

وزارة التعليم العالي
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT : G. Mecanique.

1er

PROJET DE FIN D'ETUDES

SUJET
Editeur Graphique
avec
Maillage Automatique
pour
la Méthode des Eléments Finis.

Proposé par :
TAZI Med.

Etudié par :
DOUGDAG M.

Dirigé par :
TAZI Med

PROMOTION : 1989/90

الجمهورية الجزائرية الديمقراطية الشعبية
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم العالي
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT : G. Mécanique

PROJET DE FIN D'ETUDES

SUJET
Editeur Graphique
avec
Maillage Automatique
pour
la Méthode des Eléments Finis.

Proposé par :
TAZI Med

Etudié par :
DOUGDAG M.

Dirigé par :
TAZI Med

PROMOTION : 1989/90

Dedicace

A mes defuns parents

A tous ceux que j'aime

A ma petite soeur Zora

Mourad Le 26/6/1990

Remerciements

Je tiens à remercier , Monsieur TAZI, pour ses encouragements et ses orientations qui m'ont été d'une très grande aide.

Je remercie aussi l'Association Scientifique "EL-KHAWARIZMI" de Médéa qui a mis à mon entière disposition tous ses moyens matériels. Que tous les membres de cette association trouvent ici l'expression de ma très grande gratitude.

SOMMAIRE

0	INTRODUCTION-----	1
I	LA METHODE DES ELEMENTS FINIS-----	3
I.1.	INTRODUCTION-----	3
I.1.1.	HISTORIQUE-----	3
I.1.2.	ACTUEL-----	3
I.2.	PRINCIPE DE LA METHODE-----	4
I.2.1.	IDEALISATION-----	4
I.2.2.	DISCRETISATION-----	4
I.2.3.	CONDITIONS AUX LIMITES-----	4
I.2.4.	SOLLISITATIONS-----	5
I.2.5.	FORMULATIONS-----	5
I.2.5.1.	ANALYSE STATIQUE-----	5
I.2.5.2.	ANALYSE DYNAMIQUE-----	9
I.3.	DIFFERENTSTYPES D'ELEMENTS-----	11
II.	LES MOYENS ET METHODES DE REPRESENTATION GRAPHIQUE-----	12
II.1.	INTRODUCTION CAO/DAO-----	13
II.2.	NOTIONS D'INFORMATIQUES-----	15
II.3.	L'EDITEUR GRAPHIQUE-----	18
II.4.	MODELISATION ET MAILLAGE-----	20
II.5.	GESTION DES DONNEES-----	26
III.	LA REPRESENTATION GRAPHIQUE-----	20
III.1.	REPRESENTATION DEUX DIMENSIONS (2D)-----	20
III.2.	= TROIS DIMENSIONS (3D)-----	23
III.3.	VUE 3D-----	37
III.4.	MODE DE SAISIE (La Saisie)-----	39
III.5.	TRANSFORMATIONS-----	40
III.6.	ORGANIGRAMME-----	43
III.6.	STRUCTURATION DE L'INTERFACE-----	44
	CONCLUSION-----	49
	ANNEXES	
	BIBLIOGRAPHIE	

INTRODUCTION

Pour faire face aux aléas de la nature, l'homme a toujours été en quête des meilleures méthodes qui lui permettent d'imaginer et de prévoir le comportement de ses diverses constructions.

De nos jours, plusieurs techniques concourent à la réalisation de cet objectif. L'une des techniques les plus prometteuses est celle de la méthode des éléments finis ou encore M.E.F.

Cette dernière tente de modéliser le système réel par un ensemble de modèles théoriques. Le système réel est généralement difficile à étudier, aussi il est divisé en un ensemble discrets de modèles connus et plus faciles à étudier. Cet ensemble est d'autant plus important que le système réel est complexe.

Cette technique permet de solutionner des cas complexes qu'il est pratiquement impossible de résoudre par d'autres méthodes.

Cependant, la manipulation de cet ensemble de modèles nécessite des efforts considérables lorsque le nombre de modèles devient important. Elle devient une source d'erreur difficilement détectables qui se répercutent négativement sur le rendement de l'opérateur.

Nous nous proposons d'utiliser l'outil informatique pour pallier à une partie de ces inconvénients et pour rendre la méthode plus attrayante par :

- une saisie des données plus souple
- une manipulation des données plus simple
- une représentation conviviale des structures.

Pour ce faire, nous avons réalisé un logiciel comportant :

- un éditeur graphique qui est l'outil de dialogue homme-machine
- un ensemble de bibliothèques comportant différents éléments : Elements de dessin, modélisation, affichage, etc...

Le logiciel est accompagné du présent document graphique qui se divise en trois grands chapitres.

- Chapitre I : Ce chapitre décrit la méthode des éléments finis.

- Chapitre II : Ce chapitre permet d'éclaircir certaines notions et termes de base.

- Chapitre III : Ce chapitre se rapporte au concept de la D.A.O, et les différents problèmes rencontrés au cours de ce travail, tel que :

- Passage 2D vers 3D.
- Passage 3D vers 2D.

Nous avons terminé notre travail par une conclusion qui donne:

- Les points importants.
- Les critiques (avantages/inconvénients).
- Le développement futur.
- Une comparaison avec les logiciels du marché.
- Diverses propositions .

Une bibliographie et diverses annexes complètent ce travail.

chapitre I

I. LA METHODE DES ELEMENTS FINIS [15, 16, 17, 18, 19]

I.1. INTRODUCTION

I.1.1. Historique

La période 1850 à 1875 constitue le point de départ logique de cette méthode, en effet, c'est grâce aux travaux de MAXWELL, CASTIGLIANO, MOHR et beaucoup d'autres que celle-ci a vu le jour. C'est aussi à cette époque que naquirent les premières notions d'analyse des structures. Elle n'a cependant prit forme qu'un siècle plus tard avec l'analyse par éléments finis.

C'est vers 1920 avec MANEY aux Etats-Unis et OSTENFELD au Danemark que se dégagent les idées de base d'une approche nouvelle de l'analyse des treillis et des ossatures.

I.1.2. Actuel

Aujourd'hui, cette méthode a prit des dimensions nouvelles et s'est affirmée comme outil indispensable et efficace aux différents problèmes modernes notamment avec l'apparition de l'outil informatique.

Son utilisation s'est élargie pour toucher à presque tous les domaines comme on le verra plus loin.

Les principaux avantages apportés par l'informatisation de cette méthode sont :

- la visualisation
- la rapidité de calcul
- la possibilité d'utiliser beaucoup d'éléments
- la possibilité de varier avec souplesse une multitude de paramètres, ce que ne permet pas les méthodes classiques de la Résistance des Matériaux.

Cette méthode utilise une multitude de type d'éléments qui seront définis plus tard. Une première approche sera établie dans le paragraphe suivant.

La méthode des éléments finis ne s'intéresse pas seulement aux calcul des structures, mais touche à la majeure partie des disciplines, telles que transfert de chaleur, calcul d'élasticité, mécanique des fluides, et même certains cas particuliers comme la mesure des précipitations pluviométriques dans une région donnée.

I.2. PRINCIPE DE LA METHODE

I.2.1. Idéalisisation

Afin de mieux définir l'esprit commun aux nombreuses approches possibles de la formulation des équations d'éléments finis, nous allons comparer au système réel un modèle mathématique simple de comportement d'un élément. Ce modèle constituera une caractéristique acceptable d'un élément, mais il peut en exister d'autres, de formes différentes et qui constitueront autant de manières valables d'envisager les notions sur lesquelles se fonde la méthode des éléments finis. En fait, disons dès à présent que l'approche la plus généralement empruntée n'est pas unique. Il en existe plusieurs qui sont soit variationnelles soit énergétiques.

La structure réelle subit des contraintes réparties sur l'ensemble de ses différentes régions. Un modèle mathématique consistera à représenter l'état de contraintes dans les éléments finis correspondants par des efforts -dits forces généralisées- qui seront supposées localisées aux noeuds. Les déplacements de ces mêmes noeuds, suivant les degrés de liberté, suffiront à caractériser l'état déplacé de tous les éléments.

La modélisation d'un système repose généralement sur le choix de l'élément ou groupe d'éléments parmi d'autres, ceci se fait sous l'influence de plusieurs paramètres. On citera quelques uns:

La nature des sollicitations appliquées sur le système réel, un système soumis à des flexions ne sera donc modélisé avec une barre, etc...

L'approche de la forme géométrique des systèmes réels étudiés.

beaucoup d'autres paramètres peuvent intervenir tel que l'économie de la taille mémoire, etc...

I.2.2 DISCRETISATION

Un système réel très difficilement représentable par un seul élément. Aussi, on tend généralement à le discretiser en plusieurs éléments. L'approche de ce système est réalisée par une des méthodes utilisées qu'est le maillage automatique (voir paragraphe maillage chap II).

I.2.3. CONDITIONS AUX LIMITES

Toute structure est régie par des déplacements imposés ou des restrictions, ce qu'on appelle les conditions aux limites. Elles se présentent généralement sous la forme d'appuis sur des supports de différentes natures.

I.2.4. SOLLICITATIONS

Une structure est généralement soumise à une multitude de sollicitations tant statiques que dynamiques. Ces dernières ne sont malheureusement pas simples à représenter en tant qu'efforts concentrés.

Les efforts statiques n'ont ni une valeur fixe, ni un point d'application bien déterminé; alors que les efforts dynamiques sont généralement régis par des variations aléatoires au cours du temps.

on citera les types suivant d'efforts :

- Les tractions
- Les compressions
- Les cisaillements
- Les glissements
- Les torsions
- Les flexions
- etc...

I.2.5. FORMULATIONS

II.2.5.1 Analyse statique :

L'étude du cas statique en éléments finis , se repose sur :
les déplacements et les sollicitations :

- $\{ U \}$: vecteur déplacement
- $\{ F \}$: vecteur de déplacement
- $[K]$: matrice de rigidité

la relation dans le repère Locale:
 $\{ F_i \} = [K] \{ U_i \}$ (1.1)

Le passage au repère Global :

Le passage des coordonnées locales aux coordonnées globales se fait par trois transformations

Rotation autour de OZ

$$R1(o , \alpha)$$

Rotation autour de OY

$$R2(o , \beta)$$

Rotation autour de OX

$$R3(o , \Gamma)$$

où R_i de centre (o) et d'angle α
on exprime matriciellement ces rotations :

$$R1(o , \alpha) = \begin{vmatrix} \cos(\alpha) & \sin(\alpha) & 0 \\ -\sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

$$R2(\alpha, \beta) = \begin{vmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{vmatrix}$$

$$R3(\alpha, \Gamma) = \begin{vmatrix} 0 & \cos(\Gamma) & \sin(\Gamma) \\ 0 & -\sin(\Gamma) & \cos(\Gamma) \\ 1 & 0 & 0 \end{vmatrix}$$

soit donc la transformation :

$$T = R1(\alpha, \alpha) R2(\alpha, \beta) R3(\alpha, \Gamma) \quad (1-2)$$

L'équation du comportement statique d'un élément étant :

$$[k] \{U\} = \{F\}$$

Dans le repère globale, elle prendra la forme :
le vecteur force devient :

$$\{F_g\} = [T] \{F\} \quad (1-3)$$

le vecteur déplacement devient :

$$\{U_g\} = [T] \{U\} \quad (1-4)$$

on utilisant (1-3) et (1-4) on aura :

$$\begin{aligned} [T]^{-1} \cdot \{F_g\} &= [k] \cdot [T]^{-1} \cdot \{U_g\} \\ \text{et} \\ \{F_g\} &= [T] \cdot [k] \cdot [T]^{-1} \cdot \{U_g\} \end{aligned}$$

Ainsi on définit dans le repère globale la matrice élémentaire de rigidité $[K_g]$:

$$[K_g] = [T] \cdot [k] \cdot [T]^{-1} \quad (1-5)$$

La matrice d'assemblage :

Etant donné qu'une structure est un ensemble d'éléments reliés par des noeuds, donc en interaction, ceux-ci communiquent entre eux, efforts et déplacements. Ceci est concrétisé par la matrice d'assemblage, qui est la résultante de toutes les matrices de rigidité globale des éléments de la structure.

cette opération se fait généralement par deux méthodes :

1. La méthode d'expansion des matrices de rigidité, cette méthode, est très gourmande en quantité de mémoire.

2. La méthode directe : cette méthode se base sur le principe de créer un petit sous-programme, qui aura pour rôle de placer chaque élément de la matrice de rigidité d'un élément à sa place dans la matrice d'assemblage.

Comme on sait, la matrice d'assemblage est une matrice symétrique tel que :

$$[K_a] = [K_a]^T$$

$$K_{ij} = K_{ji}$$

Le stockage pur et simple de la matrice ne sera qu'une perte de la mémoire. Le programme, par une allocation dynamique de la mémoire, fera allouer l'espace nécessaire pour le stockage la partie triangulaire supérieure de la matrice.

