الجمهورية الجزائرية الديمقراطية الشعبية
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم و البحث العلمي
MINISTERE DE L'ENSEIGNEMENT ET DE LA RECHERCHE SCIENTIFIQUE

ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT GENIE CIVIL

PROJET DE FIN D'ETUDES

SUJET

PROGRAMME D'ELEMENTS FINIS SUR MICROORDINATEUR

Proposé par :
L. T. P. C.

Etudié par :
TELMAT Zine Eddine.
HADJ-CHERIF Mahdjoub.

Dirigé par :
M^r. BARAKA.

PROMOTION : Juin 1986

E.N.P. 10, Avenue Hacen Badi – EL-HARRACH – ALGER

DEDICACES

A MON PERE A QUI JE DOIS TOUT

A MA MERE

A MES FRERES ET SOEURS

A TOUS CEUX QUI ME SONT CHERS

ZINE - EDDINE

A MON PERE

A MA MERE

A MES FRERES ET SOEURS AINSI QU'A TOUS MES AMIS.

MAHDJOUB

R E M E R C I E M E N T S

On tient à remercier avant tout Monsieur BARAKA, pour nous avoir inspiré ce sujet et pour l'aide que nous avons toujours trouvée auprès de lui.

Nous remerçions Messieurs BENDALI, OUAMEUR, CHERMOUTI ainsi que tous les responsables au niveau de L.T.P.C et ceci pour nous avoir donné les meilleures conditions de travail au sein de l'Entreprise.

Nous remerçions également tous les enseignants ayant participé à notre formation.

S O M M A I R E

Présentation du travail

Cette étude présente un programme en langage fortran de résolution de problèmes divers dont :

- Calcul des déplacements et des contraintes en élasticité linéaire (éléments réctangulaires à 3 noeuds et 2 degrés de liberté par noeud).

Calcul des pressions interticielles dans le cas de présence d'eau.

Ces calcul ont été faits dans les 3 cas suivants :

Contraintes planes, déformation planes et axisymétrie.

- Poutres tridimensionnelles en élasticité linéaire
- Poutres tridimensionnelles en plasticité

L'outil de travail étant un mini ordinateur de type olivetti M 24.

PRESENTATION DE LA METHODE DES ELEMENTS FINIS

Principe

La methode des éléments finis permet de resoudre des problèmes de mecanique des milieux continus en discretisant le milieu à étudier, c'est à dire en le considérant composé d'un grand nombre de petites parties dites elements finis, possedant chacune ses propres fonctions pour décrire contraintes et déformations, et dont l'assemblage représente au mieux le milieu réel.

On choisit ces fonctions de façon qu'elles assurent la continuité du comportement dans l'ensemble du milieu.

Type d'éléments

1/ - élément barre

Sert à décrire les poutres entreillis et les ossatures à 2 ou 3 dimenssions.

combiné avec l'élement plaque, il peut representer des tirants de raidissement.

2/ - éléments plaques minces chargées dans leur plan
l'élément triangle et quadrilatère sont des formes géométriques de cette classe.

3/ - éléments volumiques

Analyse globale

Il existe plusieurs formulations pour construire les equations albebriques de representation globale par éléments finis.

- methode des déplacements
- methode des forces
- methodes mixtes

L'utilisation de l'une ou l'autre de ces methodes depend du choix des inconnues à approcher.

En ce qui concerne notre travail, on a utilisé la méthode des déplacements qui présente une plus grande souplesse et une grande simplicité pour l'application des déplacements admissibles.

Après avoir effectué un maillage adequat, on considère les déplacements aux noeuds de ce maillage comme inconnues, et nous approcherons au sein de chaque élément le champs des déplacements par des fonctions d'interpolation polynomiales assurant la continuité des déplacements generalisés aux noeuds ou à l'interplace de deux éléments.

Formulation des éléments

Il existe deux méthodes de formulations des equations forces déplacements des élements :

La méthode directe et la méthode variationnelle

1/ - Méthode directe

Pour cette méthode, on formule les élements en combinant directement les trois equations d'élasticité :

- equation d'équilibre
- equations deformations - déplacements
- equations intrinseques du matériau

Avant de donner de plus amples détails sur cette méthode, arrêtons-nous un peu, pour expliciter les trois equations d'élasticité citées ci-dessus.

Equations differentielles de base de l'élasticité lineaire

En elasticité lineaire, on fait deux hypothèses

- les deformations sont très petites
- les lois de comportement contraintes - deformations sont lineaires

La théorie de l'élasticité peut se résumer en trois ensembles d'équations :

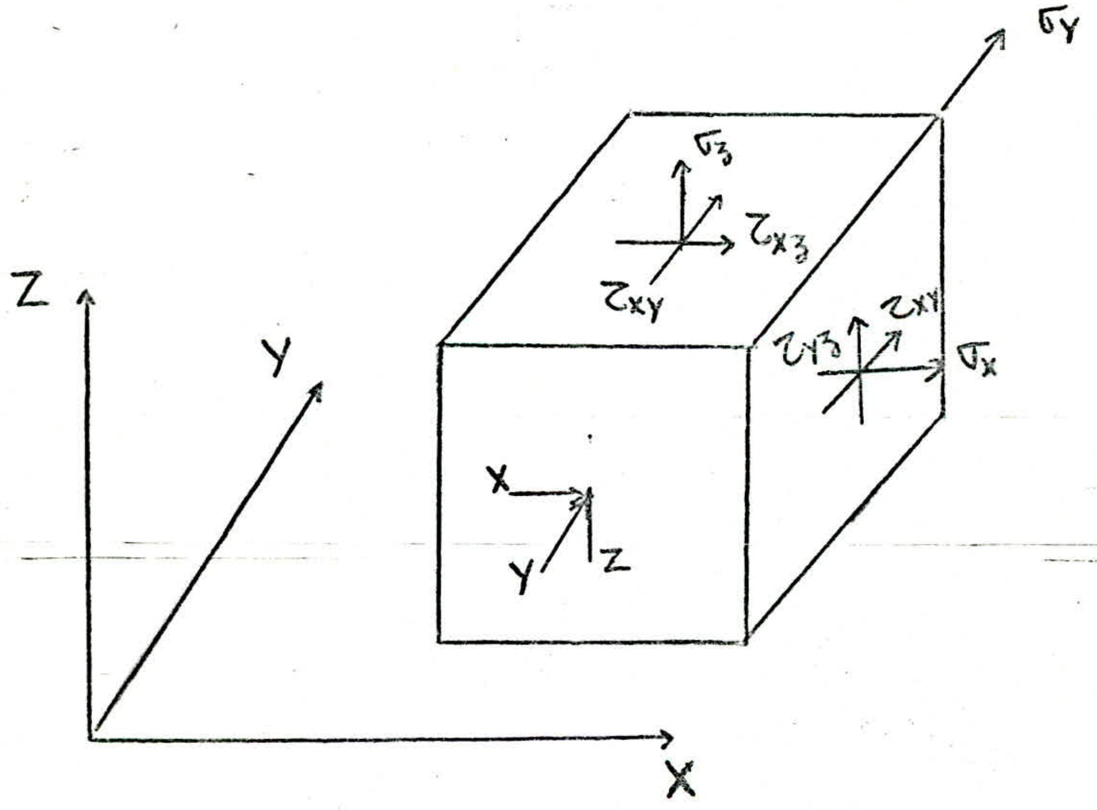
1- equations différentielles de l'équilibre

2- equations differentielles deformations - deplacements

3- lois intrinseques des matériaux

La combinaison de toutes ces equations permet de former les equations differentielles qui regissent le comportement du solide

1 - Equations differentielles de l'équilibre



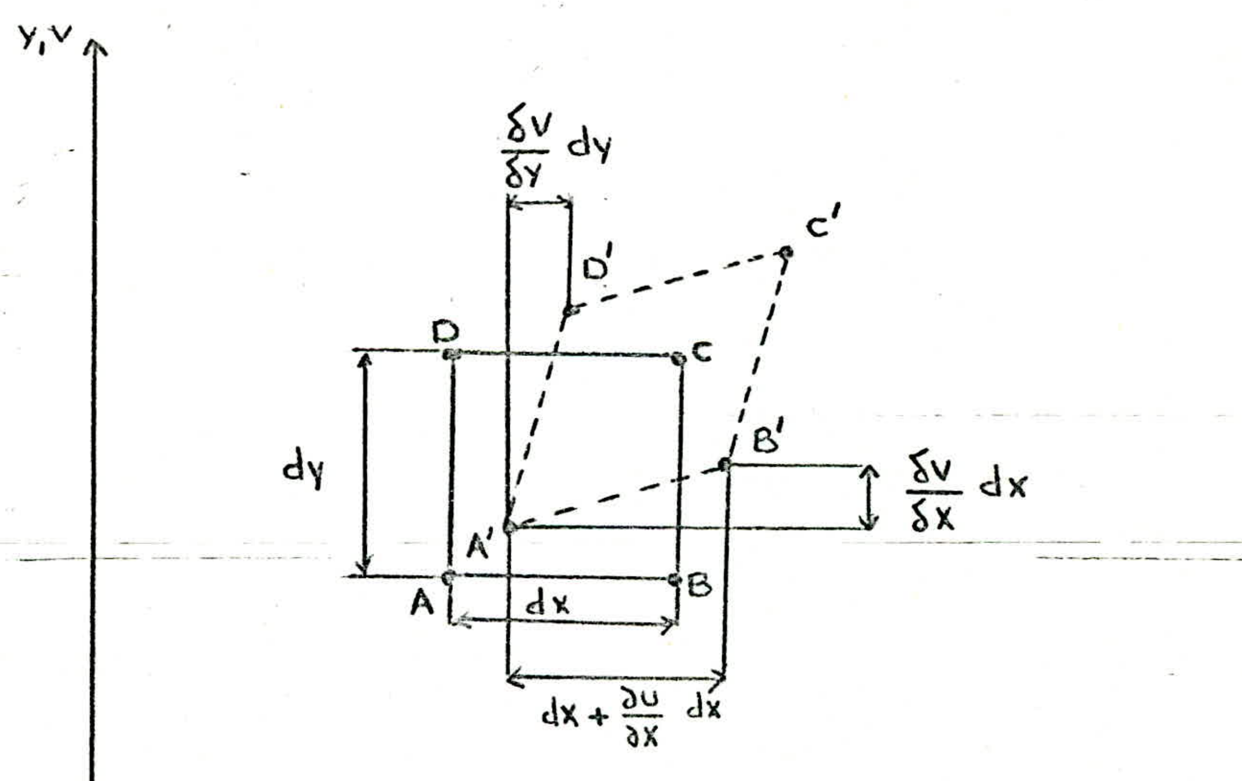
Pour un élément de volume

$$\frac{\delta \sigma_x}{\delta x} + \frac{\delta \tau_{xy}}{\delta y} + \frac{\delta \tau_{xz}}{\delta z} + X = 0$$

$$\frac{\delta \sigma_y}{\delta y} + \frac{\delta \tau_{xy}}{\delta x} + \frac{\delta \tau_{xz}}{\delta z} + Y = 0$$

$$\frac{\delta \sigma_z}{\delta z} + \frac{\delta \tau_{xz}}{\delta x} + \frac{\delta \tau_{xy}}{\delta y} + Z = 0$$

2 - Equations deformations - déplacements



$$\varepsilon_x = \frac{A'B - AB}{AB} \quad \text{avec } AB = dx$$

$$\Rightarrow A'B' = (1 + \varepsilon_x)\, dx$$

en elevant au carré et en eliminant les termes d'ordre supérieur (hypothèse de petites déformations), on aura

$$\varepsilon_x = \frac{\partial U}{\delta x} \qquad \gamma_{xy} = \frac{\delta U}{\delta y} + \frac{\delta V}{\delta x}$$

$$\varepsilon_y = \frac{\delta V}{\delta y}$$

Deformation tangentielle definie comme étant la déformation d'un angle qui est droit avant déformation.

En tridimensionnel, il suffit de completer par les equations suivantes :

$$\varepsilon_z = \frac{\delta W}{\delta z}, \quad \gamma_{xz} = \frac{\delta W}{\delta x} + \frac{\delta V}{\delta z}, \quad \gamma_{yz} = \frac{\delta W}{\delta y} + \frac{\delta V}{\delta z}$$

3 - Equations intrinseques des matériaux

Ces equations caracterisent les propriétées mécaniques du matériau.

Dans l'éssai de traction simple d'une éprouvette cylindrique, la partie linéaire du diagramme contrainte deformation s'exprime algebriquement par la loi de HOOKE

$$\sigma_x = E\,\varepsilon_x$$

et si on suppose qu'on a une certaine deformation initiale ε_x^{ini}, alors nous aurons

$$\sigma_x = E\,\varepsilon_x - E\,\varepsilon_x^{init}$$

Pour un milieu à 3 dimensions

$$\begin{bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{xz} \\ \tau_{yz} \end{bmatrix} = \begin{vmatrix} \lambda+2\mu & \lambda & \lambda & 0 & 0 & 0 \\ \lambda & \lambda+2\mu & \lambda & 0 & 0 & 0 \\ \lambda & \lambda & \lambda+2\mu & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu \end{vmatrix} \times \begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \\ \gamma_{xz} \\ \gamma_{yz} \end{bmatrix}$$

λ et μ Sont les coefficients de lame avec

$$\lambda = \frac{\nu E}{(1-2\nu)(1+\nu)} , \quad \mu = \frac{E}{2(1+\nu)}$$

.E : module de young

V : coefficient de poisson

Pour un milieu à 2 dimensions

- en contrainte plane $\tau_{xy}=0$, $\tau_{yz}=0$, $\sigma_z=0$

$$\begin{bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{bmatrix} = \frac{E}{1-\mu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{bmatrix}$$

- en deformations planes

$$\begin{bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{bmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} (1-\nu) & \nu & 0 \\ \nu & (1-\nu) & 0 \\ 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{bmatrix}$$

en axisymetrie

$$\begin{bmatrix} \sigma_r \\ \sigma_z \\ \sigma_\theta \\ \tau_{rz} \end{bmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} (1-\nu) & \nu & \nu & 0 \\ \nu & 1-\nu & \nu & 0 \\ \nu & \nu & 1-\nu & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \begin{bmatrix} e_r \\ e_z \\ e_\theta \\ \gamma_{rz} \end{bmatrix}$$

Etapes de la methode directe F = KU

1 - On exprime le champs des déplacements de l'élement en fonction d'un nombre fini de variables $\{a\}$ de preference les degrés de liberté $\{\Delta\}$ aux noeuds de l'élement.

2 - On exprime le champs ε des deformations de l'élement en fonction des degrés de liberté $\{\Delta\}$ en derivant le champ des déplacements conformement aux equations déformations deplacements de l'élasticité

3 - On introduit la loi intrinseque du matériau $\sigma = [E]\,\varepsilon$ pour établir la relation entre le champ σ et les degrés de liberté $\{\Delta\}$.

4 - On construit les equations decrivant les forces $\{F\}$ aux noeuds de l'élement comme fonction du champs de contraintes σ on obtient ainsi la relation entre $\{F\}$ et $\{\Delta\}$ qui constitue par definition les equations de rigidité de l'élement.

Methodes variationnelles

Les méthodes variationnelles ou de l'énergie, constituent en mécanique des structures une approche puissante, très utilisée pour la formulation d'équations d'éléments.

Il existe plusieurs principes variationnels.

1 - Principe des travaux virtuels.

Le principe des travaux virtuels constitue le fondement des principes variationnels.

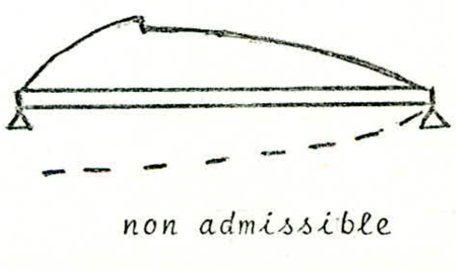
En fait, le principe des travaux virtuels constitue en lui même un moyen de formulation des equations d'éléments finis, ses 2 formes les plus courantes sont celles des déplacements virtuels et des forces virtuelles qui menent respectivement aux principes classiques de l'énergie potentielle stationnaire et de l'énergie complementaire stationnaire.

Déplacements virtuels.

Considerons un corps en équilibre sous des forces de volume et des forces extérieures, et qui subit un déplacement virtuel (fictif) dont le champ de déplacement peut être decrit par des composantes δV. δV. et δW en chaque point. Ce déplacement virtuel devra être cinematiquement admissible c'est à dire enprimable par des fonctions continues des coordonnées de l'espace, et qui satisfont aux conditions cinématiques de frontières là où celles-ci existent.

Exemple

Admissible



non admissible

Sous les hypothèses enoncées precedemment, le principe des travaux virtuels stipule que dans un déplacement virtuel δD, la somme de la variation depotentiel δV des forces appliquées et de la variation δU de l'énergie de deformation interne est égal à zéro :

$$\delta U + \delta V = 0$$

avec

$$\delta U = \int_{Vol} \sigma . \delta \varepsilon . d(Vol) \quad \text{et} \quad \delta V = [\delta \Delta]\{F\}$$

Discretisation du travail virtuel par les éléments finis

Le but est la construction d'une procédure générale de formulation de la matrice de rigidité d'un élément fini.

$$\Delta = [N]\{\Delta\}$$

où N : matrice des fonctions d'interpoletion en appliquant des equations deformations déplacements adequates :

$$\varepsilon = [B]\{\Delta\}$$

On a aussi : avec L : matrice operateur differentiel

$$\delta\Delta = [N]\{\delta\Delta\} \quad \text{et} \quad \delta\varepsilon = [B]\{\delta\Delta\} \qquad [B] = [L][N]$$

Appliquons le principe des déplacements virtuels dans le cas général où il existe des forces de volume dans ce cas les equations intrinseques du matériau s'écrivent :

$$\sigma = [D]\{\varepsilon\}$$

σ : tenseur contraintes

ε : tenseur deformations

le travail des efforts internes s'écrit

$$\delta U = \int_{vol} \sigma \, \delta\varepsilon \, d(vol)$$

$$\Rightarrow \delta U = \int_{vol} \varepsilon [D] \, \delta\varepsilon \, d(vol)$$

afin d'obtenir des formes discretisées on a :

$$\varepsilon = [B]\{\Delta\} \quad \text{et} \quad \delta\varepsilon = [B]\{\delta\Delta\}$$

$$\Rightarrow \delta U = [\delta\Delta][K]\{\Delta\}$$

$$\text{avec } [K] = \left[\int_{vol} [B]^T [D][B]\, d(vol)\right]$$

passons maintenant à la variation d'énergie potentielle des forces extérieures.

$$\delta V = [\delta\Delta]\{F\}$$

Pour tenir compte des forces de volume, il faut completer la variation du potentiel des forces appliquées δV par l'intégrale.

$$-\int_{vol} \delta\Delta . X\, d(vol) \quad \text{avec} \quad X = \begin{Bmatrix} X \\ Y \\ Z \end{Bmatrix}$$

$$\delta\Delta = [N]\{\delta\Delta\}$$

$$\Rightarrow \delta V = -[\delta\Delta]\left([N]^T F - F^b\right)$$

$$\text{avec } F^b = \int_{vol} [N]^T [X]\, d(vol)$$

Comme $X = -[\rho]\ddot{\Delta}$ avec $\ddot{\Delta} = [N]\{\ddot{\Delta}\}$

de sorte que $\{F^b\} = -[m]\{\ddot{\Delta}\}$

avec $[m] = \int_{Vol} [N]^T [\rho][N] \, d(vol)$

matrice de masse

en écrivant $\delta U + \delta V = 0$ on a :

$$[\delta\Delta]\{-[m]\{\ddot{\Delta}\} + \{F\}\} = [\delta\Delta]\{[k]\{\Delta$$

d'où

$$\boxed{\{F\} = [k]\{\Delta\} + [m]\{\ddot{\Delta}\}}$$

ANALYSE GLOBALE

Il existe 3 methodes pour construire les equations algebriques d'une representation globale par éléments finis :

1 - Methode des déplacements

- equation de rigidité

$$\{F\} = [K]\{\Delta\}$$

2 - Méthode des forces
- equation de souplesse

$$\{\Delta\} = [f] \cdot \{F\}$$

3 - Methode mixte

- equations mixtes forces - déplacements

$$\begin{Bmatrix} F_f \\ \Delta_f \end{Bmatrix} = [\Omega] \begin{Bmatrix} F_s \\ \Delta_s \end{Bmatrix}$$

On utilisera la méthode des déplacements qui est la plus simple et la plus puissante. Les inconnues étant les déplacements aux noeuds.

Méthodes des déplacements

Pour cette méthode, l'assemblage des matrices de rigidité élémentaires se fait en chaque noeud de la structure.
On obtient ainsi la matrice de rigidité globale à l'état libre.

$$\{F\} = [K]_{global} \{\Delta\}$$

Puis on introduit les conditions d'appuis

Mais à ce stade la résolution ne peut se faire en raison des mouvements de corps rigides qui ne sont pas bloqués ce qui rend la matrice de rigidité globale singulière; on doit donc on doit déterminer une matrice de rigidité qui tient compte des conditions d'appuis, ce qui nous permet la résolution du système.

En résumé, les différentes étapes de la méthode des éléments finis sont :

1 - Idealisation de la structure à étudier

2 - Discretisation de la structure au moyen d'un maillage adequat, avec pour inconnues les déplacements aux noeuds.

3 - Calcul des matrices de rigidité élémentaires

4 - Introduction des conditions aux limites

6 - Résolution

PROGRAMMATION

Etapes d'un programme d'éléments finis

1/ lecture, vérification et organisation des données décrivant le maillage (noeuds et éléments), les paramètres physiques, les sollicitations et conditions aux limites

2/ constructions des matrices et vecteurs élémentaires, puis assemblage.

3/ resolution du système d'equations après prise en compte des condition aux limites

4/ impression des resultats après calcul eventuel de variables additionnelles (gradients, contraintes...)

PRESENTATION DU PROGRAMME

Le programme B.B.M.E.F. que nous developperons et dont nous chargerons certains éléments est le programme établi dans touzot et DHATT écrit en langage fortian

Il traite des problèmes à deux dimensions avec un degré de liberté par noeud, et il utilise des élements triangulaires a 3 noeuds

Le programme est structuré de la manière suivante :

1/ - Organisation des données

- création des tables de coordonnées des connectivités et de localisation des noeuds liés

2/ - Lecture des Forces concentrées aux noeuds sollicités.

