

وزارة الجامعات والبحث العلمي
Ministère aux Universités et de la Recherche Scientifique

ECOLE NATIONALE POLYTECHNIQUE

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

DEPARTEMENT : Génie. Mécanique

PROJET DE FIN D'ETUDES

SUJET

FORMULATION DE LA METHODE
DE LANCZOS POUR LA RECHERCHE
DES VALEURS PROPRES

Proposé par F. BELBLIDIA

Etudié par : M. ZEKKOUR

Dirigé par : F. BELBLIDIA

PROMOTION

Elève Ingénieur : M. ZEKKOUR

Promoteur : F. BELBLIDIA

ملخص
يهدف هذا العمل إلى دراسة وبرمجة طريقة "LANCZOS" لإيجاد
الأشكال الذاتية والتواترات الذاتية لاستخراج المنظومات
الميكانيكية باستخدام طريقة العناصر المنتهية
لهذا الغرض استعملنا الدليل "DLEARN" كأداة للبرمجة.
في النهاية استعملنا عدة اختبارات للتحقق من حالات
التقارب والإقتصاد.

ABSTRACT :

This work is intended to study and implement the Lanczos block for extracting eigenfrequencies and eigenmodes of mechanical structures modeled by the finite element method. For this purpose we have used as a support for the implementation the Dlearn code package. Finally several tests are applied to check the states of efficiency and economy.

RESUME :

Ce travail est destiné à étudier et implémenter le block Lanczos pour l'extraction des fréquences propres et Modes propres des structures modélisées en éléments finis. Pour cette fin on a utilisé le code Dlearn comme support à l'implantation. Finalement plusieurs tests sont fait pour vérifier les états de convergence et d'économie.

*** REMERCIEMENT **



*Je tiens a remercier vivement mon promoteur monsieur
F. RELBLIDIA pour son encouragement, son aide prèsieuse et son
soutien persistant.*

*Mes remerciements vont également à tous les enseignants
du dèpartement de gènie mècanique spècialement M. A. GAHMOSSE*

*Que tous ceux qui ont contribuè de près ou de loin à ma
formation, trouvent ici ma vive reconnaissance.*

*** DEDICACES ***

Je d die ce modeste travail   :

- MES CHERS PARENTS*

- MES FRERES ET SOEUERS*

- TOUTE MA FAMILLE*

- TOUS MES AMIS*

***** S O M M A I R E *****

CHAPITRE I GENERALITES SUR LA THEORIE VIBRATOIRE

I-1 Introduction	5
I-2 Position du problème	5
I-3 Formulation	5

CHAPITRE II DYNAMIQUE DES SYSTEMES LINEAIRES CONTINUS

II-1 Introduction	13
II-2 Solutions exactes pour le cas d'une barre ..	13
II-3 Presentation M.E.F.	15
II-4 Problèmes dynamiques	18
II-5 Conclusion	21

CHAPITRE III GENERALITES SUR LE PROBLEME AUX VALEURS PROPRES (P.V.P.)

III-1 Introduction	23
III-2 Position du problème	23
III-3 Propriétés fondamentales du P.V.P.	24

CHAPITRE IV METHODES DE RESOLUTION DU P.V.P.

IV-1 Introduction	29
IV-2 Historique	29
IV-3 Choix de la méthode appropriée	30
IV-4 Classification des méthodes de résolution ..	31
IV-5 Commentaires	39
IV-6 Conclusion	40

CHAPITRE V METHODE DE LANCZOS

V-1 Introduction	42
------------------------	----

V-2 Formulation	42
V-3 Génération des vecteurs de Lanczos	43
V-4 Forme Tridiagonale	43
V-5 Le problème réduit	44
V-6 Perte et restauration d'orthogonalité	45
V-7 Résolution du problème réduit	46
V-8 Conclusion	46

CHAPITRE VI PRESENTATION DU LOGICIEL

VI-1 Introduction	48
VI-2 Options d'analyses	48
VI-3 Caractéristiques principales	49
VI-4 Intervention sur le code	53
VI-5 Résultats numériques	54
VI-6 Conclusion	54

CONCLUSION

ANNEXE 1 FORMATION DES MATRICES [K] et [M]	1
----------------------------------------------------	---

ANNEXE 2 PROCEDURES DE REDUCTION

A2-1 Procédure de Rayleigh-Ritz	6
A2-2 Condensation statique	7

ANNEXE 3 DEVELOPPEMENTS MATHEMATIQUES DE L'ALGORITHME DE LANZOS

A3-1 Construction des vecteurs de Lanczos	9
A3-2 Forme Tridiagonale	11
A3-3 Perte et Restauration d'orthogonalité	12

INTRODUCTION

Avant, qu'un système mécanique soit placé dans son environnement opératoire, une étude préalable de son comportement face aux conditions qui lui sont imposées, pendant le fonctionnement, est nécessaire.

La nature des risques peut être mécanique, chimique, thermique, ... selon le cas, le concepteur décidera si un phénomène quelconque figurera ou pas dans ses calculs, en se basant sur l'ampleur d'influence de ce dernier. Nous nous intéressons aux cas de vibrations mécaniques qui peuvent induire :

-Des petites variations de contraintes aux sein du matériau qui peuvent occasionner la fatigue.

-Des chocs qui peuvent détériorer localement le matériau.

-Des bruits émis à l'extérieur, posant des problèmes d'environnement.

Actuellement les systèmes mécaniques sont présent dans des domaines où la ruine imprévue n'est pas tolérée (Aéronautique, espace ...). pour cette raison des méthodes nouvelles d'analyse des structures sont introduites afin de rendre la marge de sûreté plus précise, et renforcer la maîtrise des installations industrielles.

Dans le domaine d'analyse des structures, la méthode des éléments finis a fait preuve d'une très grande efficacité. durant les trois dernières décennies, grâce à l'ordinateur qui a permit l'exploitation de l'outil numérique.

Parmi les problèmes importants rencontrés pendant l'analyse des structures par la MEF; le problème aux valeurs propres qu'on va essayer d'aborder en détail dans cette thèse.

La formulation de ce type de problèmes aboutit à un système d'équations lineaires susceptible de s'exprimer sous forme matricielle, dont les solutions définissent les propriétés modales

de la structure en question. L'ordre des matrices considérées est souvent élevé, et on devra pour le résoudre recourir aux méthodes itératives.

L'objectif finale est de réaliser un logiciel pour traiter ce genre de problèmes.

ORGANISATION DU MEMOIRE

Dans le but de faciliter la lecture du polycopé, les chapitres sont fait de manière résumé. Les développements mathématiques sont relégués aux annexes 1 2 et 3.

CHAPITRE 1: Décrit les aspets fondamentaux de la théorie vibratoire. Le système simple Masse-Ressort-Amortissement est étudié, afin d'entirer certaines conclusions.

CHAPITRE 2: On établi dans ce chapitre les équations décrivant le comportement de structures modélisées en éléments finis.

CHAPITRE 3: Discussion des caractéristiques générales du propbème aux valeurs propres

CHAPITRE 4: Description des plus importantes méthodes de résolution du problème aux valeurs propres employées pour le calcul des structures modélisées en éléments finis.

CHAPITRE 5: Présentation de l'algorithme de Lanczos, du point de vue fondement de la méthode et les différentes étapes à parcourir pour arriver à la solution.

CHAPITRE 6:

BLOCK 1: Une présentation du logiciel du points de vue possibilités d'analyse, organisation et propriétés principales, une comparaison avec le logiciel SAP80 est faite pour deux types d'exemples (Structure en barres, plaque)

BLOCK 2: on fait les tests de validation de la méthode de

Lanczos. du côté précision, et temps d'exécution. Les comparaisons sont faites soit par rapport au SAP80, soit à la solution exacte si elle existe.

ANNEXE 1 : Développement des matrices $[K]$ et $[M]$ pour la version déplacement, on parlera aussi de techniques de diagonalisation.

ANNEXE 2: Développement des procédures de réduction à savoir : La procédure de Rayleigh-Ritz et la technique de condensation statique.

ANNEXE 3: On présente le développement mathématique de l'algorithme de Lanczos, génération des vecteurs de Lanczos, Forme tridiagonale, Problème de perte d'orthogonalité et enfin la résolution du problème réduit.

GENERALITE SUR LA THEORIE

PRINCIPES DES SYSTEMES LINEAIRES

L'étude du système défini par la fig (I-1) présente un grand intérêt, parcequ'elle s'applique à des systèmes de technologie différente. En particulier les structures complexes dont le comportement peut être assimilé à celui du système de la fig(I-1).

I-3 FORMULATION

Autour de la position d'équilibre, l'application du principe de *D'Alembert* (la résultante des forces extérieures est égale à la résultante des forces d'inerties) conduit à l'équation du mouvement suivante:

$$\left\{ \begin{array}{l} m \frac{d^2 x}{d t^2} + c \frac{d x}{d t} + kx = f(t) \\ f_b(t) = kx + c \frac{d x}{d t} \end{array} \right. \quad (I-1)$$

Avec :

- m : la masse du solide.
- K : Raideur du ressort.
- c : coefficient d'amortissement visqueux linéaire.
- f(t) : force imposée.
- f_b(t) : force appliquée sur la base.
- x(t) : déplacement du centre de masse par rapport à la position d'équilibre.

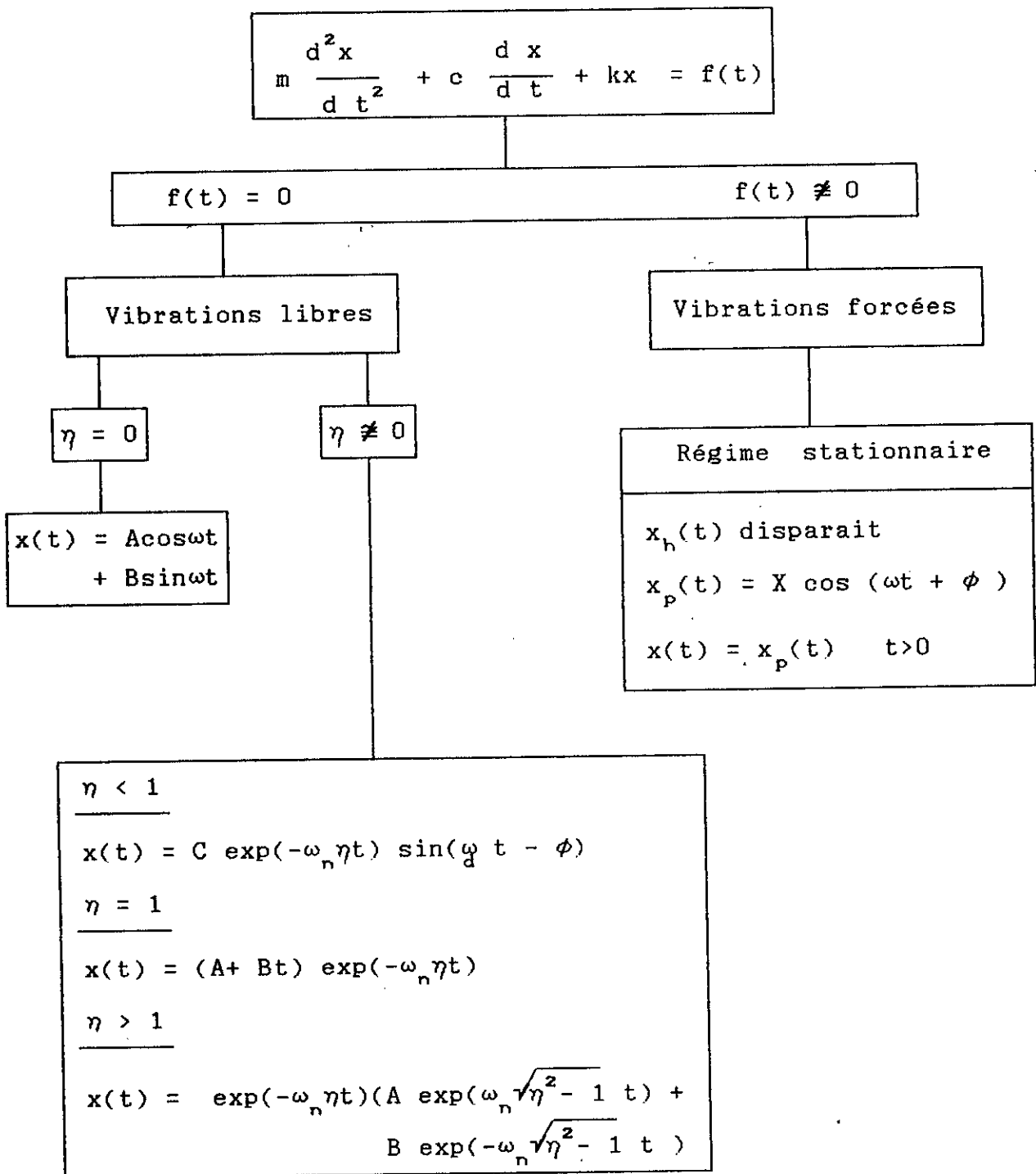
La force appliquée au système si elle n'est pas nulle, peut être harmonique simple, périodique de forme quelconque ou aléatoire. L'étude de tous ces cas dépasse le cadre de notre travail, pour cette raison nous allons nous limiter au cas de chargement harmonique simple, afin de montrer certaines caractéristiques des systèmes dynamiques. (*organigramme*)

I-3-1 VIBRATIONS LIBRES

Dans le cas où le système est écarté de sa position d'équilibre d'une quantité X₀, puis laissé à lui même son comportement sera fonction de facteur

d'amortissement η . On distingue deux cas :

Organigramme



$\eta = \frac{c}{2m\omega}$: Coefficient d'amortissement (Taux d'amortissement).

ω_n : Pulsation propre du système.

ω_d : Pulsation propre amortie.

ω : fréquence d'excitation.

a)- $0 \leq \eta < 1$

Pour un η nul, le système est instable (oscillations sans fin). En pratique un tel cas n'existe pas car l'amortissement ne peut être parfaitement nul, le système donc doit retrouver sa position d'équilibre après un certain nombre de cycles.

b)- $\eta \geq 1$

C'est le cas d'amortissements *critique* et *supercritique* respectivement. Dans ce cas le système revient à sa position d'équilibre sans osciller.

I-3-2 VIBRATIONS FORCÉES

L'action $f(t)$ évolue dans le temps suivant une loi harmonique décrite par :

$$f(t) = f_0 \sin \omega t \quad (I-2)$$

la stationnarité suppose disparu le régime transitoire consécutif à l'application de l'action. On définit trois grandeurs adimensionnelles importantes pour la description du comportement du système.

a)- Le facteur d'amplification de déplacement :

$$\frac{X_o}{X_s} = \frac{X_o}{f_o/k} = \frac{1}{\sqrt{(1 - \beta^2)^2 + 4\eta^2\beta^2}} = T_d \quad (I-3)$$

b)- Le facteur d'amplification de vitesse :

$$T_v = \beta T_d \quad (I-4)$$

c)- Le facteur d'amplification d'accélération :

$$T_a = \beta^2 T_d \quad (I-5)$$

d)- Le facteur de transmission :

$$\frac{f_b(t)}{f(t)} = T \sin \omega t$$

où T définit le facteur de transmission dynamique dont l'expression est donnée par :

$$T = \left[\frac{1 + 4\eta^2 \beta^2}{(1 - \beta^2)^2 + 4\eta^2 \beta^2} \right]^{1/2} \quad (I-6)$$

L'analyse des figures (I-3) et (I-4) montre que l'amplification se fait lentement jusqu'à ($\beta \cong 1/3$), passée cette valeur la croissance devient importante, le maximum est atteint au voisinage de ($\beta = 1$), ceci est pour les cas où ($\eta \neq 0$), par contre si $\eta = 0$ l'amplification devient infinie, On dit que le système est en état de *résonance*. La conclusion étant que la zone définie par ($1/3 < \beta < 1.5\beta$) est dangereuse, car l'amplification des efforts est très importante, chose qui peut facilement provoquer la ruine immédiate du système.

La conception des structures mécaniques chargées dynamiquement doit prévoir l'identification des fréquences propres et modes propres associés afin d'éviter les zones fréquentielles critiques.

REMARQUE: Si les chargements dynamiques appliqués à la structure sont d'une fréquences tel que $\beta < 1/3$, la conception peut être menée sur la base d'une étude statique sans grand risques.

4

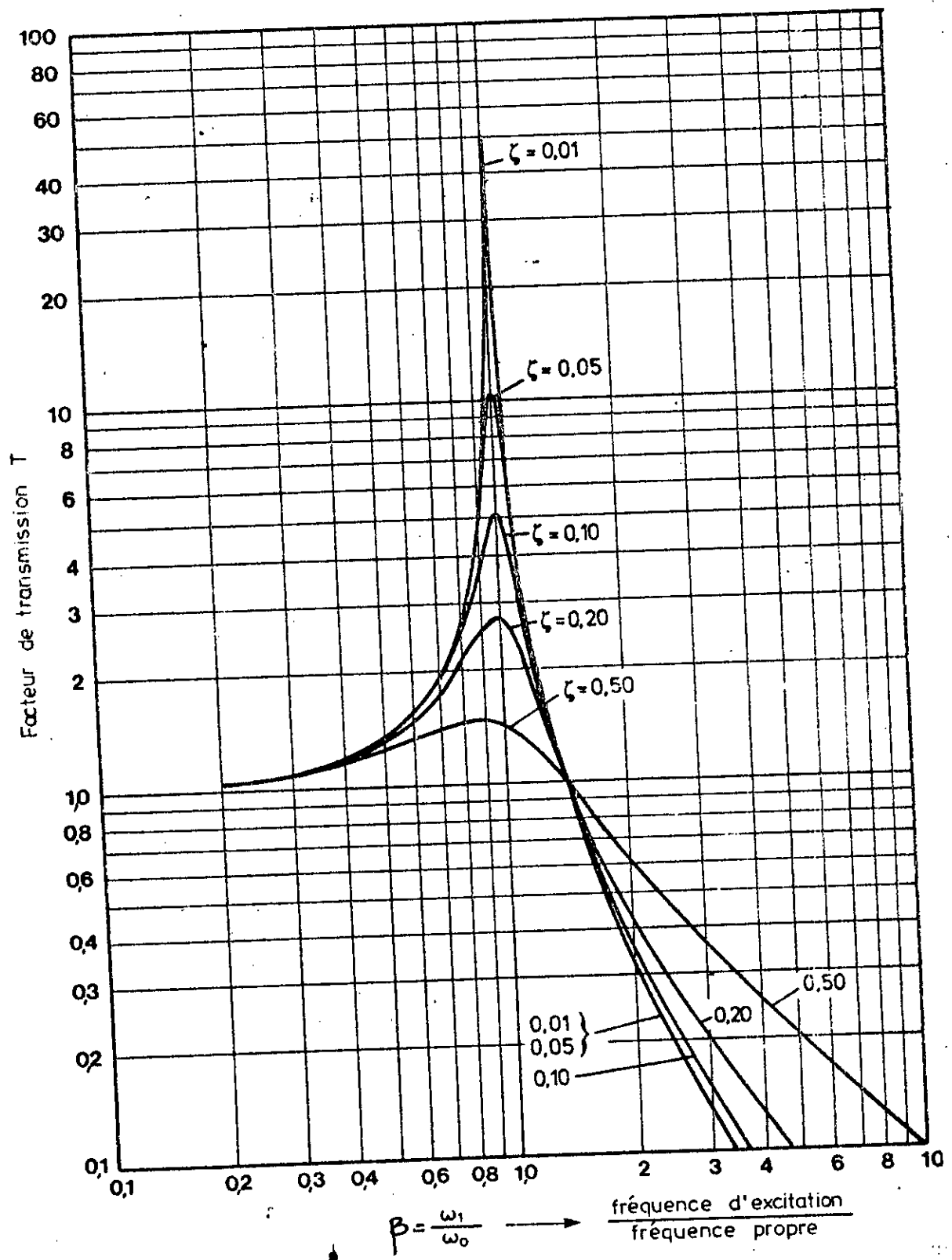


Fig.(1.B): Facteur de transmission

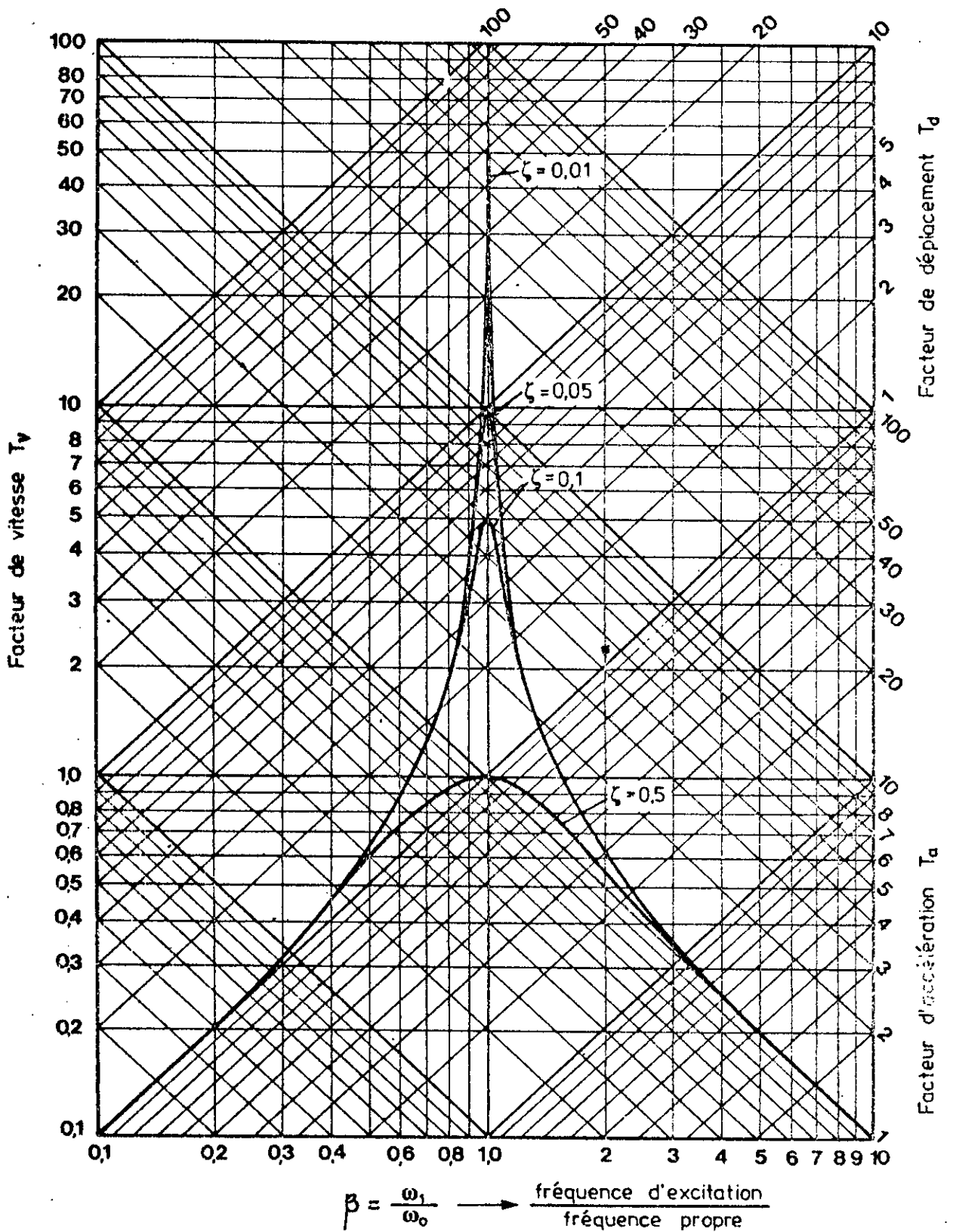


Fig.(I.4): Facteurs de déplacement, de vitesse et d'accélération.