Généralement la matrice d'assemblage contient un nombre non estimable de valeurs nulles, qu'il serait non économique de les stocker, on les éliminera grâce à la méthode de structuration que nous avons expliqué au paragraphe (gestion des données).

7. Les déformations :

Ce sont les variations relative de l'élément par rapport à son état initial, on exprimera cela par :

$$\epsilon = \frac{\Delta L}{L} \quad \text{ayant pour composantes :}$$

$$\epsilon_x = \frac{\delta U}{\delta x} \quad \text{selon OX}$$

$$\epsilon_y = \frac{\delta U}{\delta y} \quad \text{selon OY}$$

$$\epsilon_z = \frac{\delta U}{\delta z} \quad \text{selon OZ}$$

Les contraintes :

$$\text{avec : } \{ U_g \} = [T]^{-1} \{ U \}$$

$$\text{donc : } \{ U \} = [T] \{ U_g \}$$

$$\sigma_x = E \cdot \epsilon_x$$

$$\sigma_y = E \cdot \epsilon_y$$

$$\sigma z = E \cdot \epsilon z$$

II.2.5.1.2 Les méthodes de résolutions des systèmes $A.X = B.$

III.5.1 La méthode de CHOLESKEY :

elle se base sur les opérations suivantes:
 . décomposer $[K]$ en deux matrices triangulaires, l'une supérieure et l'autre inférieure tel que :

$$[K] = [L] \cdot [R]$$

$$\text{avec } [R] = [L]^T$$

. la matrice $^{\circ}R$ a pour composantes :

$$r_{ij} = 0 \quad (i > j)$$

$$r_{ij} = {}^{\circ}K_{ii} - \sum_1^{i-1} (r_{ij})^2 \quad (i > 1)$$

$$r_{11} = \sqrt{K_{11}}$$

$$r_{1j} = K_{1j} / r_{11}$$

$$r_{ij} = \frac{1}{r_{11}} \cdot K_{ij} - \sum_1^{i-1} r_{1j} \cdot r_{1i} \quad (i > 1 \text{ et } j > i)$$

. le système devient :

$$\text{posons } \begin{cases} [L] \cdot [R] \cdot [U] = [F] \\ [R] \cdot [U] = [Y] \end{cases}$$

. on obtient ainsi un système :

$$\begin{cases} [L] \cdot [Y] = [F] \\ [R] \cdot [U] = [Y] \end{cases}$$

. la résolution est très simple, elle se fait en deux étapes : on détermine $[Y]$ puis on détermine $[U]$.

III.5.2 La méthode de Gauss Jordan :

La méthode de gauss_jordon consiste à triangulariser la matrice par itérations, tel que :
ayant $[A] \cdot \{x\} = \{b\}$

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - a_{ik}^{(k)} \cdot a_{kj}^{(k)} / a_{kk}^{(k)}$$

$$b_j^{(k+1)} = b_j^{(k)} - a_{kj}^{(k)} \cdot b_k^{(k)} / a_{kk}^{(k)}$$

k : l'ordre d'itération

II.2.5.2 Analyse Dynamique

1. Réponse libre : c'est un comportement libre de la structure, sans excitations de forces extérieures.

on exprime ce ci par :

$$[M] \cdot \ddot{\{U\}} + [K] \cdot \{U\} = 0 \quad (2.1)$$

$[M]$: matrice de masse

$\{ \ddot{U} \}$: vecteur accélération

$[K]$: matrice de rigidité

$$\{U\} = -w^2 \{U\} \quad (2.2)$$

w : pulsation propre du système ; $w = 2 \pi N$

N : fréquence propre,

(2.1) devient :

$$([K] - w^2 [M]) = 0 \quad (2.3)$$

les solutions de cette équation représentent les valeurs propres et vecteurs du système. On définit alors les modes propres du système, qui sont des importantes informations ; surtout pour les ponts et bâtiments, ect..

le calcul numérique approché, de cette équation se fait par :

Stodola, Raileigh, Jaccobi, etc..

2. Cas avec force éxitatrice

si notre structure est soumise à une force éxitatrice

(2.1) devient :

$$(2.4) \quad [M] \cdot \ddot{\{U(t)\}} + [K] \{U(t)\} = \{F(t)\}$$

$\{U\}$: déplacement dependant du temps

$\{F(t)\}$: est le vecteur force éxitatrice
pour résoudre ces équations difficiles, on se limite généralement à

quelques fréquences:

on procède :

- on fixe le nombre de modes à étudier à (m)
- appliquant une transformation de simplification de calcul :

on pose : $\{U(t)\} = [\phi] \cdot \{q(t)\}$
la matrice $[\phi]$ est de l'ordre nxm

ou n : ordre de la matrice $[k]$

$\{q(t)\}$: vecteur de m composantes.

$$\text{calculant: } \ddot{\{U(t)\}} = [\phi] \cdot \ddot{\{q(t)\}}$$

- appliquant le changement de coordonnées au système de coordonnées NODALE.

$$[M] \cdot [\phi] \cdot \ddot{\{q(t)\}} + [K] \cdot [\phi] \{q(t)\} = \{F(t)\}$$

En faisant plusieurs transformations de simplification on obtient :

$$[\bar{K}] = \omega^2 [\bar{M}] \quad (2.5)$$

$[\bar{M}]$: matrice de masse Modale

$[\bar{K}]$: matrice de rigidité modale.

I.2.3. DIFFERENTS TYPES D'ELEMENTS

L'élément barre simple : (fig a)

c'est un élément composé de deux noeuds, d'un nombre de degrés de liberté dépendant du système de repère. C'est un élément non flexible qui sert à décrire des poutres en treillis et des ossatures à deux ou trois dimensions. Il est utilisé seul ou combiné avec des éléments d'un autre type.

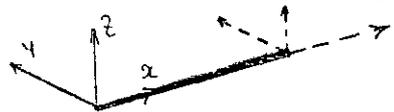
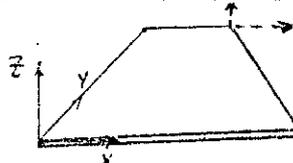


Fig (a) - Barre



Fig (b) - Plaque mince



L'élément plaque mince : (fig b)

c'est élément à quatre noeuds s'il est quadrilatère et trois noeuds s'il est triangulaire. Chacun de ses noeuds possède un nombre de degrés de liberté dépendant du type de repère. C'est un élément non flexible. Par rapport aux autres types d'éléments, ce type d'élément a une grande utilité dans un grand nombre de cas pratiques. Il est considéré comme l'élément de base de l'analyse par éléments finis. C'est un élément subissant des contraintes planes. La forme de ce type d'élément est généralement triangulaire ou quadrilatère bien qu'il existe d'autres formes géométriques qui sont surtout destinées pour des applications spécifiques.

L'élément volumique : (fig c)

C'est un élément possédant huit noeuds, au plus. Il constitue une généralisation tridimensionnelle des éléments de contraintes planes. Le tétraèdre et le parallélépipède en sont les formes des plus courantes. Cet élément est essentiel dans la modélisation des structures destinées à la production d'énergie nucléaire, ainsi que dans les problèmes de mécanique de sols et des roches.

L'élément volumique de révolution : (fig d)

Ce type d'élément a une large utilisation dans les domaines suivants :

- réservoirs en béton et en acier,
- réacteurs nucléaires,
- rotors,
- pistons,
- arbres,
- tuyères de soufflerie ou de fusée,
- etc...

Le répartition des contraintes est elle-même très souvent révolutionnaire. On peut aussi utiliser les éléments de la 2ème classe (plaques minces), pour résoudre ces problèmes.

L'élément plaque mince de flexion : (fig e)

C'est un élément de trois à quatre noeuds ayant, en plus, des degrés de liberté de l'élément plaque mince (fig b) et des degrés de libertés par flexion. Ce type d'élément est utilisé non seulement pour rendre compte du comportement des plaques planes, mais aussi pour obtenir des représentations de coques (Gauches) par facettes et pour constituer des poutres à parois minces. La géométrie de ce type d'élément se conforme à celle des éléments de contraintes planes, et en particulier les formes triangulaires et quadrilatères.

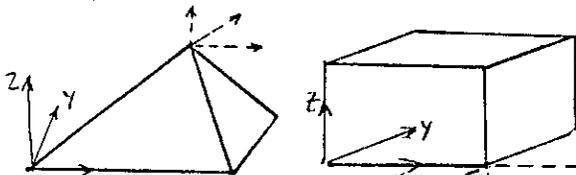


Fig (c) - élément volumique

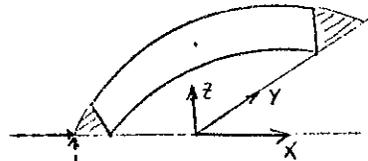


Fig (d) - Corps de révolution

L'élément coque de révolutions : (fig f)

C'est un élément disposé, en plus de l'élément (fig d), des degrés de liberté par flexion. Cet élément jouit de la même importance pratique que les coques volumiques de révolution.

Les équations proviennent des hypothèses simplificatrices de la théorie des coques minces. Les éléments de ce type sont situés à mi-chemin entre les éléments plaques planes en flexion et en tension et les éléments coques minces généraux. Ils sont très utiles dans l'étude de certains problèmes clés rencontrés avec ces derniers.

L'élément coque mince de forme générale : (fig g)

C'est un élément ressemblant aux éléments (fig e), mais ayant une forme courvéligne. Il est utilisé généralement pour ses possibilités de décrire la géométrie de la représentation de la surface réelle avec une meilleure précision, et une représentation adaptée du couplage entre tension et flexion au sein de chaque élément.

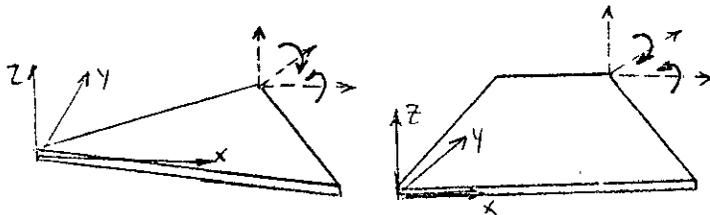


Fig (e) - Plaques minces flexibles

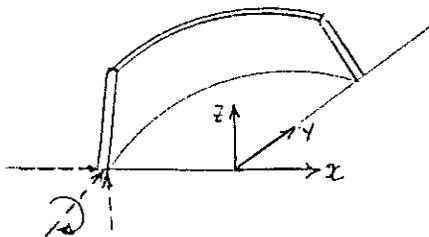


Fig (f) - Coque mince de révolution

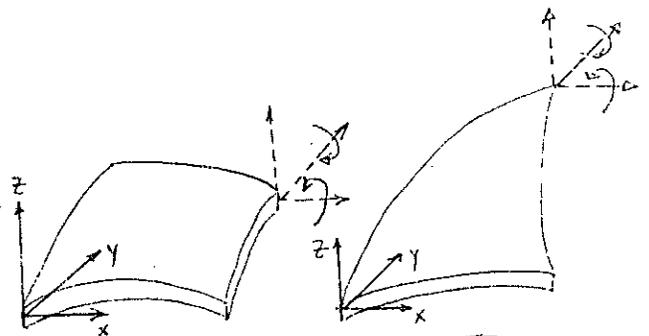


Fig (g) - Coques minces gauches

chapitre II

II.1. Introduction CAO/DAD:

En moins d'un demi siècle, les ordinateurs sont devenus des instruments indispensables dans l'industrie. Les futurs ordinateurs seront au moins dix (10) fois plus performants et fourniront une aide intellectuelle dans tous les domaines. Quand il apparut, l'ordinateur n'était qu'une curiosité de laboratoire, c'est aujourd'hui un élément vital de notre société. Les activités financières, industrielles et de transport ne pourraient plus fonctionner sans les traitements informatiques. Aujourd'hui l'ordinateur participe à la construction des matériaux, et les produits pharmaceutiques et aux recherches de physique fondamentale sur la nature de l'énergie et de la matière. L'activité économique et intellectuelle connaît grâce à cet instrument irremplaçable, un développement sans précédent. Ce sont là les moments les plus exaltants de l'histoire des sciences et des techniques.

1.1 L'industrie et les Ordinateurs :

En fait, le passage entre l'analyse théorique sur ordinateur et l'utilisation d'outils de conception assistée par ordinateur (C.A.O) et fabrication assistée par ordinateur (F.A.O) n'est pas aisé. Il ya 20 ans, les théoriciens de la dynamique des fluides utilisaient déjà les ordinateurs pour modéliser l'écoulement de l'air. On cite encore la modélisation des structures, les conséquences des contraintes et les tremblements de terre sur les ponts, etc... Les ordinateurs modernes ont joué un rôle essentiel dans l'industrie grâce à leurs grandes et diverses possibilités que l'expérience pratique ne pouvait montrer. De manière générale, les ordinateurs modernes s'imposent quand on cherche à résoudre un problème où l'on utilise des modèles mathématiques élaborés et susceptibles d'apporter des solutions nouvelles, tant pour concevoir un produit que pour améliorer son rendement.

