10: PRINT "Configuration"
    PRINT TAB(10); "-----------------------------------------"
    PRINT TAB(10); "La disquette de donnees est dans le lecteur "; lecteur$
    PRINT TAB(10); "L'imprimante est connectee au port "; imprimante$
    PRINT TAB(10); "-----------------------------------------"
    PRINT TAB(10); "1. Modifier le lecteur"
    PRINT TAB(10); "2. Modifier le port de sortie"
    PRINT TAB(10); "3. Fin des modifications"
    PRINT


120
DO
  LOCATE 13, 10: PRINT "CHOIX"
  CHOI2$ = INKEY$
LOOP UNTIL CHOI2$ = "1" OR CHOI2$ = "2" OR CHOI2$ = "3"
  CHOI2 = VAL(CHOI2$)
  ON CHOI2 GOTO 150, 180, 195
150
  CLS : LOCATE 10, 10: PRINT "Le lecteur est "; lecteur$
  LOCATE 11, 10: INPUT "Quel est le nouveau lecteur (Ex. B:)"; lecteur$
  GOTO 25
180
  CLS : IF imprimante$ = "parrallele" THEN imprimante$ = "serie" ELSE imprimante$ = "parrallele"
190
  GOTO 25
195
```

```
END SUB

SUB CONTRAINTE
'** ENTREE DES CONTRAINTES IN SITU
CLS
SHARED Np, nld

LOCATE 10, 5
PRINT "Voulez-vous inclure des contraintes in situ ?"
choi = 0
DO
  choi$ = INKEY$
  IF choi$ = "o" OR choi$ = "O" THEN choi = 1
  IF choi$ = "n" OR choi$ = "N" THEN choi = 2
LOOP UNTIL choi = 1 OR choi = 2
IF choi = 2 THEN GOTO 300
'CALL accfichier
REDIM PRESERVE R(Np, 2), nq(Np)

CLS
  LOCATE 10, 5
  INPUT "Entrez la contrainte horizontale ", sigx
  LOCATE 12, 5
  INPUT "Entrez la contrainte verticale ", sigy
CLS
  LOCATE 3, 5
  PRINT "Entrez le numero des noeuds au pourtout dans le sens antihoraire."
  LOCATE 5, 5
  PRINT "Jusqu'au noeud du depart inclusivement"
  PRINT
  N = 1
  PRINT TAB(5); "noeud "; N; TAB(15);
  INPUT pourtour1
  chk = 0
  FOR i = 1 TO nld
    IF pourtour1 = nq(i) THEN chk = 1
  NEXT i
  IF chk = 0 THEN
    nld = nld + 1
    nq(nld) = pourtour1
  END IF
  pourtourf = pourtour1
DO
  pourtour2 = pourtour1
  N = N + 1
  PRINT TAB(5); "noeud "; N; TAB(15);
  INPUT pourtour1
  forcex = (CORD(pourtour1, 2) - CORD(pourtour2, 2)) * sigx / 2
  forcey = (CORD(pourtour2, 1) - CORD(pourtour1, 1)) * sigy / 2
  chk = 0
  FOR i = 1 TO nld
    IF pourtour1 = nq(i) THEN chk = 1
  NEXT i
  IF chk = 0 THEN
```

```
    nld = nld + 1
    nq(nld) = pourtour1
  END IF
  FOR i = 1 TO nld
    IF pourtour1 = nq(i) THEN
      R(i, 1) = R(i, 1) + forcex
      R(i, 2) = R(i, 2) + forcey
    END IF
  NEXT i

  FOR i = 1 TO nld
    IF pourtour2 = nq(i) THEN
      R(i, 1) = R(i, 1) + forcex
      R(i, 2) = R(i, 2) + forcey
    END IF
  NEXT i
  IF N / 15 = INT(N / 15) THEN
    CLS
    LOCATE 5, 7
  END IF
LOOP UNTIL pourtourf = pourtour1

CALL crefichier
300
END SUB

SUB CREDON
SHARED Np, ne, NB, nld, NDF, NMAT, nom$, lecteur$, imprimante$, TITRE$
  CALL ENTNOMTRAV(TITRE$)      'Entree du nom du travail
  CALL AFFLIM                  'Affichage des limites (texte)
  CALL ENTLIM                  'Entree des limites
  CALL DIMVAR                  'Dimensionner les variables indicees
  CALL ENTDONNEES              'Entree des donnees
  CALL crefichier              'Creation du fichier de donnees
  CALL CONTRAINTE              'ENTREE DU TENSEUR DE CONTRAINTE
END SUB

SUB crefichier
REM
REM Creation du fichier de donnees
REM
  SHARED Np, ne, NB, nld, NDF, NMAT, nom$, lecteur$, TITRE$
  CLS
  LOCATE 10, 10: PRINT "Inserer la disquette de donnees dans "; lecteur$
  NOMPTMP$ = nom$
  PRINT TAB(10); "Quelle est le nom du fichier <"; NOMPTMP$; ">";
  INPUT nom$
  IF nom$ = "" THEN nom$ = NOMPTMP$
  OPEN "o", #1, lecteur$ + nom$ + ".ent"
  PRINT #1, Np, ne, NB
  PRINT #1, nld, NDF, NMAT
  FOR N = 1 TO NMAT
    PRINT #1, ORT(N, 1), ORT(N, 2), ORT(N, 3), ORT(N, 4), ORT(N, 5), poiun(N)
  NEXT N
```

```
  FOR N = 1 TO Np
    PRINT #1, CORD(N, 1), CORD(N, 2)
  NEXT N
  FOR N = 1 TO ne
    PRINT #1, nop(N, 1), nop(N, 2), nop(N, 3), IMAT(N)
  NEXT N
  FOR N = 1 TO NB
    PRINT #1, NBC(N), nfix(N)
  NEXT N
  FOR N = 1 TO nld
    PRINT #1, nq(N), R(N, 1), R(N, 2)
  NEXT N
  PRINT #1, TITRE$
  CLOSE #1
END SUB

SUB DESSGRIL
REM ************************************************************
REM Dessin de la grille
REM ************************************************************
 SHARED Np, ne, NB, nld, NDF, NMAT, nom$, lecteur$, TITRE$, chkgraf
  CALL accfichier        'Acces a un fichier de donnees

  CLS
  KEY OFF
  SCREEN 2               'Ecran en mode graphique

  CXMIN = CORD(1, 1): CYMIN = CORD(1, 2)
  CXMAX = CORD(1, 1): CYMAX = CORD(1, 2)
  FOR N = 2 TO Np
    IF CXMIN > CORD(N, 1) THEN CXMIN = CORD(N, 1)
    IF CYMIN > CORD(N, 2) THEN CYMIN = CORD(N, 2)
    IF CXMAX < CORD(N, 1) THEN CXMAX = CORD(N, 1)
    IF CYMAX < CORD(N, 2) THEN CYMAX = CORD(N, 2)
  NEXT N
  COEFX = 630 / ((CXMAX - CXMIN) * 2.18)
  COEFY = 180 / (CYMAX - CYMIN)
  IF COEFX <= COEFY THEN COEF = COEFX ELSE COEF = COEFY
  FOR N = 1 TO ne
    X1 = INT((CORD(nop(N, 1), 1) - CXMIN) * 2.18 * COEF)
    X2 = INT((CORD(nop(N, 2), 1) - CXMIN) * 2.18 * COEF)
    X3 = INT((CORD(nop(N, 3), 1) - CXMIN) * 2.18 * COEF)
    Y1 = INT(180 - ((CORD(nop(N, 1), 2) - CYMIN) * COEF))
    Y2 = INT(180 - ((CORD(nop(N, 2), 2) - CYMIN) * COEF))
    Y3 = INT(180 - ((CORD(nop(N, 3), 2) - CYMIN) * COEF))
    LINE (X1, Y1)-(X2, Y2)
    LINE (X2, Y2)-(X3, Y3)
    LINE (X3, Y3)-(X1, Y1)
  NEXT N
  WHILE INKEY$ = "": WEND 'Appuyer sur une touche

  CLS
  SCREEN 0               'Mode texte
```

```
  LOCATE 10, 10
  PRINT "Voulez-vous faire un fichier dxf avec ce fichier ?"
  DO
  tata$ = INKEY$
  LOOP UNTIL tata$ = "o" OR tata$ = "O" OR tata$ = "n" OR tata$ = "N"
  IF tata$ = "o" OR tata$ = "O" THEN
    chkgraf = 1
    CALL dxfout
  END IF
  chkgraf = 0