CHAPITRE 2

DYNAMIQUE DES SYSTEMES

LINEAIRES CONTINUS

CHAPITRE II

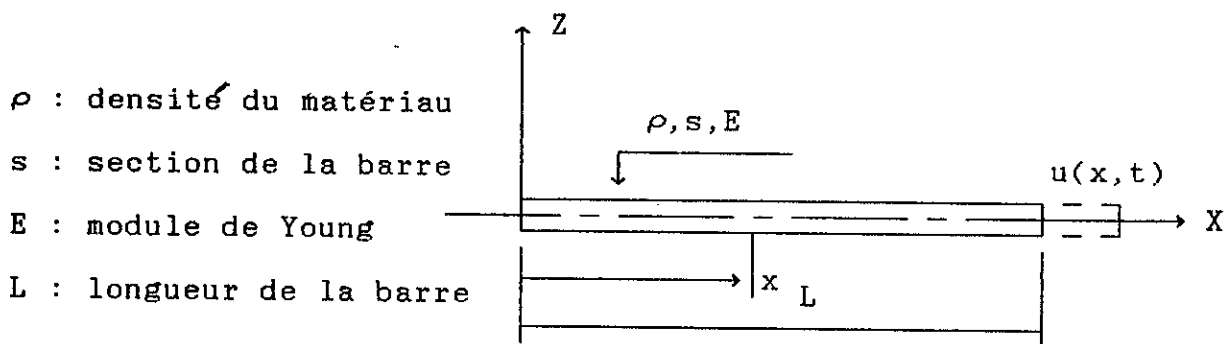
DYNAMIQUE DES SYSTEMES LINEAIRES CONTINUS

II-1 INTRODUCTION

Le comportement des systèmes continus ayant une distribution de masse, de rigidité et d'amortissement, ne peut être décrit de manière exacte que pour des cas simples, en moyennant des méthodes analytiques classiques. Sinon l'introduction d'autres approches (méthodes de simulations), pour décrire l'état du système est nécessaire. La méthode des éléments finis est devenue un outil performant pour la description de cet état. Dans ce chapitre, nous allons essayer d'analyser quelques systèmes simples, de trouver les solutions exactes des équations du mouvement, puis dans un deuxième temps nous allons étendre l'étude pour les systèmes complexes, en employant la MEF.

II-2 SOLUTIONS EXACTES POUR LE CAS D'UNE BARRE

Considérons une barre unidimensionnelle placée dans un système d'axes (X,Z) .



on suppose que la barre effectue un mouvement vibratoire selon l'axe X. L'amplitude du mouvement en un point x de la barre est $U_0(x)$, la fréquence de vibration est ω_n inconnue. Le mouvement d'un point de la barre est sous la forme :

$$U(x,t) = U_0(x) \exp(j\omega t) \quad (\text{II-1})$$

de plus l'équation caractérisant l'état du système est donnée par :

$$-\frac{\partial}{\partial x} \left[ES \frac{\partial U(x,t)}{\partial x} \right] + \rho S \frac{\partial^2 U(x,t)}{\partial t^2} = 0 \quad (\text{II-2})$$

par substitution de la relation (III-1) dans (III-2) on obtient :

$$-\frac{\partial}{\partial x} \left[ES \frac{\partial U_0(x)}{\partial x} \right] - \rho S \omega_n^2 U_0(x) = 0 \quad (\text{II-3})$$

Conditions aux limites : L'introduction des conditions aux limites va permettre l'identification de l'expression de ω_n .

1)- la barre est encadrée d'un coté.

$$\left[\frac{\partial U_0(x)}{\partial x} \right]_{x=L} = 0; \quad U_0(x) \Big|_{x=0} = 0$$

2)- Les deux extrémités encadrées.

$$U_0(x) \Big|_{x=L} = 0; \quad U_0(x) \Big|_{x=0} = 0$$

Solution générale : La solution générale de l'équation (II-2) est de la forme :

$$U_0(x) = A \cos \sqrt{\frac{\rho}{E}} \omega x + B \sin \sqrt{\frac{\rho}{E}} \omega x \quad (\text{III-4})$$

Pour le premier cas (encastrement d'une extrémité) on trouve après annulation du déterminant :

$$\cos \omega_n \sqrt{\frac{\rho}{E}} L = 0 \quad (\text{II-5})$$

par suite on déduit ω_n .

$$\omega_n = \frac{(2n+1)}{2L} \pi \sqrt{\frac{\rho}{E}} \quad (\text{II-6})$$

de la même manière on détermine la fréquence propre pour le second cas.

$$\omega_n = \frac{n}{L} \pi \sqrt{\frac{\rho}{E}} \quad (\text{II-7})$$

REMARQUE : on a fait cette application sur la barre car on aura besoins des solutions exactes pour les comparer aux solutions approchées obtenues à partir du logiciel.

II-3 PRESENTATION DE LA METHODE DES ELEMENTS FINIS [Kar 3] [Bel19]

Généralement les problèmes physiques sont décrits par des équations différentielles ou des équations intégrales, ces équations sont définies dans des domaines souvent complexes, La MEF propose de subdiviser le système globale (structure) en sous-systèmes élémentaires (éléments) dont le comportement est facilement établi, puis faire l'assemblage pour aboutir à un système d'équations algébriques qu'on résoud numériquement.

II-3-1 FORMULATION DE LA METHODE

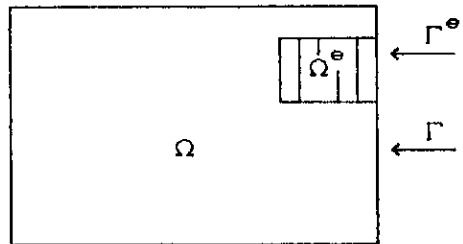
On définit en premier lieu l'approximation sur la variable (inconnue), déplacement, contrainte ..., soit :

$$u_e(x,y,z) = \langle N(x,y,z) \rangle \{u_e\} \quad (\text{II-8})$$

Ω : Domaine de définition de
de la variable $u(x,y,z)$

Γ : Frontière du domaine

Γ^e, Ω^e : Frontière et domaine
élémentaire



où $u_e(x,y,z)$ définit l'état de la variable en un point donné de coordonnées x,y et z au niveau de l'élément (Ω^e), $N(x,y,z)$ sont les fonctions de forme (d'interpolation) de type (polynomial, exponentiel, trigonométrique ou autres), et $\{u_e\}$ un vecteur contenant les inconnues en certains points de l'élément (noeuds).

La méthode compte trois principales formulations (versions) on va en présenter deux à savoir :

II-3-1-1 Version déplacement

Elle a pour principe la stationnarité de l'énergie potentielle totale Π de la structure, qui a pour expression :

$$\Pi = \frac{1}{2} \int_{\Omega} \{\varepsilon\}^T \{\sigma\} dv - \int_{\Omega} \{u\}^T \{f^b\} dv - \int_{\Gamma} \{u^s\}^T \{f^s\} ds$$

....(II-9)

où $\{u\}$: vecteur déplacement (inconnue)

$\{\sigma\}$: vecteur contrainte

$\{\varepsilon\}$: vecteur déformation

$\{f^b\}$: vecteur forces volumiques

$\{f^s\}$: vecteur forces surfaciques

par minimisation et en tenant compte de la loi de Hooke :

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

on obtient :

$$\delta\Pi = \frac{1}{2} \int_{\Omega} \delta\{\varepsilon\}^T \{\sigma\} dv - \int_{\Omega} \delta\{u\}^T \{f^b\} dv - \int_{\Gamma} \delta\{u^s\}^T \{f^s\} ds$$

(II-10)

Puisque $\Pi = \sum \Pi^e$, on décrit le domaine Ω en sous domaines Ω^e et son contour Γ en Γ^e , par conséquent l'équation de stationnarité sera même au système d'équations suivant.

$$[K] \{U\} = \{F\} \quad \text{(II-11)}$$

où $[K]$ est dite matrice de rigidité .

II-3-1-1-2 Critères de convergence

Cette formulation exige que certains critères soit vérifiés pour assurer la convergence de la méthode à savoir :

a)- *Conditions de compatibilité* :

a)-1 Le champ de déplacement doit être continu à l'intérieur de l'élément ainsi que sa (ses) première(s) dérivée(s).

a)-2 Continuité du champ de déplacement et sa (ses) première(s) dérivée(s) à la frontière entre éléments .

b)- *Conditions de complétude*: Pendant le mouvement de corps rigide l'élément doit présenter un état de déformation nul.

Si les deux premières conditions sont satisfaites l'élément est *conforme*, sinon il est dit *non conforme*. Les éléments non conformes peuvent converger mais leur utilisation est conditionnée par un "passage de tests".

La Troisième condition reste nécessaire pour éviter la singularité de la matrice de rigidité.

II-3-1-2 Version contrainte

Cette version est basée sur la stationnarité de l'énergie complémentaire Π^* (elle atteint un maximum)

$$\Pi^* = \frac{1}{2} \int_{\Omega} \{\sigma\}^T \{\epsilon\} dv - \int_{\Omega} \{f^b\}^T \{u\} dv - \int_{\Gamma} \{f^s\}^T \{u^s\} ds \quad (\text{II-12})$$

La condition de stationnarité se traduit par : $\delta\Pi^* = 0$

i.e.,

$$\delta\Pi^* = \frac{1}{2} \int_{\Omega} \delta\{\sigma\}^T [D]^{-1} \{\sigma\} d\Omega - \int_{\Omega} \{u\}^T \delta\{f^b\} d\Omega - \int_{\Gamma} \{u^s\}^T \delta\{f^s\} d\Gamma \quad (\text{II-13})$$

Pour cette version on exige que l'équilibre local (continuité des contraintes) est les conditions naturelles (forces) soient satisfaites.

REMARQUES :

- Dans la version déplacement la continuité des contraintes n'est pas exigée, réciproquement, la continuité du champ de déplacement peut ne pas être satisfaite pour la version contrainte.
- La version déplacement reste la plus utilisée dans les codes d'éléments finis.

II-3-3 PRECISION DE LA MEF

Comme toute méthode approximative le calcul par la MEF est réalisé avec une erreur définie par le corollaire [Hug 1] suivant:

$$\| e \| \leq c h^{k+1-m} \| e_{ex} \| \quad (II-14)$$

- où
- c : coefficient indépendant de l'élément .
 - h : Taille de l'élément .
 - k : Degré du polynôme d'interpolation .
 - m : Ordre de dérivées apparaissant dans l'expression des variables nodales .

Pour augmenter la précision du calcul, on peut agir sur les paramètres précédents pour réduire l'erreur, on peut donc :

- Diminuer la taille de l'élément h (augmenter le nombre d'éléments).
- Augmenter le nombre de noeuds d'interpolation .
- Augmenter le nombre de variables nodales .

II-4 PROBLEMES DYNAMIQUES

Quand la structure est sujette à des chargements invariables ou lentement variables dans le temps le problème est dit d'équilibre (statique) .

$$[K] \{U\} = \{F\} \quad (II-15)$$

- où
- {F} : vecteur chargement.
 - [K] : matrice de rigidité.

Si les forces appliquées au système sont rapidement variables dans le temps, l'effet d'inertie doit être pris en considération, et le problème est dynamique, en appliquant le principe de d'Alembert avec les forces d'inertie comme des forces de volume, et en supposant que les vitesses⁽¹⁾ [Bat 3] et les accélérations sont approximées de la même manière que les déplacements, on déduit l'expression de la résultante des forces de volume R^b .

$$R^b = \sum r_e^b = \sum_{nel} \int_{\Omega_e} [N]^T (\{f_e^b\} - \rho_e [N] \{\ddot{u}\}_e) d\Omega_e \quad (II-16)$$

(1) - pour simplifier la programmation de la méthode et réduire l'effort de calcul on prend cette hypothèse sinon $\langle u \rangle = \langle N \rangle \langle u_n \rangle$

où $\{\ddot{u}\}$: vecteur accélération nodal.
 ρ_e : masse volumique élémentaire.

La nouvelle équation d'équilibre s'écrit alors :

$$[M] \{\ddot{u}\} + [K] \{u\} = \{R\} \quad (\text{II-17})$$

où $[M]$ indique la matrice masse, elle a pour expression :

$$[M] = \sum_{\text{nel}} \int_{\Omega^e} \rho_e [N]^T [N] d\Omega^e \quad (\text{II-18})$$

REMARQUE : La formulation détaillée de $[K]$, $[M]$ et $\{F^b\}$ est faite en annexe (A 1).

En cas de présence de l'amortissement, on le représente par une matrice $[C]$, cette matrice est généralement mal connue, seul des hypothèses sont faites pour sa formulation [Clo16], la plus connue est celle due à Rayleigh déduite à partir de la condition d'orthogonalité imposée à la matrice.

$$\{\phi\}^T [C] \{\phi\} = 0; \quad m \neq n \quad (\text{II-19})$$

Par conséquent $[C]$ est une combinaison de la forme:

$$[C] = \alpha [M] + \beta [K] \quad (\text{II-20})$$

où : α et β sont des coefficients définis expérimentalement.

Ainsi on définit le problème général décrivant la dynamique des structures modélisées par éléments finis:

$$[M] \{\ddot{u}(t)\} + [C] \{\dot{u}(t)\} + [K] \{u(t)\} = \{F(t)\} \quad (\text{II-21})$$

De la même façon à ce qu'on a vu pour le système à un seul ddl on peut avoir un problème de vibrations libres ou forcées, dans le premier on détermine les modes propres de la structure, et dans le deuxième la réponse dynamique.

II-4-1 VIBRATIONS LIBRES

Le problème posé est sous la forme suivante :

$$[M] \{\ddot{u}(t)\} + [K] \{u(t)\} = 0 \quad (\text{II-22})$$

Pour résoudre ce problème on pose $\{u(t)\} = \exp(\omega t) \{\phi\}$, puis on substitue dans l'équation précédente :

$$(\lambda [M] + [K]) \{\phi\} = 0 ; \quad \lambda = \omega^2 \quad (\text{II-23})$$

Cette équation représente un problème aux valeurs propres dont les solutions sont λ et $\{\phi\}$ valeurs et vecteurs propres respectivement. Si l'amortissement n'est pas négligeable on fait intervenir la matrice d'amortissement $[C]$, et on aura à résoudre :

$$(\omega^2 [M] + \omega [C] + [K]) \{\phi\} = 0 \quad (\text{II-24})$$

C'est un problème quadratique qu'on réduit en un problème linéaire en lui doublant l'ordre.

$$\omega \begin{bmatrix} C & M \\ M & 0 \end{bmatrix} \begin{Bmatrix} \phi \\ \omega\phi \end{Bmatrix} = \begin{bmatrix} -K & 0 \\ 0 & M \end{bmatrix} \begin{Bmatrix} \phi \\ \omega\phi \end{Bmatrix} \quad (\text{II-25})$$

Qui peut être écrit sous la forme :

$$\omega [A] \{z\} = [B] \{z\} \quad (\text{II-26})$$

avec.

$$[A] = \begin{bmatrix} C & M \\ M & 0 \end{bmatrix} ; \quad [B] = \begin{bmatrix} -K & 0 \\ 0 & M \end{bmatrix} ; \quad \{z\} = \begin{Bmatrix} \phi \\ \omega\phi \end{Bmatrix}$$

Le problème défini en (II-26) est similaire à celui défini en (II-23), mais des différences importantes existent, la première est que la taille de $([A],[B])$ est double de celle de $([K] \text{ et } [M])$, en plus $[M]$ et $[K]$ sont définies positives ou au moins semi-définies positives, ce qui n'est pas le cas pour $[A]$ et $[B]$. Enfin les valeurs propres du premier système sont réelles alors que celles du deuxième sont complexes. on conclut que la résolution du dernier système s'avère plus compliquée et nécessite plus d'effort de calcul et de stockage.

II-4-1 VIBRATIONS FORCÉES

Lorsque le système es chargé dynamiquement i.e. $\{F\} = \{F(t)\}$, on aura à déterminer les variables $\{u(t)\}$, $\{\dot{u}(t)\}$ et $\{\ddot{u}(t)\}$ à partir de l'équation :

$$[M]\{\ddot{u} \} + [C]\{\dot{u} \} + [K]\{u \} = \{F(t)\} \quad (II-27)$$

Plusieurs méthodes sont disponibles pour résoudre ce problème [Huges] on peut citer Newmark α -method ...

II-5 CONCLUSION

On a vu dans ce chapitre que l'analyse des structures se partage en problèmes statiques et dynamiques, dans le premier cas on résout un système d'équations linéaires et dans le deuxième un problème aux valeurs propres et un problème dynamique.

CHAPITRE 3

GENERALITES SUR LE

PROBLEME AUX VALEURS PROPRES

GENERALITES SUR LE PROBLEME AUX VALEURS PROPRES

III-1 INTRODUCTION

On a vu au chapitre précédent que l'étude dynamique des structures chargées dynamiquement nécessite la résolution du problème aux valeurs propres, ici nous présentons quelques caractéristiques fondamentales concernant ce genre de problèmes.

III-2 POSITION DU PROBLEME

Soit le système d'équations suivant :

$$[K] \{\phi\} - \lambda [M] \{\phi\} = 0 \quad (\text{III-1})$$

Déterminer l'ensemble des paramètres λ et les vecteurs $\{\phi\}$ solutions du problème (III-1) respectivement valeur et vecteur propres. Nous nous intéresserons aux cas où les matrices $[K]$ et $[M]$ sont définies positives ou au moins semidéfinies positives, auquel cas correspondent des valeurs propres ordonnées comme suit :

$$0 \leq \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n \quad (\text{III-2})$$

Les λ_i sont les racines du polynôme caractéristique défini par la relation:

$$P(\lambda) = \text{Det} \left[[K] - \lambda [M] \right] \quad (\text{III-3})$$

De plus les vecteurs propres de (III-1) doivent vérifier les relations d'orthogonalité par rapport à $[K]$ et d'orthonormalité par rapport à $[M]$ i.e.,

$$\begin{cases} \{\phi_i\}^T [K] \{\phi_j\} = \lambda_j \delta_{ij} & (\text{Pas de sommation sur les indices}) \\ \{\phi_i\}^T [M] \{\phi_j\} = \delta_{ij} \end{cases} \quad (\text{III-4})$$

En pratique les modes supérieurs ne sont pas très importants, vu que leurs influences sur la réponse globale du système est faible, et la précision avec laquelle ils sont obtenus est médiocre [Hug 1]. On détermine donc un spectre tronqué de taille n_{mod} , avec $n_{\text{mod}} \ll n$. Dans ce qui suit on attachera plus d'attention aux procédures permettant l'extraction des solutions $(\lambda_l, \{\phi_l\})$ avec $1 \leq l \leq n_{\text{mod}}$.

III-3 PROPRIETES FONDAMENTALES DU PROBLEME AUX VALEURS PROPRES

Les procédures de résolution du problème (III-1) sont fondées sur les caractéristiques de ce dernier, on présentera ci-dessous quelques unes.

III-3-1 SEPARATION DE VALEURS PROPRES

On définit un polynôme caractéristique d'ordre (r) par la relation :

$$P^{(r)}(\lambda) = \text{Det} \left[[K^{(r)}] - \lambda [M^{(r)}] \right] \quad (\text{III-5})$$

où, $[K^{(r)}]$ et $[M^{(r)}]$ matrices avec les lignes et les colonnes $(n-r \rightarrow n)$ éliminées. Les valeurs propres de deux systèmes d'ordres r et $r+1$ peuvent être rangées de la façon suivante :

$$\lambda_1^{(r)} \leq \lambda_1^{(r+1)} \leq \lambda_2^{(r)} \leq \lambda_2^{(r+1)} \dots \leq \lambda_{n-r}^{(r)} \leq \lambda_{n-r}^{(r+1)} \quad (\text{III-6})$$

La suite de polynômes $P^{(r)}(\lambda)$ constitue la *Suite de Sturm*. Les racines du polynôme $P^{(r)}$ séparent celles du polynôme $P^{(r+1)}$.