1.2 Introduction à la D.A.O :

A la base de tout projet d'architecture ou de dessin industriel se trouve une idée que le projeteur rend compréhensible en plusieurs étapes. Des outils peuvent l'aider à trois niveaux différents : pour l'extraction de l'idée, sa projection sur une série de supports intermédiaires et pour son exécution. L'ordinateur est particulièrement adapté pour servir d'outils au niveau de la projection, cet outil travaille en 2D (un plan), en 2.5D (un plan de référence sur lequel est ajoutée artificiellement une troisième dimension) et en 3D (travail direct en volume).

L'ordinateur est d'autant plus performant qu'il dispose de deux éléments de base :

1- Un environnement matériel adéquat, soit une unité centrale puissante, un co-processeur arithmétique (si c'est possible), une souris ou un digitaliseur, un écran haute définition et une imprimante (ou une table traçante).

2- Une exploitation rationnelle de ces possibilités, autrement dit un logiciel performant et répondant aux critères de :

- souplesse de manipulation.
- convivialité de l'interface homme-machine.
- richesse de la bibliothèque graphique.
- richesse des utilitaires de base : Zooming, rotation, symétrie, translation, ajout/suppression, fusion, désignation, élimination des faces cachées, éclairage des facettes ...etc

- utilitaire de gestion de fichier permettant la création des fichiers de base de données informatique compatible avec l'existant, la possibilité d'exploiter des fichiers externes, etc...

L'apogée de cette analyse est représenté actuellement par les stations de travail inventées à la fin des années 70 par Bill PODUSKA, cofondateur de PRIME et créateur en 1980 d'Appollo computer, société actuellement leader du marché. L'idée de Bill PODUSKA fût de concevoir une machine ou mieux un système globale, qui rassemblerai les aspects positifs des deux grandes tendances : Réunir à la fois la puissance, la capacité de partage des ressources, l'interactivité, une interface homme-machine agréable et d'excellents temps de réponse. Ces exigences caractéristiques répondaient aux exigences de l'informatique graphique, ce concept de station de travail repose sur trois éléments primordiaux dont l'intégration doit être la plus complète possible:

Une unité centrale puissante pour chaque utilisateur, le graphisme.

un réseau local performant.

Autrement dit, pour que l'on puisse parler de station de travail, il faut non seulement réunir ces trois éléments, mais aussi assurer leurs intégrations totale au niveau du système d'exploitation.

II.2. NOTIONS D'INFORMATIQUE [1,11,13,14,7,2]

II.2.1 L'ordinateur :

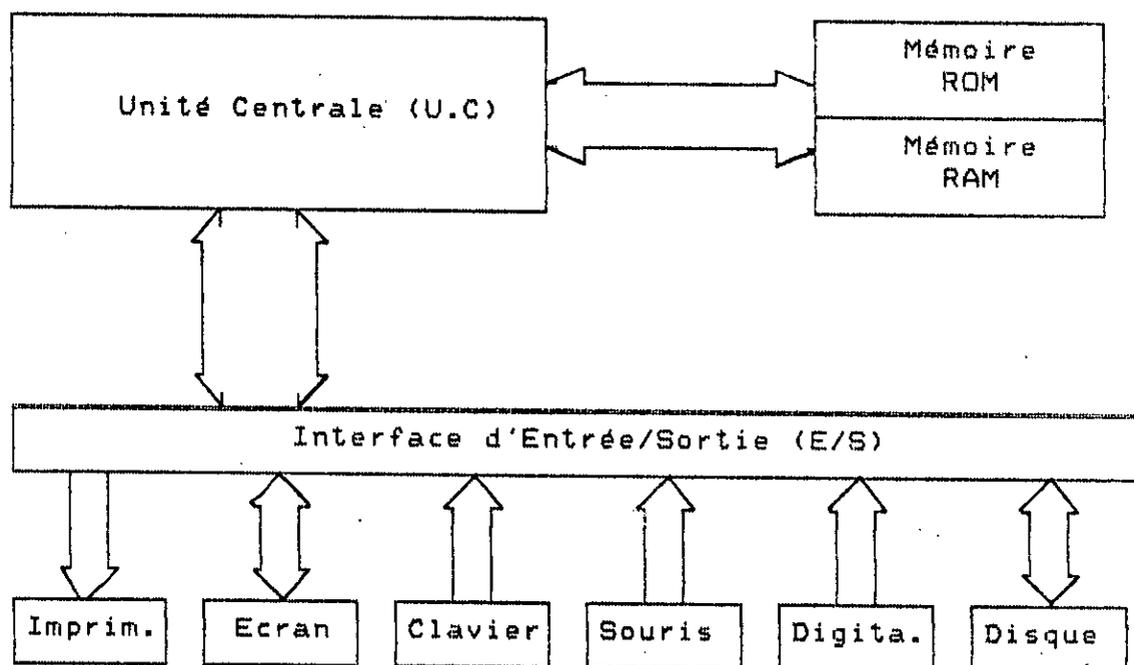


fig.1 Schéma générale d'un ordinateur.

Un ordinateur est composé alors par une unité centrale, une mémoire morte (ROM) et une mémoire vive (RAM), ainsi qu'une interface entrée/sortie ou le système de dialogue homme-machine.

1- L'unité Centrale :

C'est le cerveau de l'ordinateur, ses tâches principales sont le calcul (il est doté d'une unité arithmétique et logique), et la gestion des différents périphériques.

2- La mémoire :

Généralement, la mémoire de l'ordinateur est partagée en diverses zones (Cette nomenclature est celle d'un IBM PC):

- une est occupée par le BIOS (Ensemble de routines de base du système) et la partie permanente du système d'exploitation (DOS).
- Une s'occupe de la mémoire vidéo.
- le reste se partage entre :
 - la partie transitoire du DOS.

- Le code des programmes exécutables.
- La pile.
- et le tas (accessible par les fonctions d'allocations dynamiques de la mémoire).

512 Ko RAM Système	000000h 000500h	} réservé BIOS
BANK 0	07FFFFh	
128 Ko Extension RAM système BANK 0 ou BANK 1	080000h 09FFFFh	
128 Ko / 256 Ko RAM Vidéo	0A0000h 0BFFFFh	
32 Ko Extension BIOS	0C0000h	
RDM Vidéo VGA	0C6000h 0C7FFFh	- non accessible.
Libre extension BIOS carte extension	0C8000h 0DFFFFh	
128 Ko BIOS ROM carte mère	0E0000h 0FFFFFFh	

fig.2 Map de la RAM d'un IBM PC.

La mémoire est subdivisée en blocs mémoires dont chacun est associé à une adresse et de capacité 16 octets.

3- l'interface Entrée/Sortie :

C'est le dispositif matériel complété par le logiciel (sous forme de handler système) qui permet d'étendre les possibilités actuelles d'une gamme d'ordinateurs. C'est donc la plate forme qui vient s'intercaler entre la carte micro-processeur et le monde externe des périphériques.

On compte parmi ces derniers :

- Le clavier : C'est la première source d'introduire des données, celles-ci peut être sous forme de source d'un programme ou sous la forme de données nécessaires pour le déroulement du programme.

- La souris : Elle s'est imposé grâce à la souplesse de manipulation et depuis l'avènement du MacIntosh d'APPLE, Elle a l'apparence d'un petit boîtier qui en se déplaçant sur une surface, fera mouvoir un pointeur (généralement, sous la

forme d'une flèche) sur la surface de l'écran, le choix d'une sélection se fera à lui, en cliquant (appuyant sur une touche de la souris) sur l'objet sélectionné.

- Le digitaliseur (table à digitaliser) : C'est un outil de saisie de donnée plus précis, c'est l'outil privilégié des modeleurs et concepteurs.

- L'imprimante : C'est l'organe de sortie sur papier par excellence, on garde ainsi trace des données, des courbes et de listings, elle est néanmoins moins précise devant une table traçante pour les projets de conception, ou la précision est un facteur déterminant.

- Le Moniteur : C'est l'organe compagnon de la console, il permet de voir l'évolution du dessin, et l'affichage temporaire des données (ou résultats). Il est composé d'un ensemble de points adressables individuellement (On parle de Bit-map), ils peuvent ainsi s'allumer ou s'éteindre ou prendre n'importe quelles couleurs.

L'une des caractéristiques principales d'un moniteur est sa résolution, c'est à dire le nombre de points pouvant être adressés, qui sont en relation très étroite avec la résolution qui est d'autant plus meilleur que le nombre de points est grand.

Ces points sont gérés par des cartes vidéos de plus en plus sophistiquées .

Pour l'exemple, citons quelques exemples (Bien sur pour les types de moniteurs gérables par un IBM PC ou compatible) :

* La carte CGA (Color Graphic Array) : d'une résolution maximale de 640 x 200 en 2 couleurs seulement.

* La carte EGA (Enhanced Graphic Array) : plus récente que la CGA, elle offre une résolution maximale de 640 x 350 en 16 couleurs.

* La carte VGA (Vidéo Graphic Array) : C'est la plus récente des cartes vidéos IBM, elle offre une résolution proche des téléviseurs, d'où sa plus forte recommandation pour les travaux de C.A.O, D.A.O ... etc

Sa résolution atteint 640 x 480 en 16 couleurs. A noter, que des constructeurs indépendants offrent des cartes vidéos compatible IBM mais de meilleure résolution, citons parmi elles la carte VEGAVGA qui offre une résolution de 800 x 600 avec 256 couleurs choisies parmi une palette de 16 millions de couleurs.

III.2.2 Le fichier :

C'est un dispositif mis à la portée de l'utilisateur, il lui sert pour le stockage de ses données, ainsi que leurs réstitutions pour les traitements ultérieures.

II.3.L'EDITEUR GRAPHIQUE

II.3.1 Généralités :

L'éditeur graphique est un environnement interactif d'affichage, de sélection de tâches et de description de structures en fournissant toutes les données nécessaires. Ce type d'éditeur découle de plusieurs travaux scientifiques, de l'art de communication ou encore la science de la communication par synthèse d'image. De manière générale, c'est un outil de dialogue.

II.3.2 L'outil de dialogue :

Le dialogue est l'échange d'informations entre l'homme (opérateur) et la machine (ordinateur), le type le plus courant d'outils de dialogue est le langage de programmation (BASIC, PASCAL ...)

Pour notre cas, l'éditeur expose des messages sous forme de texte ou graphique, ces informations ne sont donc qu'un reflet de l'état de la machine en un moment donné, donc destinée à représenter à l'opérateur une situation qu'il doit analyser et interpréter.

En retour, l'opérateur donne des commandes (généralement suivi de données) à l'ordinateur en agissant sur :

- le système d'aide d'une opération de choix d'une action parmi plusieurs : on parle généralement d'utilisation de MENUS.

- Les données du système, en lui passant des données: soit graphiques, alpha-numériques ou par désignation d'objets sur leurs représentations graphiques et parfois par le choix d'une action qu'elle agira sur ces données.

Il existe plusieurs techniques élémentaires, on cite :

II.3.2.1 Le photostyle :

a/ Menu : C'est un ensemble de commandes disponible sur l'écran de l'ordinateur (voir ANNEXES), l'opérateur désigne la commande désirée en appuyant sur les touches haut/bas/gauche/droite ou par le déplacement de la souris, un témoin alors apparaît et suit les différentes directives de l'opérateur, le témoin est généralement une flèche ou une croix, d'autres formes sont envisageables, les commandes sélectionnées se démarquent par inversion vidéo.

On associe à la commande un sous-programme correspondant.

b/ L'introduction de données se fait alors sous deux aspects possibles :

- Sélection des données déjà existantes dans le programme, par exemple le type de modélisation.

- Introduction des données par le clavier alphanumérique sous un format (environnement) très souple et très limité, généralement il ne dépasse quelques données. exemple nom et valeur d'une propriété (E, E = ?) ... etc

- La collecte des coordonnées : C'est la partie la plus importante de cette méthode, on parlera ici de LA SAISIE GRAPHIQUE DES COORDONNEES.

Elle consiste généralement par l'affichage d'un témoin qui suit les directives de l'opérateur, permettant ainsi de balayer toute la zone réservée au dessin, qui peut être suivi par l'affichage simultané des coordonnées selon le type du repère, l'échelle et l'origine choisis antérieurement.

Ces applications :

1- On peut généralement dessiner une forme en saisissant ces points caractéristiques, par exemple :

- un segment est défini par deux points (les extrémités).

- un cercle est défini quand à lui par deux points aussi, à savoir le centre et un point de la périphérie.

- un triangle par les coordonnées de ses trois sommets, etc ...