END SUB

SUB DIMVAR
REM
REM Dimensionner les variables indicees
REM
  SHARED Np, ne, NB, nld, NMAT

   REDIM ORT(NMAT, 5), CORD(Np, 2), nop(ne, 3), IMAT(ne), NBC(NB)
   REDIM nfix(NB), nq(nld), R(nld, 2), POID(Np), poiun(NMAT)
END SUB

SUB dxfout
'Transfert des deplacement a un fichier DXF pour AUTOCAD

SHARED Np, ne, NB, nld, NDF, NMAT, nom$, lecteur$, TITRE$, chkgraf
CALL accfichier
DIM dep(Np, 2)
IF chkgraf = 0 THEN OPEN lecteur$ + nom$ + ".sor" FOR INPUT AS #2
FOR i = 1 TO Np
  IF chkgraf = 0 THEN INPUT #2, disx, disy
  dep(i, 1) = CORD(i, 1) + disx
  dep(i, 2) = CORD(i, 2) + disy
NEXT
IF chkgraf = 0 THEN
  FOR i = 1 TO ne
    INPUT #2, xcen, ycen, sigmax, sigmay, tauxy, sigmamax, sigmamin, angle
  NEXT
END IF

CLOSE #2

OPEN "c:\files\acad\" + nom$ + ".dxf" FOR OUTPUT AS #2

PRINT #2, " 0"
PRINT #2, "SECTION"
PRINT #2, " 2"
PRINT #2, "ENTITIES"
PRINT #2, " 0"

FOR j = 1 TO Np
```

```
    PRINT #2, "POINT"
    PRINT #2, "  8"
    PRINT #2, "NOEUDS"
    PRINT #2, " 10"
    PRINT #2, INT(dep(j, 1) * 10000 + .5) / 10000
    PRINT #2, " 20"
    PRINT #2, INT(dep(j, 2) * 10000 + .5) / 10000
    PRINT #2, " 30"
    PRINT #2, "0.0"
    PRINT #2, " 0"
NEXT j
PRINT ne
FOR j = 1 TO ne
  FOR i = 1 TO 3

    PRINT #2, "LINE"
    PRINT #2, "  8"
    PRINT #2, "LIGNE"
    PRINT #2, " 10"
    PRINT #2, INT(dep(nop(j, i), 1) * 10000 + .5) / 10000
    PRINT #2, " 20"
    PRINT #2, INT(dep(nop(j, i), 2) * 10000 + .5) / 10000
    PRINT #2, " 30"
    PRINT #2, "0"
    PRINT #2, " 11"
    PRINT #2, INT(dep(nop(j, (i + 1 - INT(i / 3) * 3)), 1) * 10000 + .5) / 10000
    PRINT #2, " 21"
    PRINT #2, INT(dep(nop(j, (i + 1 - INT(i / 3) * 3)), 2) * 10000 + .5) / 10000
    PRINT #2, " 31"
    PRINT #2, "0"
    PRINT #2, "  0"
  NEXT i
NEXT j
PRINT #2, "ENDSEC"
PRINT #2, "  0"
PRINT #2, "EOF"

CLOSE #2
END SUB

SUB ENTDONNEES
REM
REM Entree des donnees
REM
REM Caracteristiques des materiaux
REM
  SHARED Np, ne, NB, nld, NMAT


  CLS
  PRINT "Caracteristiques des materiaux"
  PRINT "Materiau   Direction 1     Direction 2     Rigidité,        Poids/volume"
  PRINT "         E(1)   v(1)    E(2)     v(2)       G2  "
FOR N = 1 TO NMAT
```

```
  LOCATE 4 + N, 2: PRINT TAB(2); N
  LOCATE 4 + N, 12: INPUT ; ORT(N, 1)
  LOCATE 4 + N, 20: INPUT ; ORT(N, 2)
  LOCATE 4 + N, 28: INPUT ; ORT(N, 3)
  LOCATE 4 + N, 38: INPUT ; ORT(N, 4)
  ORT(N, 5) = ORT(N, 1) * ORT(N, 3) / (ORT(N, 1) + ORT(N, 3) + 2 * ORT(N, 4) * ORT(N, 3))
  LOCATE 4 + N, 49: PRINT ORT(N, 5)
  LOCATE 4 + N, 49: INPUT ; dd$:
  IF dd$ <> "" THEN ORT(N, 5) = VAL(dd$)
  LOCATE 4 + N, 67: INPUT poiun(N)
NEXT N

REM
REM Coordonnees des points nodaux
REM
  CLS
  PRINT "Coordonnees des points nodaux"
  PRINT "Point nodal      X              Y  "
  FOR N = 1 TO Np
    PRINT TAB(2); N; TAB(17);
    INPUT ; CORD(N, 1): PRINT TAB(33);
    INPUT CORD(N, 2)
  NEXT N

REM
REM Caracteristiques des elements
REM
  CLS
  PRINT "Caracteristiques des elements"
  PRINT "Element    `i'        `j'        `m'          materiau "
  FOR N = 1 TO ne
    PRINT TAB(2); N; TAB(12);
    INPUT ; nop(N, 1): PRINT TAB(25);
    INPUT ; nop(N, 2): PRINT TAB(38);
    INPUT ; nop(N, 3): PRINT TAB(54);
    INPUT IMAT(N)
  NEXT N

REM
REM Conditions aux frontieres
REM
  CLS
  PRINT "Conditions aux frontieres (deplacement)"
  PRINT "Note: Y fixe 01"
  PRINT "      X fixe 10"
  PRINT "      X et Y fixes 11"
  PRINT "Point nodal            Condition "
  FOR N = 1 TO NB
    PRINT TAB(2); : INPUT ; NBC(N)
    PRINT TAB(25); : INPUT nfix(N)
  NEXT N

REM
REM Condition de charge
```

```
REM
  CLS
  PRINT "Condition de Charge"
  PRINT "Point de charge      charge X      charge Y"
  FOR N = 1 TO nld
    PRINT TAB(2); : INPUT ; nq(N)
    PRINT TAB(26); : INPUT ; R(N, 1)
    PRINT TAB(40); : INPUT R(N, 2)
  NEXT N
END SUB

SUB ENTLIM
REM
REM Entree des limites
REM
 SHARED Np, ne, NB, nld, NDF, NMAT
  LOCATE 2, 50: INPUT Np
  LOCATE 3, 50: INPUT ne
  LOCATE 4, 50: INPUT NB
  LOCATE 5, 50: INPUT nld
  LOCATE 6, 50: INPUT NDF
  LOCATE 7, 50: INPUT NMAT
END SUB

SUB ENTNOMTRAV (TITRE$)
REM ************************************************************
REM Entree du nom du travail
REM ************************************************************
  CLS
  LOCATE 10, 10
  PRINT "Quel est le nom du travail?"
  LOCATE 11, 10: INPUT TITRE$

END SUB

SUB IMPDONNEES

REM ************************************************************************
REM Impression des donnees (parrallele)
REM ************************************************************************
 SHARED Np, ne, NB, nld, NDF, NMAT, nom$, lecteur$, imprimante$, TITRE$
  CALL accfichier
  CLS
  LOCATE 10, 10
  PRINT "L'imprimante est-elle prete"
  CALL ATTENTE                    'Appuyer sur une touche
  LPRINT
  LPRINT "Travail: "; TITRE$
  LPRINT "Fichier: "; nom$
  LPRINT
  LPRINT "-------------------------------------------------------------"
  LPRINT "Nombre de points nodaux: "; Np
  LPRINT "Nombre d'elements: "; ne
  LPRINT "Nombre de points nodaux avec restriction de deplacement: "; NB
```