On peut illustrer cette propriété par l'exemple suivant.

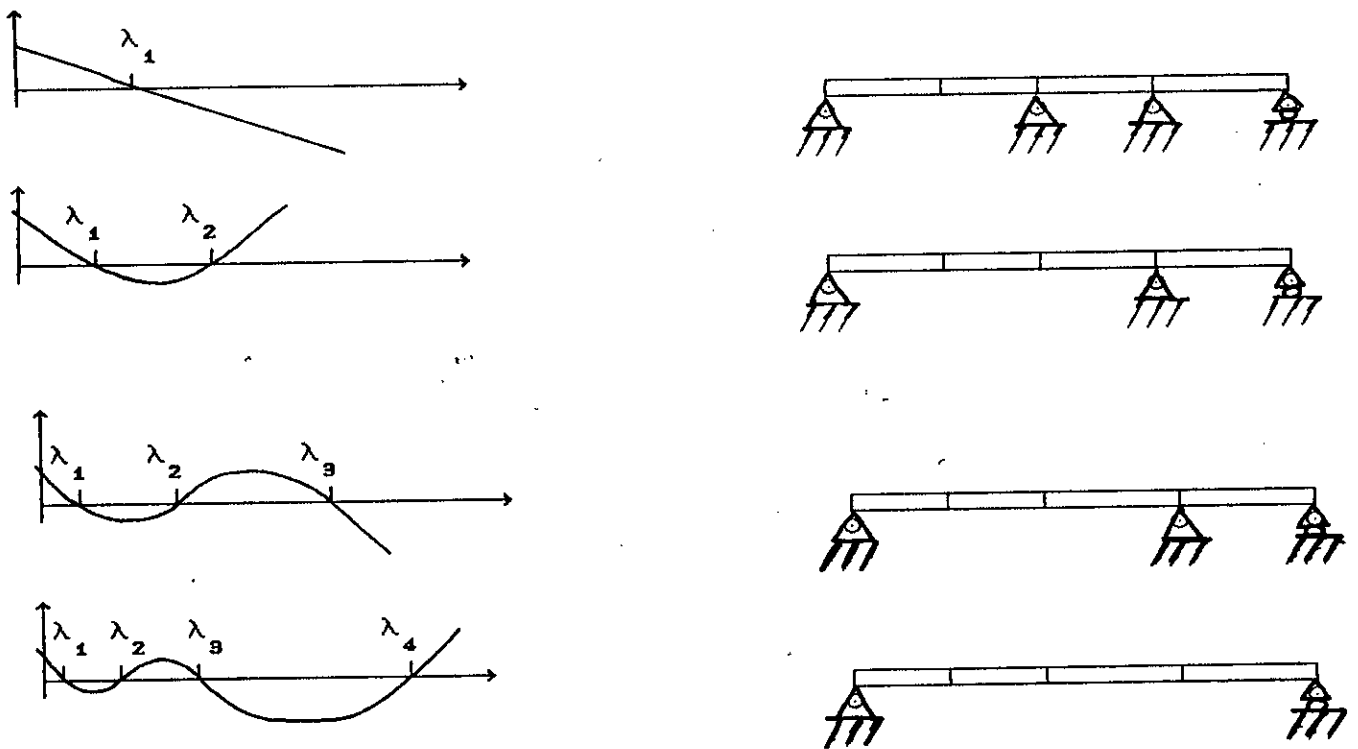


Fig (III-1)

Valeurs propres d'une poutre simplement supportée
avec différentes conditions aux limites

III-3-2 FORME STANDARD

Certains algorithmes utilisent la forme standard pour trouver les valeurs propres. l'équation dans ce cas est donnée par :

$$[A] \{ \phi \} = \lambda \{ \phi \} \quad (III-7)$$

mais en mécanique des structures le problème est toujours défini par les deux matrices $[K]$ et $[M]$, et pour avoir un système similaire à (III-7), on doit transformer (III-1), pour cela on suppose que $[M]$ est définie positive, par conséquent elle peut être décomposée sous la forme :

$$[M] = [S] [S]^T \quad (III-8)$$

Si $[S]$ est inversible (déterminant non nul) alors on peut

transformer le système (III-1) en (III-7).

$$[\bar{K}] \{\bar{\phi}\} = \lambda \{\bar{\phi}\} \quad (\text{III-8})$$

où

$$[\bar{K}] := [S]^{-1} [K] [S]^{-T}$$

$$\{\bar{\phi}\} := [S]^T \{\phi\}$$

Dans le cas où $[K]$ est définie positive et $[M]$ ne l'est pas, nous remplaçons (III-1) par :

$$[M] \{\phi\} - \frac{1}{\lambda} [K] \{\phi\} = 0$$

et la décomposition dans ce cas sera faite pour $[K]$.

REMARQUE : Quand la taille du problème augmente, le coût de la transformation domine le coût globale de résolution. dans cette situation la transformation est déconseillée.

III-3-3 MASSES NULLES SUR LA DIAGONALE

Quand des éléments nuls sont constatés sur la diagonale de la matrice $[M]$, on déduit directement la valeur et le vecteur propres :

$$\lambda_k = \infty ; \{\phi\}^T = \langle 0 \ 0 \ 0 \ \dots \underset{\uparrow}{1} \ \dots \ 0 \rangle \quad (\text{III-9})$$

la condition d'orthogonalité est violée, car : $\{\phi\} [M] \{\phi\}^T = 0$

III-3-4 DECALAGE "SHIFTING"

La technique de décalage est employée dans le but d'accélérer la convergence, ou alors éliminer la singularité de la matrice $[K]$ (éléments nuls sur la diagonale). Soit σ la valeur du décalage, on définit $[K_\sigma]$ par :

$$[K_\sigma] = [K] - \sigma [M] \quad (\text{III-10})$$

Le nouveau problème à résoudre sera posé sous la forme :

$$[K_{\sigma}]\{\phi\} - \nu [M]\{\phi\} = 0 \quad (\text{III-11})$$

avec, $\nu = \lambda - \sigma$

III-3-5 DECOMPOSITION SPECTTRALE

pour s'assurer que toutes les valeurs propres ont été calculées dans un intervalle donné, $[0, \mu]$; où μ est situé légèrement à droite de la dernière valeur propre calculée λ_j , on fait un balayage de cet intervalle, en suivant deux étapes :

1)- Factorisation de $([K] - \mu [M])$ par la technique de Crout i.e.,

$$[K] - \mu [M] = [U]^T [D] [U] \quad (\text{III-12})$$

Le paramètre μ est pris égale à $(\lambda_j(1 + \delta))$, δ étant un petit chiffre positif. Le théorème d'inertie de Sylvester stipule que : le nombre de valeurs propres au dessous de λ est égal au nombre d'éléments négatifs de $[D]$ (matrice diagonale). on a donc :

- $n_{\mu} - n_j = 0$ Toutes les valeurs sont déterminées.
- $n_{\mu} - n_j \neq 0$ Des valeurs ont été oubliées.

Cette technique est utilisée dans la méthode sous-espace et Lanczos.

CHAPITRE 4

METHODES DE RESOLUTION DU P.V.P

CHAPITRE IV

METHODES DE RESOLUTIONS DU PROBLEMES AUX VALEURS PROPRES

IV-1 INTRODUCTION

Pour un calcul efficace des solutions du problème aux valeurs propres, il n'est pas nécessaire de considérer toutes les méthodes existantes, seulement il faut savoir sélectionner dépendement de la spécificité du problème posé celle qui convient le mieux. Dans ce chapitre nous allons discuter Certaines de ces procédures qui sont des plus importantes en éléments finis.

IV-2 HISTORIQUE

Les premiers travaux portant sur le problème aux valeurs propres ont été menés [Dau20] par Courant en 1943, plus tard des contributions sur la majoration d'erreur ont été faites par Birkhoff et al (1966), suivi de Ciarlet et al (1968), Fix et Pièrre Varga (1972) qui ont apporté des améliorations sur la majoration d'erreur.

Les approximations par la méthode des éléments finis ont été analysées notamment par Babuska (1972), Fix (1973) Gregoire et al (1976) et Canuta (1978).

Concernant le cas particulier de matrices symétriques rencontrées très souvent, dans l'analyse par éléments finis des problèmes de dynamique des structures, de stabilité et de transfèrt de chaleur, des techniques ont été développées en parallèle avec le développement de la MEF à compter des années cinquantes. La méthode de Jacobi (1846) reformulée par Wilkson et Reinsh (1971) a été suivie par d'autres méthodes, nous citons LR due à Rutishauser (1955), en 1961 Francis met au point la transformation QR.

La modélisation par éléments finis donne des matrices de très grande tailles, les méthodes citées auparavant perderent d'efficacité, néanmoins, une nouvelle génération d'algorithmes basés essentiellement sur l'analyse de Rayleigh Ritz les remplaça, parmi ces méthodes on peut citer:

- La méthode sous espace
- La méthode d'itération inverse
- La méthode de Ritz
- La méthode de Lanczos

IV-3 CHOIX DE LA METHODE APPROPRIEE

Ce que l'on désire pendant l'analyse d'un système donné, est de réaliser des résultats avec une précision acceptable, dans l'espace temps le plus court, moyennant un environnement de calcul le moins couteux, afin de réaliser cet objectif nous devons définir notre problème en termes de données et de besoins.

a)- *Données* :

- [M] : Symétrique, définie positive, et Profil
- [K] : Symétrique définie / Semidéfinie, positive et Profil
- [C] : Négligeable ou non
- N : Dimension du problème
- C M : La capacité de la machine employée

b)- *Besoins*

- ω_i ($i = 1, 2, \dots, n_{\text{mod}}$) nombre de fréquences propres à déterminer (spectre tronqué / complet)
- Precision sur les résultats

IV-3-1 METHODOLOGIE

Localisation du problème selon la taille [Kar 4]

- 1 - $0 < N < 10$
- 2 - $10 < N < 250$
- 3 - $250 < N < 1000$
- 4 - $1000 < N < 10000$
- 5 - $N > 10000$

1)- Expansion directe du polynôme caractéristique

2)- Si la CM est suffisante, on utilise les méthodes standards

- Jacobi
- Réduction sous forme tridiagonale par les méthodes de

Givens et Housholder, puis Sturm pour l'extraction des valeurs propres.

- Itération inverse, s-épace ou Lanczos

3)- Si la population de [K] et [M] n'excède pas la CM, l'emploi des méthodes précédentes est toujours valable, sinon on fait la condensation de [K] et de [M], Voir (Annexe 2).

4)- La désignation de la procédure pratique dépend de la technique de mémorisation employée.

5)- Au delà du seuil de 10.000, la demande en mémoire est énorme, seulement les super-computers peuvent la satisfaire. néanmoins la technique de sous-structuration [Kar 4], [Par13], peut constituer une alternative.

REMARQUES :

-La méthode de Jacobi permet le calcul de n valeurs propres d'un système de dimension limité à 100 [Dha 5].

-Si la taille du système dépasse 100, la technique de condensation peut marcher si l'on fait une bonne sélection des degrés de libertés dynamiques "Maîtres".

-Si l'on désire déterminer la plus grande ou la plus petite fréquence propre on emploie (l'itération inverse)

-Sous-épace et Lanczos permettent le calcul d'une approximation d'une partie du spectre des fréquences propres.

IV-4 CLASSIFICATION DES METHODES DE RESOLUTION

Puisqu'il n'existe pas de formules directes pour résoudre l'équation $P(\lambda) = 0$ pour un ordre supérieur à 4 [bat 3], il est impératif de recourir aux techniques itératives, selon [bat 3] quatre groupes peuvent être distingués.

- 1)- Méthodes itératives par vecteurs
- 2)- Méthodes de transformation
- 3)- Méthodes du déterminant 'Determinant search method'
- 4)- Méthodes basées sur les propriétés de la série de Sturm

IV-4-1 ITERATION PAR VECTEURS

L'équation $[K]\{\phi_i\} = \lambda_i[M]\{\phi_i\}$ constitue la base de ce groupe de méthodes à partir de laquelle on construit le cycle suivant.

- 1)- On pose $\{X_1\}$ vecteur de dimension $n \times 1$
- 2)- On calcule $\{R_1\} = [M]\{X_1\}$
- 3)- On résoud $[K]\{X_2\} = \{R_1\}$

L'approximation $\{X_p\}$, sera de plus en plus proche de $\{\phi\}$ à chaque fois qu'on refait le cycle précédent. Parmi les méthodes les plus célèbres de ce groupe :

IV-4-1-1 L'ITERATION INVERSE

C'est une procédure permettant la détermination de la première valeur propre λ_1 et le vecteur propre lui correspondant. Associée à la technique d'orthogonalisation de Gram Schmith [Bat16] il est possible de calculer les valeurs propres consécutives. (*algorithme 1*)

IV-4-2 METHODES DE TRANSFORMATION

Les procédures de transformation sont construites à partir des relations suivantes :

$$\{\phi_i\}^T [K] \{\phi_i\} = \delta_{ij} \lambda_i \quad (\text{IV-1})$$

$$\{\phi_i\}^T [M] \{\phi_i\} = \delta_{ij} \quad (\text{IV-2})$$

Où $\{\phi_i\}$: Le $i^{\text{ème}}$ vecteur propre du système
 λ_i : La $i^{\text{ème}}$ valeur propre du système

ALGORITHME 1

1- Triangulariser $[K] = [L]^T [D] [L]$

Si $[K]$ est semi définie on lui fait un décalage σ

2- Choisir un vecteur initial $\{X_1\}$, non M-orthonormale au vecteur propre recherché

3- Calculer la sollicitation $\{R_1\} = [M]\{X_1\}$

4- Faire une boucle sur $i = 1, \dots$

4-1- Résoudre $[L]^T [K] [L] \{X_{i+1}\} = \{R_i\}$

4-2- Calculer la sollicitation $\{\bar{R}\} = [M]\{X_{i+1}\}$

4-3- Evaluer $d = \{X_{i+1}\}^T \{\bar{R}\}$

4-4- Calculer l'approximation de λ par le quotient de Rayleigh :

$$\lambda_1^{i+1} = \frac{\{X_{i+1}\}^T R_i}{d}$$

4-5- calculer le nouveau vecteur sollicitation

$$\{R_{i+1}\} = \{\bar{R}\}$$

4-6- Vérifier la convergence de la valeur calculée

$$\left| \lambda_1^{i+1} - \lambda_1^i \right| < \epsilon$$

5- Calculer le vecteur propre M-normalisé

$$\{\phi_1\} = \frac{1}{\sqrt{d}} \{X_{i+1}\}$$

ORTHOGONALISATION DE GRAM SCHMITH

$$\{X_{i+1}\} = \{X_{i+1}\} - c_1 \{\phi_1\}$$

avec:

$$c_1 = \{\phi_1\}^T [M] \{X_{i+1}\}$$

L'objectif est de ramener par le biais de transformations successives les matrices $[K]$ et $[M]$ sous forme diagonale, où chaque transformation consiste en une prémultiplication et une post-multiplication par les matrices de 'passage' $[P_k]^T$ et $[P_k]$ respectivement.

$$[K_1] = [K_1]$$

$$[M_1] = [M_1]$$

$$[K_2] = [P_1]^T [K_1] [P_1]$$

$$[M_2] = [P_1]^T [M_1] [P_1]$$

$$[K_3] = [P_2]^T [K_2] [P_2]$$

$$[M_3] = [P_2]^T [M_2] [P_2]$$

$$[K_4] = [P_3]^T [K_3] [P_3]$$

$$[M_4] = [P_3]^T [M_3] [P_3]$$

⋮

⋮

$$[K_{k+1}] = [P_k]^T [K_k] [P_k]$$

$$[M_{k+1}] = [P_k]^T [M_k] [P_k]$$

$$[K_{k+1}] \longrightarrow [\lambda]$$

et

$$[M_{k+1}] \longrightarrow [I]$$

En pratique il suffit que les matrices $[K]$ et $[M]$ convergent vers des forme diagonales, à savoir si:

$$[K_{k+1}] \longrightarrow [K_r] \quad \text{et} \quad [M_{k+1}] \longrightarrow [M_r] \quad \text{quand } k \rightarrow \infty$$

Où les matrices $[\lambda]$ et $[\phi]$ seront définies par :

$$[\lambda] = \text{diag} \left(\frac{[K_r]^{l+1}}{[M_r]^{l+1}} \right) \quad (\text{IV-3})$$

$$[\phi] = [P_1][P_2] \dots [P_l] \text{diag} \left(\frac{1}{\sqrt{[M_r]^{l+1}}} \right) \quad (\text{IV-4})$$

avec 'l' indique la dernière itération

Notons que pour obtenir les matrices diagonales $[K_r]$ et $[M_r]$ on peut effectuer une série de transformations similaires de 1 jusqu'à l comme est le cas pour la méthode de Jacobi, ou alors

$$\left(\frac{\bar{k}^k}{2} \right)^2 + \bar{k}_{ii}^k \bar{k}_{jj}^k > 0$$

ce qui signifie que x est positif et $\det (P_k) \neq \text{zero}$

CONVERGENCE

La convergence pour la méthode de Jacobi est mesurée par la comparaison des approximations successives des valeurs propres calculées à une itération quelconque, néanmoins les termes hors diagonale sont aussi testés s'ils sont suffisamment proches de zéro.

$$\frac{|\lambda_i^{l+1} - \lambda_i^l|}{\lambda_i^{l+1}} \leq 10^{-8}; \quad i = 1, \dots, n \quad (\text{IV-5})$$

où 10^{-8} représente la tolérance de la convergence exigée par l'utilisateur. La méthode de Jacobi [Dha 5] s'adapte très bien avec la technique de s-espace programmée et incorporée dans le logiciel.

REMARQUE : Il existe une autre méthode de Jacobi applicable aux problèmes sous forme standard, dont la formulation est plus simplifiée, mais généralement pour le calcul en éléments finis on emploie la version généralisée, car la transformation du problème généralisé à la forme standard est coûteuse. (l'algorithme 2)

IV-4-3 METHODE D'ITERATION PAR PAR SOUS ESPACE

La méthode sous-espace pour l'extraction de certaines fréquences propres et modes propres de structures dynamiques est très employée dans les codes de calcul de structures par éléments finis à titre d'exemple ADINA, SAP80 ... la méthode est due à K.J. Bathe [bat 3], le concept sur lequel elle est basée, consiste à faire projeter les matrices $[K]$ et $[M]$ dans une base orthogonale construite à partir de vecteurs obtenus par itération inverse. La taille du s-espace (m) est généralement très inférieure à la taille du problème initial. (l'algorithme 3)

ALGORITHME 2

1- Définition de la précision désirée et initialisation des tableaux de travail.

2- Pour chaque cycle s .

Définir la tolérance dynamique $\varepsilon_s = 10^{-2s}$

2-1- Pour chaque ligne $i = 1, 2, \dots, n$.

2-1- Pour chaque colonne $j = i+1, \dots, n$

2-2- Calculer les facteurs de couplage

$$f_k = \frac{|k_{ij}|}{\sqrt{k_{ii}k_{jj}}}; \quad f_m = \frac{|m_{ij}|}{\sqrt{m_{ii}m_{jj}}}$$

2-3- Si $f_k > \varepsilon_s$, ou $f_m > \varepsilon_s$:

2-3-1- Calculer α et γ

2-3-2- Transformer les matrices* [K] et [M].

colonne $i =$ colonne $i + \gamma$.colonne j

colonne $j =$ colonne $j + \alpha$.colonne i

ligne $i =$ ligne $i + \gamma$.ligne j

ligne $j =$ ligne $j + \alpha$.ligne i

2-3-3- Modifier les vecteurs propres.

colonne $i =$ colonne $i + \gamma$.colonne j

colonne $j =$ colonne $j + \alpha$.colonne i

3- Calculer les valeurs propres.

$$\lambda_i = \frac{k_{ii}}{m_{ii}}; \quad f_{\lambda_i} = \text{Max} \frac{|\lambda_i^s - \lambda_i^{s-1}|}{|\lambda_i^{s-1}|}$$

4- Calculer les facteurs de couplage

$$f_k = \text{Max}_{i,j} \frac{|k_{ij}|}{\sqrt{k_{ii}k_{jj}}}; \quad f_m = \text{Max}_{i,j} \frac{|m_{ij}|}{\sqrt{m_{ii}m_{jj}}}$$

5- Test de convergence : $f_k < \varepsilon$, $f_m < \varepsilon$ et $f_{\lambda} < \varepsilon$

ALGORITHME 3

1- Initialisation

1-1- Factorisation de $[K] = [L]^T [D] [L]$

1-2- Spécification du vecteur de sollicitation $[P]$ ($m \times n$)

2- Itération $i = 1, \dots, n_{\max}$ (nombre maximum d'itérations permis)

2-1- Résoudre le système linéaire $[L]^T [D] [L] \{R\} = \{P\}$

2-2- Calculer les matrices réduites de dimensions ($m \times m$)

$$[K^*] = [R]^T [K] [R]; \quad [M^*] = [R]^T [M] [R]$$

2-3- Résoudre le problème réduit

$$([K^*] - \lambda_k [M^*]) \{\psi^*\} = 0; \quad k = 1, 2, \dots, m$$

2-4 Calculer l'approximation des vecteurs propres du problème original

$$[R] [\{\psi_1^*\} \{\psi_2^*\} \dots \{\psi_m^*\}]$$

2-5 Si la convergence est atteinte, stopper l'exécution, sinon :

2-6 calculer le nouveau vecteur de sollicitation.

$$[P] = [M] [R] [\{\psi_1^*\} \{\psi_2^*\} \dots \{\psi_m^*\}]$$

REMARQUE : Il existe une version appelée *sous-espace accélérée*, "Accelerated subspace iteration method", plus performante que celle qu'on a exposée dans cette section, elle est due à Bathe, et implantée dans plusieurs logiciels récents comme ADINA.