Pour faire ces tracés, il suffit donc d'indiquer ces points caractéristiques en positionnant le témoin aux positions correspondantes, et après validation, la forme prédefini est automatiquement tracé.

2- Il ya une autre possibilité qu'est l'élastique, cette méthode généralement utilisée pour les segments est assez significative du moment qu'elle permet à l'opérateur la vue instantanée du déroulement de la forme selon ces différents changements, il peut ainsi choisir la meilleure visualisation possible. Ce procédé consiste à tracer et effacer successivement la nouvelle et la dernière position, (voir annexe sur l'élastique).

d/ L'identification d'un élément graphique représenté sur l'écran et son équivalent sur les fichiers de donnée, s'opère très facilement par le parcours de la liste des données et voir qui correspond par simple comparaison, ainsi on peut effacer, transformer ... etc

La méthode du photostyle présente donc la meilleure possibilité pour l'introduction des données, par sa souplesse et facilité d'utilisation, et l'affichage en clair de la liste des commandes, mais en contre parti représente quelques inconvénients dont on cite :

- Une réservation d'une zone permanente de l'écran dominant ainsi le champ de saisie du dessin et cela par le menu principal.

On peut y remédier par la diminution au minimum de cette zone et cela par le groupement des commandes de la même catégorie dans des options tête de fils spécifique, et ainsi de suite, c'est ce qu'on appelle les menus déroulants.

- le masquage de certaines régions de l'écran aux moments d'appel à certaines options tête de fil.

- une certaine fatigue de la main d'opérateur à force de répétitions de désignation

II.3.2.2 Le calvier alpha-numérique :

a/ Menu : Il suffit ici de taper le nom correspondant de la commande ou la lettre représentative au clavier.

b/ l'introduction des valeurs alpha-numériques par le clavier est une opération très connue.

c/ la collecte des coordonnées s'effectue généralement par une liste de valeurs de coordonnées de points, c'est visiblement un lent et fastidieux procédé si le volume de donnée est important, c'est ce qui désavantage cette méthode par rapport à la première.

II.4. MODELISATION ET MAILLAGE

II.4.1 Modelisation (Informatique):

Le modèle représente un objet conçu ou en cours de conception . C'est une représentation "informatique" d'objets du point de vue de leurs propriétés .Il peut contenir différents types de données : géométriques , technologiques, topologiques- alors qu'en M.E.F un modèle est une représentation mathématique d'objets réels- La modélisation manipule pour ça les propriétés :

géométriques : coordonnées du point ; équations des courbes, droites, cercle, ...etc.

topologiques : un objet est définie par points caractéristiques ; un segment par deux points, un cercle par deux ...etc.

structuration : un ensemble est composé par des objets de base; rectangle par quatre segments ..etc.

habillage : côtations , hachures, couleurs et divers symboles.

relations : certains relations existent ente objets : perpendicularisme, parallélisme ,etc...

visiualisation : un modèle est généralement représente sous plusieurs façons (fil de fer, faces cachées etc...).

II.4.1.1 Les principaux modèles Informatiques :

les principaux modèles utilisés se divisent en deux grandes catégories:

le bidimensionnel qui est le modèle le plus courant et le plus simple , mais il est peu adapté aux objets complexes.

le modèle tridimensionnel (à représentation virtuelle d'objets dans ces trois dimensions) , il existe trois types de modèles tridimensionnels :

- le modèle " fil de fer "
- le modèle " surfaces "
- le modèle " solide " ou " volumique "

Le modèle " fil de fer " : est une représentation en arêtes (segments), joignant les sommets de coordonnées (x,y,z) ; quoique simple ,il présente des ambiguïtés d'interprétation.

Le modèle " surface " : permet une meilleure représentation des surfaces même complexes , mais il ne lève pas toute les ambiguïtés d'interpolation.

Le modèle " solide " : est sans doute la meilleure représentation d'objets complexes , avec la cohérence de l'information qu'il permet.

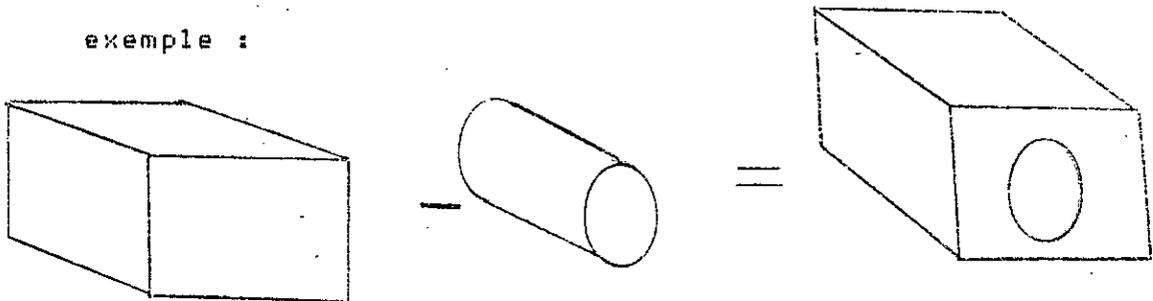
Objets de base :

Le POINT
Le SEGMENT
Le PRALLELIPIPEDE
Le CYLINDRE
 etc....

Opérations entre Objets :

UNION
INTERSECTION
DIFFERENCE
ADDITION
TRANSFORMATIONS
 etc....

exemple :



II.4.1.2 Choix du modèle :

Dans notre étude ,on s'est basé sur quatre (4) objets de base :

-
- * Le POINT
 - * Le SEGMENT
 - * Le QUADRILATAIRE
 - * L'HEXAEDRE ou PARALLELIPIPEDE

et cela contrairement à l'architecture, dont presque la totalité des Logiciels traitant ce domaine tel que : ARCHICAD et autres , ont choisi le parallélépipède comme objet de base ; pour sa grande aptitude à se transformer en plusieurs formes régulières :

Pour notre cas, et en raison de plusieurs facteurs qui sont :

- la richesse des modèles de la M.E.F .
- le critère de la taille mémoire ; car un choix unique comme le Parallélépipède aurait eu des répercussions néfastes ; du faite de la réservation d'espaces mémoire inutilement.
- commodité de manipulation .

II.4.2 Maillage automatique :

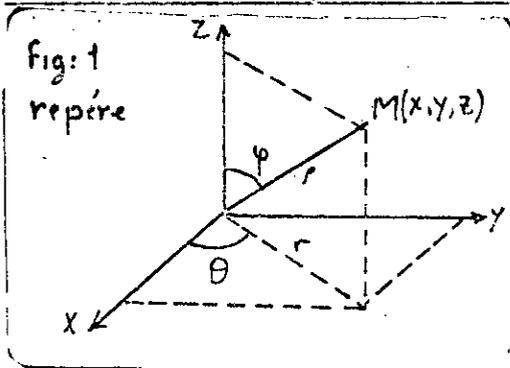
Le maillage ou la génération automatique d'éléments est l'un des atouts majeurs des techniques actuelles. Grâce à cette technique, on économise un temps précieux , comme il permet aussi à la Finition de certaines régions du maillage d'une structure , pour plus de précision dans les calculs et la représentation.

Le maillage dépend généralement de plusieurs facteurs , on citera :

- Dimensions de l'objet à mailler.
- Type de repère dont lequel sera établi le maillage .
- Directions du maillage.
- Origine du maillage (Centre générateur) .
- Nombre d'éléments à générer sur chaque direction.
- Caractéristiques géométriques de la forme à mailler.

II.4.2.1 Génération unidimensionnelle :

1. Dans le repère Cartésien :



Connaissant les deux extrémités (x_1, y_1, z_1) et (x_2, y_2, z_2) du segment, il nous est permis de définir les directions du maillage. il suffit de fournir le nombre (N) d'éléments à générer , on obtient alors les coordonnées du $(i\text{eme})$ noeud tel que

$$x_i = (i-1)(x_2-x_1)/N + x_1$$

$$y_i = (i-1)(y_2-y_1)/N + y_1$$

$$z_i = (i-1)(z_2-z_1)/N + z_1$$

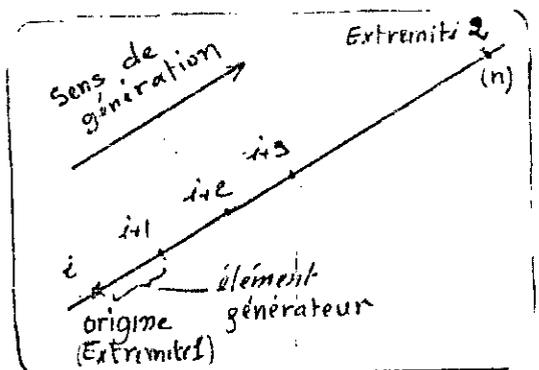


Fig.2. génération unidimensionnelle

2. Dans les repères Polaires:

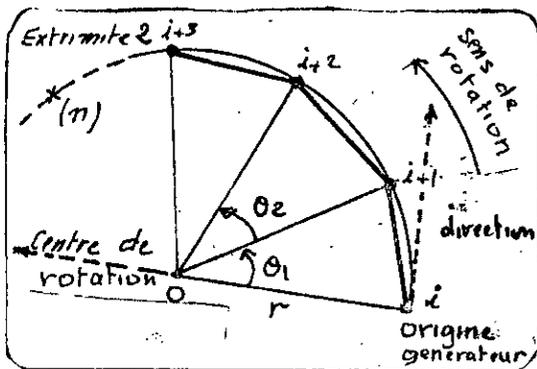


Fig:3 génération polaire

La génération ici, est une simple rotation autour d'un centre de rotation, dont l'origine générateur est (r_1, θ_1) et l'extrémité (r_2, θ_2) et (N) nombre d'éléments à générer.

Les coordonnées du $(i\text{eme})$ noeud est :

$$r_i = r_1$$

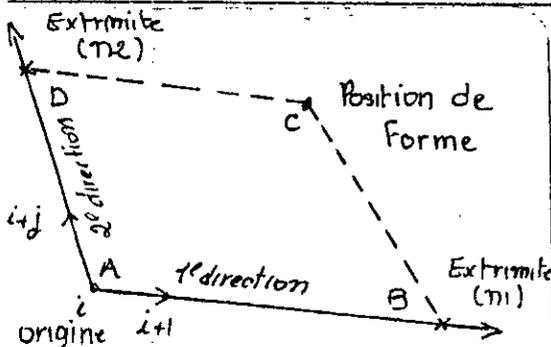
$$\theta_i = (i-1)(\theta_2 - \theta_1)/N + \theta_1$$

Pour passer au repère Cartésien, une simple transformation suffirait.

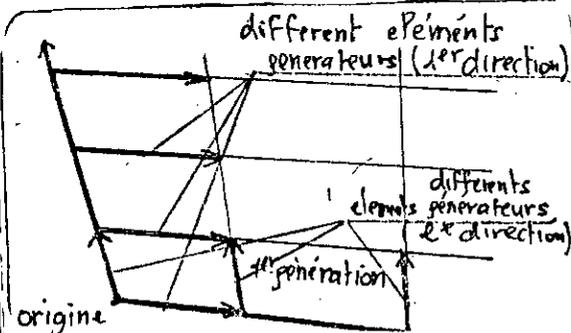
II.4.2.2 Génération bidimensionnelle :

1. Dans le repère Cartésien :

Fig:4 Principe de la génération plan



(a)



(b)

Dans ce cas il faut spécifier les deux directions du maillage, qui se définissent grâce aux quatre points caractéristiques de ce type de maillage plan (quadrilatère) :

soit A le centre générateur

soit (AB) la première direction

soit (AD) la deuxième direction

soit C la position de forme

soient respectivement N_1 et N_2 le nombre d'éléments selon la 1ere et 2eme direction.