3/ - Formation de la matrice de rigidité élementaire

4/ - Assemblage des élements

5/ - Resolution du système

6/ - Impression des résultats

Le programme présente un programme principal et cinq subrontures
subroutines

- BBMEF (P.P.)

- grille

- locef

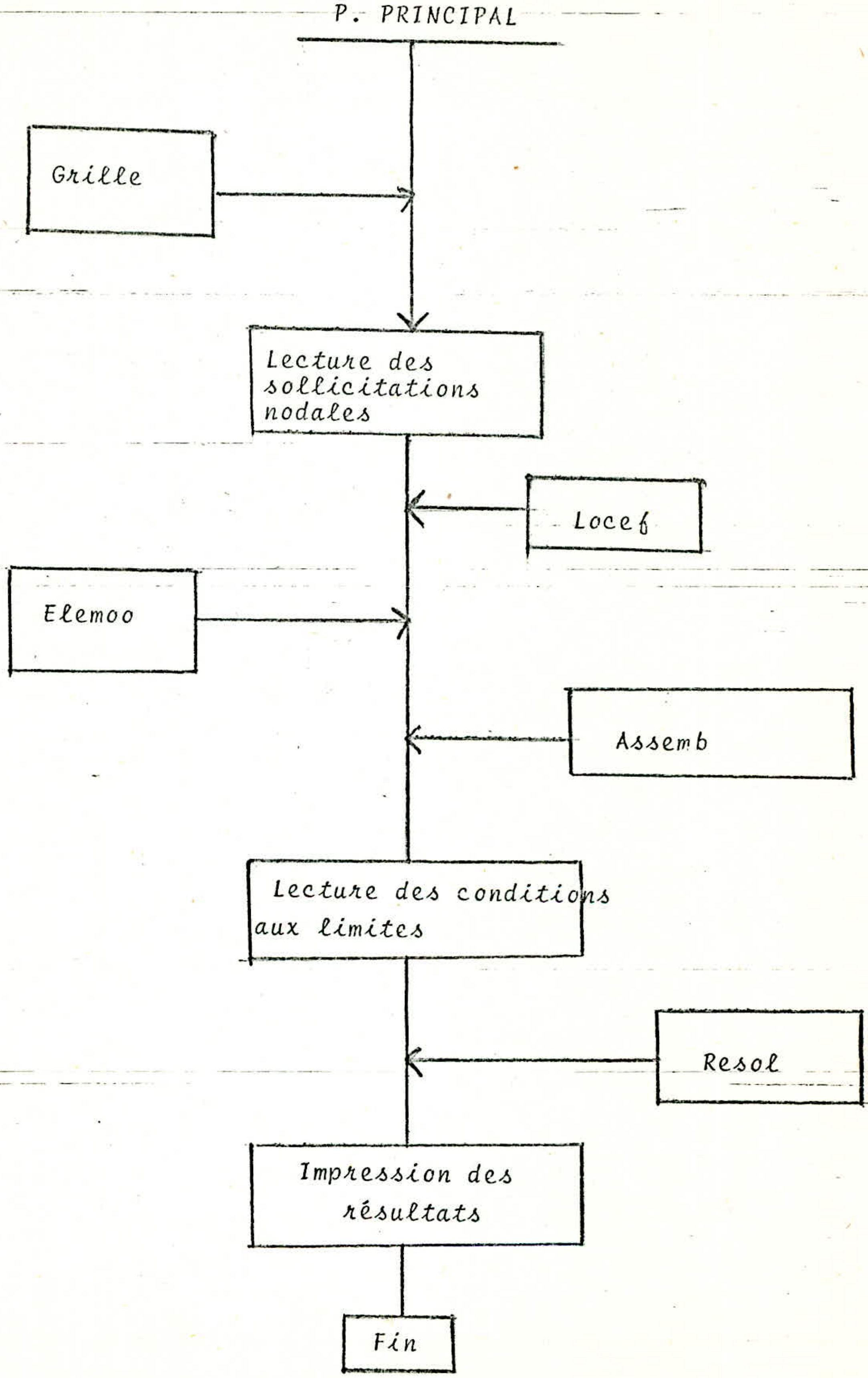
- élememoo
- assemb
- resol

leur fonctions sont présentés sur le tableau ci-dessous

Subroutines - Fonctions

NOM	FONCTION
B B M E F (P.P.)	- Lecture des sollicitations concentrées - Lecture des conditions aux limites - impression des résultats.
GRILLE	Lecture des coordonnées - Lecture des connectivités
LOCEF	- construction de la table de localisation des élements.
ELEMOO	Formation de la matrice de rigidité élementaire
ASSEMB	Assemblage des élements
RESOL	Resolution du système par triangularisation

Organigramme de l'enchainement du programme



Etude d'une structure en elasticité linéaire

(2 degrés de liberté par noeud)

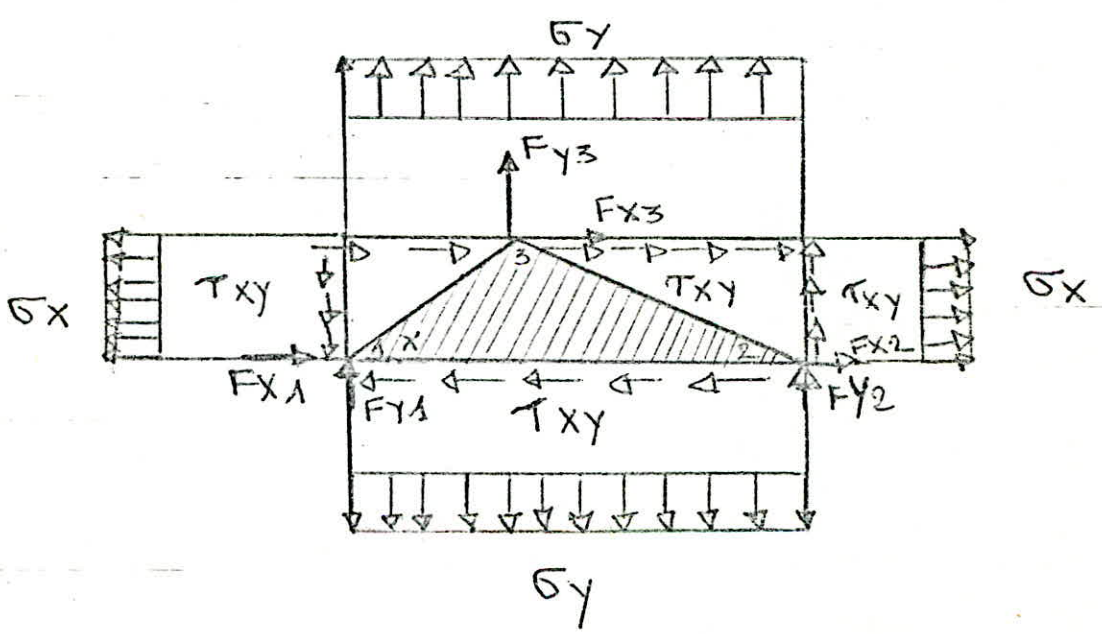
Nous étudierons d'abord le problème d'équilibre

$$[K]\{\Delta\}=\{F\}$$

On remarquera qu'étant donné qu'on est en elesticité lineaire la matrice de rigidité est independante des déplacements. la matrice de rigidité est symetrique
Le problème d'équilibre consiste à calculer les déplacements dans un cas stationnaire à partir de l'equation discretisée :

$$[K]\{\Delta\}=\{F\}$$

Après avoir decomposé notre structure en elements triangulaires agec 3 noeuds par element, on approximera les déplacements par des fonctions d'interpolation.



6 degrés de liberté decrivent le comportement de l'élement

$$\{\Delta\} = [U_1\ V_1\ \ U_2\ V_2\ \ U_3\ V_3]^T$$

et étant donné la symetrie des formules en X et Y on a :

$$U = a_1 + a_2 X + a_3 Y$$

$$V = a_4 + a_5 X + a_6 Y$$

On évalue U et V aux points 1, 2 et 3

$$\begin{cases} U_1 = a_1 + a_2 x_1 + a_3 y_1 \\ U_2 = a_1 + a_2 x_2 + a_3 y_2 \\ U_3 = a_1 + a_2 x_3 + a_3 y_3 \end{cases}$$

$$\begin{cases} V_1 = a_4 + a_5 x_1 + a_6 y_1 \\ V_2 = a_4 + a_5 x_2 + a_6 y_2 \\ V_3 = a_4 + a_5 x_3 + a_6 y_3 \end{cases}$$

En resolvant, on trouve les solutions qu'on peut mettre sous la forme :

$$U = \frac{1}{2\Delta}\Big[(C_{11} + C_{21}x + C_3 y)U_1 + (C_{12} + C_{22}x + C_{32}y)U_2$$
$$+ (C_{13} + C_{23}x + C_{33}y)U_3\Big]$$

$$V = \frac{1}{2\Delta}\Big[(C_{11} + C_{21}x + C_{31}y)V_1 + (C_{12} + C_{22}x + C_{23}y)V_2$$
$$+ (C_{13} + C_{23}x + C_{33}y)V_3\Big]$$

avec : $C_{11} = X_2 Y_3 - X_3 Y_2$ $\quad C_{12} = X_3 Y_1 - X_1 Y_3$

$C_{13} = X_1 Y_2 - X_2 Y_1$ $\quad C_{21} = Y_2 - Y_3$

$C_{22} = Y_3 - Y_1$ $\quad C_{23} = Y_1 - Y_2$

$C_{31} = X_3 - X_2$ $\quad C_{32} = X_1 - X_3$

$C_{33} = X_2 - X_1$

$$\text{et } 2\Delta = \det \begin{vmatrix} 1 & X_1 & Y_1 \\ 1 & X_2 & Y_2 \\ 1 & X_3 & Y_3 \end{vmatrix} = 2A$$

avec A : aire du triangle (l'élement)

on peut donc écrire

$$\{U\} = [N]\{\Delta\} \qquad \{U\} = \begin{Bmatrix} U \\ V \end{Bmatrix}$$

$$U = \frac{1}{2\Delta} \begin{bmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 \end{bmatrix} \begin{Bmatrix} V_1 \\ V_1 \\ V_2 \\ V_2 \\ V_3 \\ V_3 \end{Bmatrix}$$

avec

$$N_1 = (C_{11} + C_{21} X + C_{31} Y)$$
$$N_2 = (C_{12} + C_{22} X + C_{32} Y)$$
$$N_3 = (C_{13} + C_{23} X + C_{33} Y)$$

on a aussi

$$\varepsilon = LU$$

$$\begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{Bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\delta}{\partial y} \\ \frac{\delta}{\delta y} & \frac{\delta}{\delta x} \end{bmatrix} \begin{Bmatrix} U \\ V \end{Bmatrix}$$

$$\text{et } U = [N]\{\Delta\} \Rightarrow \varepsilon = [B]\{\Delta\}$$

$$\text{avec } [B] = [L][N]$$

$$B = \frac{1}{2\Delta} \begin{bmatrix} C_{21} & 0 & C_{22} & 0 & C_{23} & 0 \\ 0 & C_{31} & 0 & C_{32} & 0 & C_{33} \\ C_{31} & C_{21} & C_{32} & C_{22} & C_{33} & C_{23} \end{bmatrix}$$

On cherche la matrice (A) qui relie contraintes et forces aux noeuds

$$\{F\} = [A]\{\Delta\}$$

par exemple

$$F_{x2} = \frac{t}{2}\left[(y_3 - y_2)\sigma_x - (x_2 - x_1)\tau_{xy} + (x_2 - x_3)\tau_{xy}\right]$$

à l'aide de transformation de contraintes aux bords en forces aux noeuds on obtient la matrice (A) et finalement on aura :

$$(K) = (A)\ (D)\ (B)$$

(D) : matrice de raideur

La matrice (K) doit être symetrique, mais l'equation sur laquelle se fonde la méthode directe ne garanti pas cette symetrie. La matrice (D) est symetrique, mais (A) et (B) ne le sont pas necessairement. La difficulté de construction de matrice symetrique peut être surmontée en remplaçant la matrice (A) par la transposée de la matrice (B) ainsi donc on peut calculer la matrice de rigidité (K) par la formule :

$$K = \int_{\Omega} [B]^T [D] [B] \, d\Omega = \Delta . t . [B]^T [D] [B]$$

où Δ s'est l'aire de l'élement

t : epaisseur

La matrice de rigidité élementaire est calculée pour les 3 cas suivants :

1/ - contraintes planes

2/ - deformations planes

3/ - axisymetrie

pour les 2 premiers cas, seule la matrice de raideur (D) change

pour l'anisymetrie (D) et (B) changent avec :

$$B = \frac{1}{2\Delta}\begin{bmatrix} C_{21} & 0 & C_{22} & 0 & C_{23} & 0 \\ 0 & C_{31} & 0 & C_{32} & 0 & C_{33} \\ C_{31} & C_{21} & C_{32} & C_{22} & C_{33} & C_{23} \\ 0 & C_{1} & 0 & C_{2} & 0 & C_{3} \end{bmatrix}$$

$$C_{21} = Y_2 - Y_3$$
$$C_{22} = Y_3 - Y_1$$
$$C_{23} = Y_1 - Y_2$$
$$C_{31} = X_3 - X_2$$
$$C_{32} = X_1 - X_3$$
$$C_{33} = X_2 - X_1$$

$$C_{11} = X_2 Y_3 - X_3 Y_2$$
$$C_{12} = X_3 Y_1 - X_1 Y_3$$
$$C_{13} = X_1 Y_2 - X_2 Y_1$$
$$C_1 = \left((C_{11}/XG) + C_{21} + (C_{31}(YG/XG))\right)$$
$$C_2 = \left((C_{12}/XG) + C_{22} + (C_{32}(YG/XG))\right)$$
$$C_3 = \left([C_{13}/XG] + C_{23} + (C_{33}(YG/XG))\right]$$

(D) est donnée dans la partie "rappel d'élasticité".

Programmation

on a crée une soubroutine (ELEMO2) qui remferme tous les calculs concernant l'élement et c'est dans celle-ci qu'on a introduit la matrice de rigidité élementaire notée (VKE) .

on a introduit directement VKE après avoir fait manuellement le produit (A) (D) (B)

mais finalement on a changé en optant pour le calcul de matrice de rigidité élementaire en introduisant (D) notée DEV et (B) notée VBE.

dans notre programme on étudie 3 cas differents (contraintes planes, deformations planes et axisymetrie) dont la matrice de raideur varie d'un cas à l'autre, tandis que la matrice B est la même pour les 2 premiers cas cités mais a une ligne et une colonne de plus dans le cas de l'axisymetrie.

notre but est de chercher la matrice de rigidité VKE qu'on obtiendrai en effectuant le produit A. t. (B) (D) (B).

on a procédé de la manière suivante :

- on a introduit les matrices de raideurs qui sont d'ordre (3,3) pour deformation plane et contrainte plane et d'ordre (4,4) pour l'axisymetrie, et on a introduit la matrice (VBE) d'ordre (4,6) et suivant le cas on bouche avec un indice I = 3 ou 4
boucle

on fait dabord le produit (D) (B) on obtient ainsi la matrice (DR)

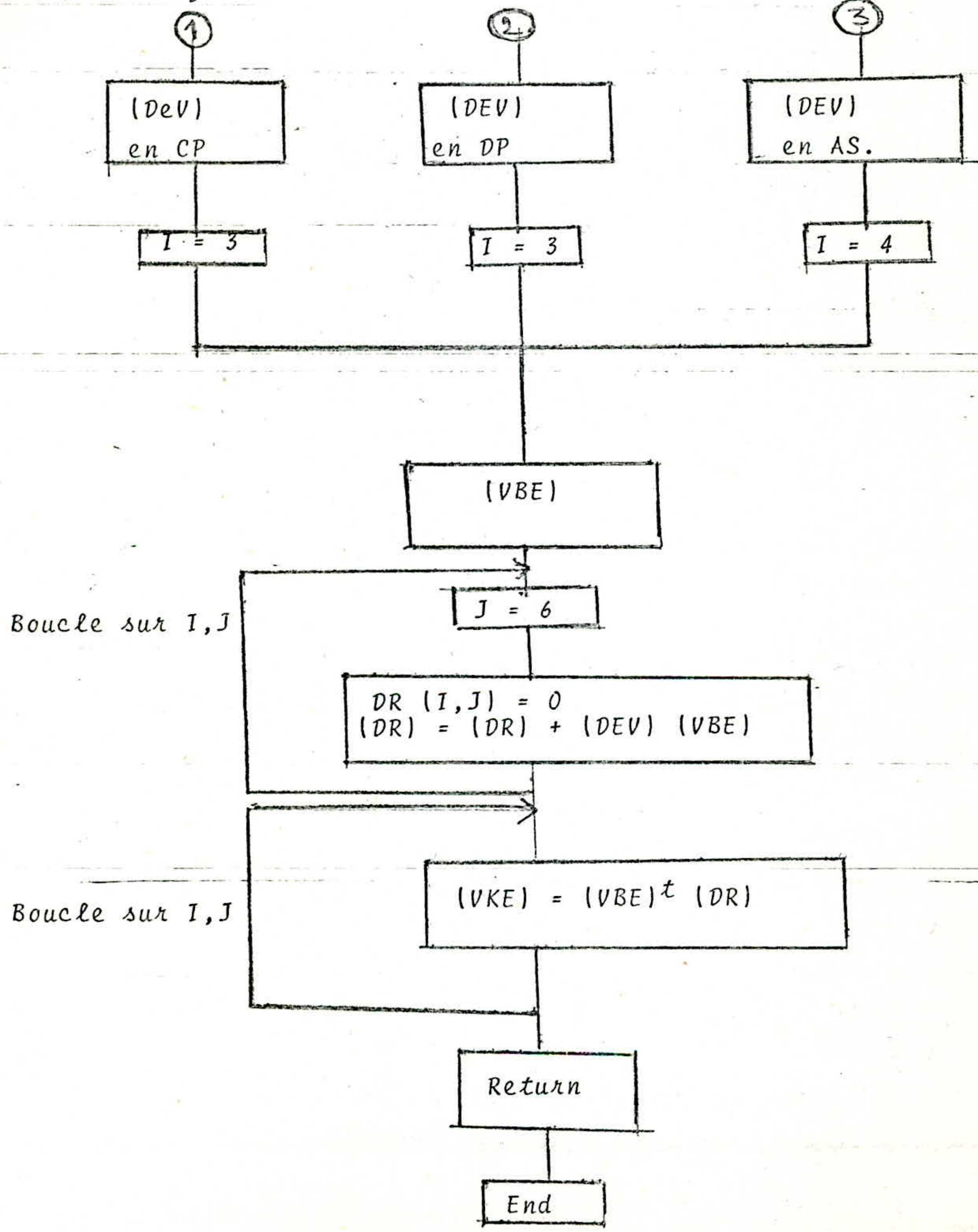
puis on obtient (VKE) = $(B]^t$ (DR)

ORGANIGRAMME

1/ - contraintes planes C P

2/ - deformations planes D P

3/ - axisymetrie A S



2/ ASSEMBLAGE DES MATRICES ELEMENTAIRES

Après avoir determiné toutes les matrices élementaires dans le repère global, il faut les assembler au moyen des conditions d'équilibre en chaque neud. Nous obtiendrons ainsi la matrice de rigidité globale à "l'état libre".

Programmation

Règles d'assemblage :

depuis une totale de localisation LOCE dans une soubroutine LOCEF qui donne la position de chaque terme (Un) dans (Dn) cette table est identique à la table de connectivités définie dans le sous programme GRILLE,

La dimenssion de la table LOCE est égale au nombre de degrés de liberté de l'élement exemple de definition de la table de localisation pour 2 degrés de liberté :

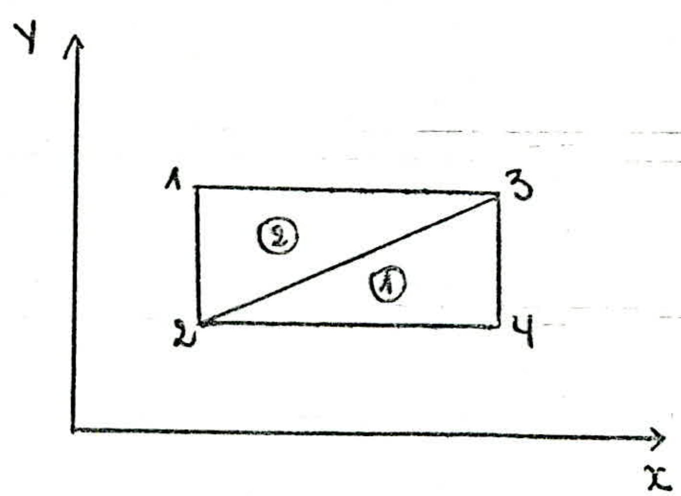


table de connection

élement	noeuds		
1	1	2	4
2	1	4	3

$$\langle \Delta n \rangle = \langle U_1 V_1 \; U_2 V_2 \; U_3 V_3 \; U_4 V_4 \rangle$$

élement 1

$$\langle \Delta n \rangle = \langle U_1 V_1 \; U_2 V_2 \; U_4 V_4 \rangle$$

$$LOCE = \langle 1 \; 2 \; 3 \; 4 \; 7 \; 8 \rangle$$

élement 2

$$\langle \Delta n \rangle = \langle U_1 V_1 \; U_4 V_4 \; U_3 V_3 \rangle$$

$$LOCE = \langle 1 . 2 . 7 . 8 . 5 . 6 \rangle$$

On introduit un sous programme ASSEMB qui assemble les élements à partir de la table de localisation.