```
  LPRINT "Nombre de points nodaux avec chargement: "; nld
  LPRINT "Nombre de degres de liberte: "; NDF
  LPRINT
  FOR N = 1 TO NMAT
    LPRINT "Proprietes du materiaux no."; N
    LPRINT TAB(6); "Module de Young: E1"; TAB(27); ORT(N, 1)
    LPRINT TAB(6); "Coefficient de Poisson: NU1"; TAB(35); ORT(N, 2)
    LPRINT TAB(6); "Module de Young: E2"; TAB(27); ORT(N, 1)
    LPRINT TAB(6); "Coefficient de Poisson: NU2"; TAB(35); ORT(N, 2)
    LPRINT TAB(6); "Module de rigidité,: G2"; "; TAB(29); ORT(N, 1)"
    LPRINT TAB(6); "Le poids par unite de volume: "; TAB(31); poiun(N)
  NEXT N
  LPRINT
  LPRINT "Coordonnees des noeuds"
  LPRINT "Numero du noeud     Coordonnee X      Coordonnee Y"
  FOR N = 1 TO Np
    LPRINT TAB(3); N; TAB(25); CORD(N, 1); TAB(44); CORD(N, 2)
  NEXT N
  LPRINT
  LPRINT "Identification des elements"
  LPRINT "Numero de l'element     Point nodaux          Materiaux"
  LPRINT TAB(25); "i"; TAB(35); "j"; TAB(45); "m"
  FOR N = 1 TO ne
    LPRINT TAB(3); N; TAB(25); nop(N, 1); TAB(35); nop(N, 2); TAB(45); nop(N, 3); TAB(55); IMAT(N)
  NEXT N
  LPRINT
  LPRINT "Restrictions sur les deplacements"
  LPRINT "Numero du noeud         Restriction"
  FOR N = 1 TO NB
    LPRINT TAB(3); NBC(N); TAB(27); nfix(N)
  NEXT N
  LPRINT
  LPRINT "Chargements"
  LPRINT "Numero du noeud       Composante X     Composante Y"
  FOR N = 1 TO nld
    LPRINT TAB(3); nq(N); TAB(25); R(N, 1); TAB(43); R(N, 2)
  NEXT N

END SUB

SUB IMPRESULTAT
REM ************************************************************
REM Impression des resultats (parrallele)
REM ************************************************************
 SHARED Np, ne, NB, nld, NDF, NMAT, nom$, lecteur$, imprimante$, TITRE$
  CALL accfichier   'Identification du fichier
  CLS
  LOCATE 10, 10
  PRINT "L'imprimante est-elle prete"
  CALL ATTENTE                    'Appuyer sur une touche
  LPRINT
  LPRINT "Travail: "; TITRE$
  LPRINT "Fichier: "; nom$: LPRINT
  LPRINT "-------------------------------------------"
```

```
LPRINT
OPEN "i", #2, lecteur$ + nom$ + ".sor"
LPRINT "Deplacements"
LPRINT "Noeud            Composante X     Composante Y"
FOR N = 1 TO Np
  INPUT #2, disx, disy
  disx = FNNT(disx)
  disy = FNNT(disy)
  LPRINT TAB(3); N; TAB(22);
  LPRINT USING "###.######"; disx; : LPRINT TAB(41);
  LPRINT USING "###.######"; disy
NEXT N
LPRINT : LPRINT
LPRINT "Contraintes suivant X et Y et contraintes principales"
LPRINT CHR$(15)
WIDTH "lpt1:", 130
LPRINT "Element      CentroideX   CentroideY   SigmaX      SigmaY      TAUXY       Sigma1
              Sigma3   Angle"
FOR N = 1 TO ne
  INPUT #2, XCENT, YCENT, sigmax, sigmay, tauxy, SIGMA1, SIGMA2, angle
  XCENT = FNNT(XCENT)
  YCENT = FNNT(YCENT)
  sigmax = FNNT(sigmax)
  sigmay = FNNT(sigmay)
  tauxy = FNNT(tauxy)
  SIGMA1 = FNNT(SIGMA1)
  SIGMA2 = FNNT(SIGMA2)
  angle = FNNT(angle)
  LPRINT TAB(3); N; TAB(15); : LPRINT USING "#####.####"; XCENT;
  LPRINT TAB(28); : LPRINT USING "#####.####"; YCENT;
  LPRINT TAB(41); : LPRINT USING "######.###"; sigmax;
  LPRINT TAB(54); : LPRINT USING "######.###"; sigmay;
  LPRINT TAB(67); : LPRINT USING "######.###"; tauxy;
  LPRINT TAB(80); : LPRINT USING "######.###"; SIGMA1;
  LPRINT TAB(93); : LPRINT USING "######.###"; SIGMA2;
  LPRINT TAB(106); : LPRINT USING "###.##"; angle
NEXT N
CLOSE #2
LPRINT CHR$(18)
END SUB

SUB MENPRINC (CHOIX)
REM ************************************************************************
REM Menu principal
REM ************************************************************************
CLS
  LOCATE 6, 10: PRINT "Menu principal"
    PRINT TAB(10); "-------------------------------------"
    PRINT TAB(10); "1. Creer un fichier de donnees"
    PRINT TAB(10); "2. Dessiner une grille"
    PRINT TAB(10); "3. Calculer des poids propres"
    PRINT TAB(10); "4. Modifier un fichier de donnees"
    PRINT TAB(10); "5. Afficher un fichier de donnees"
    PRINT TAB(10); "6. Imprimer un fichier de donnees"
```

```
    PRINT TAB(10); "7. Executer une analyse de contraintes"
    PRINT TAB(10); "8. Afficher un fichier de resultats"
    PRINT TAB(10); "9. Imprimer un fichier de resultats"
    PRINT TAB(10); "10. Creer un fichier DXF avec resultats"
    PRINT TAB(10); "11. Transferer un fichier de .NF a .ENT"
    PRINT TAB(10); "12. Quitter"
    PRINT TAB(10); "-----------------------------------------"

DO
  LOCATE 22, 10: INPUT "Choix"; CHOIX
  IF CHOIX < 1 OR CHOIX > 12 OR CHOIX <> INT(CHOIX) THEN
    CHKBCL = 0
    BEEP
    LOCATE 23, 5: PRINT "ERREUR! Mauvais choix!"
  ELSE
    CHKBCL = 1
  END IF
LOOP WHILE CHKBCL = 0
END SUB

SUB MODCONDCHAR
REM
REM Modification des conditions de charges
REM
' SHARED NP, NE, NB, NLD, NDF, NMAT, NOM$, LECTEUR$, IMPRIMANTE$, TITRE$
3480
  CALL AFFCONDCHAR
  PRINT "'0' met fin aux modifications"
  INPUT "Quelle condition voulez-vous modifier"; nu
  IF nu = 0 GOTO 3560
  INPUT "Point de charge"; nq(nu)
  INPUT "Charge en X"; R(nu, 1)
  INPUT "Charge en Y"; R(nu, 2)
  GOTO 3480
3560

END SUB

SUB MODCONDFRT
REM
REM Modifications des conditions aux frontieres
REM
 'SHARED NP, NE, NB, NLD, NDF, NMAT, NOM$, LECTEUR$, IMPRIMANTE$, TITRE$
3370
  CALL AFFCONDFRT
  PRINT "'0' met fin aux modifications"
  INPUT "quel point frontiere voulez-vous modifier"; nu
  IF nu = 0 GOTO 3440
  INPUT "Point frontiere"; NBC(nu)
  INPUT "Condition"; nfix(nu)
  GOTO 3370
3440
END SUB
```

```
SUB MODDON
' SHARED NP, NE, NB, NLD, NDF, NMAT, NOM$, LECTEUR$, IMPRIMANTE$, TITRE$

  CALL accfichier     'Acces a un fichier de donnees
DO
590
  CLS
  LOCATE 6, 10: PRINT "Quelles donnees voulez-vous modifier?"
    PRINT TAB(10); "---------------------------------------"
    PRINT TAB(10); "1. Les caracteristiques des materiaux"
    PRINT TAB(10); "2. Les conditions aux frontieres"
    PRINT TAB(10); "3. Les conditions de charges"
    PRINT TAB(10); "4. Les coordonnees des points nodaux"
    PRINT TAB(10); "5. Les elements"
    PRINT TAB(10); "6. Le nom du travail"
    PRINT TAB(10); "7. Fin des modifications"
    PRINT TAB(10); "---------------------------------------"
DO
  LOCATE 17, 10:
  PRINT "Choix"
  CHOI1 = VAL(INKEY$)
LOOP WHILE CHOI1 > 7 OR CHOI1 < 1
  IF CHOI1 = 1 THEN CALL MODMAT
  IF CHOI1 = 2 THEN CALL MODCONDFRT
  IF CHOI1 = 3 THEN CALL MODCONDCHAR
  IF CHOI1 = 4 THEN CALL MODPTND
  IF CHOI1 = 5 THEN CALL MODELE
  IF CHOI1 = 6 THEN CALL ENTNOMTRAV(TITRE$)
  IF CHOI1 = 7 THEN CALL crefichier
LOOP WHILE CHOI1 <> 7
END SUB