Alors, on peut procéder au maillage du quadrilatère, on suivra les étapes suivantes :

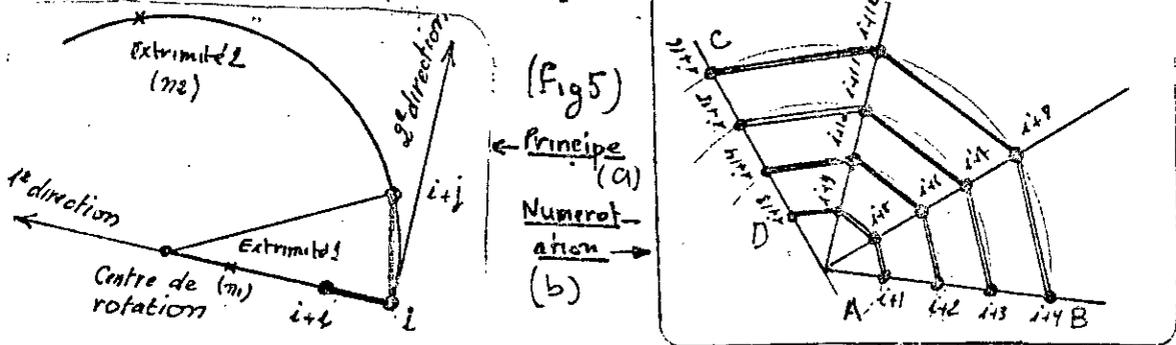
subdivisant les segments $[AB]$ et $[DC]$ par N_1 et $[AD]$ et $[BC]$ par N_2

joignant les noeuds des segments $[AD]$ et $[BC]$ entre eux selon la première direction.

subdivisant les segments déjà former par N_1 et joignant les, selon la deuxième direction

2. Dans le repère Polaire :

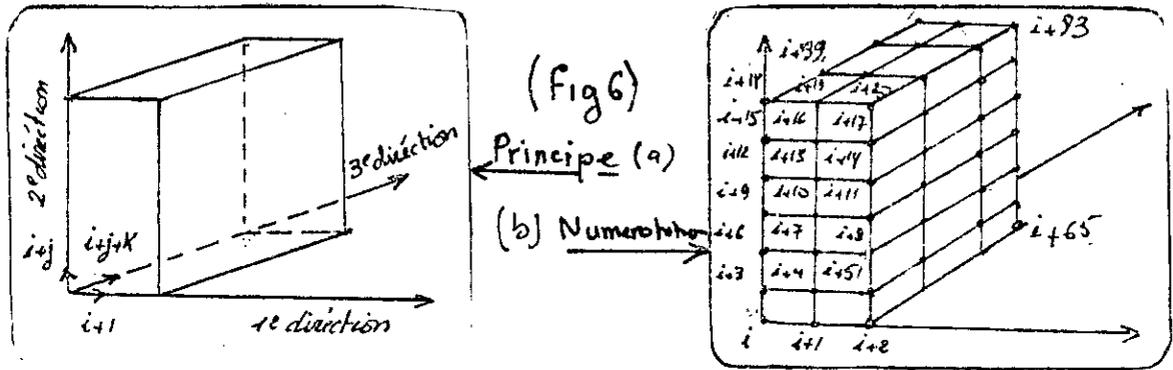
La génération se fait dans les directions: rotation et radial, le procédé de la génération suit les mêmes étapes que celles déjà vu dans le cas du repère Cartésien, il suffit simplement de faire par analogie.



II.4.2.3 Génération tridimensionnelle (en volume) :

1. Repère Cartésien :

C'est toujours le même principe bidimensionnel, quoiqu'on ajoute une 3eme direction.



3. Repère sphérique :

Dans ce cas , on aurait besoin du centre de la sphere soit (0) .

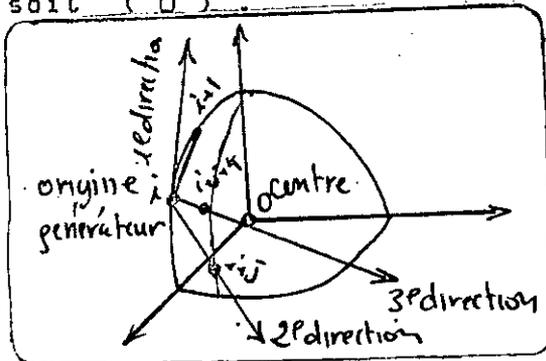
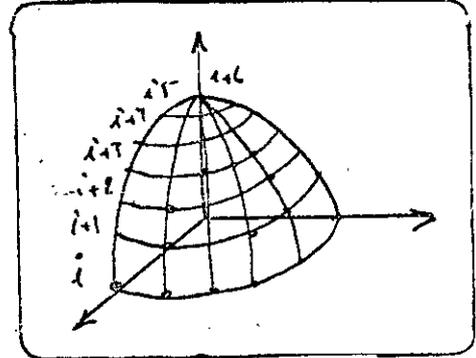


Fig:7

Principe
de numérotation



Il est évident que ces cas , ne sont pas les cas généralement rencontrés dans la pratique , du fait de la complexité des formes à mailler , mais le principe de base y'est , et pourra être réalisé dans les limites du possible.

Le problème est comment relier ces différents noeuds pour former nos éléments ?

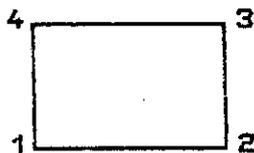
La solution à ce problème est : Puisque on connaît la méthode de numérotation des noeuds , (voir les figures ci-dessus) , il serait simple alors d'imaginer une relation entre les noeuds d'un même élément ;

pour comprendre soit l'exemple :

(voir la figure 7) .

pour le maillage bidimensionnel Cartésien, la méthode pour numérotter un élément quadrilatère est :

soit les numéros $n[1]$, $n[2]$, $n[3]$ et $n[4]$ des quatre noeuds de l'élément selon la convention suivante :



$$\begin{aligned} \text{Si } n[1] &= i \text{ alors } n[2] = i + 1, \\ & n[3] = i + j + 1, \\ \text{et } n[4] &= i + j \end{aligned}$$

où :

i : nombre du i (ème) noeud selon 1er direction

j : nombre du j (ème) noeud selon 2ème direction

II.5 Gestion et stockage des données:

Cette partie est très importante parce- qu'elle concerne un point essentiel de notre sujet. Le problème est l'optimisation de la taille mémoire en fonction de la rapidité d'exécution. En effet, Les systèmes réels sont continus et une des meilleures approches par la M.E.F pour refléter cette continuité consiste à augmenter le nombre d'éléments. Mais cette augmentation entraîne inévitablement une augmentation de la taille mémoire.

Différentes solutions sont possibles pour résoudre ce genre de problèmes. On peut, par exemple, diviser les structures en sous-structures, etc...

Pour notre part, nous avons choisi la méthode de l'allocation dynamique de la mémoire par structuration des données en enregistrements.

II.5.1 Données:

Un dessin de structure est composé d'un ensemble d'éléments ou (modèles). Ces éléments possèdent des caractéristiques géométriques et physiques différentes. Ils supportent certaines restrictions imposées et des paramètres appliqués au système.

II.5.2 Différentes méthodes de stockage

Il existe plusieurs méthodes pour le stockage de données. on citera:

Méthode de (DATA in) : Elle est utilisée dans le langage BASIC. Les données sont figées et appartiennent à la structure du programme.

Méthode de (Dimension) : c'est la méthode la plus utilisée. Elle consiste à déclarer des tableaux de variables dont la taille est fixée d'avance; c'est le cas en langage FORTRAN. Cette méthode est facile dans son utilisation mais elle est très gourmande en taille mémoire. Elle a comme inconvénient de limiter à l'avance la quantité des données.

Méthode de l'allocation dynamique de la mémoire: c'est l'une des meilleures méthodes ,elle permet une utilisation rationnelle de la mémoire ,malgré ses difficultés d'utilisation et de manipulation.

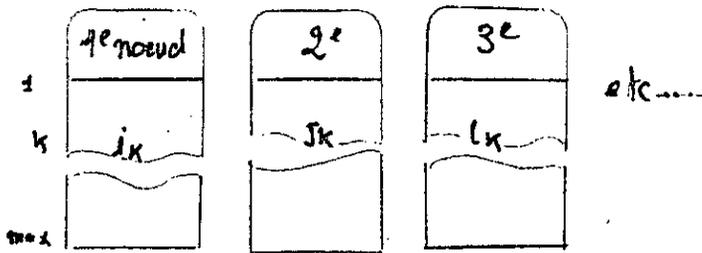
II.5.3 Différentes méthodes de gestion:

La méthode la plus classique: quoiqu'elle diffère d'un utilisateur à un autre, elle garde toujours le meme principe de base.

on présentera maintenant un exemple d'une structuration de données :

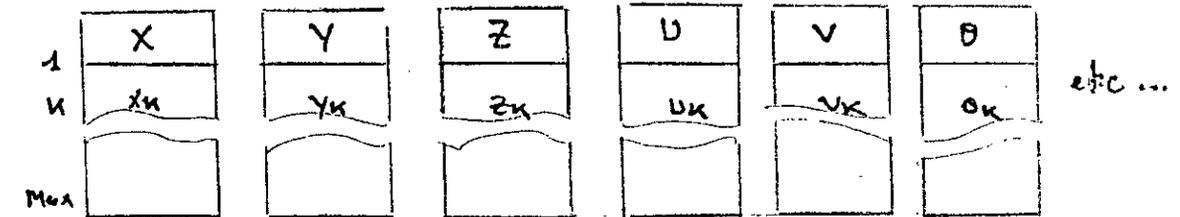
cet exemple est organisé en plusieurs structures :

* Structure des données relatives aux éléments:

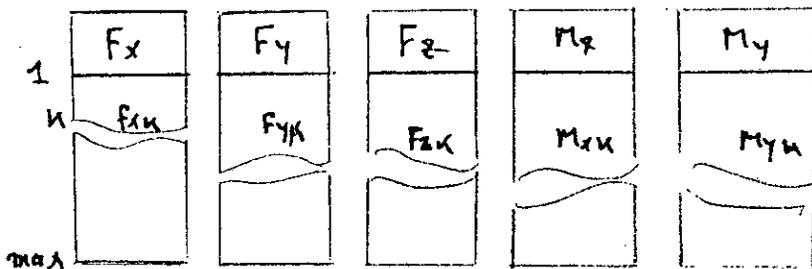


* Structure des données relatives aux noeuds:

Coordonnées et Déplacements:



Efforts:



Il existe une autre méthode assez simple, elle se base sur le principe d'enregistrement ou (RECORD en Anglais), pour plus d'éclaircissement on va présenter l'exemple suivant (en Langage PASCAL):

comme vous savez dans le FORTRAN ou dans un autre Langage on déclare généralement les variables par des identificateurs de compilation tel que REAL, INTEGER, ect..., c'est la le principe.

soit une personne quelcoque 'Mohamed', ce dernier possède les caracteristiques suivantes (le Nom, Le Prenom, L'adresse et le Numero de telephone), on a la possibilité de le déclarer comme une structure d'enregistrement suivante (cela est permisi en PASCAL, en Langage C et dans d'autres langages):

```

type                               (soit le champs personne)
  personne = RECORD;              ( l'enregistrement)
  Nom      : String;              ( chaine de caracteres )
  Prenom   : String;              ( = = )
  Adress   : String ;             ( = = )
  Tel      : Integer;             ( entier )
end;

une variable de ce type soit : Elève aura ces
caractéristiques tel que:
var élève,etudiant,travailleur ( variable )
: Personne;                       ( des variables du type personne )
en les utilisera comme suite :
with élève do
begin
  Nom      := 'Abd allah';
  Prenom   := 'Mohamed' ;
  Adress   := 'Terre' ;
  Tel      := 01 ;
end;

cette méthode très intéressante permet donc de
déclarer des types tel que :
type
  line = record ;
  x1,y1,z1,x2,y2,z2 : real ;
end;

```

Ainsi ,on a crée un ensemble d'enregistrements sur ce principe ,exemple: Noeud,Line,Quadrilataire,Hexaedre et proprietes,etc...

Actuellement,il existe une méthode très récente permise par les Langages modernes dits ' LANGAGE ORIENTE OBJET ' ,qui permettent non seulement de déclarer des champs de variable (voir exemple précédent), mais aussi les méthodes (ou les différents sous-programmes attachés à ce type de variable), On citera l'exemple suivant:

Soit une "porte de voiture" , elle est caractérisée par sa forme geometrique,ses caractéristiques mécaniques ,sont utilite...ect.

si on construit un tel enregistrement ,il doit contenir toutes ces propriétés ,une fois réalisé, plus tard ,il nous suffit de donner certains paramètres pour avoir : son dessin,son fichier de données ,ses contraintes ..ect .

Revenons maintenant aux différents avantages de la méthode de l'allocation dynamique associée à la méthode d'enregistrement,

on recitera l'exemple de la matrice d'assemblage (voir paragraphe analyse statique):

La moitié de celle-ci seulement sera mise en mémoire du fait de la symétrie, ce qui est impossible par la méthode de (Dimension).

Elimination de toutes les valeurs nulles, économisant en plus une quantité considérable, car généralement la matrice présente beaucoup de valeurs nulles.

Et aussi en maniabilité des données; grace a cette méthode : pour insérer, ordonner, écraser, modifier, etc... Qui sont des outils de base pour la saisie des données.

II.5.4 Traitement d'erreur:

D'après le réalisateur du Logiciel Turbo PASCAL, et pour tous les autres Langages, les erreurs généralement comises par les compilateurs sont de l'ordre du $1E-12$, ce qui est négligeable,

les seules erreurs qui ont de l'impacte sont les valeurs erronées qu'introduit l'utilisateur, (le Logiciel est conçu de sorte qu'il évite au maximum ces erreurs).