L'algorithme général qui effectue les deux étapes de l'assemblage est le suivant :

1/ - initialisation des termes (K) et (F) à zéro

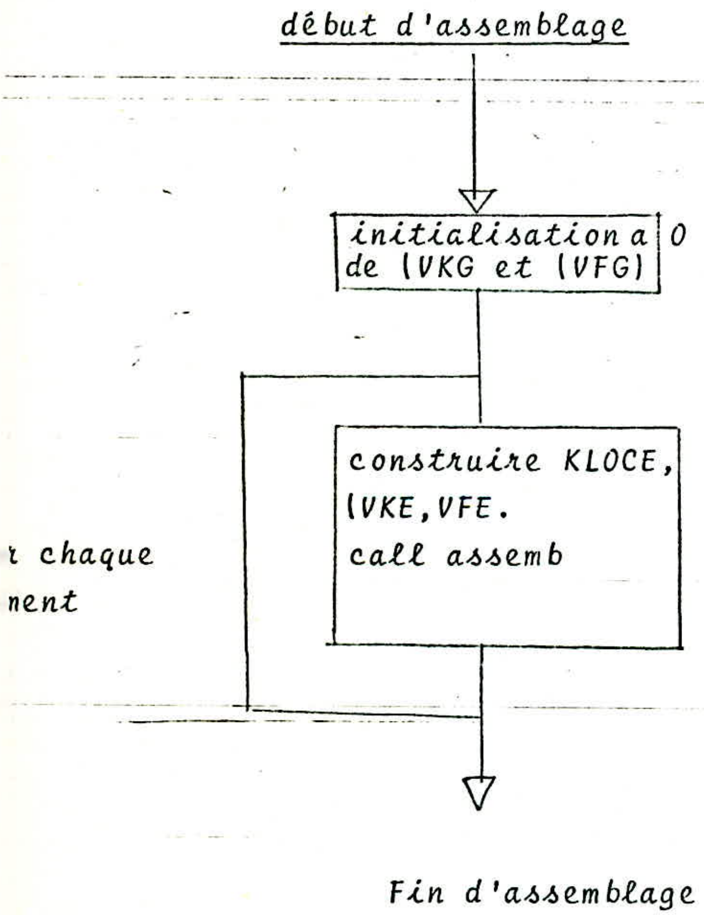
2/ - pour chaque élement

ajouter chaque terme K_{IJ} de sa matrice élementaire au terme de la matrice de ridigité globale.

ajouter chaque terme fi du vecteur élementaire des sollicitations au terme F_I du vecteur global.

$$F_I = F_I + f_i \qquad i = 1, 2 \dots n$$
$$I = LOLE\,(i).$$

Organigramme



3/ - Introduction des conditions aux limites

Pour une structure à n noeuds et étant donné qu'on a 2 degrés de liberté par noeud, on obtiendra un système de 2n equations, avec un nombre d'inconnues
inférieur à 2n en raison des conditions d'appuis. on doit déterminer une matrice de rigidité à l'état liée, qui tienne compte des conditions d'appuis :

On peut introduire les conditions d'appuis par 3 méthodes :

1/ - réarrangement de la matrice K

2/ - par l'intermédiaire du terme diagonal dominant dont le principe est le suivant :

soit le noeud bloqué cad $U_i = \bar{U}_i$

on remplace : K_{ii} par $K_{ii} + \alpha$
F_i par $\alpha \bar{U}_i$

α étant un nombre très grand par rapportaux K_{ij}

3/ - Methode du terme diagonal unitaire

<u>Principe</u> :

Pour un noeud i bloqué :

On remplace son terme diagonal par 1 et on annule la i^{eme} ligne et la i^{eme} colonne.

Donc cette méthode consiste à modifier pour chaque relation le vecteur force (F) puis la matrice de rigidité (K)

$$F_J = F_J - K_{Ji}\,\bar{U}_i \qquad J = 1, 2 \ldots n \quad , J \neq i$$

$$F_i = \bar{U}_i$$

$$K_{iJ} = K_{Ji}$$

$$K_{ii} = 1$$

$$\begin{bmatrix} K_{11} & \cdots & 0 & \cdots & K_{1n} \\ 0 \cdots 0 & \cdots & 1 & \cdots & 0 \\ K_{n1} & \cdots & 0 & \cdots & K_{nn} \end{bmatrix} \begin{Bmatrix} U_1 \\ \vdots \\ U_i \\ \vdots \\ U_n \end{Bmatrix} = \begin{bmatrix} F_1 - K_{1i}\bar{U}_i \\ \vdots \\ \bar{U}_i \\ \vdots \\ F_n - K_{ni}\bar{U}_i \end{bmatrix}$$

On a opté pour cette méthode qui donne plus de precision et qui ne présente pas de problèmes pour la programmation.

Les conditions aux limites sont introduites dans le programme principal ELAST

4/ Resolution

Après obtention de la matrice de rigidité globale et l'introduction des conditions aux limites, on obtient le système :

$$\{F\} = [K]\{\Delta\}$$

et pour la resolution, on utilise la méthode de Gauss qui est une méthode competitive en raison de petit nombre d'opérations qu'elle necessite. Pour un système N X N on a :

$N^3/3$ additions et $N^3/3$ multiplications

Programme

Lorsque la matrice (K) est stokée dans une table à deux dimensions VKE, la méthode de Gauss qu'on introduit dans un sous programme "RESOL", transforme la matrice (K) en une matrice triangulaire suppérieure.
La table VFG contient le vecteur

(F) en entrée et le vecteur (Δn) à la sortie

Contraintes dans les élements

Nous calculerons les contraintes au seins de chaque élement. Dans le cas général nous avons :

$$\sigma = [D]\{\varepsilon\} = [D][B]\{\Delta\}$$

Donc on calcule d'abord pour chaque élement le produit (D) (B), (D) et (B) étants definis dans la partie "calcul des deplacements"

Après avoir trouvé les déplacements aux noeuds, on trouve directement
$\sigma = [D]\ [B]\ \{\Delta\}$

Programmation

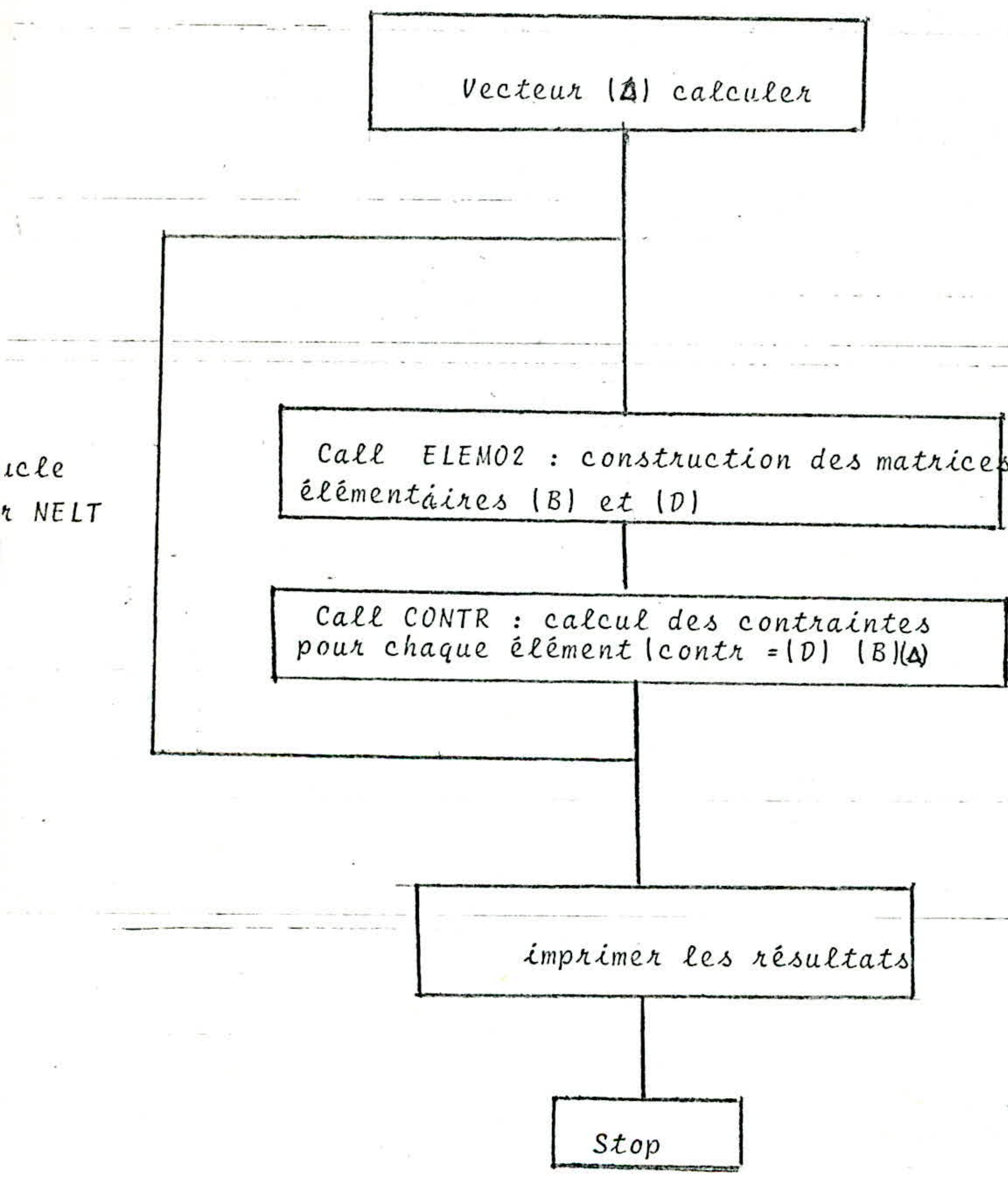
Après avoir calculé les déplacements $\{\Delta\}$, on fait de nouveau appel au sous programme ELEM02 pour nous donner les matrices $[B]$ et $[\Delta]$

Après avoir obtenu les deplacements aux noeuds c'est à dire après la soubroutine Resol, on introduit un sous programme noté CONTR par biais duquel on calcule les contraintes au sein de chaque élement en passant par les étapes suivantes :

On fait le produit Matriciel (VBE) (DEV) qu'on affecte à une matrice notée (DR)

On effectue le produit du vecteur deplacements (D) avec (DR), on obtient ainsi les contraintes notées CONT.

<u>Organigramme de calcul des contraintes</u>



CALCUL DES FORCES

Nous avons introduit 3 types de forces

1 - Forces de volume

2 - Forces concentrées aux noeuds

3 - Forces réparties.

1 - Forces de volumes :

$FG = V.G$

On prend une épaisseur unitaire d'où V = surface de l'élément (triangle) x 1 et $G = \rho \cdot g$.

Pour distribuer cette force sur les noeuds, on suppose que la surface revenant à chaque noeud est égale à 1/3 de la surface de l'élément et puisqu'on a 2 degrés de liberté par noeud donc :

$FGx = 1/6\ B.\ Gx$ $\qquad$ $FGy = 1/6\ B.\ Gy$

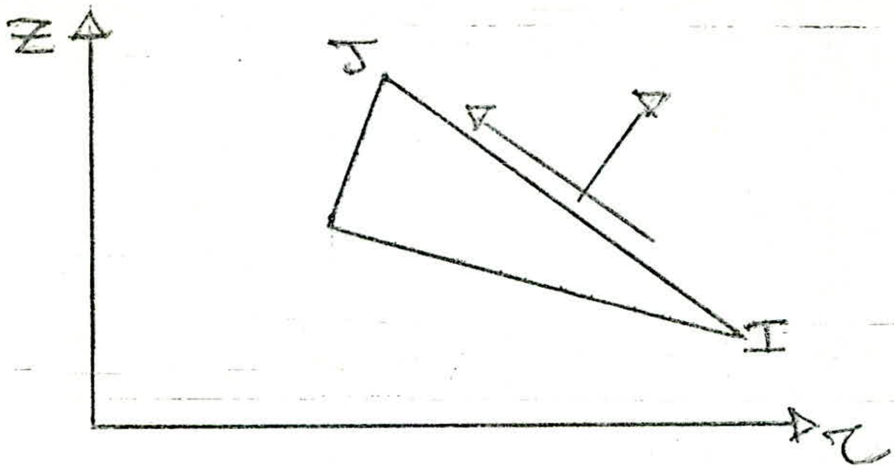
avec :

B : surface de l'élément (triangle)

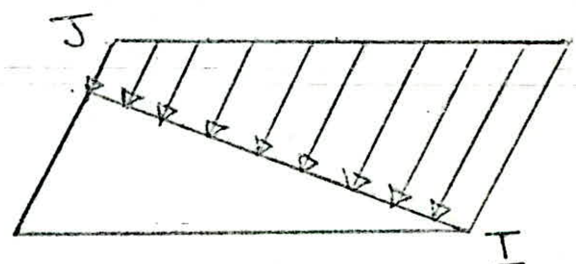
2 - Forces concentrés aux noeuds :

Pour ces forces, on les introduit directement

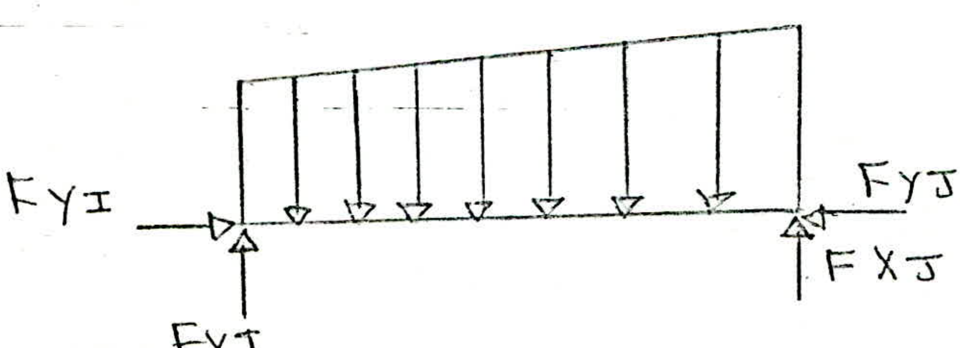
3 - Forces réparties.



Pour le calcul des forces uniformement reparties, on doit transformer les forces sur chaque face en forces normales et tangentielles en chacun des 2 noeuds de la face considérée.



On décompose les forces aux noeuds I et J en forces normales et tangentielle les forces résultantes sont les réactions d'appuis.



$$F_{XI} = F_{NI} \times \cos\alpha - F_{TI} \cdot \sin\alpha$$
$$F_{YI} = F_{NI} \cdot \sin\alpha - F_{TI} \cdot \cos\alpha$$
$$F_{XJ} = F_{NJ} \cdot \cos\alpha - F_{TJ} \cdot \sin\alpha$$
$$F_{YJ} = F_{NJ} \times \sin\alpha + F_{TJ} \cdot \sin\alpha$$

Puis on calcul les forces normales et tangentielles dans le repère global.

Suivant X

en I : $VFG = (FX^{I}/3 + FX^{j}/6) \times L$

en J : $VFG = (FX^{I}/6 + FXJ/3)\ L$

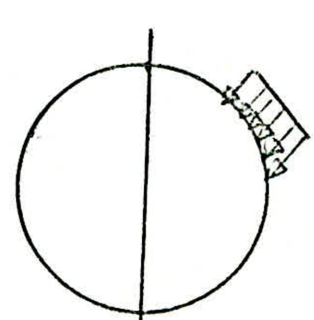
Suivant : Y

en I : $VFG = (FYI/3 + \frac{FYJ}{6} \times L$

en J $VFG = (FYI/6 + FYJ/3) \times L$

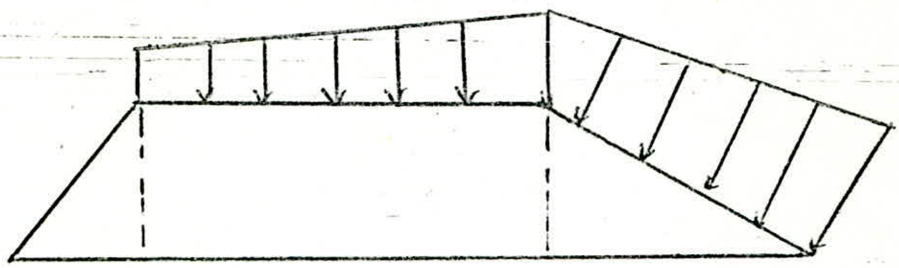
L : longueur de la face

<u>Pour l'axisymetrie</u>



Pour les forces surfacique, dans les cas de déformation planes et contraintes planes, on a pu faire une approximation, en considérant que les forces revenant aux noeuds d'une face considérée, est représentées par les réactions d'appuis, car nous avons des forces trapézoïdales sur un élément plan.

Mais pour *l'axisymetrie*, on a des forces trapézoïdales sur des surfaces trapézoïdales.



On peut décomposer la surface trapézoïdale en 1 surface rectangulaire et 2 surfaces triangulaires. Pour la surface rectangulaire, la distribution des forces se déduit simplement, mais pour les forces trapézoïdales sur les surfaces triangulaires, on peut les calculer à partir des intégrales.

voir zinkevitch annexe 3

Finalement on obtient que :

Pour I

Suivant x

$$VFG\ (2I-1) = \left(FXI \left(\frac{XI}{4} + \frac{XJ}{12} \right) + FXJ\ (XJ + XI/12) \right] 2\pi \times L$$

Suivant y

$$VFG\ (2I) = \left(FYI \left(\frac{XI}{4} + \frac{XJ}{12} \right) + FYJ \left(XJ + \frac{XI}{12} \right) \right] 2\pi \times L$$

Pour J

$$VFG\ (2J-1) = \left(FXI \left(XJ + \frac{X5}{12} \right) + FXJ \left(\frac{XJ}{4} + \frac{XI}{12} \right) \right] 2\pi \times L$$

$$VFG\ (2J) = \left(FYI\ (XI + J/12) + FYJ \left(\frac{XJ}{4} + \frac{XI}{12} \right) \right] 2\pi\ L$$

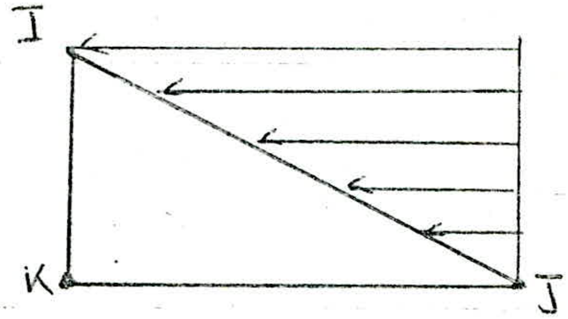
Pour le cas de cylindres : XI = XJ

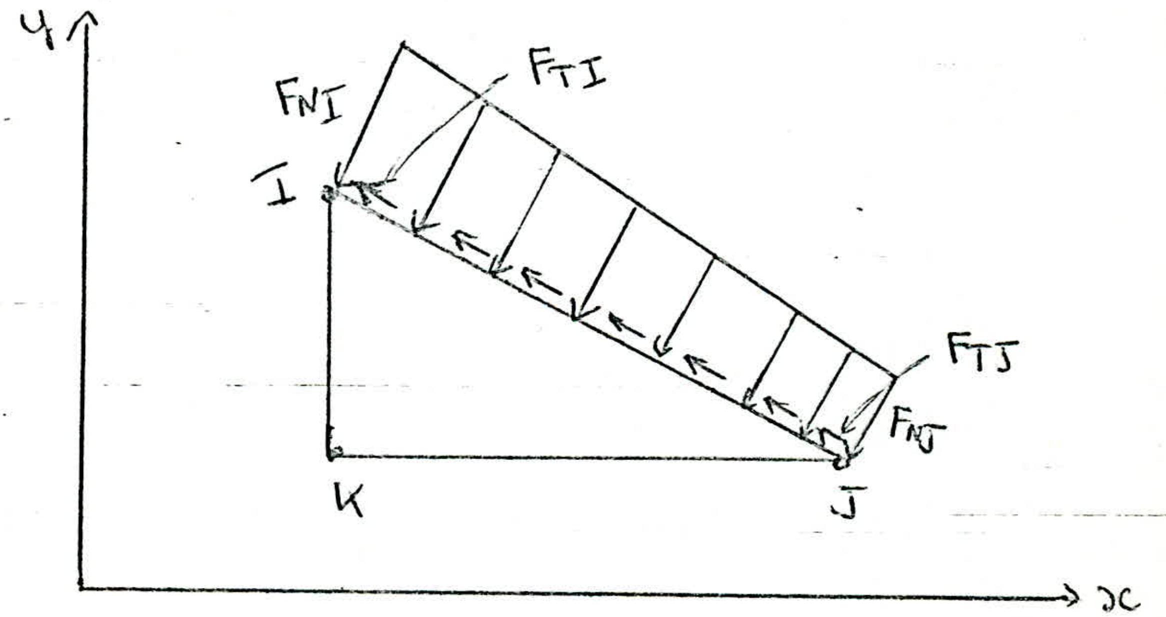
En remplaçant dans les équations précédantes on retrouve les même équations que pour les déformations planes.

Programmation des forces surfacique (uniformément réparties)

Les forces surfaciques introduites dans le programme principal sont de 2 Types

- Charges normales à la face de l'élément (FN)
- Charges tangentes à la face de l'élément (FT)





On boucle d'abord sur le nombre d'éléments chargés en introduisant le numéro de l'élément chargé puis le nombre NF de faces chargées et on boucle sur ce dernier.

Pour chaque façe chargée, on introduit les deux noeuds de celle-ci avec leurs composantes F.N.I. et F.T.J en respectant la convention de signe donnée ci-dessous:

-Donner successivement le premier puis le deuxième noeuds de la façe considérée avec leurs composantes F.N.I. et F.T.I.. Les noeuds sont donnés par ordre, suivant le parcourt de l'élément au moment de l'introduction des connectivitées.

Les charges surfaciques sont transformées en charges concentrées au noeuds I et J en des composantes F.N.I., F.N.J., F.T.J., ensuite, ces composantes sont projetées sur les axes X et Y dans le repère global.

PROGRAMMATION DES FORCES CONCENTREES:

Les forces concentrées sont des forces nodales. On a bouclé sur le nombre de noeuds, et pour chaque noeuds on lit les forces concentrées suivant X et Y notées FX et FY, ces dernières sont introduites directement dans le secteur force global(VFG) AGLORITHME.

AGLORITHME : Lecture du noeud n, Fx, Fy

I = n . NDLN

J = I - 1

VGF (J)=FX

VFG (I)=FY

PROGRAMMATION DES FORCES VOLUMIQUES

Pour la calcul des forces volumique, on avait besoin des poids volumiques (SG) suivant N et Y et cela pour chaque matériau.

On peut avaoir des matériaux différents, et pour cela on a introduit dans GRILLE une lecture du matériau noté MAT (IE) pour chaque élément avec IE variant de 1 à 5.

.../...

Dans le programme principal (ELAST), on introduit pour chaque élément le module de Youg (E), le coéfficient de poisson (μ), GAMAX et GAMAY poids volumiques suivant X et Y.

E	:	noté E
μ	:	noté P
GAMAX	:	noté GX
GAMAY	:	noté GY

Dans ELEMO 2, on introduit 6 vecteurs forces élémentaires notés VFE qui ont pour valeur le $\frac{1}{3}$ du produit surface de l'élément noté B par GX du GY suivant respectivement X et Y.