SUB MODELE
REM ************************************************************
REM Modification des elements
REM ************************************************************
'SHARED NP, NE, NB, NLD, NDF, NMAT, NOM$, LECTEUR$, IMPRIMANTE$, TITRE$
5630
  CALL AFFELEMENTS   'Affichage des elements
  PRINT : PRINT "'0' met fin aux modifications"
  INPUT "Quel element voulez-vous modifier"; nu
  IF nu = 0 THEN GOTO 5631
  INPUT "i"; nop(nu, 1)
  INPUT "j"; nop(nu, 2)
  INPUT "m"; nop(nu, 3)
  INPUT "Materiaux"; IMAT(nu)
  GOTO 5630
5631
END SUB

SUB MODMAT
REM
REM Modification des materiaux
REM
```

```
'SHARED NP, NE, NB, NLD, NDF, NMAT, NOM$, LECTEUR$, IMPRIMANTE$, TITRE$
3250
  CALL AFFCARMAT
  PRINT
  PRINT "`0` met fin aux modifications"
  INPUT "Quel materiau voulez-vous modifier"; nu
  IF nu = 0 GOTO 3330
  INPUT "Le module d'Young E(1)"; ORT(nu, 1)
  INPUT "Le coefficient de Poisson v(1)"; ORT(nu, 2)
  INPUT "Le module d'Young E(2)"; ORT(nu, 3)
  INPUT "Le coefficient de Poisson v(2)"; ORT(nu, 4)
  ORT(nu, 5) = INT((ORT(nu, 1) * ORT(nu, 3) / (ORT(nu, 1) + ORT(nu, 3) + 2 * ORT(nu, 4) * ORT(nu,
                    3))) * 100 + .5) / 100
  PRINT "Le module de rigidité, G <"; ORT(nu, 5); ">"; : INPUT ; dd$
  IF dd$ <> "" THEN ORT(nu, 5) = VAL(dd$)
  PRINT
  INPUT "Le poids par unite de volume"; poiun(nu)
  GOTO 3250
3330
END SUB

SUB MODPTND
REM ************************************************************
REM Modification des points nodaux
REM ************************************************************
 'SHARED NP, NE, NB, NLD, NDF, NMAT, NOM$, LECTEUR$, IMPRIMANTE$, TITRE$
5530
  CALL AFFCORPTND   'Affichage des coordonnees des materiaux
  PRINT : PRINT "`0' met fin aux modifications"
  INPUT "Quel point voulez-vous modifier"; nu
  IF nu = 0 THEN GOTO 5531
  INPUT "Coordonnee X"; CORD(nu, 1)
  INPUT "Coordonnee Y"; CORD(nu, 2)
  GOTO 5530
5531

END SUB

SUB present
REM ************************************************************
REM Page de presentation
REM ************************************************************
  CLS
  LOCATE 6, 33: PRINT "Programme ACCES"
  LOCATE 7, 33: PRINT "Preparation des"
  LOCATE 8, 31: PRINT "Fichiers de donnees"
  LOCATE 9, 32: PRINT "Pour ELEFIN V 1.1"
  LOCATE 19, 1: PRINT "(c) S.D.Gill et D.E. Gill, 1989"
  LOCATE 20, 1: PRINT "(c) J.-P. Basque, 1994"
  LOCATE 21, 5: PRINT "Ecole Polytechnique"
  CALL ATTENTE          'Appuyer sur une touche
  CLS
END SUB
```

```
DECLARE SUB routinedechargement ()
DECLARE SUB largeurbande ()
DECLARE SUB readres ()
DECLARE SUB detmodules (l, a, b, C, c1, d, e, F)
DECLARE SUB effichmat ()
DECLARE SUB MODYOUNG (i)
DECLARE SUB SAVFICHMAT (i)
DECLARE SUB rapfichMAT (i)
DECLARE SUB modpara (i)
DECLARE SUB printpara (i)
DECLARE SUB defpara ()
DECLARE SUB MAIN ()
DECLARE SUB LECDONPROPGEO ()
DECLARE SUB presentation ()
DECLARE SUB LECDONCHAR ()
DECLARE SUB FORMk ()
DECLARE SUB STIFFNESS (N)
DECLARE SUB SOLVE ()
DECLARE SUB STRESS ()
DECLARE SUB erreur (N)
OPTION BASE 1
COMMON NOM$, LECTEUR$, imprimante$
tmp = 150
temp = 10
DIM SHARED ort(tmp, 5), CORD(tmp, 2), nop(tmp, 4), IMAT(tmp), NBC(tmp)
DIM SHARED ESTIFM(12, 12), R(8), R1(tmp), sk(tmp, 50), NFIX(tmp), poiun(tmp)
DIM SHARED a(3, 6), b(3, 9), forcep(tmp, 3), para(temp, 5, 4)
DIM SHARED nommat$, valeurs(temp, 4, 4), erreurmoyenne(tmp, 3)
DIM SHARED typequa(temp, 5), CONCHARGE(tmp), nu(temp)
REM Programme Elefin V1.1
REM 1995-07-06
REM
REM programme principal
REM
CALL MAIN
END         'Fin du programme

SUB defpara

'*************************************************************************
' ROUTINE DE DEFINITION DES PARAMETRES POUR LES DIFFERENTS MATERIAUX
'*************************************************************************
SHARED nmat
FOR i = 1 TO nmat

  e1 = ort(i, 1)
  e2 = ort(i, 3)
  nu = (ort(i, 2) + ort(i, 4)) / 2
  para(i, 1, 1) = 1 / (3 / (2 * (1 - nu) / e1 + (1 - 4 * nu) / e2)) * 1E+12
  para(i, 2, 1) = 1 / (3 / ((1 - nu) / e1 + 2 * (1 + 2 * nu) / e2)) * 1E+12
  para(i, 5, 1) = (1 + nu) / e1 * 1E+12
  IF e1 <> e2 THEN
    para(i, 3, 1) = ((1 - nu) / 3 * (1 / e2 - 1 / e1)) * 1E+12
    para(i, 4, 1) = (2 * (1 - nu) / 3 * (1 / e2 - 1 / e1)) * 1E+12
```

```
  ELSE
    para(i, 3, 1) = 0
    para(i, 4, 1) = 0
  END IF
  FOR K = 1 TO 5
      para(i, K, 2) = 0
      para(i, K, 3) = 1
      para(i, K, 4) = 1
  NEXT K
  CLS
  LOCATE 1, 3
  PRINT "Le mat,riau "; i; " possŜde les caract,ristiques suivantes:"
  LOCATE 3
  PRINT USING "  E1 ###########,      Nu1 ##.##"; ort(i, 1); ort(i, 2)
  PRINT USING "  E2 ###########,      Nu2 ##.##"; ort(i, 3); ort(i, 4)
  PRINT USING "  G2 ###########,      "; ort(i, 5)

  LOCATE 8, 3
  PRINT "Pour le traitement, les paramŜtres ,quivalent utilis,s sont:"
  LOCATE 10
  CALL printpara(i)
  LOCATE 20
  PRINT "   1 - Pour accepter les valeurs ci-haut"
  PRINT "   2 - Pour modifier les valeurs ci-haut"

  DO
    dd$ = INKEY$
  LOOP UNTIL dd$ = "1" OR dd$ = "2"
  IF dd$ = "2" THEN
    CLS
    LOCATE 5
    PRINT "  1 - Pour entrer un module de Young et co,f. de Poisson"
    PRINT "  2 - Pour entrer des paramŜtres directement"
    PRINT "  3 - Pour ne pas faire de modification"
    DO
      dd$ = INKEY$
    LOOP UNTIL dd$ = "1" OR dd$ = "2" OR dd$ = "3"
    IF dd$ = "1" THEN CALL MODYOUNG(i)
    IF dd$ = "2" THEN CALL modpara(i)
  END IF
NEXT i
END SUB

SUB detmodules (l, SIGMAX, SIGMAZ, tauxz, e1, e2, nu, VERI)
DIM COEF(5, 3), CONTRAINTE(5), DEPSM(5)
SHARED boucle, MODETRAIT
VERI = 0
N = 0
EM2 = 0
ezz2 = 0