Ce qui reste, c'est le problème du pas minimal que permet l'écran, étant donné qu'il est le champs de saisie des données, -

ce pas et généralement estime par :

$$DL = (1 / 639) \text{ Unites le long de la largeur}$$

$$DH = (1 / 199) \text{ Unites le long de la hauteur}$$

639 et 199 : largeur et hauteur de l'écran

Unites : l'unité de mesure imposée par l'utilisateur (Micron, metre ...etc).

les solutions généralement utilisées sont : le zoom et la diminution des unites de mesure.

III. LA REPRESENTATION GRAPHIQUE [9,6,5,10,12]

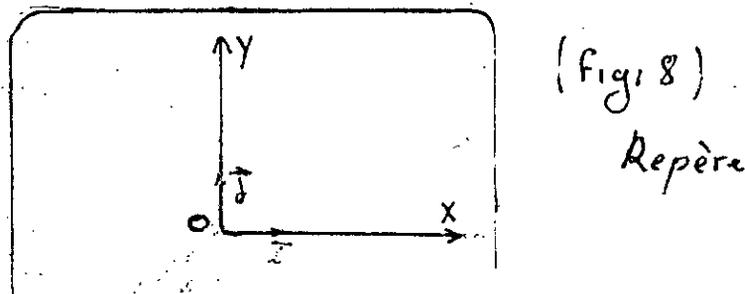
III.1 Representation deux dimensions:

C'est la méthode la plus couramment utilisée, principalement en dessin technique industriel et en architecture. Mais elle reste très peu adaptée pour les objets volumiques. Plusieurs solutions sont utilisées pour surmonter ce problème de représentation de la troisième dimension. La plus courante de ces solutions est l'ajout de vues supplémentaires (vue de gauche/droite, vue de dessous, etc...).

III.1.2 Système de repère:

Généralement, on utilise l'un des deux systèmes de repères : Cartésien ou Polaire.

III.1.2.1 Le système de coordonnées Cartésiennes:



On citera quatre types de repère cartésien:

- le repère orthonormé : dont les axes sont perpendiculaires et les unités de mesure sur ces axes sont identiques .C'est le repère le plus couramment utilisé.
- le repère perpendiculaire à unités de mesure différentes sur les deux axes.
- le repère non perpendiculaire à unités de mesure identiques.
- le repère non perpendiculaire à unités de mesure différentes.

Il faut aussi citer le cas de repère à sens inverse qu'on rencontre surtout avec les écrans d'ordinateurs.

III.1.2.2 Le système de coordonnées polaires :

Dans ce type de repère, on localise un point M par ses deux coordonnées r et θ , où r représente la distance au centre du repère et θ représente l'angle qu'il fait avec un axe de référence.

III.1.2.3 Le passage d'un système à un autre:

Ceci s'obtient très facilement avec la projection du point M de coordonnées polaires (r, θ) sur les différents axes OX et OY, on obtient:

$$X = r \times \cos (\theta)$$

$$Y = r \times \sin (\theta)$$

la transformation inverse nous donne:

$$r = \sqrt{X^2 + Y^2}$$

$$\theta = \text{arctg}(Y/X)$$

II.1.3 les fonctions élémentaires:

Ce sont les différents procédés de traçage dans le plan. On peut les classer en:

1. Point:

C'est la fonction la plus importante. Elle est à la base de pratiquement toute autre fonction. Sur un écran, elle se manifeste par les deux états d'un pixel: ALLUME ou ETEINT.

2. Ligne:

C'est un ensemble de points reliant deux extrémités. Elle obéit à l'équation bien connue $Y = a.X + b$. Sa représentation sur l'écran n'est pas aussi facile qu'il apparaît au premier abord. Elle nécessite des algorithmes adéquats.

3. Courbe:

C'est la représentation d'un ensemble de points reliés par une relation mathématique telle que $y = f(x)$.

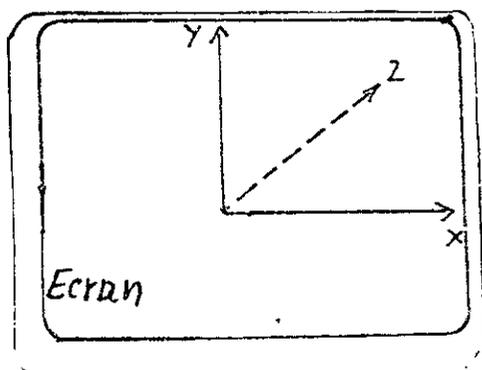
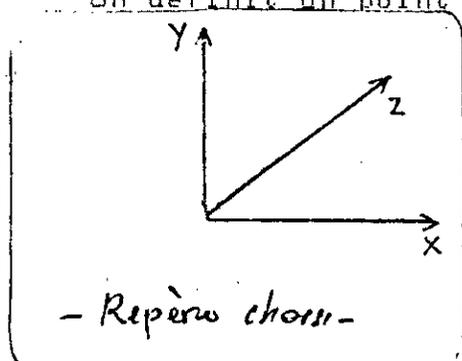
II.1.4 Les systèmes de références :

Ils se caractérisent par un centre du repère et un sens des directions.

III.2 La représentation en trois dimensions (3D):

III.2.1 Le type du repère et de l'écran de projection :

On définit un point $M(x,y,z)$ tel que :



III.2.2 La vision SPACIALE:

Dans le repère XYZ , on place un observateur, dont on définit :

1. La position.
2. La direction du regard.
3. Le type de projection.

III.2.2.1 La position de l'observateur (x_{ob}, y_{ob}, z_{ob}) :

On fait une translation à l'ensemble des points dans l'espace tel que : $M(x-x_{ob}, y-y_{ob}, z-z_{ob})$

On peut ainsi, sans changement de l'espace, moyennant ainsi la transformation .

III.2.2.2 La direction de la vision :

On définit notre repère de projection tel que :

- X : étant la largeur de l'écran
- Y : sa hauteur
- Z : fuyant derrière l'écran

Implicitement, la direction de vision est supposée selon Z. Cependant.

Lever ou baisser <<la tête>> c'est à dire Rotation autour de OX.

Tourner à gauche ou à droite <<la tête>> c'est à dire Rotation autour de OY.

Incliner << tête>> c'est à dire Rotation autour de OZ.

on a ainsi défini la position de l'observateur (x_{ob}, y_{ob}, z_{ob}), et les trois angles α , β , τ .

III.2.2.3 L'objectif utiliser :

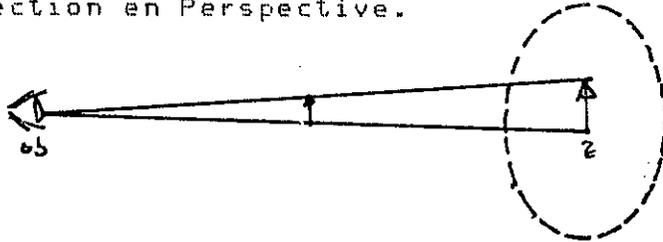
Comment se fait la projection sur l'écran plan ?

— On choisit l'axe fuyant et perpendiculaire à l'écran

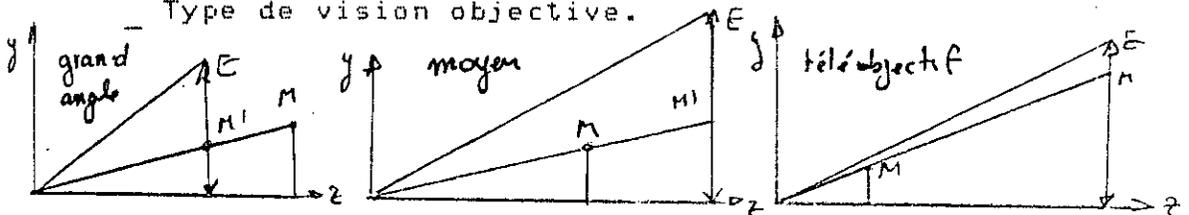
— La projection consiste simplement à ignorer la coordonnées Z .C'est la projection ORTHOGRAPHIQUE

— On envisagera une autre projection plus réaliste proche de la vision de l'oeil humain.C'est la projection en PERSPECTIVE.

— Projection en Perspective.



— Type de vision objective.



E : plan de projection et l'écran .

On remarque ainsi l'influence de l'angle de vision .

III.2.3. Les Transformations:

Pour effectuer notre projection sur l'écran ,il nous faut plusieurs transformations : translation, rotation ...etc.

III.2.3.1 Les déplacements (recentrage de l'observateur et de l'objet)

a/ l'observateur:

$$x' = x - x_{ob}$$

$$y' = y - y_{ob}$$

$$z' = z - z_{ob}$$

b/ l'objet :

$$x'' = x + dx$$

$$y'' = y + dy$$

$$z'' = z + dz$$

où: dx , dy , dz sont les déplacements relatifs d'un objet.
Après recentrage, on aura :

$$x' = x'' - x_{ob}$$

$$y' = y'' - y_{ob}$$

$$z' = z'' - z_{ob}$$

III.2.3.2 Les rotations:

On effectue trois rotations autour de OX , de OY et de OZ , respectivement d'angle α , β et γ .

III.2.3.3 La projection sur l'écran.

On assimile le cône à une pyramide.

d : distance de l'oeil à l'écran.

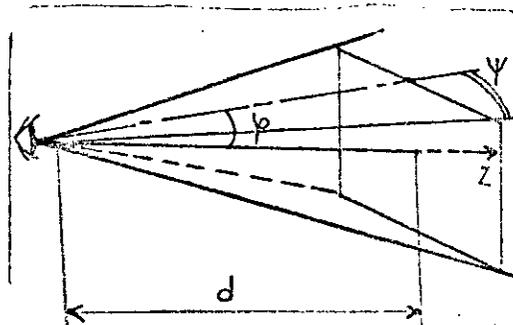
φ : le demi-angle d'ouverture dans le plan yOz .

w : (ψ) le demi-angle d'ouverture dans le plan xOy .

Définition des plans.

PLAN yOz :

(Fig.10)
(Projection)



Le point M se projète sur l'écran en M' tel que :

$$\frac{y'}{d} = \frac{y}{z} \quad \text{où} \quad y' = \frac{y d}{z}$$

PLAN xOz :

$$\text{On aura :} \quad \frac{x'}{d} = \frac{x}{z} \quad \text{où} \quad x' = \frac{x d}{z}$$

La relation existante entre φ et w (ψ)

Définissons les axes de projections sur l'écran et la taille de celui-ci:

S_x, S_y : demi-largeur (hauteur) de l'écran.

on a :

$$\text{tg } \varphi = \frac{S_y}{d}$$

$$\operatorname{tg} w = \frac{S_x}{d}$$

Les coordonnées sur les axes de projection de l'écran sont

$$X_{\text{ecran}} = \frac{x}{z} = \frac{S_x}{\operatorname{tg} w}$$

$$Y_{\text{ecran}} = \frac{y}{z} = \frac{S_y}{\operatorname{tg} \vartheta}$$

La relation entre ϑ et w est :

$$\frac{S_y}{\operatorname{tg} \vartheta} = \frac{S_x}{\operatorname{tg} w}$$

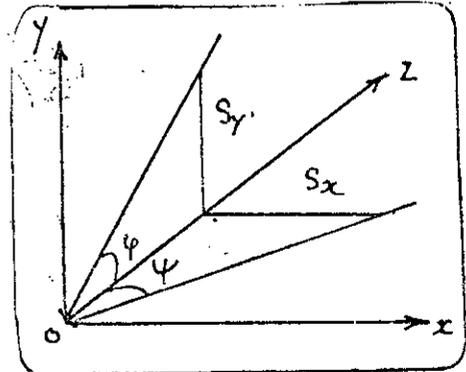


Fig: 11.

III.2.4 La visibilité:

On parlera ici des points et des segments qui appartiennent ou non à la pyramide de vision.

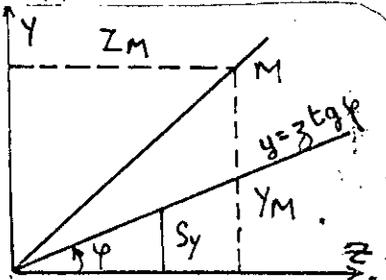


Fig: 12
(a)

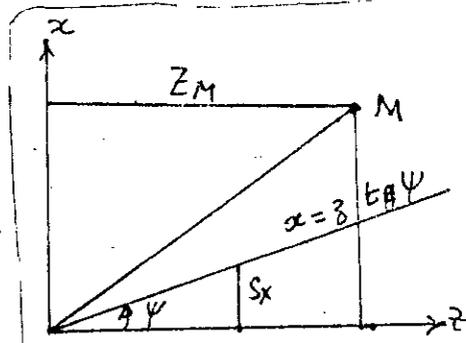


Fig: 12
(b)

On remarque ici que :

- les points dont l'ordonnée y est plus grande que $z \cdot \operatorname{tg} \vartheta$, se projettent en dehors de l'écran. Ils seront éliminés au fur et à mesure.