VECTEURS DE DEPART

Les vecteurs de départ doivent permettre la convergence vers les modes bas de la structure, avec un nombre d'itérations optimal. La démarche à suivre est présentée ci-dessous.

Soit $[P]$ le sous-espace de départ.

$$[P] = [\{P_1\} \{P_2\} \dots \{P_p\}]$$

On pose $\{P_1\} = \text{diag } [M]$, ceci assure que tous les ddl soient

excités. les autres colonnes $\{P_i\}$ ($i = 2, m$) sont des vecteurs unitaires e_i , avec les entrées +1 aux coordonnées correspondantes au plus faible rapport k_{ii}/m_{ii} . Car les modes bas sont caractérisés par des masses élevées et des rigidités faibles, ainsi tous les ddl seront excités, mais de façon plus distinguée ceux ayant des masses importantes.

Concernant la taille du s-espace il est recommandé [bat 3] que m soit donné par:

$$m = \text{Min} \{2p, p+8\} \quad (\text{IV-6})$$

où p indique le nombre de valeurs propres désirées

CONVERGENCE : Le critère de convergence employé concerne les p premières valeurs propres calculées, l'expression définissant le taux de convergence est donnée par :

$$\frac{|\lambda_i^{(k+1)} - \lambda_i^k|}{\lambda_i^{k+1}} < \text{tol} \quad (\text{IV-7})$$

où k : indique l'ordre de la valeur propre

i : indique le numéro de l'itération

tol : la tolérance exigée

REMARQUE : La méthode sous-espace utilise les vecteurs de départ pour itérations simultanées, c'est un inconvénient pour certains auteurs qui préfère utiliser un seul vecteur, comme la méthode de Lanczos ou alors plusieurs vecteurs de départ mais avec itération individuelle [Xia10]

IV-5 COMMENTAIRES

1)- En réalité il existe deux groupes de techniques pour résoudre le problème au valeurs propres, le premier c'est celui qu'on appelle techniques fondamentales, qui sert à déterminer les solutions du problème considéré, sans faire appel à d'autres techniques à titre d'exemples " Jacobi, QR, ". L'autre groupe utilise une combinaison de techniques pour concevoir une méthode finale, qui dépasse dans les performances ses constituantes. exemple sous-espace. Ce dernier groupe est notamment employé pour les systèmes de grande tailles.

2)- Le choix de la méthode appropriée est gouverné par le nombre d'opérations, et l'espace mémoire requis.

3)- La méthode de Jacobi est efficace quand tous le spectre est demandé et les éléments diagonaux sont proches de zéro, pour cette raison elle est sélectionnée pour résoudre le problème réduit formé par la méthode sous-espace [bat17].

4)- Quand la matrice masse possède un profil de bande et son ordre est élevé, la transformation au problème standard devient inefficace.

5)- D'autres méthodes sont recommandées pour des cas spécifiques par exemple la méthode de recherche par déterminant " Determinant Search Method" pour les matrices dont la largeur bande est faible [bat15], l'itération inverse pour le cas d'une seule solution désirée.

IV-6 CONCLUSION

Nous avons présenté quelques méthodes bien connues, chacune d'elle possède des avantages et des inconvénients, néanmoins l'exploitation des points forts d'un ensemble de méthodes donne de meilleurs résultats à titre d'exemple itération inverse et Jacobi qui forme la méthode sous espace. Dans le prochain chapitre on présentera la méthode de Lanczos, l'une des plus efficace actuellement.

CHAPITRE 5

METHODE DE LANCZOS

METHODE DE LANCZOS

V-1 INTRODUCTION

L'algorithme de Lanczos pour l'extraction de valeurs propres ne cesse d'augmenter de popularité, notamment dans le domaine de calcul des structures. Paru en 1950, il est considéré comme une procédure de réduction des matrices symétriques sous forme tridiagonale symétrique. Le principal inconvénient dont souffre la méthode de Lanczos est l'accumulation de l'erreur d'arrondi qui affecte sévèrement sa stabilité. En 1955 l'algorithme de Housholder remplaça celui de Lanczos, comme une méthode plus efficace pour la tridiagonalisation.

Les problèmes posés en engineering exige le calcul d'un nombre faible de valeurs propres, et l'emploi de la méthode de Housholder suppose la tridiagonalisation de la matrice toute entière, ce qui implique un calcul énorme. Pour cette raison l'algorithme de Lanczos a eu une attention particulière car il s'adapte très bien avec ce type de problèmes, en permettant le calcul des modes bas pour les systèmes de grande taille.

Notre étude sera restreinte au cas des matrices symétriques positives et semidéfinies, auxquelles correspondent des valeurs propres réelles positives ou (nulles).

V-2 FORMULATION (voir annexe 3)

L'idée de base dans la construction de l'algorithme demeure toujours comme pour la méthode sous-espace, l'obtention d'un problème réduit dont les valeurs propres sont des approximations de celles du problème initial, mais cette fois la réduction est assurée par la procédure de Rayleigh-Ritz (Annexe 2).

L'algorithme traite les problèmes du type standard, mais il peut être utilisé pour le problème généralisé, puisque ce dernier se transforme, (chap 3) néanmoins pour économiser l'effort de

transformation, on travaille directement avec:

$$[K]_{\sigma}^{-1} [M] \{ \phi \} = \nu \{ \phi \} \quad (V-1)$$

où $[K]_{\sigma} = [K] - \sigma [M]$

et σ est le décalge "shifting" employé par l'utilisateur.

V-3 GENERATION DES VECTEURS DE LANCZOS

La principale étape dans l'algorithme est la génération d'une série de vecteurs M-orthogonaux ($q_1, q_2, q_3, \dots, q_j$) appelés *vecteurs de Lanczos*. Supposons pour calculer le vecteur q_{j+1} , j vecteurs ont été déjà calculés, on a alors l'expression de q_{j+1} :

$$\{q_{j+1}\} = \frac{\{r_j\}}{\beta_{j+1}} \quad (V-2)$$

où β et r sont donnés par :

$$\beta_{j+1} = \{r_j\}^T [M] \{r_j\} \quad (V-3)$$

$$\{r_j\} = [K]_{\sigma}^{-1} [M] \{q_j\} - \alpha_j \{q_j\} - \beta_j \{q_{j-1}\} \quad (V-4)$$

les paramètres α_j et β_{j-1} sont évalués à partir des relations :

$$\alpha_j = \frac{\{q_j\}^T [M] \{\bar{r}_j\}}{\{q_j\}^T [M] \{q_j\}} \quad \beta_{j-1} = \frac{\{q_{j-1}\}^T [M] \{\bar{r}_{j-1}\}}{\{q_{j-1}\}^T [M] \{q_{j-1}\}}$$

$\{\bar{r}_j\}$ est un vecteur intermédiaire donné par :

$$\{\bar{r}_j\} = [K]_{\sigma}^{-1} [M] \{q_j\}$$

V-4 FORME TRIDIAGONALE

Les paramètres α_j et β_{j-1} calculés précédemment sont les éléments de matrice tridiagonales $[T_m]$ dont l'expression est:

$$[T_m] = [Q_m]^T [M] [K_\sigma]^{-1} [Q_m] \quad (V-5)$$

où $[Q_m] = [\{q_1\} \{q_2\} \dots \{q_m\}]$

plus explicitement: $[T_m] = \begin{bmatrix} \alpha_1 & \beta_2 & & & & \\ \beta_2 & \alpha_2 & \beta_3 & & & \\ & \beta_3 & & & & \\ & & & & & \\ & & & & & \beta_m \\ & & & & \beta_m & \alpha_m \end{bmatrix}$

V-5 LE PROBLEME REDUIT

À partir de la matrice tridiagonale obtenue précédemment on définit le problème réduit suivant :

$$[T_m] \{s_i\} - \vartheta_i \{s_i\} = 0 \quad (V-6)$$

où $(\vartheta_i, \{s_i\})$ représente la paire de solution de la matrice tridiagonale. La taille de $[T_m]$ augmente de un après chaque étape de Lanczos, et les solutions de $[T_m]$ convergent progressivement vers les solutions du problème original $(k - \nu m)z = 0$.

V-5-1 CRITERE DE CONVERGENCE

L'emploi du critère de convergence classique défini par :

$$\frac{|\vartheta^{(j-1)} - \vartheta^{(j)}|}{\vartheta^{(j)}} \quad (V-7)$$

n'est pas consistant [Hug 1], on opte pour un autre critère plus performant qui consiste à calculer le vecteur résiduel $\{r\}$:

$$\{r\} = [K_\sigma]^{-1} [M] \{y\} - \vartheta \{y\} \quad (V-8)$$

La norme du vecteur résiduel définie en (V-8) peut être utilisée pour indiquer le taux de convergence de ϑ à travers l'inégalité suivante :

$$| \nu - \vartheta | \leq \| r \|_M \quad (V-9)$$

La propriété de tridiagonalité simplifie la déduction de l'expression de $\| r \|$.

$$\rho_{mi} = \| r \| = \beta_{m+1} |\zeta_i| \quad (V-10)$$

où β_{m+1} est un scalaire calculé durant le processus de Lanczos, et $\zeta_i = \{e_i\}^T \{s_i\}^{(m)}$ représente l'élément le plus bas du vecteur $\{s_i\}^{(m)}$. La convergence est atteinte quand $\rho_{mi} < \text{tol}$. Le vecteur $y_i^{(m)}$ n'est calculé qu'une fois convergé.

V-6 PERTE ET RESTAURATION D'ORTHOGONALITE

Les processeurs numériques exécutent les opérations avec un arrondissement de chiffres, cet arrondissement est matérialisé par une entité dite " Unit Roundoff Error ", [Ral18] définie comme le plus petit chiffre ϵ tel que $1 + \epsilon > 0$.

Pendant le processus de Lanczos les vecteurs $\{q_j\}$ sont supposés être orthogonaux, mais cette propriété est vite perdue après un certain nombre d'itérations à cause de l'amplification de l'erreur introduite à chaque étape. Le taux de perte d'orthogonalité est mesurée par :

$$\eta_{ij} = \{q_i\}^T [M] \{q_j\} \quad (V-11)$$

où η_{ij} en arithmétique exacte :

$$\begin{cases} \eta_{ij} = 0 & i \neq j \\ \eta_{ij} = 1 & i = j \end{cases} \quad (V-12)$$

Des essais ont été faits sur l'algorithme de Lanczos (Version initiale), [Che 7] ont montré que η_{ij} dépasse l'ordre de ϵ et peuvent même aller jusqu'à 1. Plusieurs travaux sur l'analyse de la précision de la méthode de Lanczos ont été faits notamment par C. C. Paige [Pai 11]. Les résultats de ces travaux ont conduit à la conclusion que la perte d'orthogonalité peut être restaurée par reorthogonalisation de Gram-Schmidt par deux façons :

1)- Réorthogonalisation complète: Il faut respecter la condition:

$$| \{q_i\}^T [M] \{q_j\} | < n\epsilon ; \quad i \neq j \quad (V-13)$$

2)- Réorthogonalisation partielle : La condition précédente est relaxée, par l'exigence d'une condition moins sévère :

$$| \{q_i\}^T [M] \{q_j\} | < \sqrt{n}\epsilon ; \quad i \neq j \quad (V-14)$$

Cette condition maintient toujours un taux de convergence acceptable.

V-7 RESOLUTION DU SYSTEME REDUIT

On a vu que le résultat du processus de Lanczos est une matrice tridiagonale $[T_j]$, où j est son ordre. L'algorithme employé pour résoudre ce système est celui présenté dans [Par12]. Cet algorithme exécute entre $(20j$ à $40j)$ opérations pour finir les itérations.

V-8 CONCLUSION

L'algorithme de Lanczos semble moins coûteux comparé à la méthode sous-espace, vu le nombre réduit d'opérations qu'il lui faut pour former le système réduit, du côté théorique la formulation de l'algorithme est relativement compliquée notamment le problème d'orthogonalité, pour lequel beaucoup de travaux ont été faits. maintenant avec cette version on peut compter sur les résultats qui nous seront livrés.

CHAPITRE 6

PRESENTATION DU LOGICIEL

CHAPITRE VI

PRESENTATION DU LOGICIEL

VI-1 INTRODUCTION

DLEARN*, un code de Calcul par éléments finis écrit en Fortran 77, réalisé par T. J. R. Hugues, R. M. Ferncz et A. M. Rafskey [Hug 1], ses possibilités ont été limitées à l'analyse linéaire statique et dynamique. Par l'implantation d'un nouveau block, manquant à la version initiale, pour le calcul du problème aux valeurs propres. On est en mesure de calculer les valeurs propres par deux techniques à savoir :

- SOUS-ESPACE ET LANCZOS

Ce qui nous permettra d'étudier l'efficacité de chacune de ces techniques. (voir VII)

VI-2 OPTIONS D'ANALYSE

Le code assure deux types d'analyses généralement posés en calcul des structures à savoir: statique et dynamique. Notons qu'avant de lancer un processus quelconque il est nécessaire de faire une vérification des données, afin d'éviter des résultats erronés ou des erreurs d'exécution.

VI-2-1 ANALYSE STATIQUE

Elle consiste à calculer les contraintes (***) et les déplacements aux points nodaux ou d'intégration. L'équation à résoudre est:

$$[K]\{U \} = \{R \} \quad (VI-1)$$

où [K], {U} et {R} sont la matrice masse, le vecteur déplacement et le vecteur chargement respectivement. Les contraintes sont ensuite calculées à partir de la loi de Hooke (II)

(*)Version pédagogique [Hug 1]

(**) les quantités évaluées sont sans dimensions.

VI-2-2 ANALYSE DYNAMIQUE

L'analyse dynamique comprend :
-La réponse dynamique
-L'analyse modale

VI-2-2-1 La réponse dynamique

Pour les systèmes chargés dynamiquement, le code calcule les déplacements et les contraintes dans des intervalles de temps préalablement fixés par l'utilisateur. L'équation est la suivante :

$$[M] \{\ddot{u}\} + [C] \{\dot{u}\} + [K] \{u\} = F(t) \quad (\text{VI-2})$$

Où $[C]$ est approximée selon l'approche de Rayleigh. La méthode employée est la *Methode- α* ou ' α -Method' [Hug 1].

VI-2-2-2 L'analyse modale

Cette option permet le calcul des fréquences propres des structures et les modes propres associés, par la résolution du système:

$$[K] \{\phi\} - \lambda [M] \{\phi\} = 0 \quad (\text{VI-3})$$

Ici on peut employer au choix l'algorithme de Lanczos ou la Méthode sous-espace, Mais il est recommandé d'opter pour le premier choix. La fig(VI-1) Présente un organigramme général des différents blocks fonctionnels.

VI-3 CARACTERISTIQUES PRINCIPALES

On présente ci-dessous certaines caractéristiques importantes du logiciel qui méritent être mentionnées.

VI-3-1 ALLOCATION PSEUDO-DYNAMIQUE

L'allocation dynamique de l'espace mémoire est devenue une caractéristique standard des langages modernes comme le "C" ou le "PASCAL", mais le FORTRAN ne possède pas cette propriété. Pour palier à cet inconvénient on emploie l'allocation pseudodynamique dont le but est de réserver l'espace pendant la phase d'exécution. La procédure consiste à stocker toutes les données dans un seul vecteur colonne 'A', dimensionné au niveau du programme principal, par un "common blanc", i.e., Common A(50000)

STRUCTURE GENERALE DU CODE DLEARN

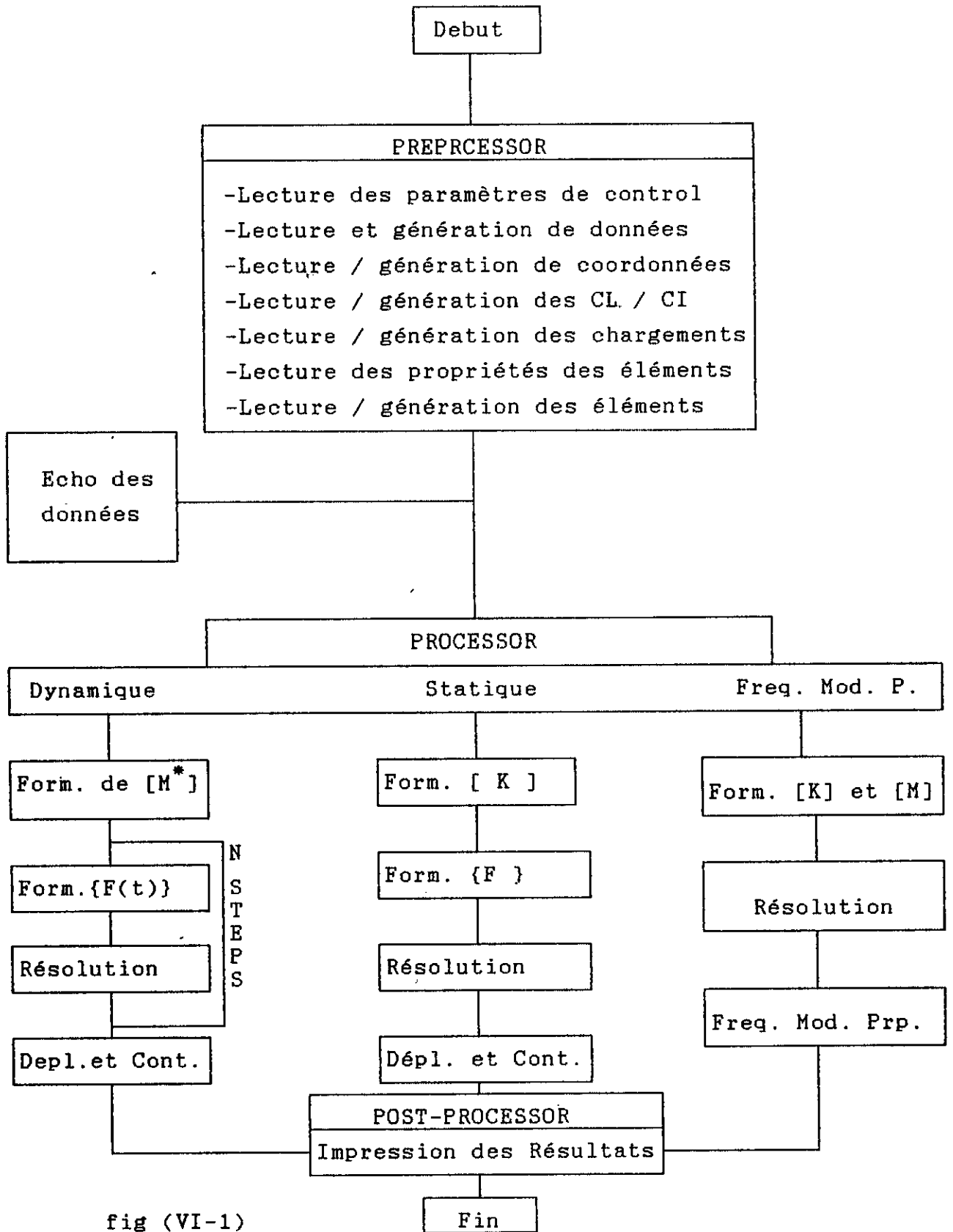
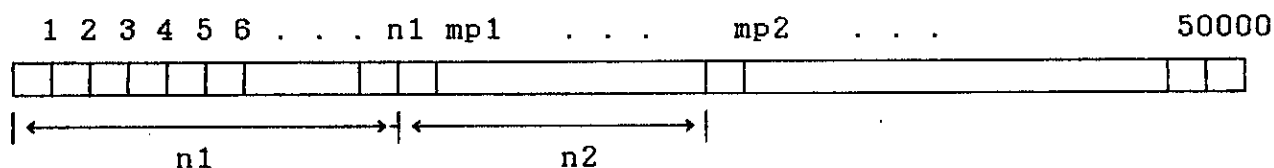


fig (VI-1)

Chaque table ou donnée quelconque (entière, simple précision ou en double précision) est repérée dans le vecteur 'A', par une adresse (pointeur). Les pointeurs sont calculés par une fonction sur le principe de superposition i.e. (l'actuel vecteur commencera à la fin de son précédent). Le schéma suivant illustre la technique :



Soit V1 et V2 deux vecteurs de dimension n1 et n2 respectivement. Supposons que v1 commence à partir de la première case et prend fin à la case n1, alors le prochain vecteur V2 occupera les cases consécutives à n1 i.e. (n1+1 — n1+n2+1), Les pointeurs des deux vecteurs seront donc '1' et 'mp1=n1+1'. Le troisième vecteur qui viendra après V2 aura comme pointeur 'mp2=n1+n2+2'.

Pour stocker dans 'A' un vecteur en double précision il faut lui réserver un nombre de cases égale au double de sa dimension, car 'A' est déclaré en simple précision. Parmi les avantages de l'emploi de cette technique :

- Une meilleure gestion de l'espace mémoire disponible.
- Le nombre d'arguments dans les appels et les déclarations de subroutine est fortement réduit.

REMARQUE : Pour les systèmes de grandes capacités les données peuvent être stockées dans la mémoire centrale "InCore Memory", cependant s'il s'agit de systèmes de capacité limitée il est nécessaire de concevoir une base de données afin de pouvoir utiliser les moyens de stockage extérieurs "Out of Core Storage devices".

VI-3-2 GENERATION DE DONNEES

Dans le but de réduire l'effort de préparation de données, il est prévu une génération automatique de :

1)- *Coordonnées / Noeuds* : On spécifie dans le fichier de données le mode de génération (linéaire, surfacique ou volumique) et le nombre de points de génération et leurs coordonnées respectives. Le Préprocesseur génère l'ensemble des noeuds du maillage et leurs coordonnées.