PRESENCE DE L'EAU

En présence de l'eau, pour chaque élément la structure (le squelette) et lleau subissent la même déformation, donc pour le calcul des déformation, on ajoute à la matrice de rigidité, une matrice diagonale (KW) dont le troisième membre est nul car l'eau n'intervient pas dans le cisaillement

on aura donc: $[D'] = [D] + [K_w]$

$$[K_w] = \begin{bmatrix} K_a & 0 & 0 & 0 \\ 0 & K_a & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & K_a \end{bmatrix} \quad \text{ou} \quad [K_w] = \begin{bmatrix} K_a & 0 & 0 \\ 0 & K_a & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

axysymetrie CP_DP

$$K_a = \frac{E}{3(1-2\nu)} 10^2$$

Calcul des contraintes

$$\sigma = \sigma' + U = D\,\varepsilon$$

$$= (D' + Kw)\,\varepsilon = D'\varepsilon + Kw\,\varepsilon$$

$$\Rightarrow \left[\begin{array}{l} \sigma' = D \\ U = Kw \end{array}\right. \qquad \text{Avec } (D) = (D') - (Kw)$$

$$U = 3\,Kw\,\frac{(\varepsilon x + \varepsilon y)}{3} = Kw\,(\varepsilon x + \varepsilon y)$$

γxy : cisaillement n'apparait pas dans la pression interstielle car la resistance de l'eau au cisaillement est nulle.

$$\begin{pmatrix} U \\ U \\ 0 \end{pmatrix} = \begin{bmatrix} & & \\ & D & \\ & & \end{bmatrix} \begin{bmatrix} \varepsilon x \\ \varepsilon y \\ \gamma xy \end{bmatrix}$$

PROGRAMMATION " PRESENCE D'EAU"

La présence d'eau dans les éléments est prise en compte dans le sous-programme GRILLE, où aprés la lecture des coordonnées, o introduit pour chaque élément un paramètre noté IEAU qui est égal à 1 si l'élément considéré est en présence d'eau, zéro sinon.

Dans ELEMO 2, on a un/noté KP, si KP=1 on a calcul des dé cements, KP=2 pour le calcul des contraintes.

code

on a $D' = [D] + [KW]$

- Si $K_p = 1$ et $IEAU = 1$

  on a $[D'] = [D] + [KW]$

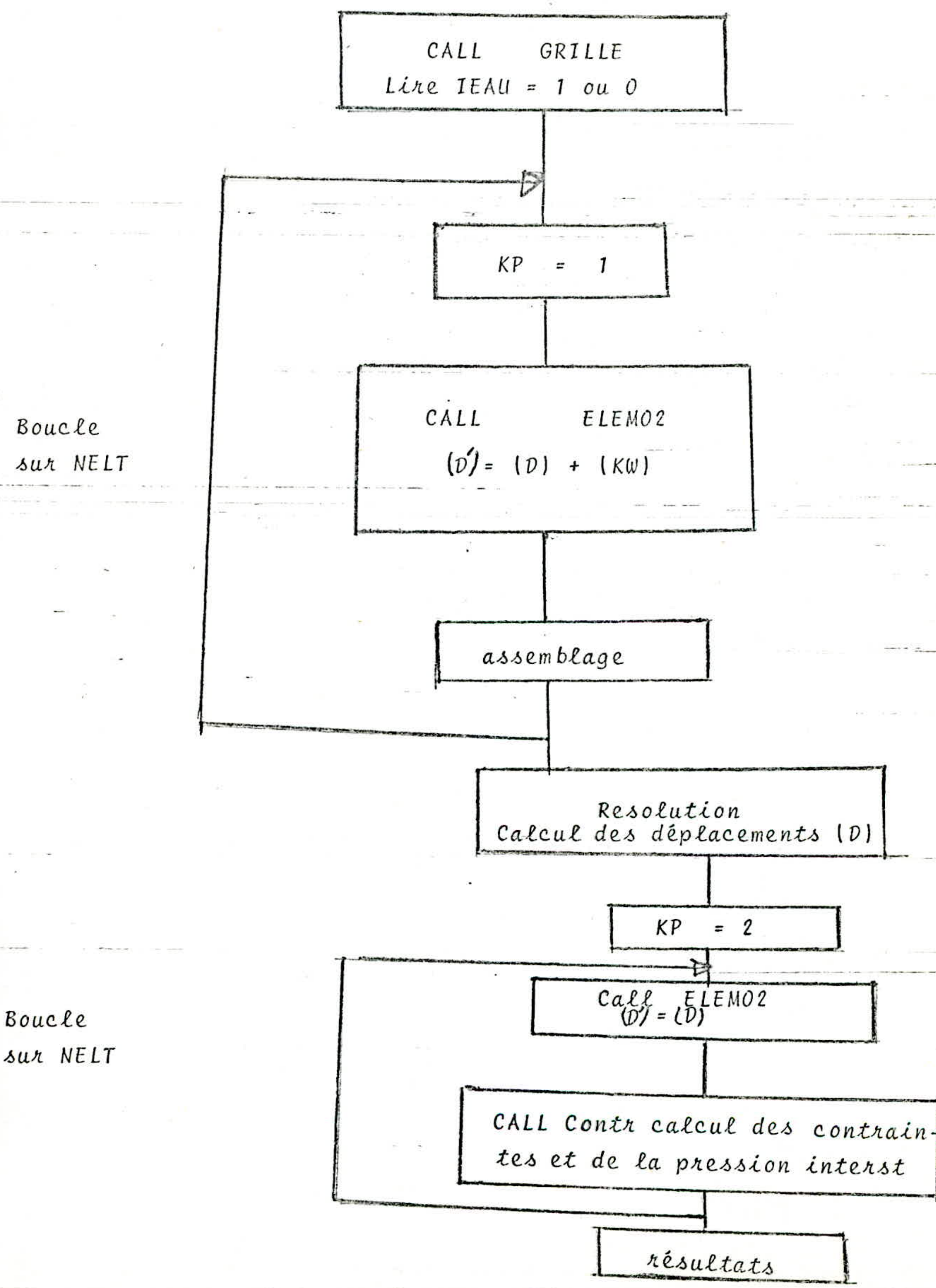
- Si $K_p = 2$ ou $IEAU = 0$

  on a $[D'] = [D]$

Dans la soubroutine (CONTR), aprés le calcul des contrain on calcule aussi la pression interticielle (U) au sein de cahque élé-ment: $\{U\} = [KW][B]\{\Delta\}$

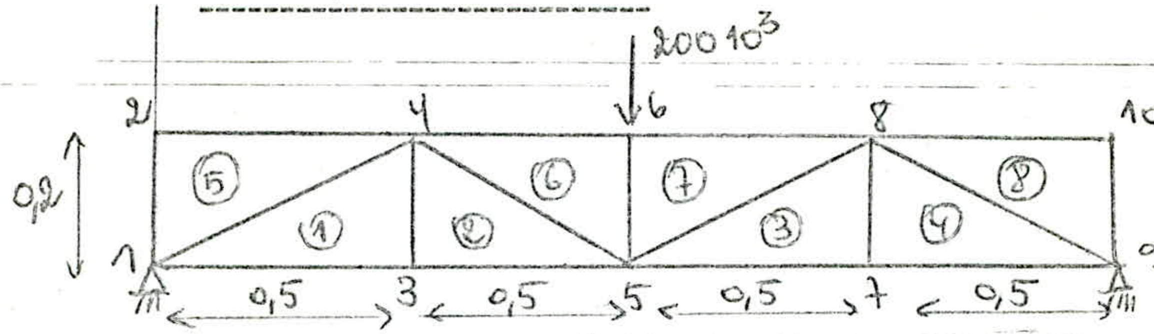
ORGANIGRAMME "PRESENCE D'EAU"

ORGANIGRAMME " PRESENCE D'EAU "



EXEMPLE (CONTRAINTES PLANES)

(1)



noeuds	U (m)		V (m)	
1	0,00		0,00	
2	0,75817	10^{-3}	0,48384	10^{-5}
3	0,11461	10^{-3}	0,19325	10^{-2}
4	0,67592	10^{-3}	0,19804	10^{-2}
5	0,39337	10^{-3}	0,29941	10^{-2}
6	0,39337	10^{-3}	0,29411	10^{-2}
7	0,67213	10^{-3}	0,19325	10^{-2}
8	0,11082	10^{-3}	0,19804	10^{-2}
9	0,78674	10^{-3}	0,00	
10	0,28571	10^{-3}	0,48384	10^{-5}

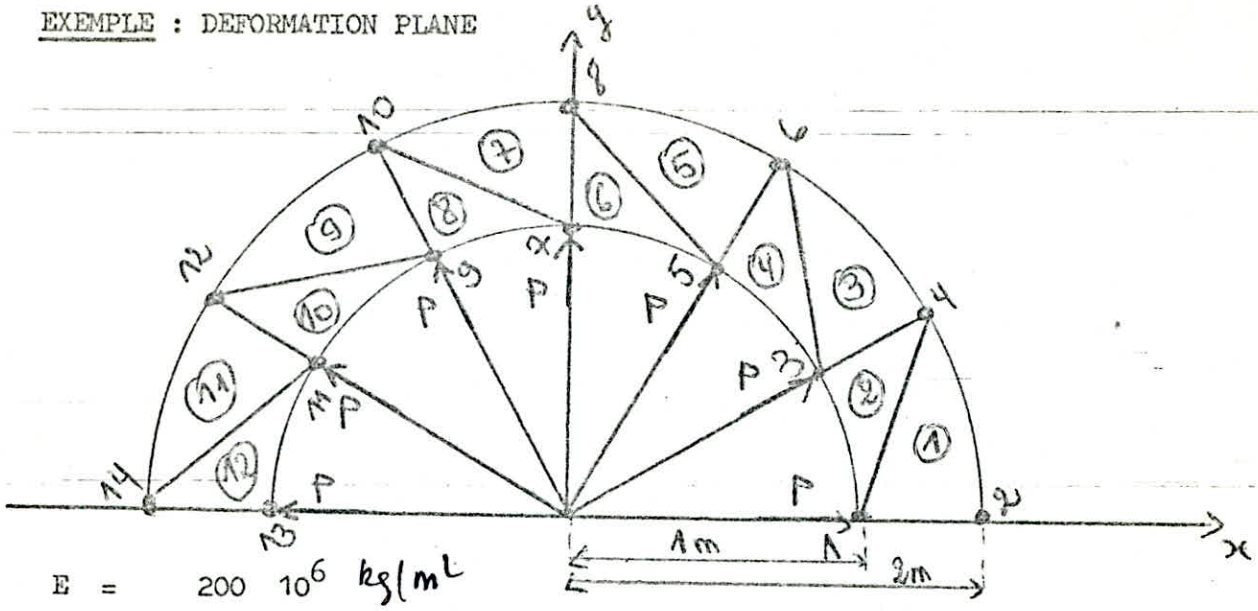
CONTRAINTES

ELEMENT	σ_x		σ_y		τ_{xy}	
1	0,36288	10^6	0,39497	10^6	0,85741	10^6
2	0,11205	10^7	0,16769	10^6	0,55378	10^6
3	0,11205	10^7	0,16769	10^6	0,55378	10^6
4	0,36288	106	0,39497	10^6	0,85741	10^6
5	0,36288	106	0,58060	105	0,14559	10^6
6	0,11205	107	0,22072	106	0,15548	107
7	0,11205	107	0,22072	106	0,15548	107
8	0,36288	107	0,58060	105	0,14559	106

(2)

Pour la meme poutre que precedemment, en remplaçant la charge concentrée par une charge uniformement repartie de 400 KG/M, on retrouve les mêmes resultats .

EXEMPLE : DEFORMATION PLANE



E = 200 10^6 kg/m²

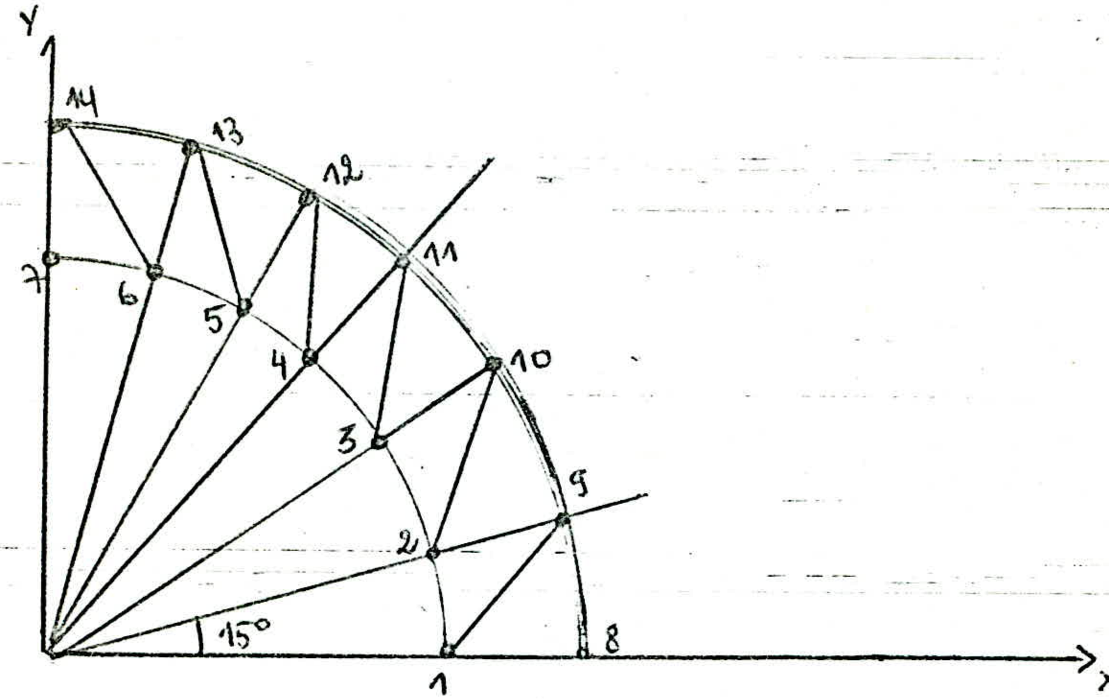
P = 1000 kg

Les resultats obtenus sont : en (m)

NOEUD .	X	Y	U	V
1	10000E + 01	00000E +00	43902E-04	00000E+00
2	20000E + 01	00000E +00	41351E-04	00000E+00
3	86600E + 00	50000E +00	38356E-04	48195E-05
4	17321E + 01	10000E +01	35065E-04	76821E-05
5	50000E + 00	86600E +00	30089E-04	79443E-05
6	10000E +01	17321E +01	26016E-04	88401E-05
7	00000E+ 00	100000E+01	20826E-04	92490E-05
8	00000E+ 00	20000E +01	19500E-04	77932E-05
9	- 50000E+ 00	86600E +00	11128E-04	87603E-05
10	- 10000E+ 01	17321E +01	13477E-04	79246E-05
11	- 86600E+00	50000E +00	19642E-05	56966E-05
12	- 17321E+01	10000E 01	54873E-05	66657E-05
13	- 10000E+01	00000E+ 00	43521E-05	00000E+00
14	- 20000E+01	00000E+ 00	00000E+00	00000E+00

EXEMPLE : en axisymetrie

SPHERE EPAISSE



P = 1000 kg   ν = 0   E = 2,1 10^6 kg/m²

DEPLACEMENT

(2)

COORDONNEES			DEPLACEMENTS (m)	
NOEUDS	X	Y	U	V
Noeuds 1	1,00	0,00	0,40226 10^{-5}	0,00
2	0,9659	0,2589	0,39950 10^{-5}	0,66281 10^{-5}
3	0,8659	0,500	0,39603 10^{-5}	0,13311 10^{-6}
4	0,7069	0,7073	0,38934 10^{-5}	0,20399 10^{-6}
5	0,4997	0,8662	0,37278 10^{-5}	0,27981 10^{-6}
6	0,2583	0,9661	0,32064 10^{-5}	0,36007 10^{-6}
7	0,00	1,00	0,00	0,44297 10^{-6}
8	2,00	0,00	0,39524 10^{-5}	0,00
9	1,9318	0,5178	0,39643 10^{-5}	0,4656 10^{-7}
10	1,7318	1,00	0,39732 10^{-5}	0,90018 10^{-7}
11	1,4138	1,4146	0,39584 10^{-5}	0,12579 10^{-6}
12	0,9993	1,7325	0,38684 10^{-5}	0,14205 10^{-6}
13	0,5167	1,9321	0,3648 10^{-5}	0,17755 10^{-6}
14	0,00	2,00	0,00	0,3342 10^{-7}

VERIFICATION

Pour le déplacement médical d'une sphére epaisse

$$U\,r = \frac{P r^3}{R^3 - r^3} \left(\frac{R^3}{2\,\rho^2} J + \rho M \right)$$

P : PRESSION INTERNE

$$M = \frac{1}{3K} \qquad J = \frac{1}{2\,G} \quad \Big\} \text{ EN ELASTICITE}$$

AVEC : K (module de compression) = $\frac{E}{3(1 - 2\,\nu)}$

G (module de cisaillement) = $\frac{G}{2(1 + \nu)}$

G : Le rayon où Ur est cherché

R,r rayon exterrieur et interrieur

Pour notre cas $\nu=0 = \underline{K} = \frac{E}{3}$ et $G = \frac{E}{2}$

D'ou $M = \frac{1}{E}$ et $G = \frac{1}{E}$

DONC

$$Ur = \frac{P\,r^3}{R^3 - r^3} \left(\frac{G}{E} + \frac{R^3}{2\rho^2} \; \frac{1}{E} \right)$$

R = 2 et R = 1 on cherche Ur pour G= 1

$$Ur = \frac{P}{7\,E} \left(1 + \frac{8}{2} \right) = \frac{5}{7} \; \frac{P}{E}$$

$$Ur = \frac{5}{200} * \frac{1000}{10^6} = 0{,}36 \; 10^{-5} \; m$$

Pour le calcul dans le programme on a trouve:

Pour le noeud 5 par exemple :

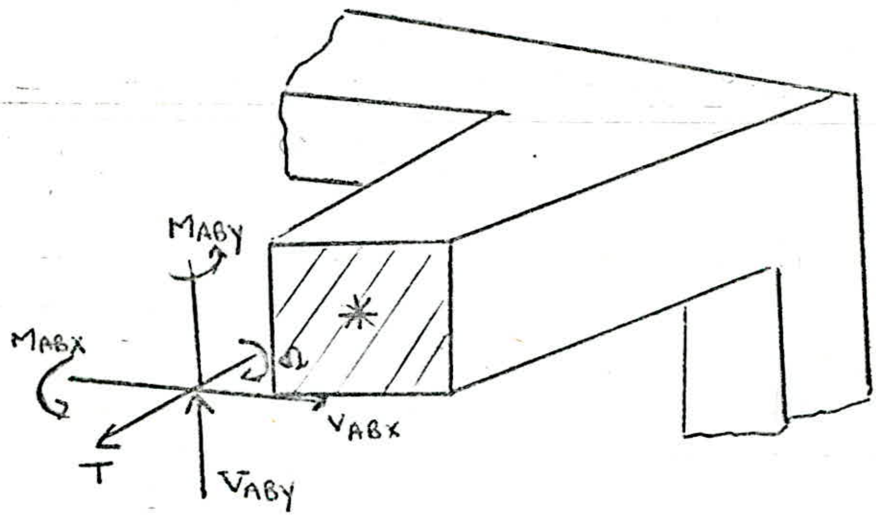
$Ur = \sqrt{(0,389\ \ 10^{5})^{2} + (\ 0,2039\ \ 10^{6})^{2}}$

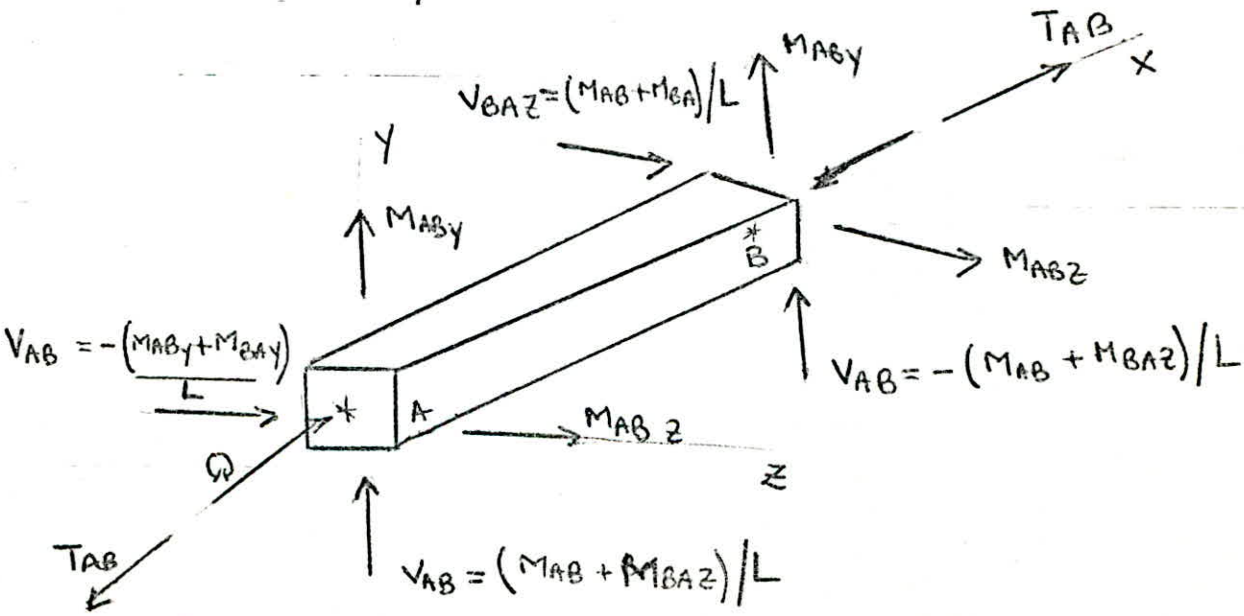
$Ur = \ 0,3895\ \ 10^{-5}\ m$

Pour tous les pointes appartenant à l'arc du cercle ou $\rho = 1$ on trouve entre 0,39 et 0,40 10^{-5} et le calcul donne 0,36 10^{-5}

POUTRES TRIDEMENSIONNELLES EN ELASTICITE LINEAIRE

Dans ce chapitre nous allons résoudres le problème de poutres à 3 degrés de liberté par noeud. Nous allons d'abord déterminer les 3 Déplacements + 3 rotations par noeud, puis nous allons déterminer les éléments de réduction sur chaque barre.