- les points dont l'abscisse x est plus grande que $z \cdot \operatorname{tg} w$ sont à éliminer.

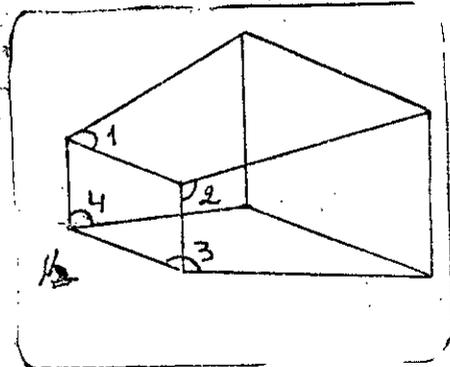
- si x , y sont plus petits que $z \cdot \operatorname{tg} w$ et $-z \cdot \operatorname{tg} \vartheta$, ils sont donc à éliminer d'où :

$$x > z \operatorname{tg} w$$

$$x < -z \operatorname{tg} w$$

$$y > z \operatorname{tg} \vartheta$$

$$y < -z \operatorname{tg} \vartheta$$



(Fig: 13)
(c)
"Espace de Travail"

- les points appartenant à ces plans se projettent au bas de l'écran.
- les inéquations retenues sont suffisantes pour les points extérieurs à la pyramide. (voir ANNEXE 3).

III.3 La vue trois dimensions (3D) :

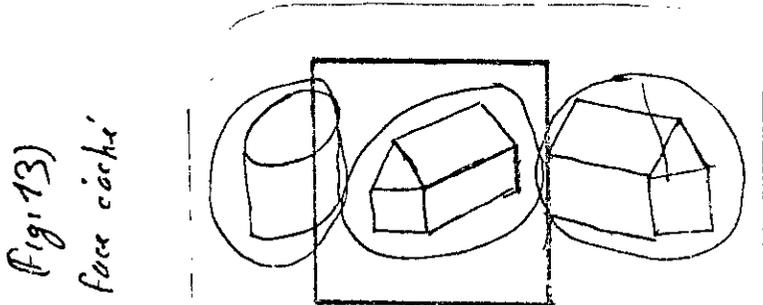
Il existe plusieurs méthodes pour la représentation d'un dessin en 3D, les plus courantes sont les méthodes en :

- fil de fer
- surface
- volume

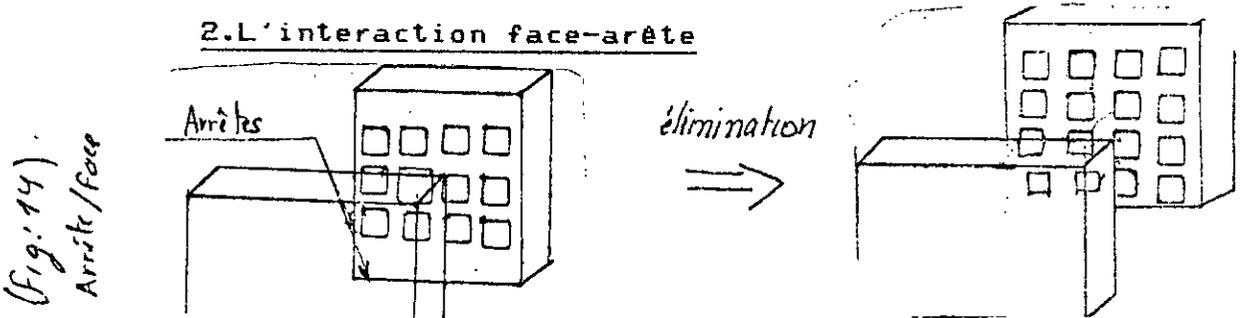
Présentant beaucoup d'ambiguïtés d'interprétation, la représentation fil de fer nous a obligé à introduire la notion de faces cachées ou objet constitué de facettes :

1. L'élimination des faces cachées (algorithme du peintre)

cette méthode consiste à noircir les facettes postérieures, voir plus loin.



2. L'interaction face-arête



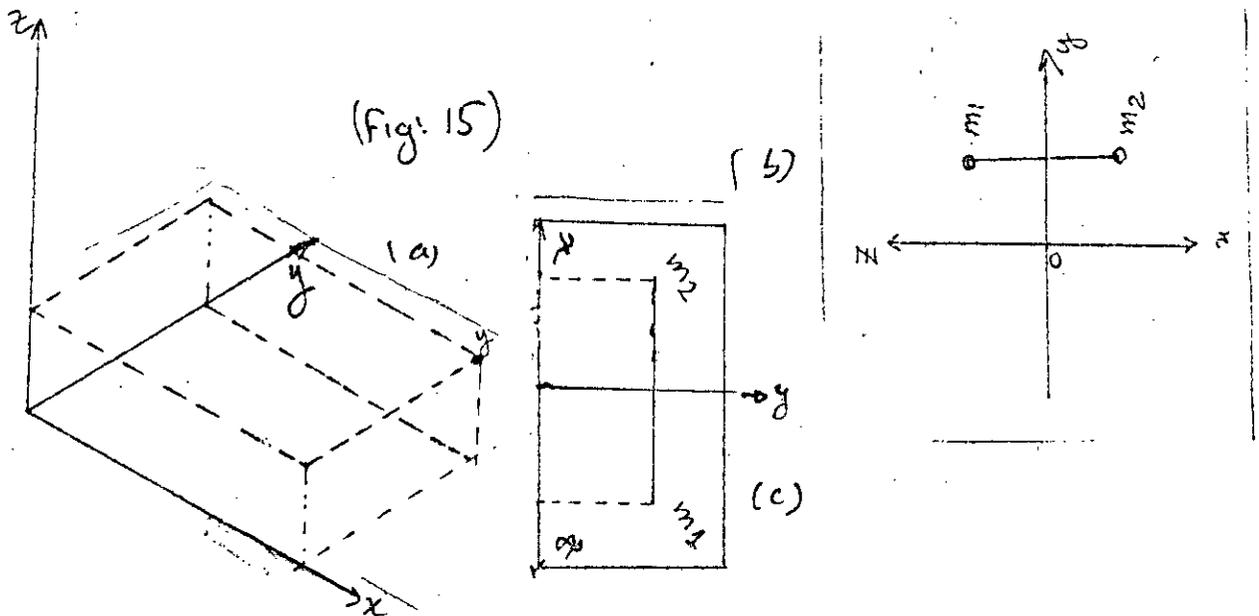
Cette méthode consiste à éliminer les segments d'intersection entre les objets, en faisant laborieusement interagir chaque arête, chaque segment avec chaque face après un tri. Il s'avère que cette méthode est très lente, au vue du très grand nombre d'éléments manipulés par rapport à la première méthode.

III.4 Mode de travail (La saisie):

L'introduction des données (EN DESSINANT) manuellement est source de nombreuses erreurs et difficultés , les moyens les plus efficaces sont : Le digitaliseur ou le numériseur (ANNEXE 2), la souris reste aussi un outil irréprochable, d'autres méthodes modernes existent tel que : Le scanner à image...etc.

III.4.1 Le mode de travail (passage 2D ----> 3D)

a/ La géométrie descriptive:



Cette méthode consiste à simuler un repère tel (fig 15 b) ou (fig 15 c); où un point quelconque $M(x,y,z)$.

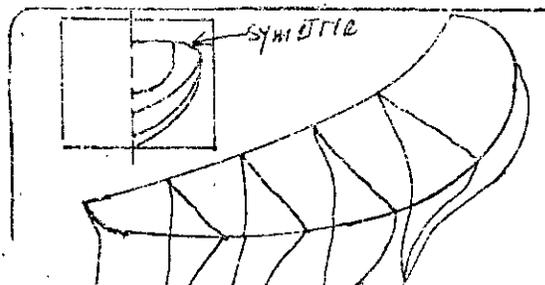
À sa projection $m2$ sur le plan XOZ et $m1$ sur XOY, il suffit maintenant de trouver les coordonnées réels de M et ce comme l'illustre la (fig 15 a).

b/ La géométrie Cotée:

Ici une cotation en Z, on réalise un croquis coté en (x,y) avec mention pour chaque point de sa cote particulière en Z, point par point (entre généralement par clavier sous forme de valeurs alpha-numérique), ce qui est très

ennuyeux. D'autres astuces peuvent remplacer cette méthode d'introduction de la 3ème coordonnée.

Fig 16

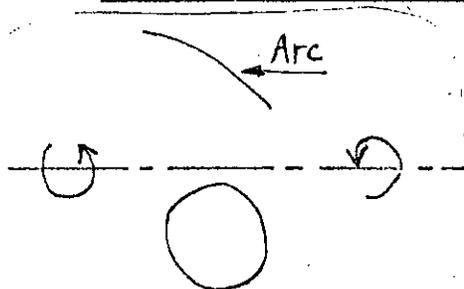


c/ La saisie d'une coque:

Cette méthode intéressante permet de saisir les polygones méridiens, (un plan de forme de navir correspond à des coques longitudinales); seront donc saisi par plusieurs coupes (voir fig 16) on générera en parallèle des facettes ou faces polygonales planes, (malheureusement couteuses en matière de mémoire) pour la représentation en faces cachées .

La méthode qui consiste à l'engendrement de facettes doit créer à chaque déplacement un nombre de facettes quadrilataires , ce contour n'est pas forcément plan , s'il y a courbure , les facettes seront considérées comme des éléments plan , mais pour les plus grande précision on utilisera des éléments triangulaires nécessairement plans; cela se répercute par une élévation du nombre de facettes, causant ainsi un coup additionnel en mémoire.

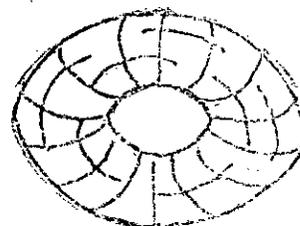
d/ La symétrie révolutionnaire:



(Fig: 17)

Principe

Exemple



à base circulaire

Cette méthode permet de réaliser des formes très variées , par une simple répétition d'un ensemble de rotation à des formes de base (voir fig 17).

e/ La saisie du 3D vers la 3D:

Cette méthode est l'une des plus modernes découvertes scientifiques de ces dernières années , elle consiste à saisir les coordonnées directement sur des maquettes par des systèmes optiques très sophistiqués (voir ANNEXE 3.)

III.5 Les transformations :

Tout Logiciel de DAO ou CAO doit nécessairement utiliser des transformations. Ces dernières sont d'une très grande utilité.

III.5.1 Le zoom

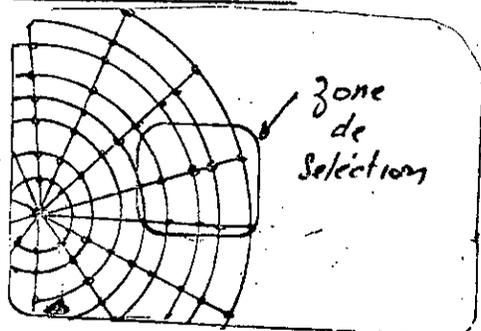
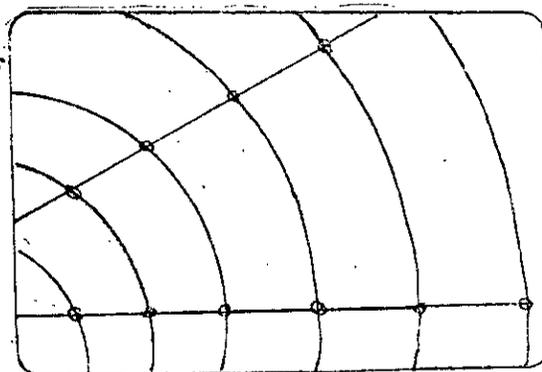


Fig: 18
zoom
⇒



Cette technique consiste en l'agrandissement d'une partie de la structure sélectionnée par un rectangle élastique. On procède pour la 2D par :

— traduire les points sélectionnés vers le milieu de l'écran.

— calculer le facteur d'agrandissement, soit k tel que :

$$k = \frac{\text{taille de la fenêtre réceptrice (l'écran)}}{\text{taille de la fenêtre source (zone sélectionnée)}}$$

— on multiplie tous les points par le facteur k et on procède au dessin.

Dans la 3D, le procédé est identique mais la multiplication se fait au niveau du module de projection sur l'écran.

III.5.2 La translation

Cette option permet le déplacement d'une zone sélectionnée ou toute la structure d'une distance définie par l'utilisateur.

soit l'expression de cette transformation:

$$X = X + d$$

X : vecteur (x,y,z)

d : vecteur déplacement (dx,dy,dz) définie par l'utilisateur.