2)- *Conditions aux limites*: Généralement un grand nombre de noeuds possède le même type de conditions aux limites. Dans la préparation de fichier de données il suffit d'indiquer le premier et le dernier noeuds dans la série et le type de conditions imposées.

3)- *Éléments*: Une technique similaire à celle employée pour le cas des noeuds est utilisée pour la génération d'éléments (numéro de l'élément et les noeuds qui le définient). [Hug 1].

VI-3-2 STOCKAGE DES MATRICES

Afin d'optimiser l'espace mémoire alloué aux matrices [K] et [M], il est prévu un stockage en ligne de ciel "sky line" pour le cas de matrice bandes et en vecteur pour les matrices diagonales.

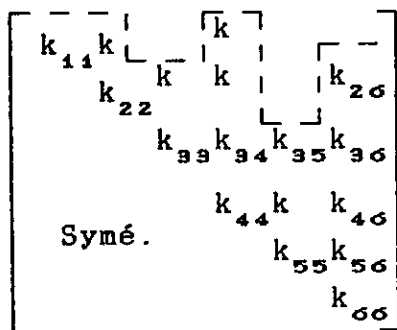


fig (VI-2) - Stocage en ligne de ciel

Les éléments de la matrice sont ordonnés en colonnes descendantes et leurs adresses sont localisées par les positions des éléments diagonaux, les nombres indiquant ces positions sont stockés dans un vecteur appelé en cas de nécessité.

Le profil de la matrice est déterminé avant que les éléments de raideur soit calculés. Les éléments nuls au dessous de la ligne de ciel sont aussi stockés, car après factorisation il deviennent non nuls.

VI-3-3 BLOCK DE RESOLUTION

Deux étapes sont nécessaires pour la résolution de systèmes linéaires $[A]\{U \} = \{R \}$ à savoir :

1)- Etape de factorisation.

On factorise la matrice $[A]$ supposée définie-positve sous la forme :

$$[A] = [U]^T [D] [U] \quad (VI-4)$$

où $[U]$ est une matrice triangulaire supérieure dont les éléments diagonaux sont égaux à l'unité, cette propriété permet de stocker les éléments de $[D]$ (matrice diagonale) sur la diagonale de $[U]$. L'autre avantage de cette technique est que le profil de $[U]$ est le même que celui de $[A]$, ce qui signifie qu'un espace supplémentaire pour $[U]$ ou $[D]$ n'est pas nécessaire.

2)- Etape d'élimination.

il est fait usage de la procédure de Crout, considérée comme une variante de la méthode d'élimination de Gauss. Le vecteur solution est déterminé en 3 phases :

- Résoudre le système : $[U]^T \{Z \} = \{R \}$
- Résoudre : $[D] \{Y \} = \{Z \}$
- vecteur solution : $[U] \{X \} = \{Y \}$

REMARQUE : Si un problème nécessite plusieurs résolutions, on fait une seule factorisation de la matrice. Les prochaines résolutions il suffit de faire seulement l'élimination.

VI-4 INTERVENTION SUR LE CODE

Pour implanter les modules manquants d'analyse modale (recherche des valeurs et vecteurs propres), à savoir; les blocks Lanczos et Sous-espace il a fallu :

- Créer des interfaces entre le code et les modules précédents.
- Faire des modifications sur DLEARN pour permettre le calcul de

[K] et [M], car dans l'ancienne version, une seule matrice est utilisée, [K] pour le cas statique et [M*]⁽¹⁾ pour le cas dynamique [Hug 1]

VI-5 RESULTATS NUMERIQUES :

L'efficacité d'un algorithme peut être mesurée par les critères de précision et d'économie, dans cette section nous allons essayer d'étudier les deux aspects ensemble.

VI-5-1 PRECISION : La précision sur les résultats finaux ne dépend pas seulement de l'algorithme utilisé, mais aussi de la formulation du problème comme nous aurons à le constater plus tard. Maintenant essayons d'analyser l'erreur finale à partir des tests (1), (2), (3) et (4).

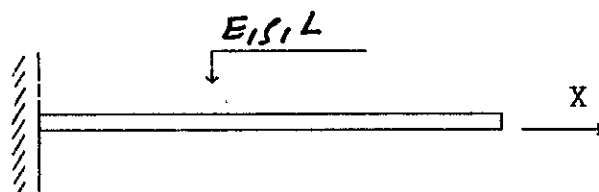
Pour vérifier la précision de l'algorithme de Lanczos, nous avons effectué une comparaison avec la solution exacte pour le cas d'une barre encastrée (Fig. ci-dessous). Les résultats obtenus montrent que les premiers modes (1-10) sont près proches de la solution exacte (Fig.VI-3). au delà du 10^{ème} mode l'erreur commence à grandir. On constate que cette erreur est due principalement à la formulation concentrée de la matrice masse. (Fig.VI-12)

Avec :

$$E = 0.717 \text{ E}+011 \text{ N/m}^2$$

$$\rho = 2768.0 \text{ kg/m}^3$$

$$L = 5.0 \text{ m}$$



^{*}
(1) [M*] est dite " effective mass matrix "

L'expression de la solution exacte est donnée par :

$$\omega_n = \frac{2n + 1}{2L} \sqrt{E/\rho}$$

VI-5-2 ANALYSE DE L'ERREUR : Dans la fig.(VI-5) nous avons tracé avec une échelle agrandie (10 fois plus) l'erreur due au deux algorithmes (de Lanczos et sous-espace(sap80)), on remarque que l'évolution de l'erreur pour la méthode de Lanczos est sous forme parabolique, tandis que pour sous-espace la forme de l'erreur n'est pas stable, mais puisque le taux max. de cette erreur est faible (< 10 %), on adoptera comme référence pour les exemples prochains la solution donnée par le SAP80.

VI-5-3 INFLUENCE DE LA FORMULATION DE LA MATRICE MASSE : A travers les figures (VI-12) et (VI-13) on remarque que l'effet du type de la matrice masse commence à apparaitre pour les éléments barre à partir du 10^{ème} mode, et pour les éléments quadrilatères à partir du 5^{ème} mode. On peut dire que pour un calcul plus précis des modes supérieurs de la structure, il est recommandé d'utiliser [M] consistante.

Ces résultats ont été confirmés par Zienkiewicz [Zei 2] qui recommande l'utilisation de la matrice consistante.

VI-5-4 COMPORTEMENT DE L'ALGORITHME DE LANCZOS :

Nous avons fait des tests sur l'évolution de la convergence pour les problèmes 1 et 2 (barre et plaque), les résultats sont reportés sur les figures (VI-8) et (VI-11). On remarque que le nombre d'itérations nécessaire à la convergence d'une valeur propre est relativement grand au départ (paliers), mais à partir d'un certain point la convergence s'accélère pour atteindre une itération pour la convergence d'une nouvelle valeur. En moyenne on peut dire que le nombre d'itérations requis pour la convergence d'une valeur est de 2.

Les figures (VI-10) et (VI-9) montrent l'allure générale de la convergence.

VI-5-5 EFFET DU DECALAGE :

On a fait des tests de calcul de 15 valeurs propres pour le problème N° 3 (Fig.VI-17), les résultats obtenus montrent que le décalage réduit le temps de calcul en accélérant la convergence de la méthode.

VI-5-6 COMPARAISON AVEC LA METHODE DU SOUS-ESPACE :

On a programmé la méthode du sous-espace mais on n'a pas pu obtenir la convergence pour un nombre suffisant de valeurs propres, pour cette raison nous avons pris les résultats de [Nou 8].

La figure (VI-16) montre la différence dans le nombre d'opérations entre les deux méthodes.

Il est clair donc que la méthode de Lanczos demande moins d'opérations arithmétiques donc moins de temps d'exécution, i.e : Lanczos est plus économique que la méthode du sous-espace.

REMARQUE :

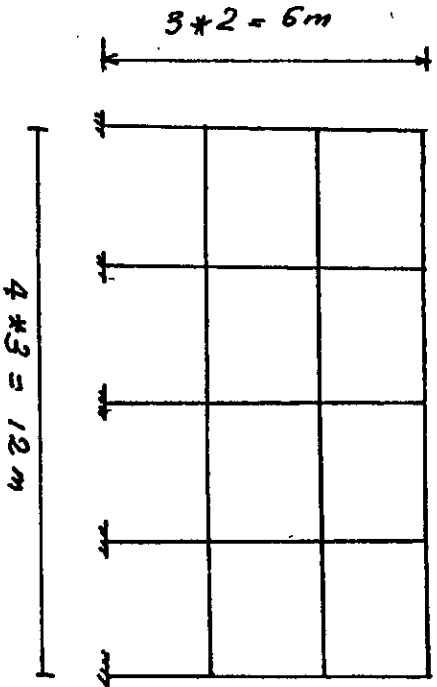
Les calculs (exécutions) ont été faits sur la station micro-VAX de l'ENP. La taille du logiciel ne nous a pas permis l'utilisation du PC, ce qui nous a fait perdre beaucoup de temps car le micro-VAX n'était pas fonctionnel jusqu'à la fin du mois de juin.

VI-6 CONCLUSION

Ce code 'DLEARN', sur lequel nous avons travaillé n'est pas conçu pour être un logiciel " clef en main ", vu le nombre d'éléments qu'il contient et la classe des problèmes qu'il peut

traiter " Lineaires ", néanmoins sa structure modulaire offre une grande possibilité d'extension, par additions de nouveaux types d'éléments manquants (éléments flexionnels par exemple), où alors étendre la classe des problèmes au cas non linéaire.

Test 4

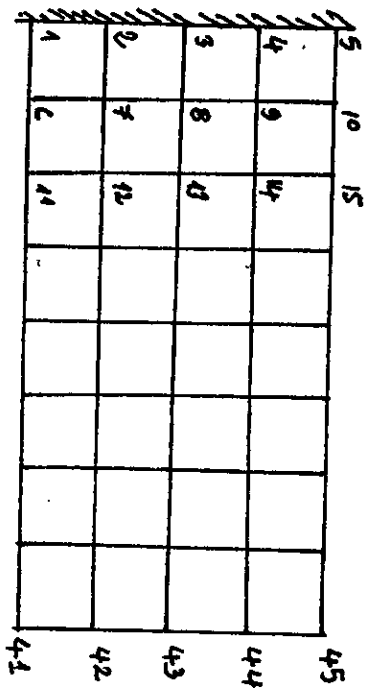


$E = 1.0 \text{ E} + 0.3 \text{ N/m}^2$
 $\rho = 1 \text{ kg/m}^3$
 $A = 1.0 \text{ m}^2$
 Moment d'inertie = 1.0 m^4

Nb. d'éléments : 27
 Nb. de Nœuds : 20
 Nb. d'équation : 45

Fig (IV-1)

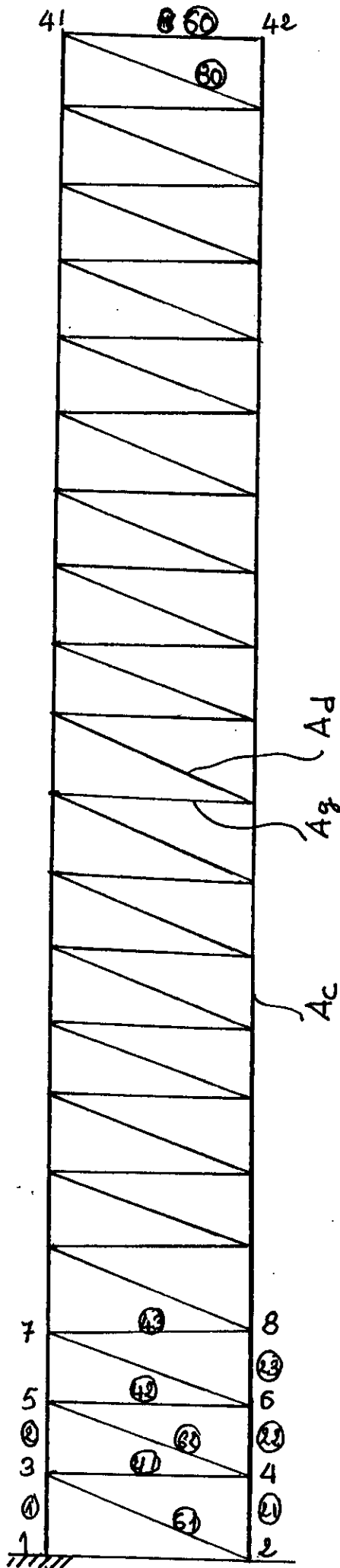
Test 2



$E = 7.17.10^{10} \text{ N/m}^2$
 $\rho = 2768 \text{ kg/m}^3$
 $Lh = 0.2 \text{ m}$

Nb. d'éléments : 32
 Nb. de Nœuds : 45
 N° d'éq : 80

Test No 3



$$A_c = 80 \cdot 10^{-6} \text{ m}^2$$

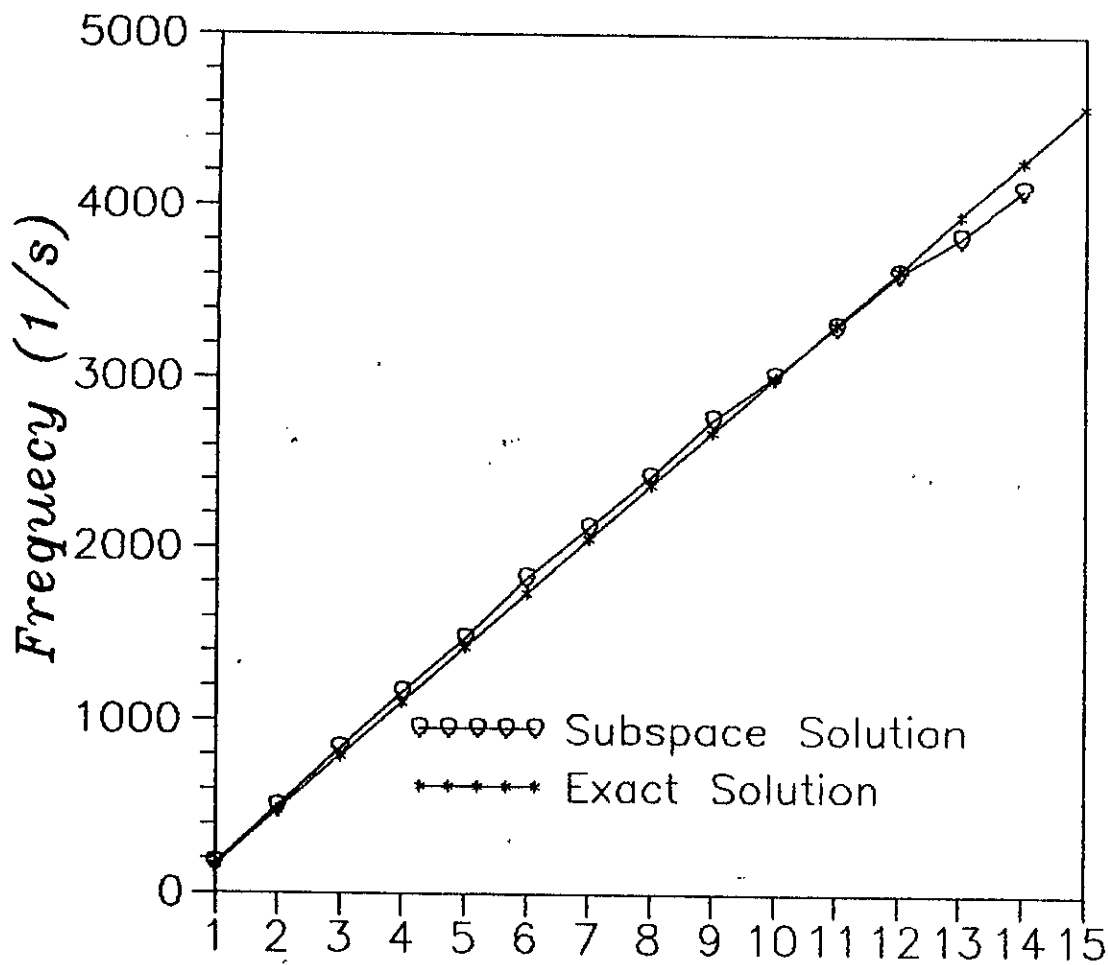
$$A_d = 40 \cdot 0 \cdot 10^{-6} \text{ m}^2$$

$$A_g = 60 \cdot 10^{-6} \text{ m}^2$$

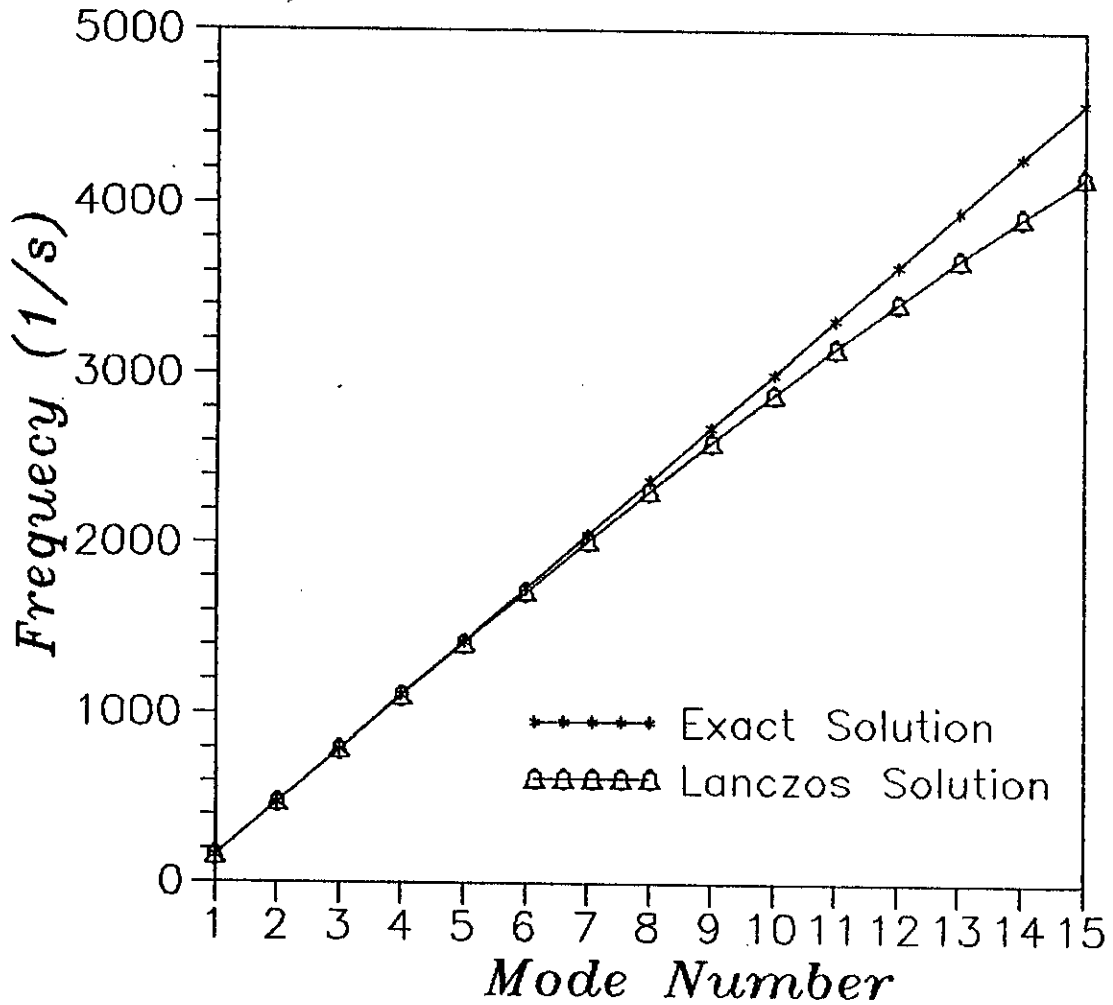
$$\rho = 2768 \text{ kg/m}^3$$

$$E = 7.77 \cdot 10^{10} \text{ N/m}^2$$

Fig. (IV. 2)