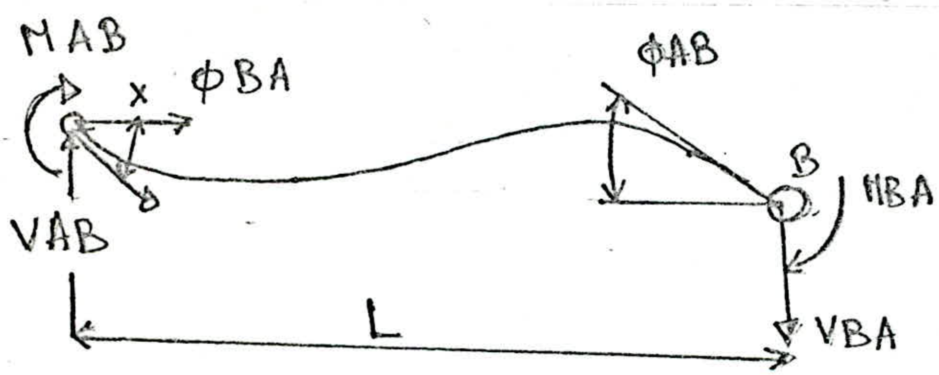
1- CALCUL DE LA MATRICE DE RIGIDITE (K)

La matrice de rigidité élémentaire se calcule par le tripe produit [A] [S] [A] T.

S Notée VKR représente la matrice de régidité réduite qui relié les déplacements et rotation locaux aux elements de reductions.

Les elements de reductions pris en compte sont seulement ceux qui sont independants, les autres peuvent êtres ensuite trouvés à partir de la matrice de transformation. Donc, ont doit avant tout calculer les déplacements et rotations de la barre, et pour cela il faudra chercher la matrice de rigidité élementaire (K).

- calcul de la matrice S (VKR)



on a $V_{AB} = V_{BA} = -\dfrac{(M_{AB} + M_{BA})}{L}$

Calculons le moment à la distance n du point A

$$M(x) = MAB + VAB.\ x$$

au point d'abcise x

on a : $\frac{M(x)}{EI} = -\frac{d^2y}{dx^2}$

$$\frac{1}{EI}\ (MAB + VAB.\ x) = \frac{d^2y}{dx^2}$$

La solution de cette equation différentielle nous donne :

$$y = \frac{1}{6EIL}\left[(MAB + MBA)x^3 - 3\ MABL\ x^2 + (2MAB - MBA)L^2 x\right]$$

on a :

$$OAB = \left[\frac{dy}{dx}\right]_{x=0} = 1/_{6EI}(2\bar{M}AB - \bar{M}BA)L$$

$$OBA = \left[\frac{dy}{dx}\right]_{x=L} = 1/_{6EI}(2MBA - MAB)L$$

et d'après la loi de HOOKE en élasticité linéaire on a : $TAB = \frac{EAe}{L}$

on a aussi

$$Q = \frac{GJ}{L}\theta$$

Q : torsion

finalement on aura :

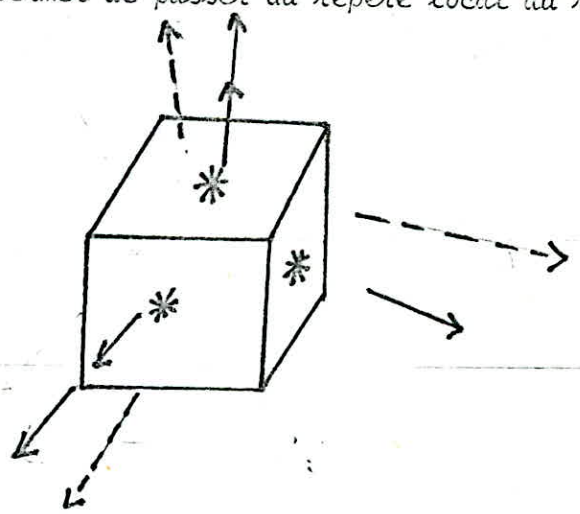
$$\begin{bmatrix} TAB \\ MABZ \\ MBAZ \\ BABY \\ MBAY \\ Q \end{bmatrix} = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{4EIZ}{L} & \frac{2EIZ}{L} & 0 & 0 & 0 \\ 0 & \frac{2EIZ}{L} & \frac{4EIZ}{L} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{4EIY}{L} & \frac{2EIY}{L} & 0 \\ 0 & 0 & 0 & \frac{2EIY}{L} & \frac{4EIY}{L} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{GJ}{L} \end{bmatrix} \begin{bmatrix} Q \\ QABZ \\ QBAZ \\ QABY \\ QBAY \\ 0 \end{bmatrix}$$

$$(S\,r) = (VKR)\,(n)$$

Calcule de la matrice A

- La matrice (A) est le produit de la matrice de passage VPE par la matrice de transformation VME

- La matrice VPE nous permet de passer du repère local au repère global.



FAX,FAY,FAZ dans le repère global
FAYm,FAYm, FAZm dans le repère local.

$$\begin{bmatrix} FAX \\ FAY \\ FAZ \end{bmatrix} = \begin{bmatrix} -\cos\alpha\cos\beta & \sin\alpha\cos\gamma - \cos\alpha\sin\beta\sin\gamma & -\sin\alpha\sin\gamma - \cos\alpha\sin\beta\cos\gamma \\ -\sin\alpha\cos\beta & -\cos\alpha\cos\gamma - \sin\alpha\sin\beta\sin\gamma & \cos\alpha\sin\alpha - \sin\alpha\sin\beta\cos\gamma \\ \sin\beta & -\cos\beta\sin\gamma & -\cos\beta\cos\gamma \end{bmatrix} \cdot \begin{bmatrix} FAXm \\ FAYm \\ FAZm \end{bmatrix}$$

$$\sin\alpha = \frac{Y_B - Y_A}{L_1} \qquad \cos\alpha = \frac{X_B - X_A}{L_1}$$

$$\sin\beta = \frac{Z_A - Z_B}{L} \qquad \cos\beta = \frac{L_1}{L}$$

$$L_1 = [(X_B - X_A)^2 + (Y_B - Y_A)^2]^{1/2}$$

$$L = [(X_B - X_A)^2 + (Y_B - Y_A)^2 + (Z_B - Z_A)^2]^{1/2}$$

donc $F = [RM][FM]$

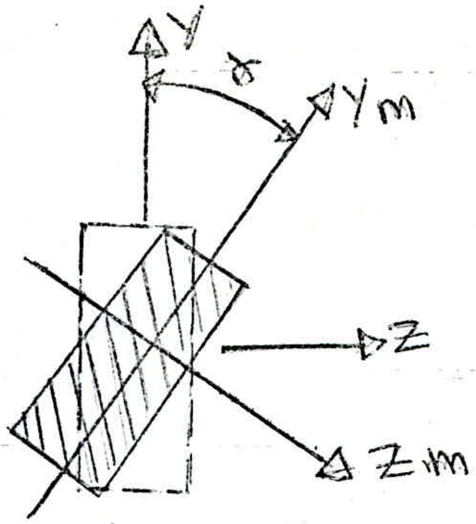
Donc Finalement:

$$\begin{bmatrix} FA \\ MA \\ FB \\ MB \end{bmatrix} = \begin{bmatrix} [RM] & & & \\ & [RM] & & \\ & & [RM] & \\ & & & [RM] \end{bmatrix} \cdot \begin{bmatrix} FAm \\ MAm \\ FBm \\ MBm \end{bmatrix}$$

$$(12\times1) \qquad (12\times12) \qquad (12\times1)$$

$$W = [VPE][SRE\}$$

avec $F = \begin{bmatrix} Fx \\ Fy \\ Fz \end{bmatrix}$ $\qquad M = \begin{bmatrix} Mx \\ My \\ Mz \end{bmatrix}$



La matrice VME nous permet de trouver les forces et moment dans le repère local et ceci à partir des elements de réductions.

D'aprés la Figure 1 on voit que :

$$V\ ABZ = \frac{(MABY + MBAY)}{Z}$$

$$V\ BAY = \frac{(MABZ + MBAZ)}{L}$$

$$FAXM = TAB$$

$$FBXM = -TAB$$

et d'aprés cette figure on peut déduire les autres forces et moments.

Finalement on aura :

$$
\begin{bmatrix} FAXm \\ FAYm \\ FAZm \\ MAXm \\ MAYm \\ MAZm \\ FBXm \\ FBYm \\ FBZm \\ MBYm \\ MBYm \\ MBZm \end{bmatrix}
=
\begin{vmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & -1/L & -1/L & 0 & 0 & 0 \\
0 & 0 & 0 & 1/L & 1/L & 0 \\
0 & 0 & 0 & 0 & 0 & -1 \\
0 & 0 & 0 & -1 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1/L & 1/L & 0 & 0 & 0 \\
0 & 0 & 0 & -1/L & -1/L & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & -1 & 0 \\
0 & 0 & -1 & 0 & 0 & 0
\end{vmatrix}
\cdot
\begin{bmatrix} TAB \\ MABZ \\ MBAZ \\ MABY \\ MBAY \\ Q \end{bmatrix}
$$

(12×6)

donc $(W) = (VPE)\ (VME)\ (Sr) = (A).(Sr)$

donc on a :

$(A) = (VPE)\ (VME)$

$(W) = (A).(Sr)$

$(Sr) = (VKR)\ (x) \qquad (W) = (A)\ (VKR)\ (x)$

dans le repère global :

$(X) = (A)^t\ (x)$

finalement : $(W) = (A)\ (VKR)\ (A^t)\ (x)$

Dans notre programme (A) est notée (VAE)

$$\{X\} = [\Delta x \ \Delta y \ \Delta z \ \theta x \ \theta y \ \theta z]^T$$

$$\text{Donc } [k] = [VAE][VKR][VAE]^T$$

(K) matrice de rigidité élémentaire

2- Aprés l'obtention des matrices de régidité élémentaires, on passe à l'assemblage, afin d'obtenir la matrice de régidité global

3- On introduit des conditions aux limites qui sont de 6 par noeud.

On pose : 1- mouvement possible
0- mouvement bloqué

par exemple:

On pose 0 0 0 0 0 0
tous les deplacements et rotation son bloqués.

on pose 1 0 0 1 1 1
on a bloqué les deplacements suivant Y et Z

Pour les conventions de signe pour les forces et les moments:

Sont positives les forces dans le sens des axes
Sont positifs les moments qui tournent dans le sens des aiguil les d'une montre.

Aprés obtention de la matrice de régidité globale et introductions d'appuis, on résoud par la méthode de GAUSS l'équation.

On obtient ainsi les 3 déplacements et les 3 rotations par noeud et ceci dans les directions X,Y et Z.

CALCUL DES EFFORTS DANS LES POUTRES

Aprés l'obtention des déplacements et des rotations pour chaque noeud, on peut déduire les éléments de réductions dans les barres, à partir des étapes suivantes:

1- Puisqu'on doit calculer les éléments de réduction pour chaque poutre, donc on doit déduire les déplacements et les rotations

pour chacune d'elles

Pour chaque poutre donc :

$Ul = (VAE)^t \times U\ global$

U global : deplacement et rotations déjà calculés

2 - $Fl = (VKR) . Ul$

$$\begin{bmatrix} TAB \\ MABZ \\ MBAZ \\ MABY \\ MBAY \\ Q \end{bmatrix} = (VKR) . \begin{bmatrix} e \\ \phi ABZ \\ \phi BAZ \\ \phi ABY \\ \phi BAY \\ \theta \end{bmatrix}$$

$$(SR) = (VKR) . (Ul)$$

3 - Après le calcul de (SR) on a :

$$(W) = (VAE) . (SR)$$

donc :

$$\begin{bmatrix} FAX \\ FAY \\ FAZ \\ MAX \\ MAY \\ MAY \\ MAZ \\ FBX \\ FBY \\ FBZ \\ MBX \\ MBY \\ MBZ \end{bmatrix} = (VAE) . \begin{bmatrix} TAB \\ MABZ \\ MBAZ \\ MABY \\ MBAY \\ Q \end{bmatrix}$$

REMARQUES :

On a utilisé la matrice de régidité réduite qui ne tient compte que des éléments de réduction qui sont indépendants. En élasticité linéaire, on peut prendre en compte tous les éléments même s'il sont dépendants, sans le résultat, mais on a procédé ainsi pour pouvoir éventuellement introduire avec une méthode simple, le problème de plasticité.

PROGRAMMATION

I - CALCUL DES DEPLACEMENTS

Les trois matrices qui sont respectivement :

- matrice de passage notée (VPE)
- matrice de transformation notée (VME)
- matrice de régidité notée (VRE)

sont introduites dans la soubroutine ELEMO 2 on, pour obtenir la matrice de régidité élémentaire, VKE on fait les produits suivants

$$(VAE) = (VPE).\ (VME)$$
$$(VSE) = (VAE).\ (VRE)$$
$$(VKE) = (VSE)\ \ (VAE)^t$$

Les matrices de régidité élémentaire et les vecteurs élémentaires (VFG) (3 déplacements et 3 rotations) sont introduits dans le programme principal ELAST, sont assemblés dans le sous-programme ASSEMB. Aprés introductions de 6 conditions d'appuis possible, on résoud léquation (F) = (K) (Δ) dans le sous programme resol, et cela pour obtenir les déplacements et les rotations aux noeuds.

$$\{\Delta\} = \begin{Bmatrix} U_x \\ U_y \\ U_z \\ \theta_x \\ \theta_y \\ \theta_z \end{Bmatrix}$$

-CALCUL DES ELEMENTS DE REDUCTION

Le procédé est analogue à celui utilisé pour le calcul des contraintes pour l'elesticité linéaire.

Aprés le calcul des déplacements, on appelle une deuxième fois le sous programme ELEMO 2 (avec l'indice KP=2) d'où on tiré (VAE) et (VRE) et dans le sous programme CONTR, on fait les produits suivants:

$$[VAE]^T \{\Delta\} = \{\Delta l\}$$

$$[VRE] \{\Delta l\} = \{S_r\}$$

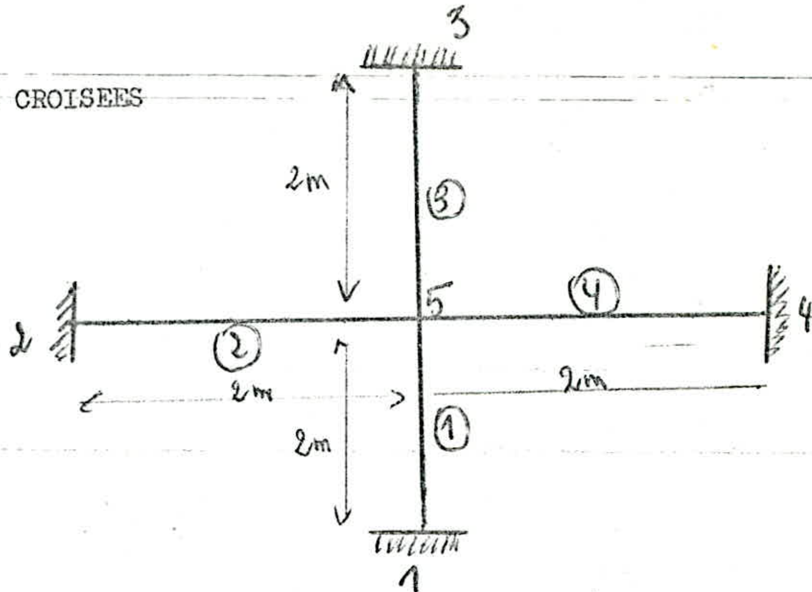
(déplacements dans le repère local)

Avec (SR) qui sont les éléments de réduction élémentaires indépendants, on obtient tous les autres éléments de réduction(W) pour chaque élément, en faisant dans le programme principal ELAST, le produit suivant:

$$\{W\} = [VAE] \{SR\}$$

(1)

EXEMPLE : POUTRES CROISEES



DONNEES :

P = 1000 KG J = 0,00015 M^4 S= 0,02 m^2 IZ = 0,001 m^4

1- DEPLACEMENTS IY= 0,00004 m^4 E= 200 10^6 KG/M^2 G= 80 10^6 KG/M^2

en (m)

NOEUD	UX	UY	YZ	ΘX	ΘY	ΘZ
1	0,00000	0,00000	0,0000	0,0000	0,0000	0,0000
2	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
3	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
4	0,0000	0,0000	0,00000	0,0000	0,0000	0,0000
5	0,0000	0,0000	0,20833 10^{-1}	0,0000	0,0000	0,0000

- Effort dans les Barres

(2)

ement (arre)	TAB	MABZ	MBAZ	MABY	MBAY	Mt de Tortion
1	0,00	0,00	0,00	- 0,25 10^3	- 0,25 10^3	0,00
2	0,00	0,00	0,00	- 0,25 10^3	- 0,25 10^3	0,00
3	0,00	0,00	0,00	- 0,25 10^3	- 0,25 10^3	0,00
4	0,00	0,00	0,00	- 0,25 10^3	- 0,25 10^3	0,00

- Réaction et

t	Noeud	FX	FY	FZ	MX	MY	MZ
	1	0,00	0,00	0,25$10^3$	0,25 10^3	0,00	0,00
	5	0,00	0,00	0,25$10^3$	0,25 10^3	0,00	0,00
	2	0,00	0,00	0,25$10^3$	0,00	- 0,25$10^3$	0,00
	5	0,00	0,00	0,25$10^3$	0,00	- 0,25$10^3$	0,00
	3	0,00	0,00	0,25$10^3$	0,25$10^3$	0,00	0,00
	5	0,00	0,00	0,25$10^3$	0,25$10^3$	0,00	0,00
	4	0,00	0,00	0,25$10^3$	0,00	0,25$10^3$	0,00
	5	0,00	0,00	0,25$10^3$	0,00	0,25$10^3$	0,00

VERIFICATION

Pour la verification on prend l'une des deux poutres croisées et dans l'aide memoire RDM, on trouve la formule de la fleche pour une poutre doublement encastrée.

deformée : $YP = \dfrac{P\, A^3 B^3}{3\, E\, I\, L^3}$

pour notre cas, la force est au milieu donc

$A = B = \dfrac{L}{2}$

$\Rightarrow YP = - \dfrac{P\, L9/64}{3EI\ L3} = \dfrac{P\ L\ 3}{3X64\ 01} = \dfrac{P\ L\ 3}{192EI}$

Dans notre exemple :

P= 500 kg I= IY = 0,00004 E= 200 10^6 kg/m²

(4)

$$YP = \frac{500, \times 4^3}{192 \times 200 \quad 10^6 \times 4 \quad 10^5}$$

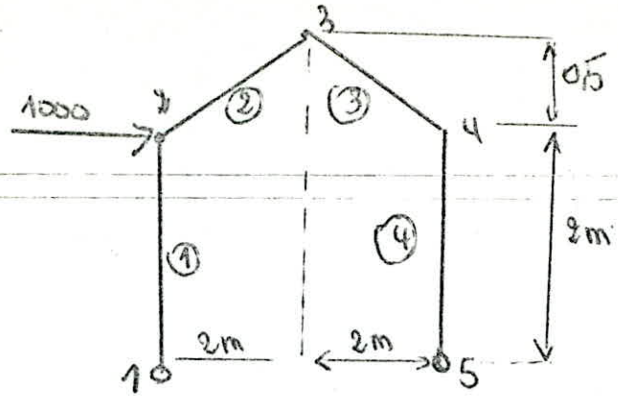
$$YP = \frac{5 \times 4^2}{192 \times 20} = \frac{4}{192} = 0{,}0208 \text{ m.}$$

ET LE PROGRAMME DONNE LA VALEUR :

$Y_5 = 0{,}20833 \quad 10\text{-}1$

POUR LA REACTION D'APPUIS $R = \frac{P}{2} = 250$ kg

EXEMPLE :



(1)

H = 2 M

L = 4 M

F = 0,5

S = 0,080 m^2

J = 0,00015 m^4

IZ= 0,001 "

IY= 0,00004 "

E = 200 10^6 kg/m^2

G = 80 10^6 " "

RESULTATS OBTENUS :

deplacement

NOEUDS	DEPL. SUIVANT X	DEPL. SUIVANT Y	DEPL. S. Z	NOTATION S. X	NOTATION S. Y	NOTATION S. Z
1	0,0	0,0	0,0	0,0	0,0	0,0089
2	0,01415	0,00025	0,0	0,0	0,0	0,003418
3	0,0139	0,00048	0,0	0,0	0,0	- 0,00158
4	0,0134	0,00025	0,0;	0,0	0,0	-0,0037
5	0,0	0,0	0,0	0,0	0,0	-0,00822

(01)

EFFORTS NORMAUX, TRANCHANT ET LES MOMENTS AUX NOEUDS

NOEUD	F X	F Y	F Z	M X	M Y	M Z
1	549,07	- 500,0	0,0	0,0	0,0	0,0
2	450,09	- 500,0	0,0	0,0	0,0	1098,1
3	450,93	- 500,0	0,0	0,0	0,0	127,33
4	450,93	- 500,0	0,0	0,0	0,0	901,8
5	450,93	- 500,0	0,0	0,0	0,0	0,0

Methode RDM Nous donne :
d'aprés aide memoire RDM.

Noeuds	FX	FY	FZ	MX	MY	MZ
1	545,7	-500	0,0	0,0	0,0	0,0
2	545,7	-500	0,0	0,0	0,0	1091,48
3	545,7	-500	0,0	0,0	0,0	135,05
4	454,26	-500	0,00	0,0	0,0	908,52
5	454,26	500	0,0	0,0	0,0	0,0

ON REMARQUE : Que les valeurs obtenues par la methodes des elements finis sont trés proché des valeurs obtenues par les formules de la R D M .

(2)

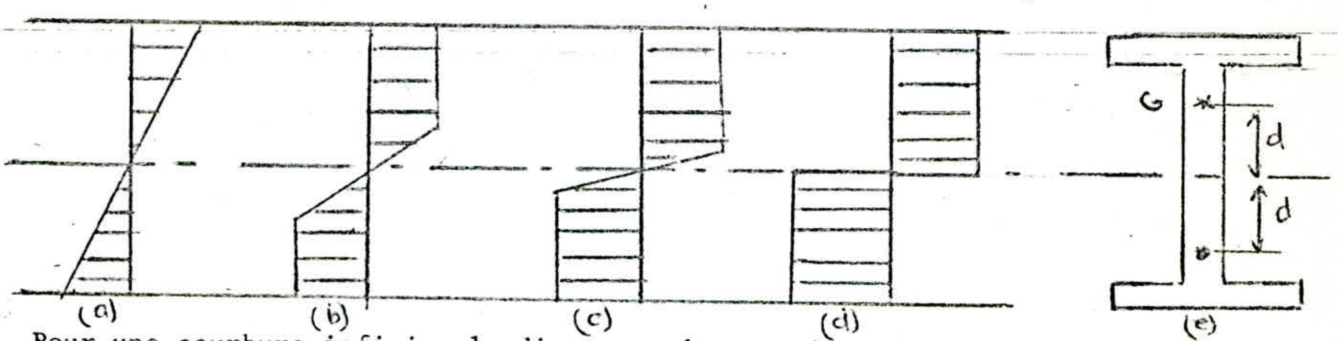
POUTRES TRIDIMENSIONNELLES EN PLASTICITE

Le moment fléchissant manimum que cette poutre peut supporter élastiquement pour valeur :

$$M_e = \sigma_e \frac{I}{V}$$

$\frac{I}{V}$: Module de fléxion de la section droite.