20 REM Determination des pentes a l'origine

sx = -(SIGMAX)
```

```
sz = -(SIGMAZ)
IF sx = 0 THEN sx = 3000000!
IF sz = 0 THEN sz = 3000000!
txz = ABS(tauxz)
sx = sx / 1000000!
sz = sz / 1000000!
txz = txz / 1000000!
IF sx <> sz THEN ANGLE = ATN(2 * txz / (sx - sz)) * 90 / 3.1415926537# / 2 ELSE ANGLE = 45
IF sx < sz THEN ANGLE = ANGLE + 90
IF sx = sz THEN ANGLE = 90
S1 = (sx + sz) / 2 + (sx - sz) / 2 * COS(2 * ANGLE / 180 * 3.1415926537#) + txz * SIN(2 * ANGLE / 180
                * 3.1415926537#)
S3 = (sx + sz) / 2 - (sx - sz) / 2 * COS(2 * ANGLE / 180 * 3.1415926537#) - txz * SIN(2 * ANGLE / 180 *
                3.1415926537#)
E1P = 1E+10
E2P = 1E+10


SIGM = (sx + sz) / 3
  S11 = S1 - SIGM
  S33 = S3 - SIGM
  SZZ = sz - SIGM
  SYY = sy - SIGM
  SXX = sx - SIGM
  SIJMAX = ABS(S11)
  IF ABS(S33) > SIJMAX THEN SIJMAX = ABS(S33)
  IF ABS(SYY) > SIJMAX THEN SIJMAX = ABS(SYY)

  CONTRAINTE(1) = (SIGM)
  CONTRAINTE(2) = (SIJMAX)
  CONTRAINTE(3) = (SZZ)
  CONTRAINTE(4) = (SIGM)
  CONTRAINTE(5) = (SXX - SYY)

50 REM REGRESSIONS A L'AIDE DES FONCTIONS LISSEES

FOR M = 1 TO 5
    IF CONTRAINTE(M) > 0 THEN
      DEPSM(M) = (para(l, M, 1) * CONTRAINTE(M) + para(l, M, 2) * (1 - EXP(-para(l, M, 3) *
                CONTRAINTE(M))) ^ para(l, M, 4))
    ELSE
      DEPSM(M) = para(l, M, 1) + para(l, M, 2) * para(l, M, 3) * para(l, M, 4)
    END IF
NEXT M
EM1 = DEPSM(1)
ezz1 = DEPSM(2) * SZZ / SIJMAX
EM2 = DEPSM(3)
ezz2 = DEPSM(4)
exx1 = (DEPSM(5) - ezz1) / 2
exx2 = -ezz2 / 2
eyy1 = -(exx1 + ezz1)
eyy2 = -(exx2 + ezz2)

30 REM CALCUL DU MODULE SECANT
```

```
ex = (EM1 + EM2 + exx1 + exx2) / 1000000!
ez = (EM1 + EM2 + ezz1 + ezz2) / 1000000!
ey = (EM1 + EM2 + eyy1 + eyy2) / 1000000!
sx = sx * 10000!
sy = sy * 10000!
sz = sz * 10000!

e2 = ABS((sz - nu * (sx + sy)) / ez)
IF e2 > 1E+13 THEN e2 = 1E+13
IF e2 < 1E+08 THEN e2 = 1E+08
e1 = ABS((sx - nu * sy) / (ex + (nu * sz) / e2))
IF e1 > 1E+13 THEN e1 = 1E+13
IF e1 < 1E+08 THEN e1 = 1E+08

REM PRINT USING "### ###############,.# ##############,.#  #.####
                 ###########.## ###.##"; N; E1; E2; NU; sx; sz

IF INKEY$ = CHR$(27) THEN END
END SUB

SUB effichmat
CLS
LOCATE 5
OPEN "C:\ELEFIN\DONNEES.v8" FOR INPUT AS #1
INPUT #1, nomat
REDIM nommat$(nomat), valeurs(nomat, 4, 4)
FOR J = 1 TO nomat
  INPUT #1, nommat$(J)
  INPUT #1, valeurs(J, 1, 1), valeurs(J, 1, 2), valeurs(J, 1, 3), valeurs(J, 1, 4)
  INPUT #1, valeurs(J, 2, 1), valeurs(J, 2, 2), valeurs(J, 2, 3), valeurs(J, 2, 4)
  INPUT #1, valeurs(J, 3, 1), valeurs(J, 3, 2), valeurs(J, 3, 3), valeurs(J, 3, 4)
  INPUT #1, valeurs(J, 4, 1), valeurs(J, 4, 2), valeurs(J, 4, 3), valeurs(J, 4, 4)
  INPUT #1, valeurs(J, 5, 1), valeurs(J, 5, 2), valeurs(J, 5, 3), valeurs(J, 5, 4)

  PRINT USING " ##  \                \"; J; nommat$(J)
NEXT J
CLOSE #1
INPUT "Quel materiau voulez-vous effacer (0 pour aucun)", nmat

IF nmat > 0 THEN
  OPEN "C:\ELEFIN\DONNEES.v8" FOR OUTPUT AS #1
  PRINT #1, nomat - 1
  FOR J = 1 TO nomat
    IF J <> nmat THEN
      PRINT #1, nommat$(J)
      PRINT #1, valeurs(J, 1, 1), valeurs(J, 1, 2), valeurs(J, 1, 3), valeurs(J, 1, 4)
      PRINT #1, valeurs(J, 2, 1), valeurs(J, 2, 2), valeurs(J, 2, 3), valeurs(J, 2, 4)
      PRINT #1, valeurs(J, 3, 1), valeurs(J, 3, 2), valeurs(J, 3, 3), valeurs(J, 3, 4)
      PRINT #1, valeurs(J, 4, 1), valeurs(J, 4, 2), valeurs(J, 4, 3), valeurs(J, 4, 4)
      PRINT #1, valeurs(J, 5, 1), valeurs(J, 5, 2), valeurs(J, 5, 3), valeurs(J, 5, 4)
    END IF
  NEXT J
  CLOSE #1
END IF
```

```
END SUB

SUB erreur (N)
PRINT "Erreur dans l'element "; N
PRINT
PRINT

PRINT "    pressez une touche pour continuer..."
DO
LOOP WHILE INKEY$ = ""

END SUB

SUB FORMk
REM
REM Formk
REM
SHARED np, ne, NB, NLD, NDF, nmat, nszf, nband, NCN, MODETRAIT, boucle
IF boucle = -1 THEN CALL largeurbande
FOR N = 1 TO nszf
  FOR M = 1 TO nband
    sk(N, M) = 0
NEXT M, N
VERI = 0
FOR N = 1 TO ne
  CALL STIFFNESS(N)
  VERI = 1
  IF ctr = 0 THEN
    PRINT "/";
    ctr = 1
  ELSE
    PRINT "\";
    ctr = 0
  END IF

  FOR JJ = 1 TO NCN
    nrowb = (nop(N, JJ) - 1) * NDF
      FOR J = 1 TO NDF
        nrowb = nrowb + 1
        i = (JJ - 1) * NDF + J
        FOR KK = 1 TO NCN
          NCOLB = (nop(N, KK) - 1) * NDF
          FOR K = 1 TO NDF
            l = (KK - 1) * NDF + K
            ncol = NCOLB + K + 1 - nrowb
            IF ncol > 0 THEN sk(nrowb, ncol) = sk(nrowb, ncol) + ESTIFM(i, l)
          NEXT K
        NEXT KK
  NEXT J, JJ
NEXT N
FOR N = 1 TO NB
  NX = 10 ^ (NDF - 1)
  i = NBC(N)
  nrowb = (i - 1) * NDF
```

```
  FOR M = 1 TO NDF
    nrowb = nrowb + 1
    ICON = INT(NFIX(N) / NX)
    IF ICON > 0 THEN
      sk(nrowb, 1) = 1
      FOR J = 2 TO nband
        sk(nrowb, J) = 0
        NR = nrowb + 1 - J
        IF NR > 0 THEN sk(NR, J) = 0
      NEXT J
      NFIX(N) = NFIX(N) - NX * ICON
    END IF
    NX = NX / 10
  NEXT M
NEXT N
END SUB

SUB largeurbande
SHARED np, ne, NB, NLD, NDF, nmat, nszf, nband, NCN
FOR N = 1 TO ne
  CALL STIFFNESS(N)
  FOR JJ = 1 TO NCN
    REM PRINT ".";
    nrowb = (nop(N, JJ) - 1) * NDF
      FOR J = 1 TO NDF
        nrowb = nrowb + 1
        i = (JJ - 1) * NDF + J
        FOR KK = 1 TO NCN
          NCOLB = (nop(N, KK) - 1) * NDF
          FOR K = 1 TO NDF
            l = (KK - 1) * NDF + K
            ncol = NCOLB + K + 1 - nrowb
            IF ncol > nband THEN nband = ncol
            IF INKEY$ = CHR$(27) THEN END
          NEXT K
        NEXT KK
  NEXT J, JJ
NEXT N
PRINT nband
REDIM sk(nszf, nband)
END SUB