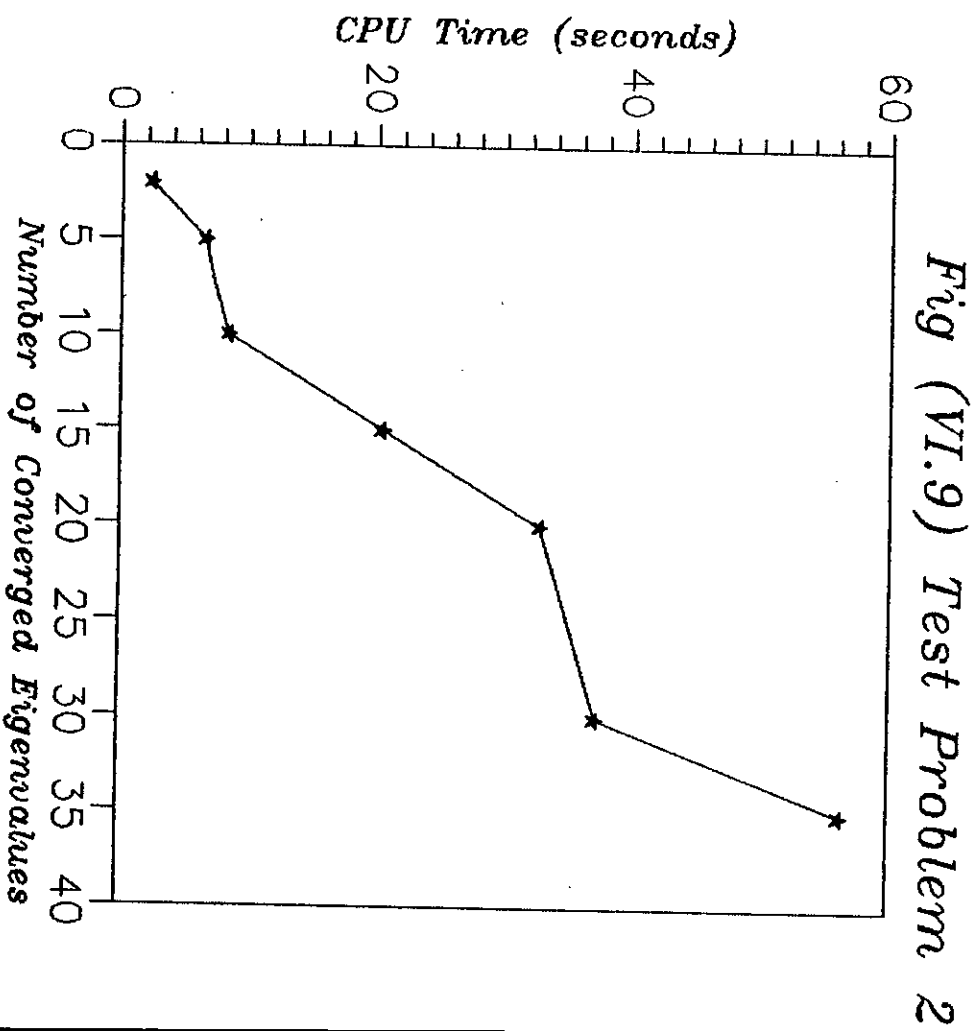
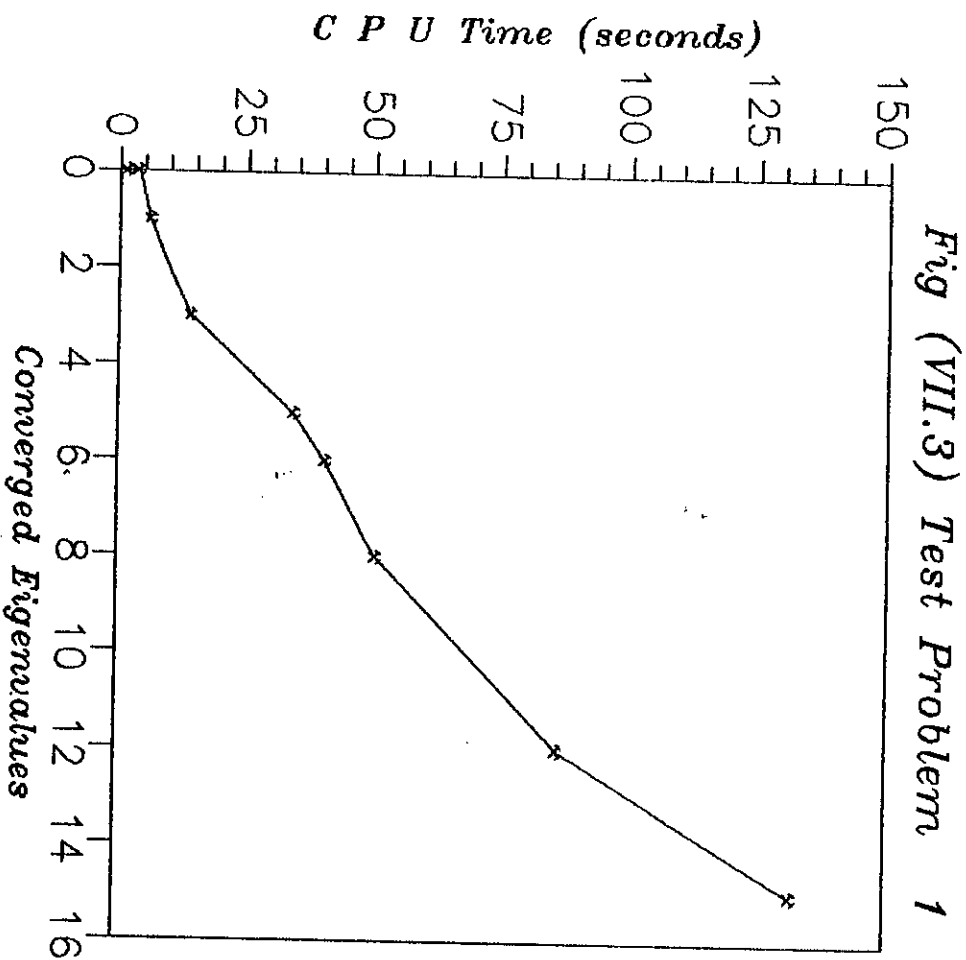


*Fig(VI.4)
Subspace*



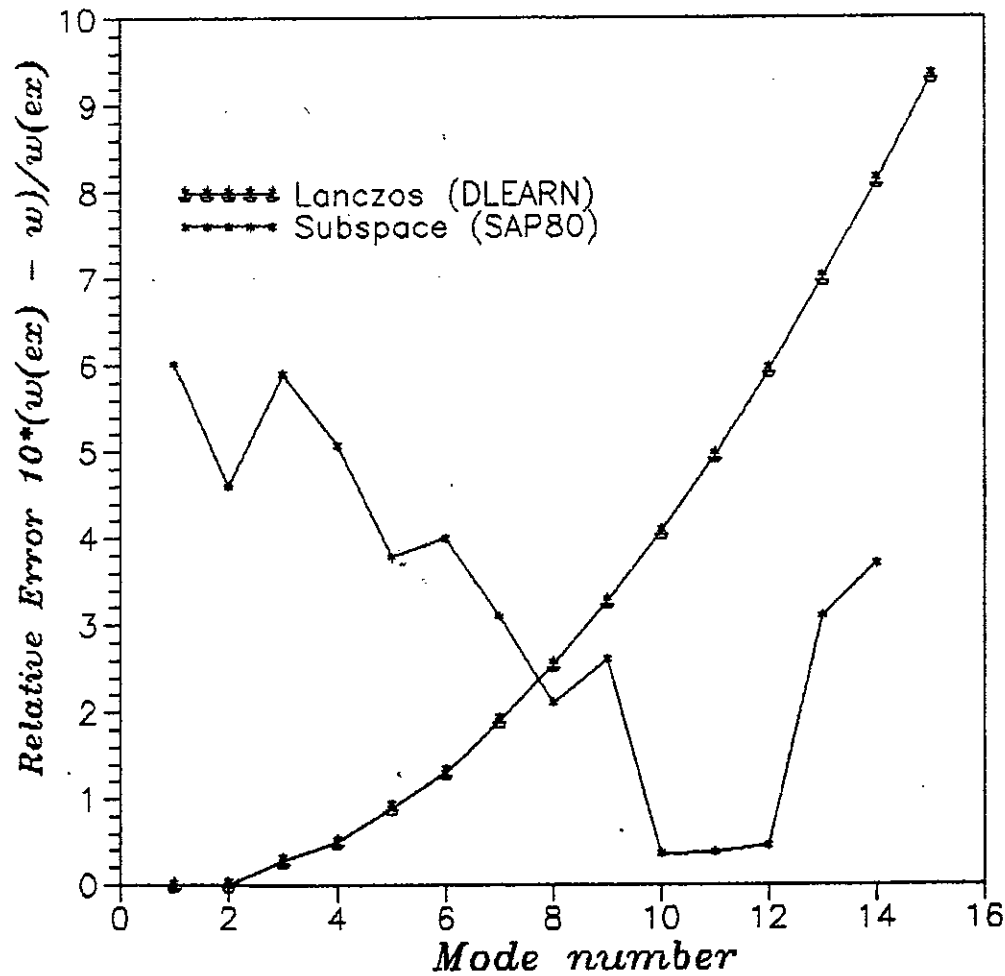
*Fig(VI.3)
Lanczos*

Convergence Speed



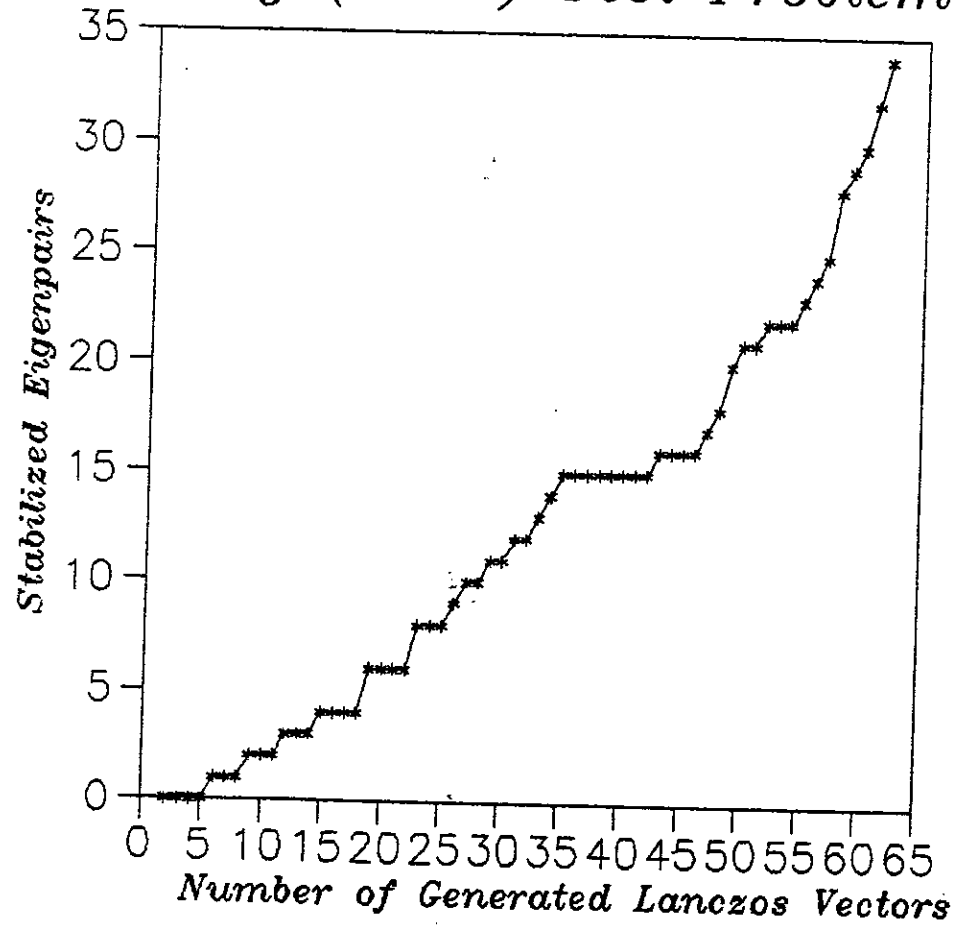
Fig(VI.5) Test Problem 1
Error Analysis for Lumped Masses

(Error max. = 10%)