Si l'on dépasse ce moment (M_e), les fibres extrémes de la poutre se plastifient ; au fur et à mesure que le moment augmente, les zones plastifiées s'étendent vers l'axe de la poutre comme on le voit sur la figure ci-dessous.



Pour une courbure infinie, le diagramme des tensions de fléxions tend vers la distribution birectangulaire (figure I d). Le moment fléchissant correspondant est le plus grand moment que peut supporter la poutre, c'est le moment plastique noté

D'après la figure I on a :

$$M_p = 2 . \sigma_e \frac{\Omega}{2} d \quad \text{on ecrit } M_p = M_p = \sigma_e Z$$

$$Z = \Omega . d$$

Z : module plastique de la section droite.

On sait aussi que $\frac{\Omega}{2}d$ est le moment statique (S_X) de la section droite.

dont $Z = 2\,S_{xp}$

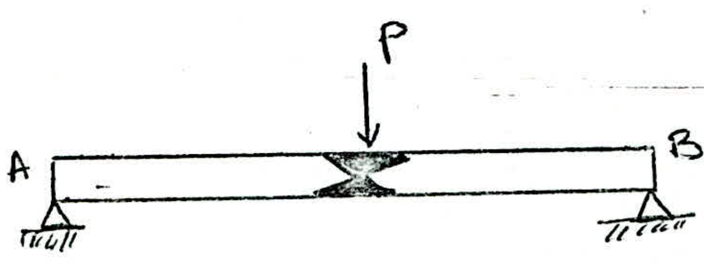
Le rapport :

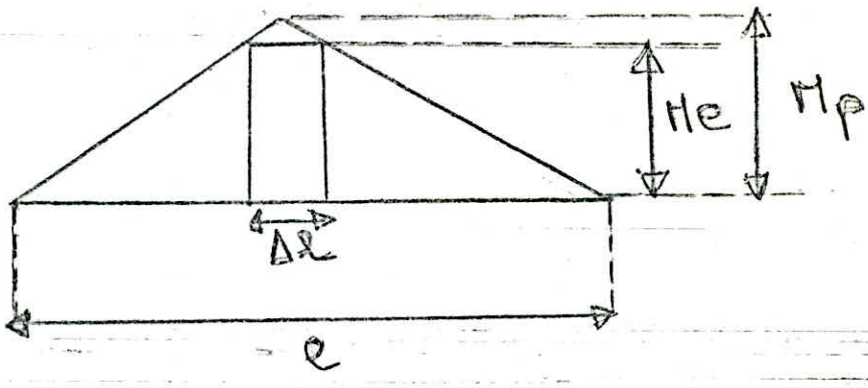
$$\frac{Mp}{Me} = F \quad \text{est appelé facteur de forme.}$$

Section					
F	2,37	2,00	I,70	I,50	I,27

2 - Notion de rotule plastique

Considérons une poutre en double Té sur 2 appuis simples avec une force concentrée P en son milieu. Le diagramme des moments correspondant est représenté sur la figure ci-dessous .





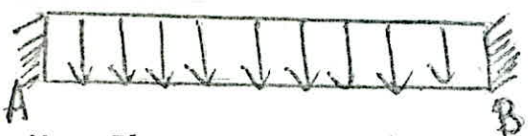
La poutre s'éffondre quant le moment en C atteint la valeur Mp.

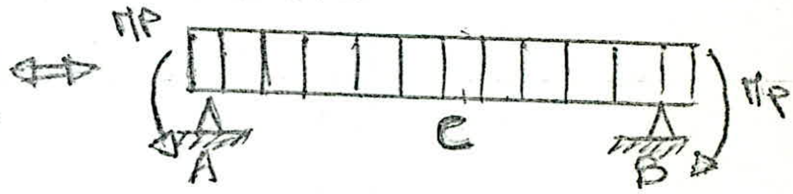
La zone de déformations importantes est toujours très localisée et ne peut admettre pratiquement que la poutre déformée se compose de deux tronçons rigides articulés au point C.

On peut dire que tout se passe comme s'il éxistait dans la section où le moment est maximum, une rotule à frottement qui resterait rigide tant que M < Mp et qui permettrait la rotation relative des deux tronçons de poutres dès que le moment atteint sa valeur plastique Mp. Une telle rotule s'appelle rotule plastique.

Méthode des contraintes initiales

La méthode des contraintes initiales, consiste à remplacer les efforts dans les barres par des efforts entérieurs Mp d'où la notion de contraintes initiales.





Mc : $\frac{Pl}{8}$ - Mp = Mp

Mp : moment hyperstatique

$\frac{Pl^2}{8}$: moment isostatique.

Mais la méthode qu'on utilise, est une méthode simplifiée, dérivée de la méthode des contraintes initiales.

Pour cette méthode, on calcule $\lambda p = \frac{Mp}{Me}$ et cela pour chaque noeud, et le noeud qui entre en plasticité, est celui dont le λp_1 est le minimum.

2 - Ou les nouveaux Mp qui sont représentées pas les capacitées résiduelles et après avoir eu les nouveaux Me, on recalcule λp_2 et $\lambda p = \Sigma \lambda P_i$

Donc au noeud I on a une rotule plastique

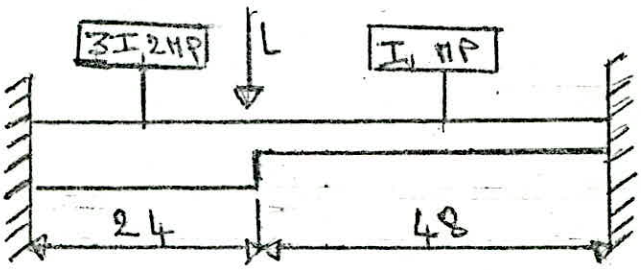
On avait 

$$\begin{bmatrix} T \\ MAB \\ MBA \end{bmatrix} = \begin{bmatrix} \frac{EA}{L} & 0 & 0 \\ 0 & 4EI/L & 2EI/L \\ 0 & \frac{2EI}{L} & \frac{4EI}{L} \end{bmatrix}$$

Maintenant

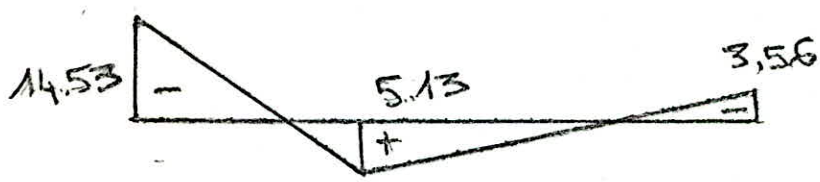
$$\begin{bmatrix} T \\ MAB \\ MBA \end{bmatrix} = \begin{bmatrix} \frac{EA}{L} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{4EI}{L} \end{bmatrix}$$

On recalcule K et on tire Δ puis les moments Me



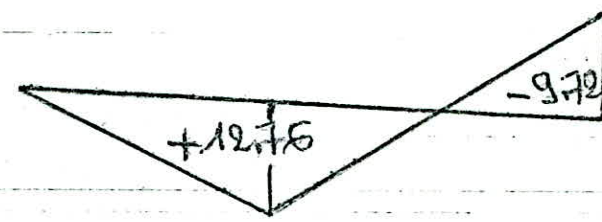
Pour mieux assimiler cette méthode, on va l'illustrer par une exemple en bidimensionnel.

MP = 12

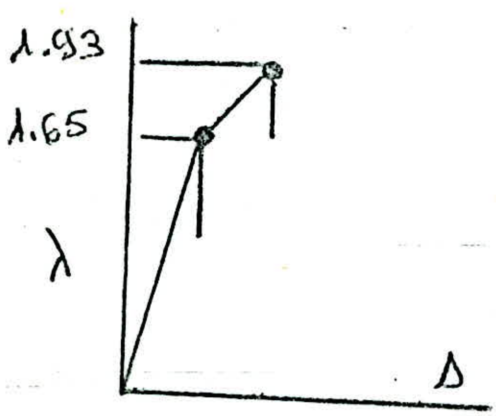


Me	I4,53	5,I3	3,56
Mp	24,00	I2,00	I2,00
Xp	I,65	2,34	3,37
niv Xp	I,65		
Capacité résiduelle	0,00	3,56	6,I3

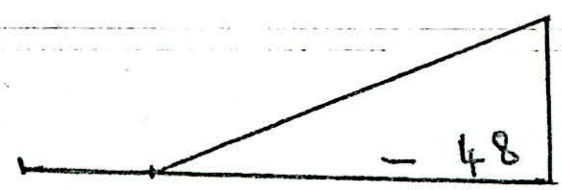
Me	0,00	I2,76	9,72
Mp	0,00	3,54	6,I3
Xp	-	0,28	0,63
Min Xp	-	0,28	-
Capacité résiduelle	0,θ0	0,00	3,4I
Xp	I,93		

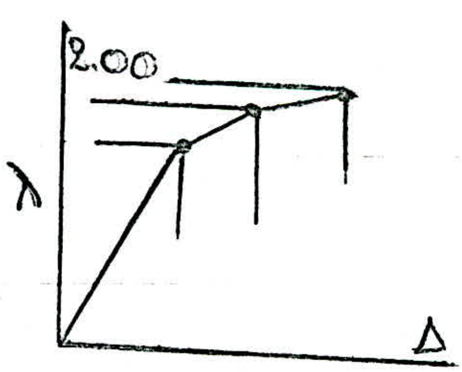


Le noeud 2 entre en plasticité.



Me			48
Mp	0,00	0,00	3,4I
Xp	-	-	0,07
Min Xp	0	0	3,4I
Capacité résiduelle	0,00	0,00	0,00
Xp	2,00		





Programmation

Au premier pas le calcul se fait comme pour l'elasticité on multiplie seulement les composants de la matrice (VRE) ~~par des facteurs Wij~~ qui sont égaux à un debut.

On introduit en plus le degré d'hypestaticité et les moments plas-tiques noté AMP pour chaque noeud et dans les 3 directions X,Y,Z après l'obtention des moments suivant toutes les directions et pour chaque noeud dans le repère local, on calcul les rapports des moments plastiques sur les moments elastiques

correspondants, qu'on note RP (NELT,K)

NELT = element correspondant

K = 1,5

K = 1 noeud 1 et moment suivant Z

K = 2 noeud 2 et moment suivant Z

K = 3 noeud 1 zt moment suivant Y

K = 4 noeud 2 et moment suivant Y

K = 5 noeud 1m2 et moment suivant X (TORSION)

On test R.P. pour obtenir le minimum noté Test avec celui ci on calcul les capacités residuelles pour chaque noeud qui deviennent les nouveaux moments plastiques M_{p_2}

$$M_{p_2} = M_{p_1} - \text{Test} . M_e$$

Test : le minimum

Me : moment elastique pour chaque noeud

M_{p_1} : moment plastique (donné) pour chaque noeud

On revient et on fait une autre (itteration) en annulent les Wij correspondant au noeud ou apparait la rotule plastique c'est à dire là ou le rapport M_p/Me est le minimum.

On obtient un nouveau minimum le nombre d'itteration est égal au degré d'hyperstaticité plus un. Ces minimum sont stockés et sommés leur somme nous donne λ P.

IC = 5

IPL = I

NNT = Nombre de noeud Total

I = Nombre délément total

J = Rang.de la compésante de VRE non nul.

LT = O

Lecture des AMP pour chaque Direc.
Lecture du degré d'hyperstaticité H.

Boucle sur NNT

Wij = I

Boucle sur I,J

Wij := O correspondant au minimum

Call ELEMO 2
VRE x Wij

Call CONTR calcul des moments élactique

Calcul des RP et le minimum des RP TES
Calcul des nouveaux
AMP = AMP - M x TES

IPL = O

IF (LT.EQ.H)

NE

Xp = minimums

ANNEXE

PROGRAMME PRINCIPALE ET SOUBROUTINES

ELAST (F , F)

- LECTURE DES CAVITES ET LES SECTIONS (TRIDIMENSIONNEL)
- LECTURE DES MOMENTS PLASTIQUES. (TRIDIMENSIONNEL)
- LECTURE DU DEGRE D'HYPERSTATICITE
- LECTURE DES CHARGES CONCENTREES ET UNIFORMES
- IMPRIME LES RESULTATS

GRILLE

- LECTURE DES COORDONNEES
- LECTURE DES CONNECTIVITES
- LECTURE DU COEFFICIENT RELATIF A LA PRESENCE D'EAU
- LECTURE DU NUMERO DU MATERIAU UTILISE