SUB LECDONCHAR
REM
REM Lecture dans le fichier de donnees des conditions de charges
REM
SHARED np, ne, NB, NLD, NDF, nmat, nszf, nband, NCN, NOMBREBOUCLE, CYCLE
FOR J = 1 TO nszf
  R1(J) = 0
NEXT J
FOR M = 1 TO NLD
  INPUT #1, NQ, R(1), R(2)
  IF NOMBREBOUCLE > 0 THEN
    R(1) = R(1) * CONCHARGE(CYCLE)
```

```
    R(2) = R(2) * CONCHARGE(CYCLE)
  END IF
  FOR K = 1 TO NDF
   IC = (NQ - 1) * NDF + K
   R1(IC) = R(K) + R1(IC)
  NEXT K
NEXT M
END SUB

SUB LECDONPROPGEO
REM
REM Lecture dans le fichier de donnees des proprietes et de la geometrie
REM
SHARED np, ne, NB, NLD, NDF, nmat, nszf, nband, NCN, boucle
INPUT #1, np, ne, NB
INPUT #1, NLD, NDF, nmat
nszf = np * NDF
IF boucle = -1 THEN REDIM forcep(ne, 3), ort(nmat, 5), nu(nmat)
REDIM CORD(np, 2)
REDIM nop(ne, 4)
REDIM IMAT(ne)
REDIM NBC(NB)
REDIM R1(nszf)
REDIM sk(nszf, nband)
REDIM poiun(nmat)
REDIM NFIX(NB)

NCN = 3
REM Dimensionner les variables indicees
FOR N = 1 TO nmat
  INPUT #1, ort(N, 1), ort(N, 2), ort(N, 3), ort(N, 4), ort(N, 5), poiun(N)
IF boucle = -1 THEN nu(nmat) = ort(N, 2) / 2 + ort(N, 4) / 2
NEXT N

FOR N = 1 TO np
  INPUT #1, CORD(N, 1), CORD(N, 2)
NEXT N

FOR N = 1 TO ne
  INPUT #1, nop(N, 1), nop(N, 2), nop(N, 3), IMAT(N)
NEXT N

FOR N = 1 TO NB
  INPUT #1, NBC(N), NFIX(N)
NEXT N
END SUB

SUB MAIN
'************************************
'     ROUTINE PRINCIPALE
'************************************
REM   CALL PRESENTATION
CLS
```

```
SHARED NOM$, LECTEUR$, MODETRAIT, boucle, ALPHA, nband, CTRL, NOMBREBOUCLE,
                CYCLE
LOCATE 10, 10
PRINT "Inserer la disquette de donnees dans "; LECTEUR$
  NOMPTMP$ = NOM$
  LOCATE 11, 10: PRINT "Quel est le nom du fichier <"; NOMPTMP$; ">";
  INPUT NOM$
  IF NOM$ = "" THEN NOM$ = NOMPTMP$

  LOCATE 13, 10: PRINT "Voulez-vous traiter ce problŠme en"
  LOCATE 15, 20: PRINT "1. Etat plan de contrainte (Plane stress)"
  LOCATE 16, 20: PRINT "2. Etat plan de d,formation (Plane strain)  "
DO
  dd$ = INKEY$
LOOP UNTIL dd$ = "1" OR dd$ = "2"
MODETRAIT = VAL(dd$)
CLS
CALL routinedechargement
boucle = -1
nband = 5
OPEN "c:\elefin\young1.prn" FOR OUTPUT AS #3
OPEN "C:\ELEFIN\PARAMETR.PRN" FOR OUTPUT AS #4
OPEN "C:\ELEFIN\DEFORM.PRN" FOR OUTPUT AS #5
CYCLE = 0
DO
  CYCLE = CYCLE + 1
  ALPHA = 0
  CTRL = 0
  DO
    ALPHA = ALPHA + 1
    OPEN "i", #1, LECTEUR$ + NOM$ + ".ent"
    CALL LECDONPROPGEO

    CALL LECDONCHAR
    CLOSE #1
    IF boucle = -1 THEN
      CALL defpara
      verification = 0
    END IF
    OPEN "o", #2, LECTEUR$ + NOM$ + ".tmp"
    CALL FORMk
    CALL SOLVE
    CLOSE #2
    IF NOMBREBOUCLE > 0 THEN EXT$ = RIGHT$(STR$(CYCLE), INT(LOG(CYCLE) / LOG(10) +
                1)) ELSE EXT$ = "SOR"

    OPEN "o", #1, LECTEUR$ + NOM$ + "." + EXT$
    OPEN "i", #2, LECTEUR$ + NOM$ + ".tmp"
    CALL STRESS
    CLOSE #2, #1
    PRINT "BOUCLE NUMERO "; ALPHA
    OPEN "c:\elefin\erreur.sor" FOR OUTPUT AS #1
    FOR a = 1 TO ALPHA
      PRINT #1, a, erreurmoyenne(a, 1), erreurmoyenne(a, 2), erreurmoyenne(a, 3)
```

```
    NEXT a
    CLOSE #1
  LOOP UNTIL boucle = 1
  boucle = 0
LOOP UNTIL CYCLE >= NOMBREBOUCLE
KILL LECTEUR$ + NOM$ + ".tmp"
CLOSE #3, #4, #5
END SUB

SUB modpara (i)
DO
  CLS
  LOCATE 3
  CALL printpara(i)
  PRINT
  PRINT
  PRINT "Voulez-vous:  1 - Modifier ces paramŠtres"
  PRINT "              2 - Rappeler un fichier mat,riau"
  PRINT "              3 - Effacer un fichier mat,riau"
  PRINT "              4 - Accepter ces paramŠtres"
  DO
    dd$ = INKEY$
  LOOP UNTIL VAL(dd$) >= 1 AND VAL(dd$) <= 4
  IF dd$ = "2" THEN CALL rapfichMAT(i)
  IF dd$ = "3" THEN CALL effichmat
LOOP UNTIL dd$ = "1" OR dd$ = "4"
IF dd$ = "1" THEN
  DO
    CLS
    LOCATE 3
    CALL printpara(i)
    LOCATE 13
    PRINT "   Entrez l',quation ... modifier "
    DO
      EQUA$ = INKEY$
      eqm = VAL(EQUA$)
    LOOP UNTIL eqm >= 1 AND eqm <= 5 OR EQUA$ = CHR$(27)
    IF EQUA$ <> CHR$(27) THEN
      LOCATE 15
      PRINT "   Entrez le paramŠtre ... modifier "
      DO
        PAR = VAL(INKEY$)
      LOOP UNTIL PAR >= 1 AND PAR <= 4
      LOCATE 17

      PRINT "ParamŠtre ... modifier <"; para(i, eqm, PAR); ">"
      LOCATE 17, 35: INPUT " ", dd$
      IF dd$ <> "" THEN para(i, eqm, PAR) = VAL(dd$)
      CLS
      LOCATE 3
      CALL printpara(i)
    END IF
  LOOP UNTIL EQUA$ = CHR$(27)
  LOCATE 17
```

```
  PRINT "Voulez-vous sauvegarder ce mat,riau (o/n)"
  DO
    dd$ = INKEY$
  LOOP UNTIL dd$ = "o" OR dd$ = "O" OR dd$ = "n" OR dd$ = "N"
  IF dd$ = "o" OR dd$ = "O" THEN CALL SAVFICHMAT(i)
END IF
nomod:
END SUB

SUB MODYOUNG (i)
  CLS
  LOCATE 5
  PRINT "  Entrez E1  <"; ort(i, 1); ">": LOCATE 5, 35: INPUT " ", dd$
  IF dd$ <> "" THEN ort(i, 1) = VAL(dd$)
  PRINT "  Entrez E2  <"; ort(i, 3); ">": LOCATE 6, 35: INPUT " ", dd$
  IF dd$ <> "" THEN ort(i, 3) = VAL(dd$)