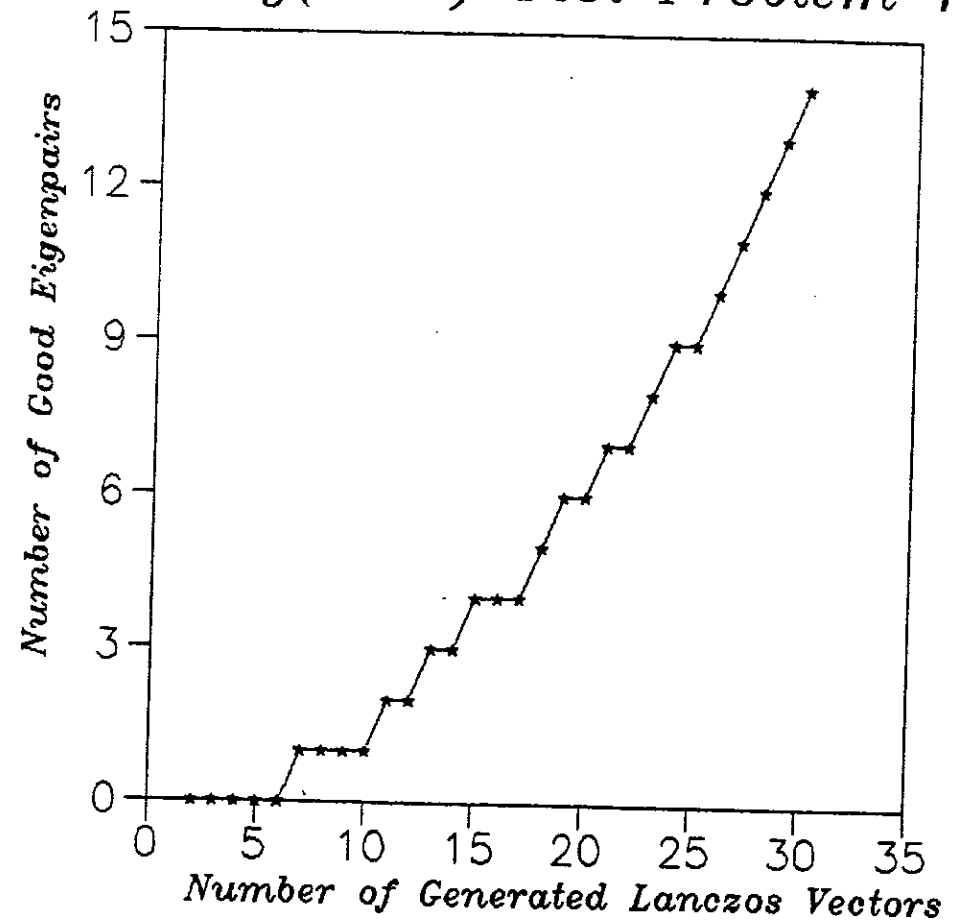


Evolution of the Convergence Process

Fig (VI.11) Test Problem 2

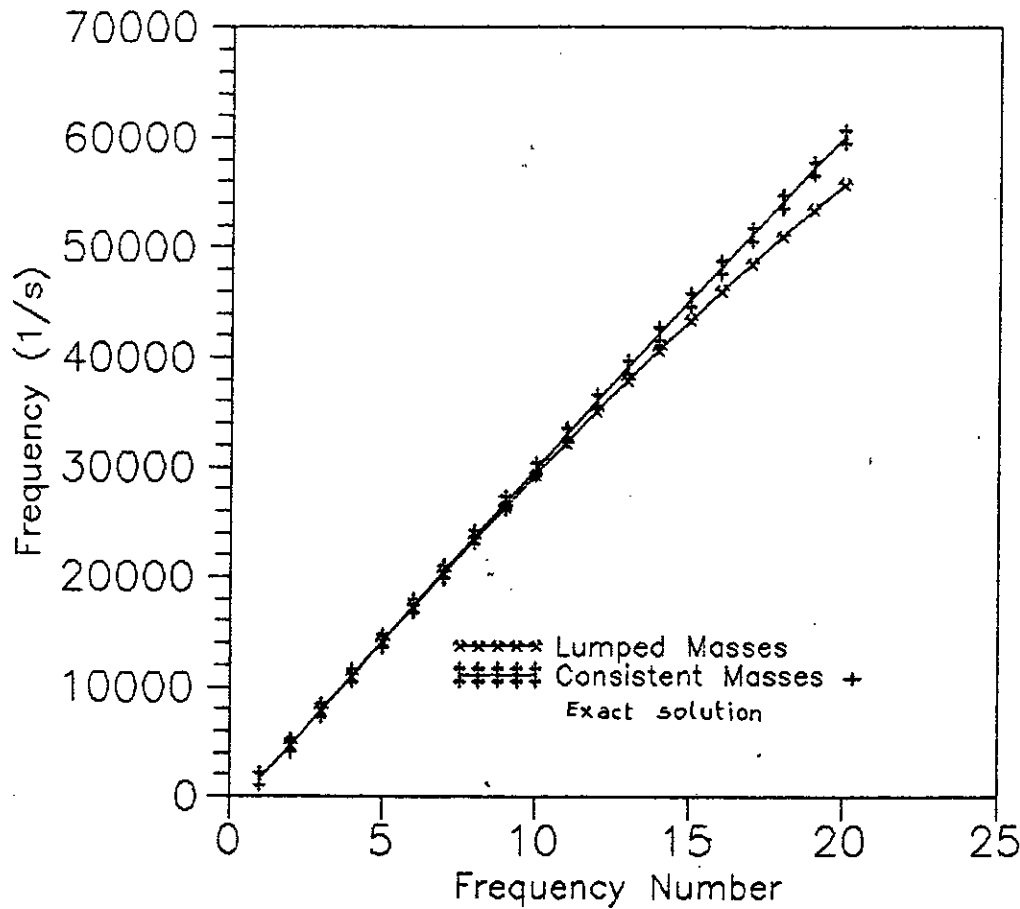


Fig(VI-8) Test Problem 1

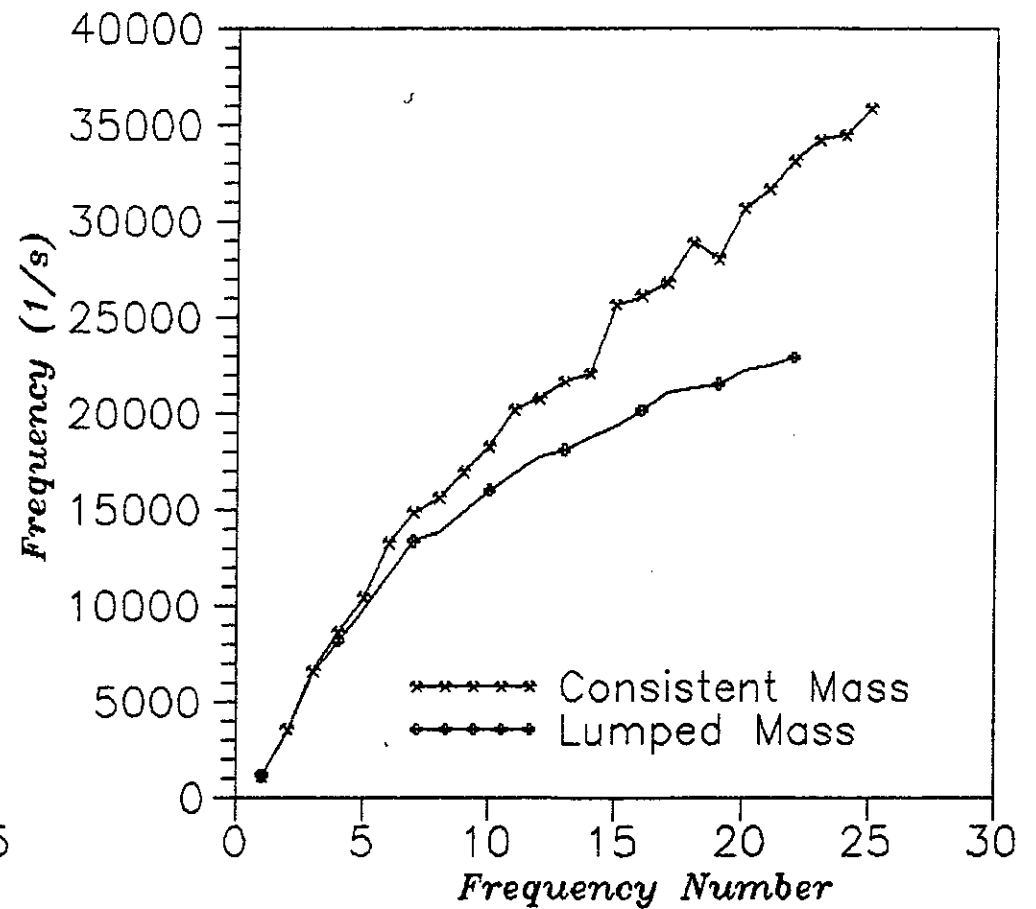


Convergence of Consistent and lumped Masses

Fig(VI.12)
Test Problem 1

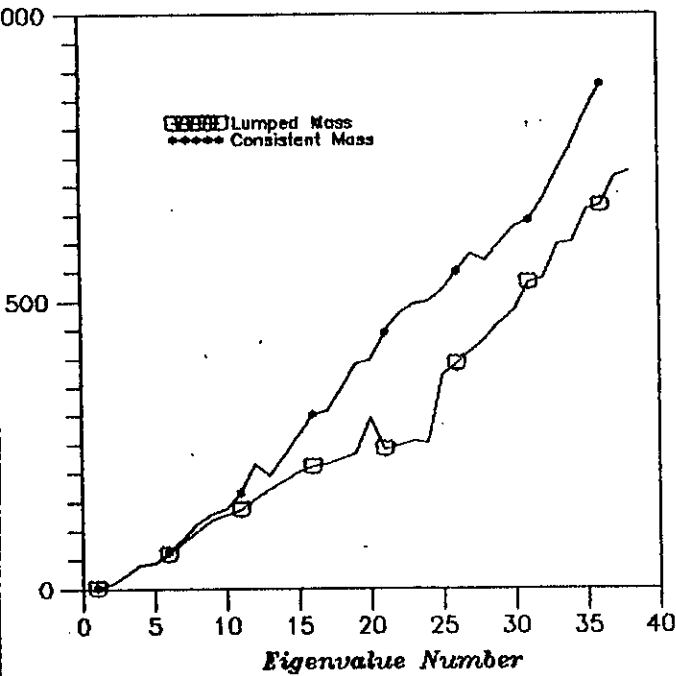


Fig(VI.13)
Test Problem 2

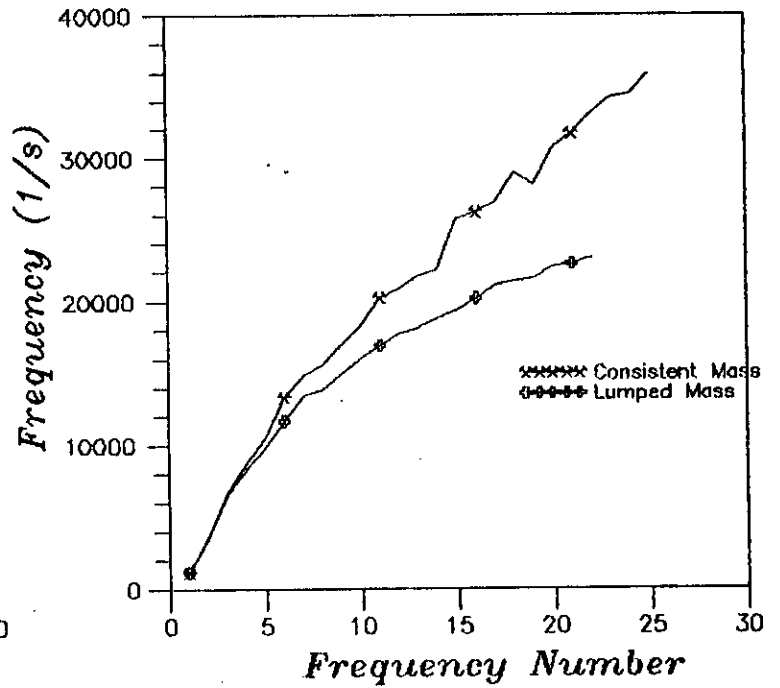


Lumped to Consistent Masses

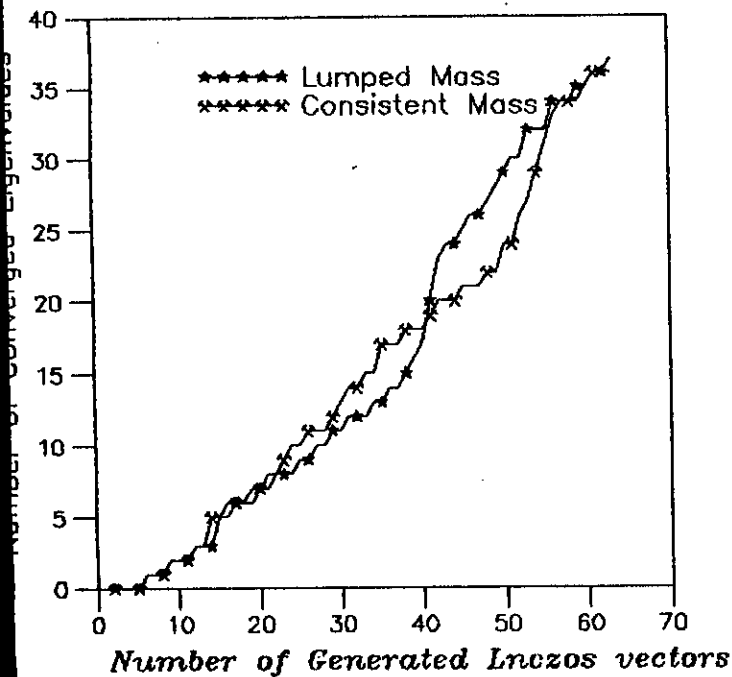
Test Problem 3



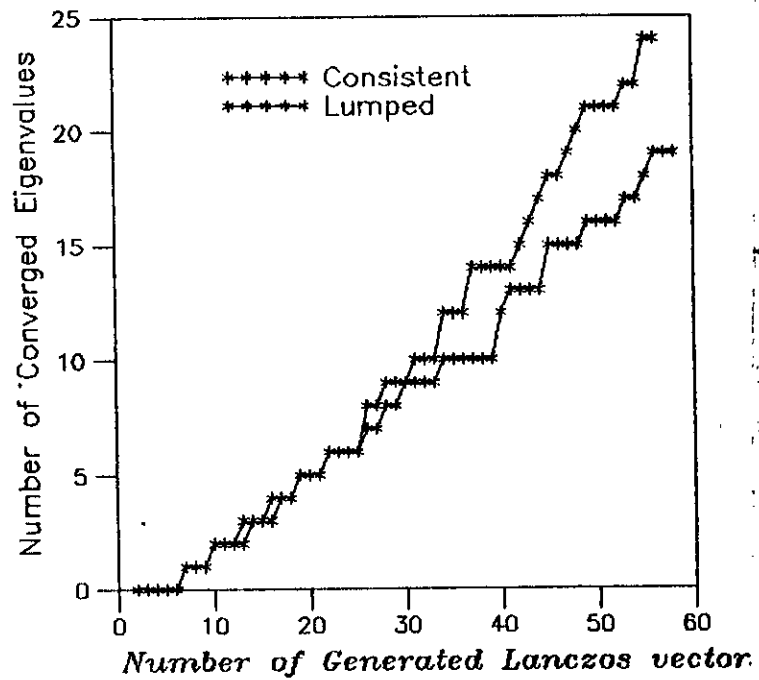
Test Problem 2



Test Problem 3

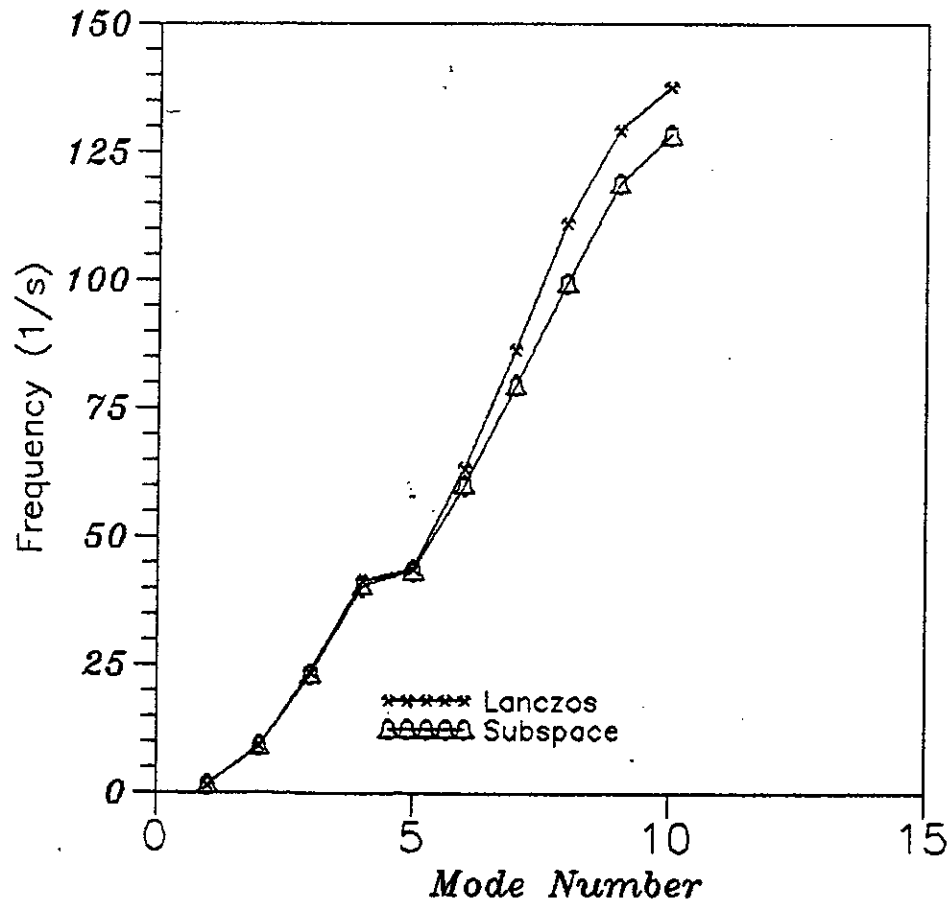


Test Problem 2

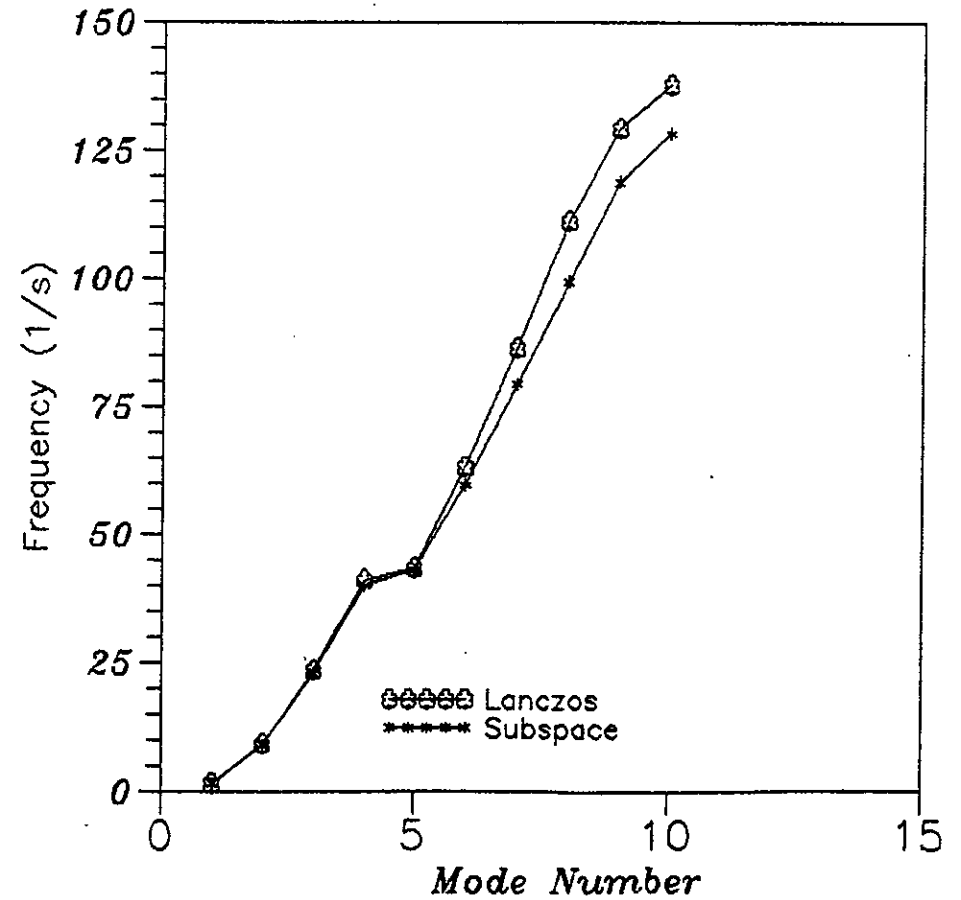


Lanczos (Dlearn) To Subspace (Sap80)

Fig(VI.14)
Test Problem 3
Lumped Masses

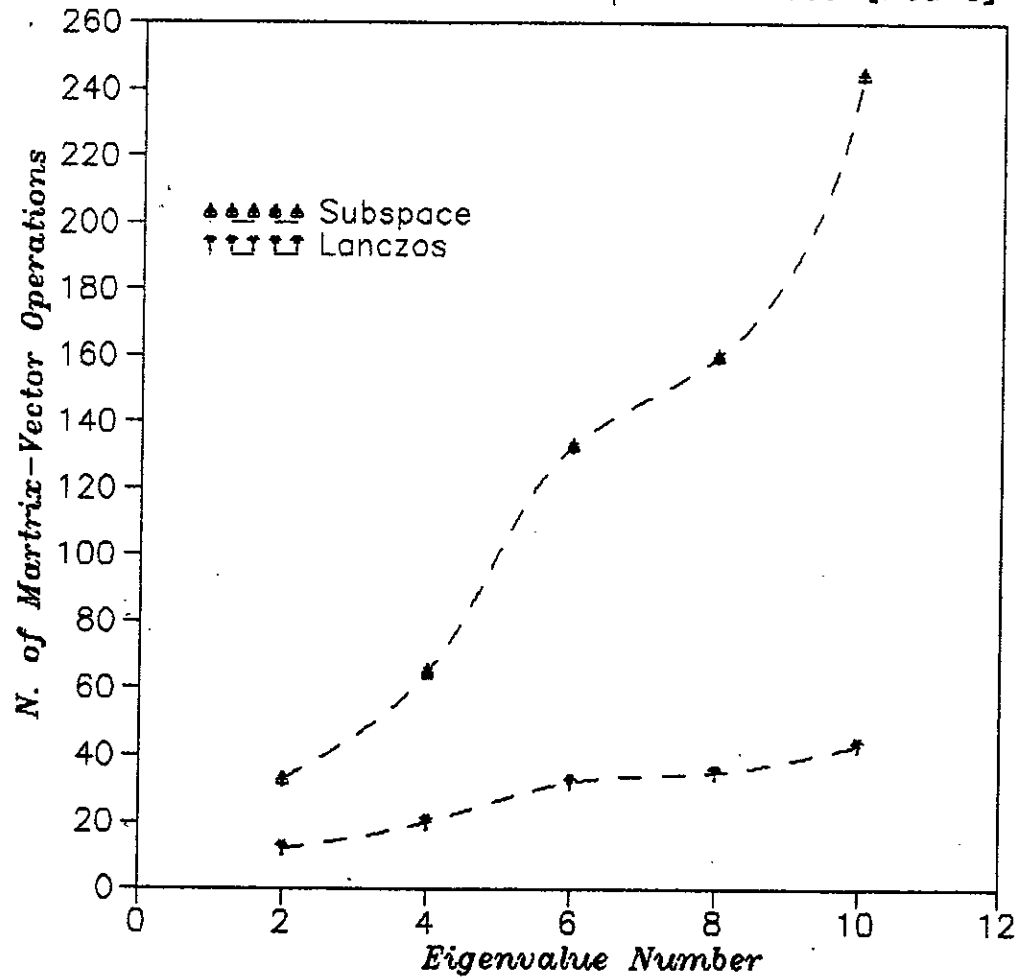


Fig(VI.15)
Test Problem 3
Consistent Masses



Fig(VI.16)

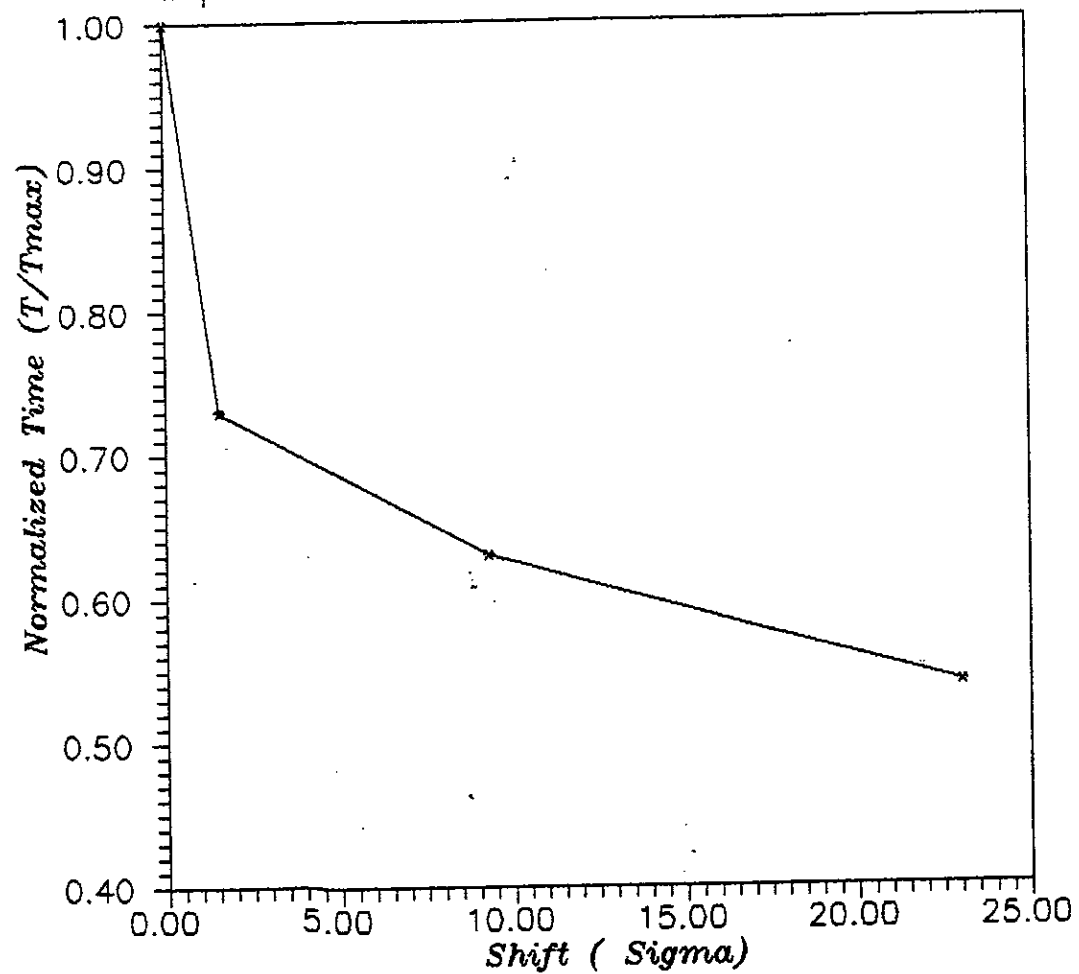
Test Problem 4
Comparison of the Matrix vector Operations
Between Lanczos and Subspace Methods [Nou 8]



Fig(VI.17)

Test Problem 3

Improvement of Solution Times for Obtaining the First (15) Modes



CONCLUSION

Dans cette étude, on a essayé d'étudier certaines procédures numériques, utilisées pour la résolution du problème aux valeurs propres, posé en dynamique des structures modélisées en éléments finis. Les systèmes résultants d'une telle analyse sont souvent de tailles importantes, pour cette raison des procédures combinant les techniques de réduction et les méthodes classiques de résolution du problème aux valeurs propres, sont conçues pour répondre à cette exigence.

Parmi les méthodes les plus connues, on trouve *Sous-espace* et *Lanczos*. On a eu comme objectif principal l'implantation de la méthode de Lanczos dans le code *DLEARN*. Cette méthode a suscité beaucoup de travaux pour lui éliminer l'instabilité numérique due à la précision limitée des processeurs de calcul. Les résultats de ces travaux ont encouragés le retour à l'exploitation de la méthode qui constitue actuellement l'une des plus performantes.

L'idée de travailler avec la source d'un code préexistant, est choisie pour mieux rentabiliser notre travail. De plus ça nous a permis d'appréhender les différentes techniques de programmation en usage dans les codes MEF, ce que je considère comme un grand acquis.

Le code est loin d'être un logiciel complet, puisqu'il manque beaucoup d'autres options d'analyse, pour cette raison j'espère qu'il y aura dans les années avenir des étudiants qui contribueront à l'extention de ses possibilités. Toutefois il est impératif que les moyens de calcul nécessaires soient disponibles.

ANNEXES

FORMATION DES MATRICES [K] ET [M]

On a vu que le calcul par éléments finis des structures nécessite le calcul des matrices Masse, Rigidité et chargement, pour cette raison on a prévu cet annexe. Notons que la formulation sera faite pour la version déplacements⁽¹⁾

A1-1 FORMATION DE LA MATRICE [K] :

Dans la première phase on définit le polynôme d'interpolation (dépendant de la nature de l'élément), $P(x,y,z)$, Ce polynôme approxime le champ de déplacement au niveau d'un élément i.e.,

$$u_{ex} \cong u(x,y,z) = \langle P(x,y,z) \rangle \{a\} \quad (A1-1)$$

où,

u_{ex} : solution exacte

u : solution approchée

$\{a\}$: Paramètres de combinaison polynomiale.

Cette approximation est valable pour toutes les variables nodales par conséquent on généralise :

$$\{u_n\} = [P_n]\{a\}; \text{ avec } [P_n] = [\langle P(x_i, y_i, z_i) \rangle] \quad (A1-2)$$

si, $[P_n]$ est inversible on peut déterminer les paramètres $\{a\}$ tel que: $\{a\} = [P_n]^{-1}\{u_n\}$, et approximer le champ de déplacement en fonction des variables nodales à savoir :

$$u(x) = \langle N_1(x) \quad N_2(x) \quad \dots \quad N_n(x) \rangle \begin{Bmatrix} u_1 \\ \vdots \\ u_n \end{Bmatrix} \quad (A1-3)$$

$$[\langle N_1(x) \quad N_2(x) \quad \dots \quad N_n(x) \rangle] = \langle p(x,y,z) \rangle [P_n]^{-1} \quad (A1-4)$$

(1) version la plus populaire

Les fonctions $N_i(x)$, avec $x = \langle x \ y \ z \rangle$, sont appelées fonctions de forme (d'interpolation), elles doivent vérifier :

- 1)- $N_i(x_i, y_i, z_i) = \delta_{ij}$
- 2)- $\sum_{i=1}^n N_i(x, y, z) = 1$; n nombre de nœuds.
- 3)- $\sum_{i=1}^n N_i(x, y, z) L(x_i, y_i, z_i) = L(x, y, z)$;
L : polynôme quelconque

Sachant que les relations liant déformations aux déplacements : $\{\epsilon\} = [L] \{u\}$

où $[L]$ est un opérateur différentiel. En utilisant la relation (A1-3) on peut écrire :

$$\{\epsilon\} = [L][N] \{u_i\}$$

ou alors:

$$\{\epsilon\} = [B] \{u_i\}; \quad [B] = [L][N] \quad (A1-4)$$

D'autre part d'après la loi de Hooke : $\{\sigma\} = [D] \{\epsilon\}$, $[D]$ est fonction du matériau seulement.