LOCEL

- CREATION DE LA TABLE DE LOCALISATION

ASSEM

- ASSEMBLAGE DES VKE ET VFE

RESOL

- RESOLUTION DE L'EQUATION F=K U

CONTR

- CALCUL DES CONTRAINTES
- CALCUL DES PRESSIONS INTERSTICIELLES EN PRESENCE D'EAU

ELEMO2

- CALCUL DES MATRICES DE RIGIDITE ELEMENTAIRES

PARAMETRES UTILISES DANS LE PROGRAMME

NNT	:	NOMBRE DE NOEUDS TOTAL
NELT	:	NOMBRE D'ELEMENTS TOTAL
NNEL	:	NOMBRE DE NOEUDS PAR ELEMENT
NDLE	:	NOMBRE DE DEGRES DE LIBETE PAR ELEMENT
NDLN	:	NOMBRE DE DEGRES DE LIBERTE PAR NOEUD
NDIM	:	DIMENSION DU PROBLEME
VCORE	:	TABLE DE COORDONNEES ELEMENTAIRES
VCORG	:	TABLE DE COORDONNEES GLOBALES
KCONEC	:	TABLE DES CONNECTIONS
KLOCE	:	TABLE DE LOCALISATION
VFE	:	VECTEUR FORCES ELEMENTAIRE
VFG	:	VECTEUR FORCES GLOBAL
VKE	:	MATRICE DE RIGIDITE ELEMENTAIRE
VKG	:	MATRICE DE RIGIDITE GLOBAL
VME	:	MATRICE DE TRANSFORMATION ELEMENTAIRE
VPE	:	MATRICE DE PASSAGE
VKE	:	MATRICE DE RIGIDITE REDUITE
DEV	:	MATRICE INTRINSEQUE DU MATERIAU
AMP	:	MOMENTS PLASTIQUES
KP	:	FACTEUR DE FORME

```
      PROGRAMME ELAST
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION VCORE(3,3),KLOCE(12),VKE(12,12),VKG(3000),VPE(144)
      DIMENSION KCONEC(3,50),VCORG(3,50),VFG(100),MAT(50),AMP(10,3)
      DIMENSION E(5),P(5),GX(5),GY(5),DONT(6),VBE(4,6),DEV(4,4),UL(6),
     1 SEC(50),A12(50),GAMA(50),VFE(12),AJY(50),EJA(50),IEAU(50),IS(5),
     1 DIV(16),DOIT(4),GZ(5),VAE(12,6),VRE(6,6),FMN(12,10),VFP(100),
     1 IT(5),RP(10,5),CONT(10,6),TES2(10),W2(10,9),W(9),COIT(30,4)
      DATA VKG/3000*0.D0/,GRAND/1.D12/,VFG/100*0.D0/,VPE/144*0.D0/
      DATA NDLE/6/,NDLN/2/,NNEL/3/,NDIM/2/,NSYM/0/,PI/3.14159265/,
     1 DIV/16*0.D0/,DOIT/4*0.D0/,VFP/100*0.D0/,APE/1.E-10/
C              BLOC DE LECTURE DES DONNEES
      WRITE(*,10)
10    FORMAT(1X,'PROGRAMME D ELEMENT FINIS',/,/,1X,'- SI VOUS VOULEZ TR
     1AITER UN PROBLEME DE CONTRAINTE PLANE TAPEZ 1',/,1X,'-SI VOUS  VOU
     1LEZ TRAITER UN PROBLEME DE DEFORMATION PLANE TAPEZ 2',/,1X,' -SI V
     1OUS VOULEZ TRAITER UN PROBLEME D AXISYMETRIE TAPEZ 3',/,1X,'-SI VO
     1US VOULEZ TRAITER UN PROBLEME D ELASTICITE TRIDIMENSIONNEL TAPEZ 4
     1',/,1X,'-SI VOUS VOULEZ TRAITER UN PROBLEME DE PLASTICITE TRIDIMEN
     1SIONNEL TAPEZ 5')
      READ(*,20)IC
20    FORMAT(I2)
      IF(IC.NE.5)GOTO 2110
      WRITE(*,2312)
2312  FORMAT(1X,'DONNER LE DEGRÉ D HYPERSTATICITE')
      READ(*,2113)JPL
2113  FORMAT(I3)
      IT=0
2110  IF(IC.LT.4)GOTO 29
      IPE=1
      NDIM=3
      NNEL=2
      NDLN=6
      NDLE=12
29    WRITE(*,30)
30    FORMAT(1X,'TYPE DE CHARGEMENT DU PROBLEME',/,/,1X,'-FORCES UNIFOR
     1MEMENT REPARTIES TAPEZ 1',/,1X,'-FORCES CONCENTREES TAPEZ 2',/,1X
     1,'-LES 2 EN MEME TEMPS TAPEZ 3',/,1X,'-PAS DE CHARGES TAPEZ 4')
      READ(*,40)IF
40    FORMAT(I2)
      CALL GRILLE(NDIM,NNEL,NNT,NELT,VCORG,KCONEC,MAT,IEAU,IC)
      IF(IC.NE.5)GOTO 2102
      DO 2200 IO=1,NNT
      WRITE(*,2103)IO
2103  FORMAT(1X,'DONNER LES MOMENTS LIMITES PLASTIQUES MPX,MPY,MPZ POUR
```

```
      1 LE NOEUD(EN F10.3) :',I5)
        READ(*,2104)(AMP(IU,JY),JY=1,3)
2104    FORMAT(3F10.3)
2200    CONTINUE
2102    IF(IC.LT.4) GOTO 45
        WRITE(*,32)
32      FORMAT(1X,'G=? EN F14.2,')
        READ(*,33)G
33      FORMAT(F14.2)
        DO 44 IU=1,NELT
        WRITE(*,42)IU
42      FORMAT(1X,'DONNER LA SECTION ,L INETIE(Iz),L INERTIE(IY),MOMENT D
      1E TORSION(J) ET ANGLE GAMA EN (4F10.5,F6.2) POUR L ELEMENT ',I5)
        READ(*,43)SEC(IU),AIZ(IU),AIY(IU),EJA(IU),GAMA(IU)
43      FORMAT(4F10.5,F6.2)
44      CONTINUE
45      NEQ = NNT*NDLN
        WRITE(*,50)
50      FORMAT(1H,1X,'COMBIEN DE MATERIAUX ST UTILISES EN I2 ')
        READ(*,60) NM
60      FORMAT (I2)
        WRITE(*,70) NM
70      FORMAT(1H,1X,'NOMBRE DE MATERIAUX = ',I2)
        DO 120 I=1,NM
        WRITE(*,80)
80      FORMAT(1H,1X,'DONNER PAR ORDRE :',/,1X,'E(modul de young) en F14
      1 .2',/,1X,'u(coeff de poisson) en F6.4 ',/,1X,'GAMA(X) en F9.3
        WRITE(*,90)I
90      FORMAT(1X,'GAMA(Y) GAMA(Z)en 2F9.3',/,1X,'POUR LE MATERIAU N°
      1 I2)
        READ(*,100)E(I),P(I),GX(I),GY(I),GZ(I)
100     FORMAT(F14.2,F6.4,3F9.3)
        WRITE(*,110)E(I),P(I),GX(I),GY(I),GZ(I)
110     FORMAT(1H,1X,'E=',F14.2,/,1X,'u=',F6.4,/,1X,'GAMA(X)=',F9.3,
      1  'GAMA(Y)=',F9.3,/,1X,'GAMA(Z)=',F9.3)
120     CONTINUE
C        LECTURE DES SOLLICITATION NODALES(DONNEES)
        IF(JF.EQ.4) GOTO 420
        IF(JF.EQ.1)GOTO 170
        IF(IC.LT.4) GOTO 121
        DO 930 I=1,NNT
        WRITE(*,900)
900     FORMAT(1H,2X,'N° DU  NOEUD-SOLLICITATIONS (FX,FY,FZ,Mx,My,Mz)
      1'TAPEZ ZERO S IL N Y EN A  PLUS EN I5 ,6F10.2')
        READ (*,910) IN,FX,FY,FZ,FMX,FMY,FMZ
910     FORMAT (I5,6F10.2)
        IF(IN.EQ.0)GOTO 170
        WRITE(*,920) IN,FX,FY,FZ,FMX,FMY,FMZ
920     FORMAT (1H,1X, 'SOLLICITIONS AU NOEUD' I5,'=',6(E12.5,1X))
        VFG(IN*6)=FMZ
        VFG(IN*6-1)=FMY
        VFG(IN*6-2)=FMX
        VFG(IN*6-3)=FZ
        VFG(IN*6-4)=FY
```

```
      VFG(IN*6-5)=FX
      VFP(IN*6)=FMZ
      VFP(IN*6-1)=FMY
      VFP(IN*6-2)=FMX
      VFP(IN*6-3)=FZ
      VFP(IN*6-4)=FY
      VFP(IN*6-5)=FX
120   CONTINUE
      GOTO 170
121   DO 160 I=1,NN
      WRITE(*,130)
130   FORMAT(1H,2X,'N° DU  NOEUD-SOLLICITATIONS (FX,FY)
     1 TAPEZ ZERO S IL N Y EN A  PLUS EN I5 ,2F10.2')
      READ (*,140) IN,FX FY
140   FORMAT (I5,2F10.2)
      IF(IN.EQ.0)GOTO 170
      IF(IC.NE.5)GOTO 148
      H=2*PI*VCORG(1,IN)
      GOTO 149
148   H=1.
149   WRITE(* 150) IN,FX,FY
150   FORMAT (1H,1X, 'SOLLICITATION AU NOEUD ',I5,'=',E12.5,1X,E12.5)
      VFG(IN*2-1)=FX*H
      VFG(IN*2)=FY*H
160   CONTINUE
170   IF(JF.EQ.2)GOTO 420
C      CHARGES UNIFORMEMENT REPARTIES
P100  DO 410 IJ=1,NELT
      WRITE(*,180)
180   FORMAT(1X,'DONNER L-ELEMENT CHARGE UNIFORMEMENT ')
      READ(*,190)IEL
190   FORMAT(I3)
      IF(IEL.EQ.0)GOTO 420
      WRITE(*,200)
200   FORMAT(1X,'NOMBRE DE FACES CHARGEES -? ')
      READ(*,210)NF
210   FORMAT(I3)
      DO 400 I=1,NF
      WRITE(*,220)
220   FORMAT(1X,'-CHARGE QN(NORMALE A LA FACE)POSITIVE SI RENTRANTE DS
     1 ELEMENT',/,1X,'-CHARGE QT(TANGENTE A LA FACE) POSITIVE SI ELLE PA
     1 RCOURE LA FACE DANS LE SENS DU PARCOUR DE LA MAILLE ',/,/)
      WRITE(*,230)I
230   FORMAT(1X,'DONNER LE PREMIER NOEUD D APRES LE SENS DU PARCOUR DE
     1LA MAILLE ET CES Q(CHARGE NORMALE) ET QT (CHARGE TANGENTIELLE) POU
     1R LA FACE',I5,'EN (I3,2F10.2)')
      READ(*,240)JN,FN,FT
240   FORMAT(I3,2F10.2)
      WRITE(*,250)I
250   FORMAT(1X,'DONNER LE DEUXIEME NOEUD ET CES Q(CHARGE NORMALE) ET
     1 QT(CHARGE TANGENTE) POUR LA FACE',I5,'EN (I3,2F10.2)')
      READ(*,260)JM,FM,FG
260   FORMAT(I3,2F10.2)
      X1=VCORG(1,JN)
```

```
      Y1=VCORG(2,JN)
      X2=VCORG(1,JM)
      Y2=VCORG(2,JM)
      D=((X2-X1)*(X2-X1)+(Y2-Y1)*(Y2-Y1))**0.5
      CS=(X1-X2)/D
      SN=(Y1-Y2)/D
      FXI=(FN*CS-FT*SN)
      FYI=(FN*SN+FT*CS)
      FXJ=(FM*CS-FG*SN)
      FYJ=(FM*SN+FG*CS)
      IF(IC.EQ.3)GOTO 310
      VFG(JN*2-1)=VFG(JN*2-1)+(FXI/3.+FXJ/6.)*D
      VFG(JN*2)=VFG(JN*2)+(FYI/3.+FYJ/6.)*D
      VFG(JM*2-1)=VFG(JM*2-1)+(FXI/6.+FXJ/3.)*D
      VFG(JM*2)=VFG(JM*2)+(FYI/6.+FYJ/3.)*D
      GOTO 400
310   DE=2.*FT*D
      VFG(JN*2-1)=VFG(JN*2-1)+(FXI*(X1/4.+X2/12.)+FXJ*(X2+X1/12.))*DE
      VFG(JN*2)=VFG(JN*2)+(FYI*(X1/4.+X2/12.)+FYJ*(X2+X1/12.))*DE
      VFG(JM*2-1)=VFG(JM*2-1)+(FXI*(X1+X2/12.)+FXJ*(X2/4.+X1/12.))*DE
      VFG(JM*2)=VFG(JM*2)+(FYI*(X1+X2/12.)+FYJ*(X2/4.+X1/12.))*DE
400   CONTINUE
410   CONTINUE
420   IF(IC.LT.4)GOTO 2900
      DO 2201 I=1,9
2201  W(I)=1.
      DO 2202 I=1,NELT
      DO 2202 J=1,9
2202  W2(I,J)=1.
2900  DO 440 IE=1,NELT
      DO 430 IN=1,NNEL
      J=KCONEC(IN,IE)
      VCORE(1,IN)=VCORG(1,J)
      VCORE(3,IN)=VCORG(3,J)
430   VCORE(2,IN)=VCORG(2,J)
      CALL LOCEF(KCONEC(1,IE),NNEL,NDLN,KLOCE)
      KP=1
      GAM=GAMA(IE)
      EJ=EJA(IE)
      AJ=AJY(IE)
      ITT=IEAU(IE)
      AI=AIZ(IE)
      SE=SEC(IE)
      A1=P(MAT(IE))
      A2=E(MAT(IE))
      GXE=GX(MAT(IE))
      GYE=GY(MAT(IE))
      GZE=GZ(MAT(IE))
      IF(IC.NE.5)GOTO 2125
      IF(IPL.NE.0)GOTO 2125
      DO 2118 LP=1,N
      IF(IS(LP).NE.IE)GOTO 2119
      IF(IT(LP).NE.1)GOTO 2121
      W2(IE,1)=0.
```

```
      W2(IE,2)=0.
      W2(IE,3)=0.
 2121 IF(IT(LP).NE.2)GOTO  2122
      W2(IE,2)=0.
      W2(IE,3)=0.
      W2(IE,4)=0.
 2122 IF(IT(LP).NE.3)GOTO  2123
      W2(IE,5)=0.
      W2(IE,6)=0.
      W2(IE,7)=0.
 2123 IF(IT(LP).NE.4)GOTO  2124
      W2(IE,7)=0.
      W2(IE,6)=0.
      W2(IE,8)=0.
 2124 IF(IT(LP).NE.5)GOTO 2118
      W2(IE,9)=0.
 2118 CONTINUE
 2119 DO 2206 I=1,9
 2206 W(I)=W2(IE,I)
 2125 CALL ELEM02(VCORE,VKE,VFE,RELT DEV,VBE,GXE,GYE,KP,IC,NEQ,A1,A2,
     1 VPE,A1,GAM,NDIM,NDLE,SE,EJ,AJ,ITT,G,GZE,VAE,VRE,W)
      CALL ASSEMB(VKE,VFE,KLOCE,NDLE,NEQ,VKG,VFG)
  540 CONTINUE
C      CONDITIONS AUX LIMITES
      IF(IC.LT.4)GOTO 851
      DO 820 I=1,NEQ
      WRITE(X,720)
  720 FORMAT(2X,'DONNER LE NOEUD , AYANT DES DEPLACEMENTS ET ROTATINS
     1IMPOSES SUIVANT RESPECTIVEMENT X,Y,Z'
     1,/,1X,'DONNER ZERO SI CEST IMPOSE ,1 SI CEST LIBRE ')
      READ(X,730) IX,J1,J2,J3,J4,J5,J6
  730 FORMAT(I5,6I1)
      IF(IX.LE.0) GOTO 540
      WRITE(X,740) IX,J1,J2,J3,J4,J5,J6
  740 FORMAT(1X,' VALEUR IMPOSEE AU NOEUD ',I5,'=',6(1X,I1))
      GOTO 850
  860 IN3=NEQ*NEQ
      IN1=(IQ-1)*NEQ+IQ
      IN1=IN3-NEQ+IQ
      DO 840 J=IQ,IN1,NEQ
      VKG(J)=0.
  840 CONTINUE
      DO 830 J=1,NEQ
      IN2=IN1-IQ+J
      VKG(IN2)=0.
  830 CONTINUE
      VKG(IN1)=1.
      VFG(IQ)=0.
  850 IF(J1.EQ.1) GOTO 750
      IQ=IX*6-5
      J1=1
      GOTO 860
  750 IF(J2.EQ.1) GOTO 760
      IQ=IX*6-4
```

```
      J2=1
      GOTO 860
760   IF(J3.EQ.1) GOTO 770
      IQ=IX*6-3
      J3=1
      GOTO 860
770   IF(J4.EQ.1) GOTO 780
      IQ=IX*6-2
      J4=1
      GOTO 860
780   IF(J5.EQ.1) GOTO 7°0
      IQ=IX*6-1
      J5=1
      GOTO 860
790   IF(J6.EQ.1) GOTO 820
      IQ=IX*6
      J6=1
      GOTO 860
820   CONTINUE
      GOTO 540
451   DO 530 I=1,NEQ
      WRITE(*,450)
450   FORMAT(2X,'DONNER LE NOEUD,AYANT DES DEPACEMENTS.IMPOSES SUIVANT
     1X ET Y
     1 ',/,1X,'-00 NOEUD FIXE DANS LES DEUX DIRECTIONS',/,1X,'-10 NOEUD
     1 FIXE DANS SUIVANT Y',/,1X,'-01 NOEUD FIXE SUIVANT X',/,1X,'-DONNEZ
     1 0 LA VALEUR DU NOEUD POUR SORTIR')
      READ(*,460) IV,MX2,MY2
460   FORMAT(I5,2I1)
      IF(IV.LE.0) GOTO 540
      WRITE(*,470) IV,MX2,MY2
470   FORMAT(' VALEUR IMPOSEE AU NOEUD ',I5,'=',I1,1X,I1)
      IF(MX2.EQ.1) GOTO 480
      IF(MY2.EQ.1) GOTO 490
      IU1=IV*2
      IU2=IV*2-1
      GOTO 500
480   IU1=IV*2
      IU2=IU1
      GOTO 500
490   IU1=IV*2-1
      IU2=IU1
500   DO 530 IQ=IU2,IU1
      IN3=NEQ*NEQ
      INI=(IQ-1)*NEQ+IQ
      IN1=IN3-NEQ+IQ
      DO 510 J=IQ,IN1,NEQ
      VKG(J)=0.
510   CONTINUE
      DO 520 J=1,NEQ
      IN2=INI-IQ+J
      VKG(IN2)=0.
520   CONTINUE
      VKG(INI)=1.
```

```
550   VFG(IQ)=0.
540   CALL RESOL(RSTM,NEQ,VFG,VFG)
      IF(IC.LT.4)GOTO 549
      DO 561 IZ=1,NNT
      I1=(IZ-1)*NDLN+1
      I2=IZ*NDLN
561   WRITE(*,571) IZ,VCORG(1,IZ),VCORG(2,IZ),VCORG(3,IZ),(VFG(I),I=I1,
     1 I2)
571   FORMAT (I5,3E12.5,/,1X,6E12.5,/)
      GOTO 1513
549   WRITE(*,550)
550   FORMAT(' NOEUD      X            Y            U            V '/)
      DO 560 IZ=1,NNT
      I1=(IZ-1)*NDLN+1
      I2=IZ*NDLN
560   WRITE(*,570) IZ,VCORG(1,IZ),VCORG(2,IZ),(VFG(I),I=I1,I2)
570   FORMAT (I5,4E12.5)
1513  WRITE(*,580)
580   FORMAT(1X,'SI CONTR DONNER VALEUR DIFFERENT DE ZERO ')
      READ(*,590)IB
590   FORMAT(I3)
      IF(IB.EQ.0) GOTO 690
      KP=2
      DO 610 IM=1,NELT
      DO 600 ID=1,NNEL
      J=KCONEC(ID,IM)
      VCORE(1,ID)=VCORG(1,J)
      VCORE(3,ID)=VCORG(3,J)
600   VCORE(2,ID)=VCORG(2,J)
      A1=P(MAT(IM))
      A2=E(MAT(IM))
      A3=A2/(3-6*A1)
      ITT=IEAU(IM)
      GAM=GAMA(IM)
      EJ=EJA(IM)
      AJ=AJY(IM)
      AI=AIZ(IM)
      SE=SEC(IM)
      A1=P(MAT(IM))
      A2=E(MAT(IM))
      GXE=GX(MAT(IM))
      GYE=GY(MAT(IM))
      GZE=GZ(MAT(IM))
      IF(IPL.NE.0)GOTO 1137
      DO 1131 I=1,9
1131  W(I)=W2(IM,I)
1137  CALL ELEM02(VCORE,VFE,VFL,NELT,DEV,VBE,GXE,GYE,KP,IC,NEQ,A1,A2,
     1 VPT,AI,GAM,NDIM,NDLE,SE,EJ,AJ,ITT,G,GZE,VAE,VRE,W)
      L3=KCONEC(3,IM)*NDLN
      L1=KCONEC(1,IM)*NDLN
      L2=KCONEC(2,IM)*NDLN
      L4=KCONEC(1,IM)
      L5=KCONEC(2,IM)
      CALL CONTR(VBE,DEV,VFG,L1,L2,L3,DONT,IC,ITT,DIV,A3,VAE,VRE,DOIT)
```

```
      CONT(IM,1)=DONT(1)
      CONT(IM,2)=DONT(2)
      CONT(IM,3)=DONT(3)
      CONT(IM,4)=DONT(4)
      CONT(IM,5)=DONT(5)
      CONT(IM,6)=DONT(6)
      COIT(IM,1)=DOIT(1)
      COIT(IM,2)=DOIT(2)
      COIT(IM,3)=DOIT(3)
      COIT(IM,4)=DOIT(4)
      IF(IC.LT.4)GOTO 610
      DO 1528 I=1,12
      FMN(I,IM)=0.
      DO 1528 J=1,6
      FMN(I,IM)=FMN(I,IM)+VAE(I,J)*DONT(J)
1528  CONTINUE
      IF(IC.NE.5)GOTO 610
      DO 5528 J=1,6
5528  DONT(J)=DABS(DONT(J))
      IF(DONT(2).LE.APE)GOTO 2400
      RP(IM,1)=DABS(AMP(L4,3)/DONT(2))
      GOTO 2401
2400  RP(IM,1)=GRAND
2401  IF(DONT(3).LE.APE)GOTO 2402
      RP(IM,2)=DABS(AMP(L5,3)/DONT(3))
      GOTO 2404
2402  RP(IM,2)=GRAND
2404  IF(DONT(4).LE.APE)GOTO 2405
      RP(IM,3)=DABS(AMP(L4,2)/DONT(4))
      GOTO 2406
2405  RP(IM,3)=GRAND
2406  IF(DONT(5).LE.APE)GOTO 2407
      RP(IM,4)=DABS(AMP(L5,2)/DONT(5))
      GOTO 2408
2407  RP(IM,4)=GRAND
2408  IF(DONT(6).LE.APE)GOTO 2409
      RP(IM,5)=DABS(AMP(L4,1)/DONT(6))
      GOTO 610
2409  RP(IM,5)=GRAND
610   CONTINUE
      IF(IC.NE.4)GOTO 1520
      IO=6
      WRITE(*,1521)
1521  FORMAT(1X,'ELEMENT       TAB            MABZ           MBAZ           MABY
     1       MBAY        M-TORTION')
      DO 2713 IM=1,NELT
      WRITE(*,1523)IM,(CONT(IM,JL),JL=1,10)
1523  FORMAT(1X,I5,6E12.5)
2713  CONTINUE
      WRITE(*,1529)
1529  FORMAT(1X,'       NOEUD,       FX             FY             FZ             MX
     1          MY             MZ')
      DO 2613 IM=1,NELT
      WRITE(*,1530)IM,L1,(FMN(JL,IM),JL=1,10)
```

```
1530  FORMAT(1H,1X,I3,1X,I5,6E12.5)
      WRITE(*,1531)IM,L5,(SIGMA(JL,IM),JL=19,12)
1531  FORMAT(1H,1X,I5,1X,I5,6E12.5)
1813  CONTINUE
      GOTO 690
1520  IF(IC.NE.3)GOTO 8500
      IO=4
      WRITE(*,620)
620   FORMAT(1X,' ELEMENT    SIGMA X     SIGMA Y     GAMA XY       E`XY ')
      DO 2731 IM=1,NELT
      WRITE(*,640)IM,(SIGMA(IM,JL),JL=1,10)
640   FORMAT(1H,1X,I5,4E12.5)
2731  CONTINUE
      WRITE(*,621)
621   FORMAT(1X,'              UX          UY          UXY         EUXY ')
      DO 2732 IM=1,NELT
      WRITE(* 641)IM,(COTT(IM,JL),JL=1,10)
641   FORMAT(1H,1X,I5,4E12.5)
2732  CONTINUE
      GOTO 690
8500  IF(IC.GE.3)GOTO 5114
      IO=5
      WRITE(*,922)
922   FORMAT(1X,'  ELEMENT     SIGMA X      SIGMA Y       GAMA XY  ')
      DO 2733 IM=1,NELT
      WRITE(*,941)IM,(CONT(IM,JL),JL=1,10)
941   FORMAT(1H,1X,I5,3E12.5)
2733  CONTINUE
      WRITE(*,661)
661   FORMAT(1X,'                UX           UY            UXY   ')
      DO 2734 IM=1,NELT
      WRITE(*,681)IM,(COTT(IM,JL),JL=1,10)
681   FORMAT(1H,1X,I5,3E12.5)
2734  CONTINUE
      GOTO 690
5114  IF(IC.NE.5)GOTO 690
      TES=RF(1,1)+1
      DO 2113 IM=1,NELT
      DO 2113 K=1,5
      IF(RF(IM,K).LT.TES)GOTO 2114
      TES=TES
      GOTO 2113
2114  TES=RF(IM,K)
2113  CONTINUE
      N=0
      DO 2115 IM=1,NELT
      DO 2115 K=1,5
      IF(RF(IM,K).NE.TES)GOTO 2115
      N=N+1
      I5=KCONEC(2,IM)
      I4=KCONEC(1,IM)
      IF(K.NE.1)GOTO 4101
      AMP(I4,3)=GRARD
4101  IF(K.NE.2)GOTO 4102
```

```
      [illegible]
      [illegible] RP(IM,5),RP(IM,3),RP(IM,2)
      [illegible] ,I3,' ',3E12.5)
      AMP(L5,3)=GRAND
4102  IF(K.NE.3)GOTO 4103
      AMP(L4,2)=GRAND
4103  IF(K.NE.4)GOTO 4104
      AMP(L5,2)=GRAND
4104  IF(K.NE.5)GOTO 4105
      AMP(L4,1)=GRAND
4105  IS(N)=IM
      IT(N)=K
2115  CONTINUE
      DO 2430 IM=1,NELT
      L4=KCONEC(1,IM)
      L5=KCONEC(2,IM)
      IF(IM.EQ.1)GOTO 2981
      L8=IM-1
      L6=KCONEC(1,L8)
      L7=KCONEC(2,L8)
      IF(L4.EQ.L7)GOTO 2983
2981  AMP(L4,1)=AMP(L4,1)-TES*DABS(CONT(IM,6))
      AMP(L4,2)=AMP(L4,2)-TES*DABS(CONT(IM,4))
      AMP(L4,3)=AMP(L4,3)-TES*DABS(CONT(IM,2))
      IF(IM.EQ.1)GOTO 2983
2983  AMP(L5,2)=AMP(L5,2)-TES*DABS(CONT(IM,5))
      AMP(L5,3)=AMP(L5,3)-TES*DABS(CONT(IM,3))
2430  CONTINUE
      IO=6
      WRITE(*,1192)
1192  FORMAT(1X,'ELEMENT      TAB          MABZ          MBAZ          MABY
     1      MBAY        M-TORTION')
      DO 2191 IM=1,NELT
      WRITE(*,1193)IM,(CONT(IM,JL),JL=1,10)
1193  FORMAT(1X,I5,6E12.5)
2191  CONTINUE
      WRITE(*,2140)
2140  FORMAT(1X,'CAPACITE RESIDUELLE          X          Y          Z ')
      DO 2431 IM=1,NELT
      L4=KCONEC(1,IM)
      L5=KCONEC(2,IM)
      WRITE(*,2141)L4,(AMP(L4,LO),LO=1,3)
2141  FORMAT(9X,'NOEUD ',I3,' ',3E12.5)
      WRITE(*,2541)L5,(AMP(L5,LO),LO=1,3)
2541  FORMAT(9X,'NOEUD ',I3,' ',3E12.5)
2431  CONTINUE
      WRITE(*,2592)
2592  FORMAT(1X,'FACT FORME SUIVANT          X          Y          Z ')
      DO 2861 IM=1,NELT
      L4=KCONEC(1,IM)
      L5=KCONEC(2,IM)
      WRITE(*,2171)L4,RP(IM,5),RP(IM,3),RP(IM,1)
2171  FORMAT(9X,'NOUED ',I3,' ',3E12.5)
      WRITE(*,2761)L5,RP(IM,5),RP(IM,4),RP(IM,2)
2761  FORMAT(9X,'NOUED ',I3,' ',3E12.5)
2861  CONTINUE
      WRITE(*,2142)
```

```
2142  FORMAT(1X,'  MINIMUM                        ')
      WRITE(*,2143)TES
2143  FORMAT(1X,F7.4)
      LT=LT+1
      TES2(LT)=TES
      IPL=0     5
      DO 2116 L=1,NEQ*NEQ
2116  VKG(L)=0.
      DO 2450 L=1,NEQ
2450  VFG(L)=VFP(L)
      IF(LT.EQ.JPL+1)GOTO 2148
      GOTO 2900
2148  S=0.
      DO 2147 I=1,LT
2147  S=S+TES2(I)
      WRITE(*,2145)
2145  FORMAT(1X,'  FACTEUR DE PLASTICITE       ')
      WRITE(*,2146)S
2146  FORMAT(1X,F7.4)
2900  STOP
      END    6
```

```
      SUBROUTINE ELEM02(VCORE,VKE,VFE,NELT,DEV,VBE,GXE,GYE,KP,IC,
     1 NEQ,A1,A2,VPE,AI,GAM,NDIM,NDLE,SE,EJ,AJ,ITT,G,GZE,VAE,VRE,W)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION VCORE(3,3),VKE(NDLE,NDLE),VFE(NDLE),VBE(4,6),
     1 VAE(12,6),VSE(6,12),W(9)
      DIMENSION DBE(4,6),VRE(6,6),VME(12,6),VPE(12,12),DEV(4,4)
      DATA PI/3.141593/
C     MATRICE ELEMENTAIRE
      UN=-1
      IF(IC.LT.4)GOTO 300
      X1=VCORE(1,1)
      X2=VCORE(1,2)
      Y1=VCORE(2,1)
      Y2=VCORE(2,2)
      Z1=VCORE(3,1)
      Z2=VCORE(3,2)
      C23=Y2-Y1
      C33=X2-X1
      C21=Z2-Z1
      C13=X1*Y2-X2*Y1
      EL1=(C33*C33+C23*C23)**0.5
      EL=(EL1*EL1+C21*C21)**0.5
      CS=DCOS(GAM*PI/180)
      SN=DSIN(GAM*PI/180)
C     G=A2/(2*(1+A1))
      DO 302 I=1,6
      DO 302 J=1,6
302   VRE(I,J)=0.
      VRE(1,1)=A2*SE/EL
      VRE(2,2)=4*A2*AI*W(1)/EL
      VRE(2,3)=2*A2*AI*W(2)/EL
      VRE(3,2)=2*A2*AI*W(3)/EL
      VRE(3,3)=4*A2*AI*W(4)/EL
      VRE(4,4)=4*A2*AJ*W(5)/EL
      VRE(4,5)=2*A2*AJ*W(6)/EL
      VRE(5,4)=2*A2*AJ*W(7)/EL
      VRE(5,5)=4*A2*AJ*W(8)/EL
      VRE(6,6)=G*EJ*W(9)/EL
      DO 301 I=1,12
      DO 301 J=1,6
301   VME(I,J)=0.
      VME(1,1)=1.
      VME(2,2)=UN/EL
      VME(2,3)=UN/EL
      VME(3,4)=1./EL
```

```
      VME(3,5)=1./EL
      VME(4,6)=UN
      VME(5,4)=UN
      VME(6,2)=UN
      VME(7,1)=UN
      VME(8,2)=1./EL
      VME(8,3)=1./EL
      VME(9,4)=UN/EL
      VME(9,5)=UN/EL
      VME(10,6)=1.
      VME(11,5)=UN
      VME(12,3)=UN
      VPE(1,1)=-C33/EL
      VPE(1,2)=C23*CS/EL1+C33*C21*SN/(EL1*EL)
      VPE(1,3)=-C23*SN/EL1+C33*C21*CS/(EL1*EL)
      VPE(2,1)=-C23/EL
      VPE(2,2)=-C33*CS/EL1+C23*C21*SN/(EL1*EL)
      VPE(2,3)=C33*SN/EL1+C23*C21*CS/(EL1*EL)
      VPE(3,1)=-C21/EL
      VPE(3,2)=-EL1*SN/EL
      VPE(3,3)=-EL1*CS/EL
      VPE(4,4)=VPE(1,1)
      VPE(4,5)=VPE(1,2)
      VPE(4,6)=VPE(1,3)
      VPE(5,4)=VPE(2,1)
      VPE(5,5)=VPE(2,2)
      VPE(5,6)=VPE(2,3)
      VPE(6,4)=VPE(3,1)
      VPE(6,5)=VPE(3,2)
      VPE(6,6)=VPE(3,3)
      VPE(7,7)=VPE(1,1)
      VPE(7,8)=VPE(1,2)
      VPE(7,9)=VPE(1,3)
      VPE(8,7)=VPE(2,1)
      VPE(8,8)=VPE(2,2)
      VPE(8,9)=VPE(2,3)
      VPE(9,7)=VPE(3,1)
      VPE(9,8)=VPE(3,2)
      VPE(9,9)=VPE(3,3)
      VPE(10,10)=VPE(1,1)
      VPE(10,11)=VPE(1,2)
      VPE(10,12)=VPE(1,3)
      VPE(11,10)=VPE(2,1)
      VPE(11,11)=VPE(2,2)
      VPE(11,12)=VPE(2,3)
      VPE(12,10)=VPE(3,1)
      VPE(12,11)=VPE(3,2)
      VPE(12,12)=VPE(3,3)
      DO 55 J=1,12
      DO 55 K=1,6
      VAE(J,K)=0.
      DO 55 L=1,12
      VAE(J,K)=VAE(J,K)+VPE(J,L)*VME(L,K)
55    CONTINUE
```

```
      DO 66 K=1,12
      VSE(J,K)=0.
      DO 66 L=1,6
      VSE(J,K)=VSE(J,K)+VRE(J,L)*VHE(K,L)
66    CONTINUE
      DO 77 J=1,12
      DO 77 K=1,12
      VKE(J,K)=0.
      DO 77 L=1,6
      VKE(J,K)=VKE(J,K)+VHE(J,L)*VSE(L,K)
77    CONTINUE
      GOTO 40
300   X1=VCORE(1,1)
      X2=VCORE(1,2)
      X3=VCORE(1,3)
      Y1=VCORE(2,1)
      Y2=VCORE(2,2)
      Y3=VCORE(2,3)
      C23=Y1-Y2
      C21=Y2-Y3
      C33=X2-X1
      C22=Y3-Y1
      C32=X1-X3
      C31=X3-X2
      C11=X2*Y3-X3*Y2
      C12=X3*Y1-X1*Y3
      C13=X1*Y2-X2*Y1
      XG=(X1+X2+X3)/3
      YG=(Y1+Y2+Y3)/3
      EP=2*PI*XG
C     CONTRAINTES PLANES
      B1=1.0
      B2=B1-A1
      B3=B2/2
      IF(IC.EQ.1) GOTO 15
C     DEFORMATIONS PLANES
      B1=B1-A1
      B2=B1-A1
      B3=B2/2
15    EB=A2/((1.0+A1)*B2)
      IF(KP.EQ.2)GOTO 251
      IF(IT1.NE.1) GOTO 251
      AK=100*A2/(3-3*A1)
      GOTO 253
251   AK=0
C     MATRICE DEV
253   DEV(1,1)=EB*B1+AK
      DEV(1,2)=EB*A1
      DEV(1,3)=0.
      DEV(2,1)=DEV(1,2)
      DEV(2,2)=DEV(1,1)
      DEV(2,3)=0.
      DEV(3,1)=0.
```

```
      DEV(3,2)=0.
      DEV(3,3)=EB*B3
      IF(IC.NE.3) GOTO 20
      DEV(1,1)=EB*B1+AK
      DEV(1,2)=EB*A1
      DEV(1,3)=EB*A1
      DEV(1,4)=0.
      DEV(2,1)=EB*A1
      DEV(2,2)=EB*B1+AK
      DEV(2,3)=EB*A1
      DEV(2,4)=0.
      DEV(3,1)=EB*A1
      DEV(3,2)=EB*A1
      DEV(3,3)=EB*B1
      DEV(3,4)=0.
      DEV(4,1)=0.
      DEV(4,2)=0.
      DEV(4,3)=0.
      DEV(4,4)=EB*B3+AK
20    B=X2*Y3+X1*Y2+X3*Y1-X2*Y1-X1*Y3-Y2*X3
      IF(IC.NE.3)GOTO 25
      T=EP
      GOTO 35
25    T=1.
35    VBE(1,1)=T*C21/B
      VBE(1,2)=0.
      VBE(1,3)=T*C22/B
      VBE(1,4)=0.
      VBE(1,5)=T*C23/B
      VBE(1,6)=0.
      VBE(2,1)=0.
      VBE(2,2)=T*C31/B
      VBE(2,3)=0.
      VBE(2,4)=T*C32/B
      VBE(2,5)=0.
      VBE(2,6)=T*C33/B
      VBE(3,1)=T*C31/B
      VBE(3,2)=T*C21/B
      VBE(3,3)=T*C32/B
      VBE(3,4)=T*C22/B
      VBE(3,5)=T*C33/B
      VBE(3,6)=T*C23/B
      VBE(4,1)=0.
      VBE(4,2)=((C11/XG)+C21+(C31*(YG/XG)))*T/B
      VBE(4,3)=0.
      VBE(4,4)=((C12/XG)+C22+(C32*(YG/XG)))*T/B
      VBE(4,5)=0.
      VBE(4,6)=((C13/XG)+C23+(C33*(YG/XG)))*T/B
      IF(KP.EQ.2)GOTO 40
      FG1=B*GXE/6
      FG2=B*GYE/6
      VFE(1)=FG1
      VFE(2)=FG2
      VFE(3)=FG1
```

```
      VFE(4)=FG2
      VFE(5)=FG1
      VFE(6)=FG2
      IF(IC.NE.3)GOTO 50
      IZ=6
      IH=4
      GOTO 60
50    IZ=6
      IH=3
60    DO 70 I=1,IH
      DO 70 K=1,IZ
      DBE(I,K)=0.
      DO 70 J=1,IH
      DBE(I,K)=DBE(I,K)+(DEV(I,J)*VBE(J,K)*B)/2
70    CONTINUE
      DO 80 I=1,IZ
      DO 80 K=1,IZ
      VKE(I,K)=0.
      DO 80 J=1,IH
      VKE(I,K)=VKE(I,K)+VBE(J,I)*DBE(J,K)
80    CONTINUE
40    RETURN
      END
```

```
      SUBROUTINE CONTR(VBE,DEV,VFG,L1,L2,L3,DONT,IC,ITT,DIV,A3,VAE,VRE,
     1DOTT)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION BR(4,6),VBE(4,6),DOTT(4),DIV(4,4),UL(6),VRE(6,6)
      DIMENSION DONT(6),DEV(4,4),VFG(100),U(12),VAE(12,6)
      IF(IC.LT.4)GOTO 51
      U(1)=VFG(L1-5)
      U(2)=VFG(L1-4)
      U(3)=VFG(L1-3)
      U(4)=VFG(L1-2)
      U(5)=VFG(L1-1)
      U(6)=VFG(L1)
      U(7)=VFG(L2-5)
      U(8)=VFG(L2-4)
      U(9)=VFG(L2-3)
      U(10)=VFG(L2-2)
      U(11)=VFG(L2-1)
      U(12)=VFG(L2)
      DO 52 I=1,6
      UL(I)=0.
      DO 52 K=1,12
      UL(I)=UL(I)+VAE(K,I)*U(K)
52    CONTINUE
      DO 53 I=1,6
      DONT(I)=0.
      DO 53 K=1,6
      DONT(I)=DONT(I)+VRE(I,K)*UL(K)
53    CONTINUE
      GOTO 50
51    U(1)=VFG(L1-1)
      U(2)=VFG(L1)
      U(3)=VFG(L2-1)
      U(4)=VFG(L2)
      U(5)=VFG(L3-1)
      U(6)=VFG(L3)
      IF(IC.NE.3)GOTO 30
      IH=4
      IZ=6
      GOTO 40
30    IH=3
      IZ=6
40    DO 10 I=1,IH
      DO 10 K=1,IZ
      BR(I,K)=0.
      DO 10 J=1,IH
```