  PRINT "  Entrez nu1 <"; ort(i, 2); ">": LOCATE 7, 35: INPUT " ", dd$
  IF dd$ <> "" THEN ort(i, 2) = VAL(dd$)
  PRINT "  Entrez nu2 <"; ort(i, 4); ">": LOCATE 8, 35: INPUT " ", dd$
  IF dd$ <> "" THEN ort(i, 4) = VAL(dd$)
  g2 = (ort(i, 1) * ort(i, 3)) / (ort(i, 1) + ort(i, 3) + (ort(i, 2) + ort(i, 4)) + ort(i, 3))
  PRINT "  Entrez G2 <"; g2; ">": LOCATE 9, 35: INPUT " ", dd$
  IF dd$ = "" THEN ort(i, 5) = g2 ELSE ort(i, 5) = VAL(dd$)
  PRINT
  PRINT "  Il est a noter que pour le traitement, le programme considSre un seul"
  PRINT "  co,fficient de Poisson et prend la moyenne des deux co,fficient."
  ort(i, 2) = (ort(i, 2) + ort(i, 4)) / 2
  ort(i, 4) = ort(i, 2)
  nu(i) = ort(i, 2)
  e1 = ort(i, 1)
  e2 = ort(i, 3)
  nu = (ort(i, 2) + ort(i, 4)) / 2
  para(i, 1, 1) = 1 / (3 / (2 * (1 - nu) / e1 + (1 - 4 * nu) / e2)) * 1E+12
  para(i, 2, 1) = 1 / (3 / ((1 - nu) / e1 + 2 * (1 + 2 * nu) / e2)) * 1E+12
  para(i, 5, 1) = (1 + nu) / e1 * 1E+12
  IF e1 <> e2 THEN
    para(i, 3, 1) = ((1 - nu) / 3 * (1 / e2 - 1 / e1)) * 1E+12
    para(i, 4, 1) = (2 * (1 - nu) / 3 * (1 / e2 - 1 / e1)) * 1E+12
  ELSE
    para(i, 3, 1) = 0
    para(i, 4, 1) = 0
  END IF
  FOR K = 1 TO 5
      para(i, K, 2) = 0
      para(i, K, 3) = 1
      para(i, K, 4) = 1
  NEXT K
  CALL printpara(i)
END SUB

SUB presentation
REM
REM Page de presentation
```

```
REM
CLS
LOCATE 10, 28: PRINT "Programme ELEFIN V1.1"
LOCATE 21, 5: PRINT "(c) S.D. Gill et D.E. Gill, 1989"
LOCATE 22, 5: PRINT "Ecole Polytechnique"
PRINT : PRINT "Appuyer sur une touche"
WHILE INKEY$ = ""
WEND
END SUB

SUB printpara (i)
    PRINT "        parametre m      parametre b      parametre a      parametre c"
PRINT USING "1 im1      ##.###^^^^      ##.###^^^^      ##.###^^^^      ##.###^^^^ "; para(i,
                1, 1); para(i, 1, 2); para(i, 1, 3); para(i, 1, 4)
PRINT USING "2 ezz1     ##.###^^^^      ##.###^^^^      ##.###^^^^      ##.###^^^^ "; para(i,
                2, 1); para(i, 2, 2); para(i, 2, 3); para(i, 2, 4)
PRINT USING "3 im2      ##.###^^^^      ##.###^^^^      ##.###^^^^      ##.###^^^^ "; para(i,
                3, 1); para(i, 3, 2); para(i, 3, 3); para(i, 3, 4)
PRINT USING "4 ezz2     ##.###^^^^      ##.###^^^^      ##.###^^^^      ##.###^^^^ "; para(i,
                4, 1); para(i, 4, 2); para(i, 4, 3); para(i, 4, 4)
PRINT USING "5 exx1-eyy1  ##.###^^^^      ##.###^^^^      ##.###^^^^      ##.###^^^^ ";
                para(i, 5, 1); para(i, 5, 2); para(i, 5, 3); para(i, 5, 4)

PRINT
PRINT " NOTE: l',quation est du type Y = m * X + b * (1 - exp(-a * X))^c."
'DO: LOOP UNTIL INKEY$ <> ""
END SUB

SUB rapfichMAT (i)
CLS
OPEN "C:\ELEFIN\DONNEES.v8" FOR INPUT AS #1
INPUT #1, nomat
REDIM nommat$(nomat), valeurs(nomat, 5, 4), NULU(nomat)
FOR J = 1 TO nomat
  INPUT #1, nommat$(J)
  INPUT #1, NULU(J)
  INPUT #1, valeurs(J, 1, 1), valeurs(J, 1, 2), valeurs(J, 1, 3), valeurs(J, 1, 4)
  INPUT #1, valeurs(J, 2, 1), valeurs(J, 2, 2), valeurs(J, 2, 3), valeurs(J, 2, 4)
  INPUT #1, valeurs(J, 3, 1), valeurs(J, 3, 2), valeurs(J, 3, 3), valeurs(J, 3, 4)
  INPUT #1, valeurs(J, 4, 1), valeurs(J, 4, 2), valeurs(J, 4, 3), valeurs(J, 4, 4)
  INPUT #1, valeurs(J, 5, 1), valeurs(J, 5, 2), valeurs(J, 5, 3), valeurs(J, 5, 4)
NEXT J
CLOSE #1

LOCATE 5
PRINT "Choississez un mat,riau"
LOCATE 7
FOR J = 1 TO nomat
  IF nomat < 16 THEN
    PRINT USING "  ## \            \"; J; nommat$(J)
  ELSE
    PRINT USING "                    ## \            \"; J; nommat$(J)
  END IF
NEXT J
```

```
LOCATE 5, 30
INPUT " ", CHOIXMAT
nu(i) = NULU(CHOIXMAT)
FOR J = 1 TO 5
  FOR K = 1 TO 4
    para(i, J, K) = valeurs(CHOIXMAT, J, K)
  NEXT K
NEXT J

END SUB

SUB routinedechargement
SHARED NOMBREBOUCLE
LOCATE 5, 5
PRINT "Voulez faire un execution avec plusieurs condition de chargement (o/n)"
DO
  dd$ = INKEY$
LOOP UNTIL dd$ = "o" OR dd$ = "O" OR dd$ = "n" OR dd$ = "N"
IF dd$ = "o" OR dd$ = "O" THEN
  LOCATE 7, 5
  INPUT "Quel est le nom de fichier de chargement "; nomchar$
  OPEN "c:\elefin\" + nomchar$ + ".prn" FOR INPUT AS #1
  INPUT #1, NOMBREBOUCLE
  REDIM CONCHARGE(NOMBREBOUCLE)
  FOR i = 1 TO NOMBREBOUCLE
    INPUT #1, CONCHARGE(i)
    PRINT CONCHARGE(i)
  NEXT i
END IF
CLOSE #1
END SUB

SUB SAVFICHMAT (i)
LOCATE 22
INPUT "Quel nom voulez-vous donner ... votre mat,riau "; nommat1$
OPEN "C:\ELEFIN\DONNEES.v8" FOR INPUT AS #1
INPUT #1, nomat
IF nomat <> 0 THEN REDIM nommat$(nomat), valeurs(nomat, 5, 4), NULU(nomat)
FOR J = 1 TO nomat
  INPUT #1, nommat$(J)
  INPUT #1, NULU(J)
  INPUT #1, valeurs(J, 1, 1), valeurs(J, 1, 2), valeurs(J, 1, 3), valeurs(J, 1, 4)
  INPUT #1, valeurs(J, 2, 1), valeurs(J, 2, 2), valeurs(J, 2, 3), valeurs(J, 2, 4)
  INPUT #1, valeurs(J, 3, 1), valeurs(J, 3, 2), valeurs(J, 3, 3), valeurs(J, 3, 4)
  INPUT #1, valeurs(J, 4, 1), valeurs(J, 4, 2), valeurs(J, 4, 3), valeurs(J, 4, 4)
  INPUT #1, valeurs(J, 5, 1), valeurs(J, 5, 2), valeurs(J, 5, 3), valeurs(J, 5, 4)
NEXT J
CLOSE #1
OPEN "C:\ELEFIN\DONNEES.v8" FOR OUTPUT AS #1
PRINT #1, nomat + 1
FOR J = 1 TO nomat
  PRINT #1, nommat$(J)
  PRINT #1, NULU(J)
```

```
  PRINT #1, valeurs(J, 1, 1), valeurs(J, 1, 2), valeurs(J, 1, 3), valeurs(J, 1, 4)
  PRINT #1, valeurs(J, 2, 1), valeurs(J, 2, 2), valeurs(J, 2, 3), valeurs(J, 2, 4)
  PRINT #1, valeurs(J, 3, 1), valeurs(J, 3, 2), valeurs(J, 3, 3), valeurs(J, 3, 4)
  PRINT #1, valeurs(J, 4, 1), valeurs(J, 4, 2), valeurs(J, 4, 3), valeurs(J, 4, 4)
  PRINT #1, valeurs(J, 5, 1), valeurs(J, 5, 2), valeurs(J, 5, 3), valeurs(J, 5, 4)