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{xz} \end{Bmatrix} = \begin{bmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ & 1-\nu & \nu & 0 & 0 & 0 \\ & & 1-\nu & 0 & 0 & 0 \\ & & & .5-\nu & 0 & 0 \\ \text{symétrique} & & & & .5-\nu & 0 \\ & & & & & .5-\nu \end{bmatrix} \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \epsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{xz} \end{Bmatrix} \quad (A1-5)$$

Ainsi on arrive à formuler l'expression de la matrice $[K]^\circ$ en utilisant les formules (A1-4) et (A1-5).

$$[K]^\circ = \int_{\Omega^\circ} [B][D][B] d\Omega^\circ \quad (A1-6)$$

A1-2 FORMATION DE LA MATRICE $[M]$

On a vu au chapitre II que la formulation de $[M]$ peut être

faite par deux techniques différentes à savoir :

a)- *FORMULATION CONSISTANTE* :

Cette formulation découle des concepts théoriques de la méthode des éléments finis, elle bénéficie d'un appui mathématique rigoureux d'après Zeinkeiwicz

$$[M]^\circ = \int_{\Omega^\circ} \rho^\circ [N]^\top [N] d\Omega^\circ \quad (A1-7)$$

où [N] sont les fonctions de forme

En réalité la matrice [M] doit être approximée par rapport à la vitesse et non pas les déplacements [Zei2],

$$[M]^\circ = \int_{\Omega^\circ} \rho^\circ [N']^\top [N'] d\Omega^\circ \quad (A1-8)$$

où [N'] sont les fonctions de forme sur les vitesses, mais pour des raisons de faciliter l'implantation de la méthode on prend [N']=[N].

b)- *FORMULATION CONCENTREE*

Le principale avantage de cette technique est le profil diagonal de la matrice résultante. Le concept sur lequel e basée était au départ physique, cependant un appui mathématique est nécessaire selon certains auteurs [Zei 2].

Première Approche : A partir du principe de conservation , on réparti la masse de l'élément sur les noeuds de façon équitable, par exemple :



Fig (A1-1) : Conservation de masse sur le principe de conservation.

Cette technique est presque abandonnée actuellement, elle est remplacée par d'autres techniques de *Diagonalisation* de la matrice consistante [Hug 1] et [Zei 2]. Parmi ces techniques On peut citer " *Diagonalisation par Quadrature* ", " *Sommation des linges* ", ces deux dernières peuvent produire des masses nulles ou

négatives. Celle développée par Hinton et al semble être la meilleure puisque elle permet d'obtenir des masses diagonales positives.

$$m_{pq}^e = \begin{cases} \alpha \delta_{ij} \int_{\Omega^e} \rho_e [N_k]^2 d\Omega^e & p=q \\ 0 & p \neq q \end{cases} \quad (A1-9)$$

où α est^{un} coefficient défini par :

$$\alpha = \frac{\int_{\Omega^e} \rho_e d\Omega^e}{\sum_k^{nen} \int_{\Omega^e} \rho_e [N_k]^2 d\Omega^e} \quad (A1-10)$$

où nen : Nombre de noeuds de l'élément .
 ρ_e : Densité massique de l'élément.
 $[N]$: Fonction de forme

Cette technique est adoptée dans le code DLEARN pour la formulation des masses concentrées des différents éléments.

REMARQUE : La technique de diagonalisation converge moins comparée à la formulation consistante, pour cette raison on adopte la dernière formulation pour les analyses de grande précision.

PROCEDURES DE REDUCTION

Cet annexe sera consacré à certaines procédures de réduction employées en dynamique et en statique des structures.

A1-1 PROCEDURE DE RAYLEIGH RITZ [Hug 1]

Cette procédure permet d'approximer certains vecteurs propres du problème original défini en (III-23), en fonction des vecteurs propres d'un problème réduit, de taille $m \ll n$. Soit $(\{y\}, \vartheta)$ une solution approchée du problème :

$$([K] - \lambda [M])\{\phi\} = 0 \quad (A2-1)$$

On définit le vecteur résiduel $\{r\}$ par :

$$\{r\} = [K]\{y\} - \vartheta [M]\{y\} \quad (A2-2)$$

Soit $Q, [q_1, q_2, \dots, q_n]$ une base orthogonale quelconque, on exprime $\{y\}$ dans cette base :

$$\{y\} = [Q]\{s\} \quad (A2-3)$$

où $\{s\}$ un vecteur contenant les paramètres de combinaison. La procédure de Rayleigh-Ritz exige que le vecteur $\{r\}$ soit orthogonal à tous les vecteurs $\{q_i\}$ i.e.,

$$[Q]^T\{r\} = [Q]^T[K]\{y\} - \vartheta [Q]^T[M]\{y\} = 0 \quad (A2-4)$$

En substituant l'expression de $\{y\}$ dans (A2-4) et en tenant compte de la condition d'orthogonalité de Ritz on obtient :

$$[K^*]\{s\} = \vartheta [M^*]\{s\} \quad (A2-4)$$

avec.

$$[K^*] = [Q]^T[K][Q]; \quad [M^*] = [Q]^T[M][Q]$$

Une fois la paire $(\vartheta, \{s\})$ est calculée on déduit les solutions du problème initial en posant :

$$\lambda \cong \vartheta ; \quad \{\phi\} \cong \{y\} = [Q] \{s\} \quad (\text{A2-5})$$

A2-2 CONDENSATION STATIQUE (dite de GUYAN)

En pratique on rencontre souvent des problèmes aux valeurs propres avec des éléments diagonaux de la matrice masse nuls, dans ce cas pour appliquer la condensation statique on restructure le système en blocks :

$$\left(\begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} - \lambda \begin{bmatrix} M_{11} & 0 \\ 0 & 0 \end{bmatrix} \right) \begin{Bmatrix} \phi_1 \\ \phi_2 \end{Bmatrix} = 0 \quad (\text{A2-6})$$

où $[M_{11}]$ est une matrice symétrique définie-positive. on peut régulariser la matrice masse globale on ajoutant des petites valeurs positives aux éléments diagonaux nuls [Hug 1], ou alors éliminer les zéros par condensation statique. On développe (A2-6) par blocks :

$$[K_{11}] \{\phi_1\} + [K_{12}]\{\phi_2\} - \lambda [M_{11}]\{\phi_1\} = 0 \quad (\text{A2-7})$$

$$[K_{21}] \{\phi_1\} + [K_{22}]\{\phi_2\} = 0 \quad (\text{A2-8})$$

L'étape suivante est de résoudre (A2-8) pour $\{\phi_2\}$ puis substituer dans (A2-7), Le résultat est le suivant :

$$([K_{11}^*] - \lambda [M_{11}]) \{\phi_1\} = 0 \quad (\text{A2-9})$$

où $[K_{11}^*]$ est la matrice condensée dont l'expression est :

$$[K_{11}^*] = [K_{11}] - [K_{12}][K_{22}]^{-1}[K_{21}] \quad (\text{A2-10})$$

REMARQUE :

La condensation fait perdre à $[K]$ son profil, et produit une matrice qui tend à être pleine, pour cette raison la méthode n'est pas économique pour les grands systèmes.

DEVELOPPEMENT MATHEMATIQUE DE L'ALGORITHME DE LANCZOS

En partant d'un vecteur initial l'algorithme de Lanczos génère une série de vecteurs M-orthogonaux appelés vecteurs de Lanczos, ces vecteurs sont utilisés dans la procédure de Rayleigh-Ritz pour réduire le problème original (III-23) en un problème réduit dont la matrice est tridiagonale .

A3-1 CONSTRUCTION DES VECTEURS DE LANCZOS

On décrit ici comment générer un vecteur de Lanczos $\{q_{j+1}\}$, en supposant que j vecteurs ont été déjà calculés. Il faut se rappeler que ces vecteurs sont orthogonaux et que leurs formulations est faite sur la base de la série de Krylov définie par l'opérateur $[D] = [K_\sigma]^{-1} [M]^{(1)}$ et un vecteur arbitraire $\{q\}$ [TAYLOR] à savoir $([D]\{q\}, [D]^2\{q\}, \dots [D]^j\{q\})$. Le calcul de $\{q_{j+1}\}$ passe par le calcul d'un vecteur préliminaire $\{\bar{r}_j\}$.

$$\{\bar{r}_j\} = [D] \{q_j\} \quad (A3-1)$$

ce vecteur peut être exprimé comme une combinaison linéaire des vecteurs de Lanczos et un vecteur résiduel $\{r_j\}$,

$$\begin{aligned} \{\bar{r}_j\} = \{r_j\} + \alpha_j \{q_j\} + \beta_{j-1} \{q_{j-1}\} \\ + \epsilon_{j-2} \{q_{j-2}\} + \dots \end{aligned} \quad (A3-2)$$

M-orthogonal par rapport au vecteurs de Lanczos précédents. Les paramètres α_j et β_{j-1} et ϵ_{j-2} sont les composantes de $\{\bar{r}_j\}$ dans les directions des vecteurs $\{q_j\}$, ces paramètres sont obtenus par l'imposition de la condition d'orthogonalité, multiplions à gauche par $\{q\}^T [M]$, on obtient :

(1) $[K_\sigma] = [K] - \sigma [M]$, σ est le décalage.

$$\{q_j\}^T [M] \{\bar{r}_j\} = \{q_j\}^T [M] \{r_j\} + \alpha_j \{q_j\}^T [M] \{q_j\} + \beta_{j-1} \{q_j\}^T [M] \{q_{j-1}\} + \varepsilon_{j-2} \{q_j\}^T [M] \{q_{j-2}\} + \dots \quad (A3-2)$$

Du fait de l'orthogonalité *ie.* ($\{q_j\} [M] \{q_i\} = 0$ pour tout $i \neq j$), les termes à droite disparaissent sauf le deuxième, par suite on détermine la composante de $\{\bar{r}_j\}$ selon $\{q_j\}$:

$$\alpha_j = \frac{\{q_j\}^T [M] \{\bar{r}_j\}}{\{q_j\}^T [M] \{q_j\}} \quad (A3-3)$$

De façon similaire on détermine la valeur de β_{j-1} , en multipliant par $\{q_{j-1}\} [M]$, on obtient :

$$\beta_{j-1} = \frac{\{q_{j-1}\}^T [M] \{\bar{r}_j\}}{\{q_{j-1}\}^T [M] \{q_{j-1}\}} \quad (A3-4)$$

et ε_{j-1} aussi .

$$\varepsilon_{j-1} = \frac{\{q_{j-2}\}^T [M] \{\bar{r}_j\}}{\{q_{j-2}\}^T [M] \{q_{j-2}\}} \quad (A3-5)$$

Maintenant on va montrer que ε_{j-2} est en réalité nul, en utilisant la formule (A3-1).

$$\{q_{j-2}\}^T [M] \{\bar{r}_j\} = \{q_{j-2}\}^T [M] [D] \{q_j\}$$

$$\text{puisque : } \{q_{j-1}\} [A] \{q_j\} = \{q_j\}^T [A] \{q_{j-1}\}$$

$$\begin{aligned} \text{donc : } \{q_{j-2}\}^T [M] [D] \{q_j\} &= \{q_j\}^T [M] [D] \{q_{j-2}\} \\ &= \{q_j\}^T [M] \{\bar{r}_{j-1}\} \end{aligned}$$

En développant $\{\bar{r}_{j-1}\}$ de la même façon qu'en (A3-2) on a :

$$\{q_j\}^T [M] \{\bar{r}_{j-1}\} = \{q_j\}^T [M] \left[\{r_{j-1}\} + \alpha_{j-3} \{q_j\} + \beta_{j-3} \{q_{j-3}\} + \varepsilon_{j-2} \{q_{j-4}\} + \dots \right]$$

A cause de l'orthogonalité tous les termes à droite valent zero, par conséquent ($\varepsilon_{j-2} = 0$). Avec le même cheminement on démontre que tous les coefficients consécutifs à ε_{j-2} sont nuls. Ce résultat conduit à dire que $\{r_j\}$ est exprimé en fonction des deux derniers vecteurs de Lanczos, *ie.*,

$$\{r_j\} = [D] \{q_j\} - \alpha_j \{q_j\} - \beta_j \{q_{j-1}\} \quad (A3-6)$$

Enfin le nouveau vecteur de Lanczos est obtenu par :

$$\{q_{j+1}\} = \frac{\{r_j\}}{\beta_{j+1}} \quad (A3-7)$$

avec :

$$\beta_{j+1}^2 = \{r_j\}^T [M] \{r_j\}$$

Dans cette situation on doit avoir un β_{j+1} nul, mais si ce fait se produit, deux causes peuvent être à l'origine [chen] à savoir :

- $\{r_j\} = 0$: dans ce cas on a capturé un sous-espace invariant; les valeurs et les vecteurs de Ritz sont des solutions exactes du problème original.

- $\{r_j\}$ n'est pas nul : on a choisi un mauvais vecteur de départ.

A3-2 FORME TRIDIAGONALE

Après m itérations on aura calculé m vecteurs de Lanczos, m coefficients α et $m-1$ coefficients β . compte tenu de l'équation (A3-6) on a :

$$\begin{bmatrix} D_m \\ \text{nxn} \end{bmatrix} \begin{bmatrix} Q_m \\ \text{nxm} \end{bmatrix} - \begin{bmatrix} Q \\ \text{nxm} \end{bmatrix} \begin{bmatrix} T_m \\ \text{mxm} \end{bmatrix} = \begin{bmatrix} 0 & | & 0 & | & \dots & | & \{r_m\} \\ \text{nxm} \end{bmatrix} = \{r\} \{e_m\}^T \quad (A3-8)$$

où $\{e_m\}^T = \{0 \ 0 \ 0 \ \dots \ 1\}$ vecteur unitaire, et $[T]$ est la matrice tridiagonale. La condition d'orthogonalité ($[Q]^T [M] [Q] = [I]$), peut être employée pour obtenir :

$$[T_m] = [Q_m]^T [M] [K_0]^{-1} [Q_m] \quad (A3-9)$$

La forme explicite de $[T_m]$ est la suivante :

$$[T_m] = \begin{bmatrix} \alpha_1 & \beta_2 & 0 & & 0 \\ & \beta_2 & \alpha_2 & & \\ & 0 & & \ddots & \\ & & & & \beta_m \\ 0 & 0 & 0 & \beta_m & \alpha_m \end{bmatrix} \quad (A3-10)$$

On a trouvé la matrice tridiagonale, voyons maintenant comment elle peut approximer les solutions du problème de départ, pour cette fin, multiplions d'abord l'équation $[K]\{\phi\} - \lambda[M]\{\phi\}$ à gauche par $[M][K_0]^{-1}$ on obtient :

$$[Q]^T [M] [K_0]^{-1} [M] \{\phi\} = [M] \{\phi\} \quad (A3-11)$$

Appliquons la procédure de Rayleigh-Ritz au système précédent en choisissant comme base $[Q]$ formée par les vecteurs de Lanczos pendant m itérations. On arrive ainsi au problème réduit :

$$[T_m] \{s_i\} - \vartheta_i \{s_i\} = 0 \quad (A3-12)$$

où $(\vartheta_i, \{s_i\})$ représente la paire de solution de la matrice tridiagonale. La taille de $[T_m]$ augmente de un après chaque étape de Lanczos, et les solutions de $[T_m]$ convergent progressivement vers les solutions du problème original. D'après Ritz (annexe 2) :

$$\begin{cases} \lambda_i \cong \vartheta_i^{-1} (1 + \sigma) \\ \{\phi_i\} = [Q] \{s_i\} \end{cases} \quad (A3-14)$$

A3-3 PERTE ET RESTAURATION D'ORTHOGONALITE

Comme nous l'avons annoncé au départ l'algorithme de Lanczos devient instable après un certain nombre d'étapes dû au fait d'accumulation de l'erreur d'arrondi. Pour résoudre ce problème on doit évaluer le taux de perte d'orthogonalité, le comparer au seuil admissible et déduire si la réorthogonalisation est nécessaire ou pas. Deux alternatives sont possibles :

A3-3-1 *ORTHOGONALISATION PAR RAPPORT AUX VECTEURS DE RITZ* : Paige dans ces travaux sur la précision de l'algorithme de Lanczos [Pia11] a formulé l'expression donnant le taux d'orthogonalité (τ_j) entre les vecteurs de Ritz ayant convergés et les vecteurs de Lanczos.

$$\tau_j = \{y_i\}^T [M] \{q_j\} \quad (A3-15)$$

on multiplie l'équation (A3-6) à gauche par $\{y_i\}^T [M]$ on obtient :

$$\begin{aligned} \beta_{j+1} \{y_i\}^T [M] \{q_{j+1}\} &= \{y_i\}^T [M] [K_o]^{-1} [M] \{q_j\} - \\ &\quad - \alpha_j \{y_i\}^T [M] \{q_j\} - \beta_j \{y_i\}^T [M] \{q_{j-1}\} \end{aligned} \quad (A3-16)$$

où $\{y_i\}$ est un vecteur propre i.e., $[K_o]^{-1} [M] \{y_i\} = \vartheta_i \{y_i\}$. On déduit par conséquent la relation de récurrence suivante :

$$\tau_{j+1} = \frac{(\vartheta_i - \alpha_j) \tau_j - \beta_j \tau_{j-1}}{\beta_{j+1}}$$

Quand τ_{j+1} est plus grand que $\sqrt{\varepsilon}$, cela veut dire qu'une perte d'orthogonalité est produite, on reorthogonalise par Gram-schmith.

ORTHOGONALISATION PAR RAPPORT AUX VECTEURS DE LANCZOS : Cette fois on mesure le taux de perte d'orthogonalité par la formule (V-11) du chapitre V.

ORGANIGRAMME

Etape No 1

Donner un vecteur de départ $\{r\}$

Calculer $\{p\} = [M]\{r\}$

Résoudre pour q : $[K]\{q\} = \{p\}$

$\beta_1 = \text{Sqrt}(\text{Abs}\{q\}^T\{p\})$

$\{p\} = \{p\} / \beta_1$

$\{q\} = \{q\} / \beta_1$

Résoudre pour r : $[K]\{r\} = \{p\}$

$\alpha_1 = \{r\}^T\{p\}$

$\{r\} = \{r\} - \alpha_1\{q\}$

$\{\text{oldp}\} = [M]\{r\}$

$\beta_2 = \text{sqrt}(\text{abs}(\{r\}^T\{\text{oldp}\}))$

stocker $\{q\}$ comme $\{q_1\}$

Etape 2 $j = 2, 3, \dots$

$\{\text{oldq}\} = \{q\}$

$\{\text{oldp}\} = \{p\}$

$\{q\} = \{r\} / \beta_j$

Résoudre $[K]\{r\} = \{p\}$

$\alpha_j = \{r\}^T\{p\}$

$\beta_{j-1} = \{r\}^T\{\text{oldp}\}$

$\{r\} = \{r\} - \alpha_j\{q\}$

$\{r\} = \{r\} - \beta_{j-1}\{\text{oldq}\}$

$\{\text{oldp}\} = [M]\{r\}$

$\beta_{j+1} = \text{sqrt}(\text{abs}(\{r\}^T\{\text{oldp}\}))$

Stocker $\{q\}$ comme $\{q_j\}$

REMARQUE : Cet Algorithme génère seulement les vecteurs de Lanczos, et calcule les éléments de la matrice réduite. On l'appelle Algorithme Simple de Lanczos.

BIBLIOGRAPHIE

- [Hug 1] T. J. R. Huges, "The Finite Element Method - Linear Static and Dynamic Finite Element Analysis ", Prentice-Hall
- [Zei 2] O. C. Zeinkiewicz, "The Finite Element Method ", Fourth Edn. McGraw-Hill, London (1989)
- [Bat 3] K. J. Bathe, " Finite Element Procedures In Engineering Analysis ", Prentice-Hall, (1982)
- [Kar 4] H. Kardestuncer and H. Nourrie, "Finite Element Handbook", McGraw-Hill, (1987)
- [Dha 5] G. Dhatt, G. Touzot " Une Presentation de la Methode des Elements Finis ", Deuxième Edn., Maloine (1984)
- [Gib 6] R. J. Gibert, " Vibration des Structures Interaction avec les Fluides ", Ecole d'été d'Analyse Numérique, C.E.A.-E.D.F. INREA (1988)
- [Che 7] H. C. Chen and R. L. Taylor " Solution of Eigenproblems for Damped Structural Systems By The LANCZOS Algorithm", Computer & Structures Vol. 30 No. 1/2, PP. 151-161, (1988)
- [Nou 8] B. Nour-Omid, B. N. Parlett, and R. L. Taylor, "Lanczos Versus Subspace Iteration for Solution of Eigenvalue problems", Int.J.Num.Meth.Engng., Vol 19, PP. 859-871 (1983)
- [Bat 9] K. J. Bathe and L. W. Edward, " Solution Methods for Eigenvalue Problems in Structural Mechanics ", Int. J. Num. Meth. Engng., Vol 6, PP. 213-225 (1973)
- [Jia10] Xu Jiaxian, " An Improved Method for Partial Eigensolution of Large Structures ", Computer &

Structures Vol. 32 No. 5, PP. 1055-1060, (1989)

- [Pai11] C. C. Paige, "Accuracy and Effectiveness of Lanczos Algorithm for Symmetric Eigenproblems", Linear Algebra and its Applications, 34, PP. 235-258, (1980)
- [Par12] B. N. Parlett and B. Nour-Omid, "The use of Refined Bounds When Updating Eigenvalues of Tridiagonals", Linear Algebra and its Applications, 68, PP. 179-219 (1985)
- [Pra13] B. G. Prakash and M. S. S. Prabhu, "Reduction Techniques in Dynamic Substructures for Large Problems", Computers & Structures Vol. 22, No 4, PP. 539-552, (1986)
- [Xia14] Xian Lu, "Simplified Dynamic Condensation in Multi-Substructure Systems", Vol. 30, No 4, PP. 851-854, (1988)
- [Bat15] K. J. Bathe and E. L. Wilson "Eigensolution of Large Structural Systems With Small Bandwidth", ASCE, J. Engng. Mech. Div., Vol. 99, PP. 213-226, (1973)
- [Clo16] R. W. Clough, "Dynamics of Structures", McGraw-Hill (1975)
- [Smi17] I. M. Smith and D. V. Griffiths, "Programming the Finite Element Method", Second Edn. JOHN WILEY & SONS (1988)
- [Ral18] A. Ralston and P. Rabinowitz, "A First Course in Numerical Analysis", McGraw-Hill (1985)