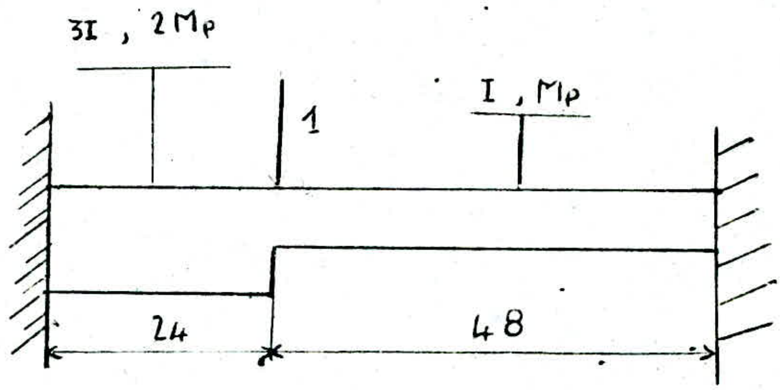
```
      DR(I,K)=DR(I,K)+DEV(I,J)*VBE(J,K)
10    CONTINUE
      DO 20 I=1,IH
      DONT(I)=0.
      DO 20 K=1,IZ
      DONT(I)=DONT(I)+DR(I,K)*U(K)
20    CONTINUE
      IF(IIT.NE.1)GOTO 50
      DIV(1,1)=100*A3
      DIV(2,2)=100*A3
      DIV(3,3)=0.
      DIV(4,4)=100*A3
      IF(IC.NE.3)GOTO 31
      IH=4
      IZ=6
      GOTO 41
31    IH=3
      IZ=6
41    DO 11 I=1,IH
      DO 11 K=1,IZ
      DR(I,K)=0.
      DO 11 J=1,IH
      DR(I,K)=DR(I,K)+DIV(I,J)*VBE(J,K)
11    CONTINUE
      DO 21 I=1,IH
      DONT(I)=0.
      DO 21 K=1,IZ
      DOTT(I)=DOTT(I)+DR(I,K)*U(K)
21    CONTINUE
50    RETURN
      END
```

```
      SUBROUTINE ASSEMB(VKE,VFE,KLOCE,IDLE,NEQ,VKG,VFG)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION VKE(IDLE,IDLE),VFE(IDLE),KLOCE(IDLE)
      DIMENSION VKG(NEQ,NEQ),VFG(NEQ)
      DO 10 ID=1,IDLE
      I=KLOCE(ID)
      VFG(I)=VFG(I)+VFE(ID)
      DO 10 JD=1,IDLE
      J=KLOCE(JD)
10    VKG(I,J)=VKG(I,J)+VKE(ID,JD)
      RETURN
      END


      SUBROUTINE LOCEF(KCONEC,NNEL,NDLN,KLOCE)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION KCONEC(1),KLOCE(1)
      J=0
C     BOUCLE SUR LES NNEL
      DO 10 IN=1,NNEL
      IDO=(KCONEC(IN)-1)*NDLN
C     BOUCLE SUR LES NDLN
      DO 10 ID=1,NDLN
      J=J+1
10    KLOCE(J)=ID+IDO
      RETURN
      END
```

```
      SUBROUTINE RESOL (NSYM,NEQ,VKG,VFG)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION VKG(NEQ,NEQ),VFG(NEQ)
      DATA ZERO/0.D0/
      N1=NEQ-1
      DO 50 IS=1,N1
      PIV=VKG(IS,IS)
      IF(PIV) 20,10,20
10    WRITE(*,2000) IS
2000  FORMAT(' PIVOT NUL EQUATION ',I5)
      STOP
20    IS1=IS+1
      DO 50 II=IS1,NEQ
      CL=VKG(II,IS)/PIV
      IF(CL.EQ.ZERO) GO TO 50
      VFG(II)=VFG(II)-CL*VFG(IS)
      IF(NSYM.NE.1) GO TO 32
      DO 30 IJ=IS1,NEQ
30    VKG(II,IJ)=VKG(II,IJ)-CL*VKG(IS,IJ)
      GOTO 50
32    DO 40 IJ=IS,NEQ
40    VKG(II,IJ)=VKG(II,IJ)-CL*VKG(IS,IJ)
50    CONTINUE
      VFG(NEQ)=VFG(NEQ)/VKG(NEQ,NEQ)
      DO 70 II = 1,N1
      IS1=IS1-1
      CL=ZERO
      IJ1=IS1+1
      DO 60 IJ = IJ1,NEQ
60    CL=CL+VKG(IS1,IJ)*VFG(IJ)
70    VFG(IS1)=(VFG(IS1)-CL)/VKG(IS1,IS1)
      RETURN
      END
```

EXEMPLE



Données $M_p = 12$, $I_y = 0{,}4\ 10^{-3}$, $I_z = 0{,}001$

$J = 0{,}00015$, $A = 0{,}02$, $E = 200\ 10^6$

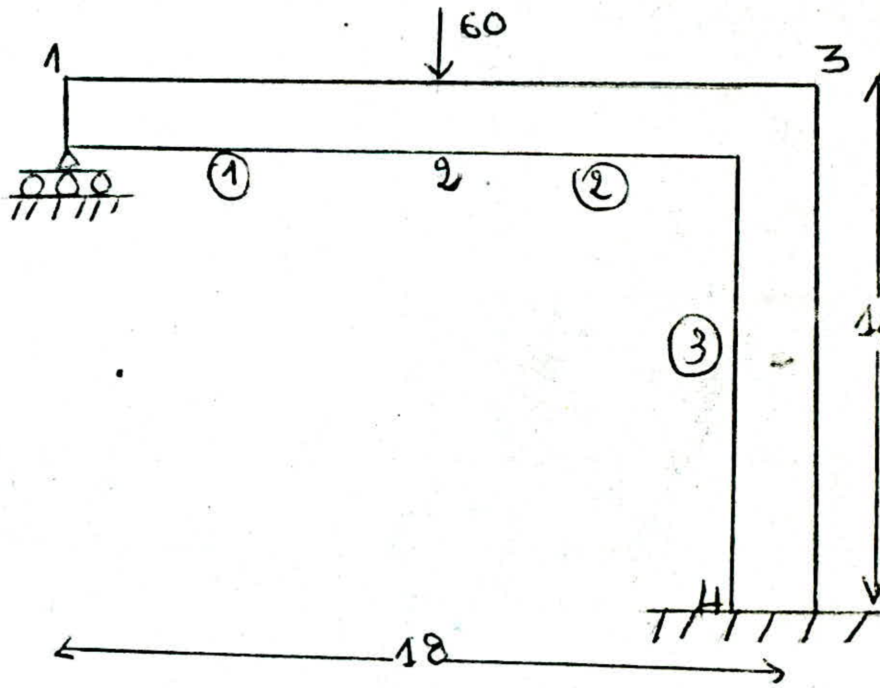
$G = 80\ 10^6$

```
ELEMENT      NAB           MABZ          MBAZ          MABY          MBAY         M-TORTI
   1     .00000E+00   .14528E+02    .51276E+01   .00000E+00   .00000E+00   .00000E+0
   2     .00000E+00  -.51276E+01   -.35508E+01   .00000E+00   .00000E+00   .00000E+0
CAPACITE RESIDUELLE                 X             Y             Z
           NOEUD    1.       .10000E+09    .10000E+09    .10000E+13
           NOEUD    2.       .10000E+09    .10000E+09    .35294E+01
           NOEUD    2.       .10000E+09    .10000E+09    .35294E+01
           NOEUD    3.       .10000E+09    .10000E+09    .61176E+01
COUT FORME SUIVANT   :            X             Y             Z
           NOUED    1.       .10000E+13    .10000E+13   -.16520E+01
           NOUED    2.       .10000E+13    .10000E+13    .23403E+01
           NOUED    2.       .10000E+13    .10000E+13    .23403E+01
           NOUED    3.       .10000E+13    .10000E+13    .33700E+01
  MINIMUM
 1.6520
```

```
ELEMENT      TAB          MABZ          MBAZ          MABY          MBAY          M-TOR
    1  .00000E+00  .00000E+00  .12800E+02  .00000E+00  .00000E+00  .00000E
    2  .00000E+00 -.12800E+02 -.96000E+01  .00000E+00  .00000E+00  .00000E
CAPACITE RESIDUELLE          X            Y            Z
        NOEUD   1    .10000E+09  .10000E+09  .10000E+13
        NOEUD   2    .10000E+09  .10000E+09  .10000E+13
        NOEUD   2    .10000E+09  .10000E+09  .10000E+13
        NOEUD   3    .10000E+09  .10000E+09  .34706E+01
FACT FORME SUIVANT           X            Y            Z
        NOUED   1    .10000E+13  .10000E+13  .10000E+13
        NOUED   2    .10000E+13  .10000E+13  .27574E+00
        NOUED   2    .10000E+13  .10000E+13  .27574E+00
        NOUED   3    .10000E+13  .10000E+13  .63725E+00
  MINIMUM
  .2757
 1
 VALEUR IMPOSEE AU NOEUD      1= 0 0 0 0 0 0
 DONNER LE NOEUD , AYANT DES DEPLACEMENTS ET ROTATINS  IMPOSES SUIVANT RESP
-DONNER ZERO SI CEST IMPOSE  1 SI CEST LIBRE

ELEMENT      TAB          MABZ          MBAZ          MABY          MBAY          M-TORT
    1  .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+00  .00000E+
    2  .00000E+00  .47551E-15 -.40000E+02  .00000E+00  .00000E+00  .00000E+
CAPACITE RESIDUELLE          X            Y            Z
        NOEUD   1    .10000E+09  .10000E+09  .10000E+13
        NOEUD   2    .10000E+09  .10000E+09  .10000E+13
        NOEUD   2    .10000E+09  .10000E+09  .10000E+13
        NOEUD   3    .10000E+09  .10000E+09  .10000E+13
FACT FORME SUIVANT           X            Y            Z
        NOUED   1    .10000E+13  .10000E+13  .10000E+13
        NOUED   2    .10000E+13  .10000E+13  .10000E+13
        NOUED   2    .10000E+13  .10000E+13  .10000E+13
        NOUED   3    .10000E+13  .10000E+13  .72304E-01
  MINIMUM
  .0723
 FACTEUR DE PLASTICITE
 2.0000
Stop - Program terminated.
```

EXEMPLE



ients

-3 $I_y = 0{,}00004$, $I_z = 0{,}001$, $J = 0{,}00015$ $A = 0{,}02$

-4 $I_y = 0{,}00033$, $I_z = 0{,}002$, J $0{,}00096$ $A = 0{,}03$

$E = 200.10^6$ $G = 80.10^6$

ELEMENT	TAB	MABZ	MBAZ	MABY	MBAY	M-TORT
1	.00000E+00	[illegible]	.00000E+00	.00000E+00	.00000E+00	.00000E+[illegible]
2	.00000E+00	.50125E+[illegible]	[illegible]0000E+03	.00000E+00	.00000E+00	.00000E+[illegible]
3	.00000E+00	.54000E+03	.54000E+03	.00000E+00	.00000E+00	.00000E+[illegible]

RAIDEUR RESIDUELLE

		X	Y	Z
NOEUD	1	.10000E+09	.10000E+09	.30000E+03
NOEUD	2	.10000E+09	.10000E+09	.10000E+13
NOEUD	2	.10000E+09	.10000E+09	.10000E+13
NOEUD	3	.10000E+09	.10000E+09	.10000E+13
NOEUD	3	.10000E+09	.10000E+09	.10000E+12
NOEUD	4	.10000E+09	.10000E+09	.10000E+03

DEFORME SUIVANT

		X	Y	Z
NOEUD	1	.10000E+13	.10000E+13	.10000E+13
NOEUD	2	.10000E+13	.10000E+13	.10000E+13
NOEUD	2	.10000E+13	.10000E+13	.10000E+13
NOEUD	3	.10000E+13	.10000E+13	.32321E+00
NOEUD	3	.10000E+13	.10000E+13	.32321E+00
NOEUD	4	.10000E+13	.10000E+13	.50839E+00

minimum
.4232

FACTEUR DE PLASTICITE
1.0667

Stop - Program terminated.

BIBLIOGRAPHIE

1 - H. GALLAGHER, *Introduction aux éléments finis*, 1976

2 - G. DHATT et G. TOUZOT, *une présentation de la méthode des éléments finis*, 1979.

3 - HB. HARRISON *Structural Alalysis* DESIGN 1980

4 - DEMIDOVICH, *Elements de calcul numérique*, 1979.