NEXT J
PRINT #1, nommat1$
PRINT #1, ort(i, 2)
PRINT #1, para(i, 1, 1), para(i, 1, 2), para(i, 1, 3), para(i, 1, 4)
PRINT #1, para(i, 2, 1), para(i, 2, 2), para(i, 2, 3), para(i, 2, 4)
PRINT #1, para(i, 3, 1), para(i, 3, 2), para(i, 3, 3), para(i, 3, 4)
PRINT #1, para(i, 4, 1), para(i, 4, 2), para(i, 4, 3), para(i, 4, 4)
PRINT #1, para(i, 5, 1), para(i, 5, 2), para(i, 5, 3), para(i, 5, 4)
CLOSE #1
END SUB

SUB SOLVE
REM
REM solve
REM
SHARED np, ne, NB, NLD, NDF, nmat, nszf, nband, NCN
FOR N = 1 TO nszf
  IF INKEY$ = CHR$(27) THEN END
  PRINT ".";
  i = N
  FOR l = 2 TO nband
    i = i + 1
    IF sk(N, l) <> 0 THEN
      C = sk(N, l) / sk(N, 1)
      J = 0
      FOR K = l TO nband
        J = J + 1
        IF sk(N, K) <> 0 THEN sk(i, J) = sk(i, J) - C * sk(N, K)
      NEXT K
      sk(N, l) = C
      R1(i) = R1(i) - C * R1(N)
    END IF
  NEXT l
  R1(N) = R1(N) / sk(N, 1)
NEXT N
N = nszf
DO
  N = N - 1
IF N <= 0 THEN GOTO 2400
  l = N
  FOR K = 2 TO nband
    l = l + 1
    IF sk(N, K) <> 0 THEN R1(N) = R1(N) - sk(N, K) * R1(l)
  NEXT K
LOOP

2400
END SUB
```

```
SUB STIFFNESS (N)
REM
REM Stiffness Generation
REM
SHARED np, ne, NB, NLD, NDF, nmat, nszf, nband, NCN, MODETRAIT, ALPHA, VERI
stif:
i = nop(N, 1)
j = nop(N, 2)
K = nop(N, 3)
l = IMAT(N)
AJ = CORD(J, 1) - CORD(i, 1)
AK = CORD(K, 1) - CORD(i, 1)
BJ = CORD(J, 2) - CORD(i, 2)
BK = CORD(K, 2) - CORD(i, 2)
area = (AJ * BK - AK * BJ) / 2
IF area = 0 THEN CALL erreur(N)
IF area < 0 THEN
  SWAP nop(N, 1), nop(N, 2)
  GOTO stif
END IF
a(1, 1) = BJ - BK
a(1, 2) = 0
a(1, 3) = BK
a(1, 4) = 0
a(1, 5) = -BJ
a(1, 6) = 0
a(2, 1) = 0
a(2, 2) = AK - AJ
a(2, 3) = 0
a(2, 4) = -AK
a(2, 5) = 0
a(2, 6) = AJ
a(3, 1) = AK - AJ
a(3, 2) = BJ - BK
a(3, 3) = -AK
a(3, 4) = BK
a(3, 5) = AJ
a(3, 6) = -BJ
CALL detmodules(l, forcep(N, 1), forcep(N, 2), forcep(N, 3), ort(l, 1), ort(l, 3), nu(l), VERI)

IF N = 1 AND ALPHA = 2 THEN PRINT #3, forcep(N, 1); ort(l, 1), ort(l, 3), ort(l, 2)
ort(l, 4) = ort(l, 2)
ort(l, 5) = (ort(l, 1) * ort(l, 3)) / (ort(l, 1) + ort(l, 3) + (ort(l, 2) + ort(l, 4)) * ort(l, 3))

g2 = ort(l, 5)
'IF g2 = 0 THEN DO: LOOP UNTIL INKEY$ <> ""
NN = ort(l, 1) / ort(l, 3)
M = g2 / ort(l, 3)
IF MODETRAIT = 1 THEN
  COMM = ort(l, 3) / ((1 - NN * ort(l, 4) ^ 2) * area)
  ESTIFM(1, 1) = COMM * NN
  ESTIFM(1, 2) = COMM * NN * ort(l, 4)
  ESTIFM(1, 3) = 0
```

```
  ESTIFM(2, 1) = COMM * NN * ort(l, 4)
  ESTIFM(2, 2) = COMM
  ESTIFM(2, 3) = 0
  ESTIFM(3, 1) = 0
  ESTIFM(3, 2) = 0
  ESTIFM(3, 3) = COMM * M * (1 - NN * ort(l, 4) ^ 2)
ELSE
  COMM = ort(l, 3) / ((1 + ort(l, 2)) * (1 - ort(l, 2) - 2 * NN * ort(l, 4) ^ 2) * area)
  ESTIFM(1, 1) = COMM * NN * (1 - NN * ort(l, 4) ^ 2)
  ESTIFM(1, 2) = COMM * NN * ort(l, 4) * (1 + ort(l, 2))
  ESTIFM(1, 3) = 0
  ESTIFM(2, 1) = COMM * NN * ort(l, 4) * (1 + ort(l, 2))
  ESTIFM(2, 2) = COMM * (1 - ort(l, 2) ^ 2)
  ESTIFM(2, 3) = 0
  ESTIFM(3, 1) = 0
  ESTIFM(3, 2) = 0
  ESTIFM(3, 3) = COMM * M * (1 + ort(l, 2)) * (1 - ort(l, 2) - 2 * NN * ort(l, 4) ^ 2)
END IF
FOR i = 1 TO 3
  FOR J = 1 TO 6
    b(i, J) = 0
    FOR K = 1 TO 3
      b(i, J) = b(i, J) + ESTIFM(i, K) / 2 * a(K, J)
NEXT K, J, i
PRINT #2, N
FOR J = 1 TO 6
  PRINT #2, b(1, J), b(2, J), b(3, J)
NEXT J
FOR i = 1 TO 6
  FOR J = 1 TO 6
    ESTIFM(i, J) = 0
    FOR K = 1 TO 3
      ESTIFM(i, J) = ESTIFM(i, J) + b(K, i) / 2 * a(K, J)
NEXT K, J, i
END SUB

SUB STRESS
REM
REM stress
REM
SHARED np, ne, NB, NLD, NDF, nmat, nszf, nband, NCN, boucle, ALPHA, CTRL
DIM force(ne, 3), DIS(2, np)

FOR M = 1 TO np * 2 STEP 2
  PRINT #1, R1(M), R1(M + 1)
'  PRINT R1(M), R1(M + 1)

  DIS(1, (M + 1) / 2) = R1(M)
  DIS(2, (M + 1) / 2) = R1(M + 1)
NEXT M
FOR NC = 1 TO ne
  INPUT #2, N
FOR J = 1 TO 6
  INPUT #2, b(1, J), b(2, J), b(3, J)
```

```
NEXT J
FOR i = 1 TO NCN
  M = nop(N, i)
  IF M = 0 GOTO 2680
  K = (i - 1) * NDF
  FOR J = 1 TO NDF
    IJ = J + K
    R(IJ) = DIS(J, M)
2680
NEXT J, i
IA = K + NDF
FOR i = 1 TO 3
force(N, i) = 0
  FOR J = 1 TO IA
    force(N, i) = force(N, i) + b(i, J) * R(J)
  NEXT J, i
NEXT NC
boucle = 1
meanerr = 0
compteur = 0
FOR N = 1 TO ne
  C = (force(N, 1) + force(N, 2)) / 2
  a = SQR(((force(N, 2) - force(N, 1)) / 2) ^ 2 + force(N, 3) ^ 2)
  SMAX = C + a
  SMIN = C - a
  IF ABS(1 - (ABS(forcep(N, 1)) + ABS(forcep(N, 2))) / (ABS(force(N, 1)) + ABS(force(N, 2)))) > .005
                THEN
    boucle = 0
    COLOR 28
    compteur = compteur + 1
  ELSE
    COLOR 15
  END IF
  REM PRINT USING " #### sx, ##########, sxp, ##########, sz, ##########, szp,
              ##########,"; N; force(N, 1); forcep(N, 1); force(N, 2); forcep(N, 2)
  IF force(N, 2) = SMIN GOTO 2840
  ANG = 57.29578 * ATN(force(N, 3) / (force(N, 2) - SMIN))
  PRINT ".";
  GOTO 2850
```