III.5.3 Les rotations

Ce sont les relations les plus intéressantes et les plus utilisées. Elles sont employées pour tourner l'objet sur lequel on travaille afin d'une meilleure interprétation et compréhension du dessin. Elles sont aussi pour d'autres desseins comme des déplacements imposés.

— Rotation par rapport à l'axe OZ :

$$\begin{aligned} X_t &= X \sin(\theta) + Y \cos(\theta) \\ Y_t &= -X \cos(\theta) + Y \sin(\theta) \\ Z_t &= Z \end{aligned}$$

— Rotation par rapport à l'axe OY :

$$\begin{aligned} X_t &= X \sin(\theta) + Z \cos(\theta) \\ Y_t &= Y \\ Z_t &= -X \cos(\theta) + Z \sin(\theta) \end{aligned}$$

— Rotation par rapport à l'axe OX :

$$\begin{aligned} X_t &= X \\ Y_t &= Y \sin(\theta) + Z \cos(\theta) \\ Z_t &= -Y \cos(\theta) + Z \sin(\theta) \end{aligned}$$

III.5.4 L'homothétie (H):

C'est une option qui permet d'agrandir ou inverser de réduire la partie sélectionnée sans affecter les angles de l'objet. Elle peut être aussi utilisée comme moyen d'étirer certaines parties d'une structure.

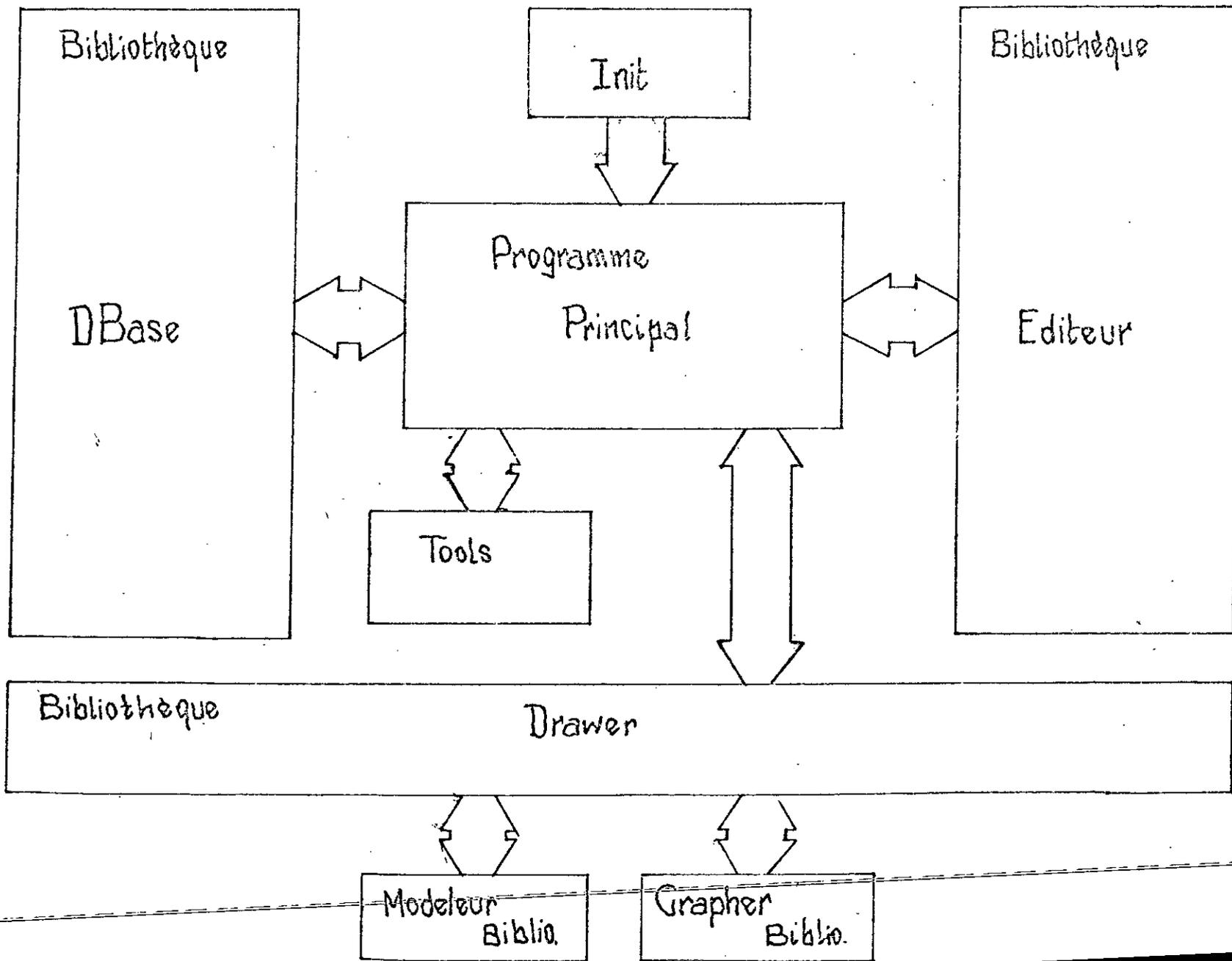
Elle est régie par l'expression :

$$X = X \times H_0$$

X : vecteur (x,y,z)

H₀: facteur d'homothétie

Il existe d'autres transformations telles que la
Torsion, l'Affinité, la Similitude planne directe et
indirecte, etc.



Organigramme
général
du
Programme

III.6 ORGANIGRAMME

Presentation des unites

- 1 : une unite :est un ensemble de fonctions.
- 2 : notre programme est lier avec un ensemble d'unités
elles sont :
- A : Init : cette unité contient les fonctions et procedures
d'initialisation du programme.
- B : DBase: cette unite rassemble toutes les fonctions qui
gèrent les données du programme.
- C : Editeur: cette unité contient toutes les fonctions
d'affichage et presentation de l'ecran
- D : Drawer : c'est l'unité responsable de la grille
d'aide (voir le shema gnerale du programme).
- E : Modleur : c'est l'unité responsable du maillage
automatique.
- F : grapher : cette unite contient toute les fonctions
de dessin et de representation 3D.

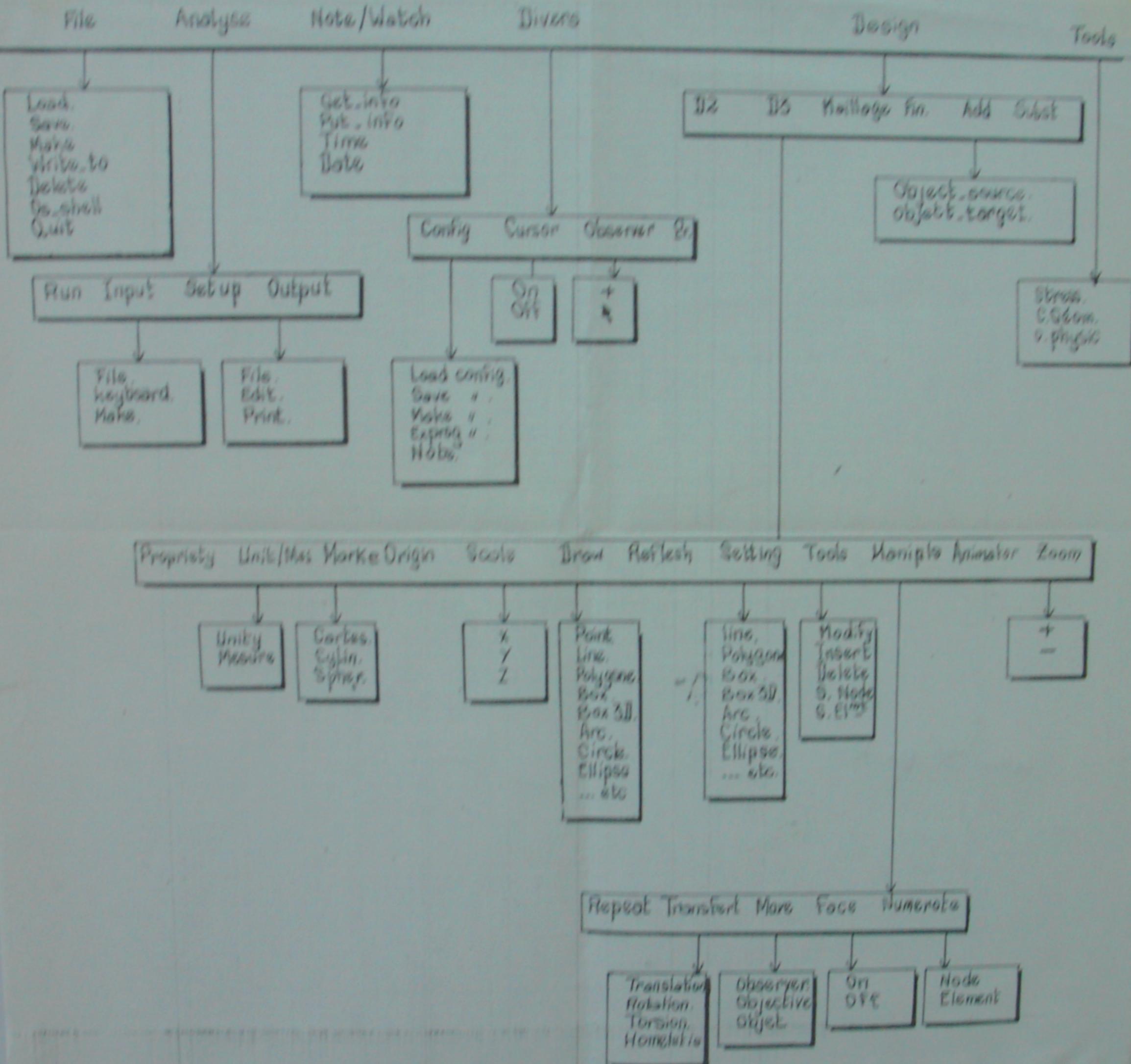


Schéma fonctionnel de l'Editeur

PM008 30
Avant p.44

III.7. Structuration de l'interface (l'editeur) :

Comme tous les editeurs qui existent actuellement, la structure la plus courante est la structure en pyramide. Un menu principal (racine) est composé de plusieurs têtes de file. Chaque tête de file regroupe plusieurs autres commandes à execution directe ou d'autres têtes de file et ainsi de suite.

Beaucoup d'autres outils existent de dialogue telles les touches de fonctions.

III.7.1.Généralités :

L'évolution actuelle des choses dans le domaine de la C.A.O nous pousse à chercher le plus de facilité et de souplesse possible dans la saisie des données, leurs représentations et la possibilité de les traiter sous toutes les formes. On étudie ici les différentes options du logiciel :

III.7.1.1 Organisation de la zone d'affichage :

Dans presque tous les logiciels graphiques actuels, on partage l'écran généralement en trois zones possibles :

- Zone du menu principale : contenant les principaux commandes ou têtes de file, sans lesquels l'opérateur ne peut ni entrer ni sortir du programme.
- Zone de saisie graphique du dessin, sinon le logiciel n'a aucune valeur, c'est la plus importante partie de l'écran.
- Une zone réservée à la grille d'aide, qui englobe toutes les fonctions de saisie telles que l'origine, l'échelle, etc..., et sans lesquelles le dessin n'est plus saisissable.

III.7.1.2 Menu principale (Racine) :

Comme on l'a déjà vu dans les différents paragraphes, le menu principale est la compression de plusieurs commandes en menus déroulants classés par groupes ou classes de même équivalence, pour notre cas, on a classé notre menu en :

'File' 'Analyse' 'Design' 'Divers' 'Tools' et 'Note/Watch'.
a/ 'File' : Cette option regroupe les principaux commandes des fichiers :

- 'Load' : charge un fichier à partir du disque dans la mémoire pour son exploitation possible.

- 'Save' : sauvegarde du présent fichier de travail sur une disquette.
- 'Make' : crée un fichier pour le présent travail (pour sa sauvegarde prochaine).
- 'Write to' : change le nom actuel du fichier en un autre nom.
- 'Ch Directory' : change le répertoire courant en un autre.
- 'Directory' : liste les différents fichiers existants dans le répertoire courant
- 'Os-shell' : retourne au système d'exploitation en gardant résidant le programme courant.
- 'Quit' : permet de quitter le programme.

Comme on le remarque, notre interface est basée sur un format traditionnel, généralement utilisé par tous les Logiciels du marché.

Analysons maintenant le reste du menu Principal.

'Analyse': cette option permet, sur le choix déjà fait par l'utilisateur, de lancer l'exécution d'un programme extérieur ou spécifique au programme; (on fait ici allusion au module de calcul de structure ou autre), voir l'option 'Tools', et aussi le type du système d'entrée que ce soit par clavier, fichiers ou saisie graphique, et enfin le type du système de sortie fichier, écran ou imprimante.

'Design' : cette option est la clef d'entrée au mode graphique c'est à dire à la grille d'aide pour la saisie graphique sur écran, et aussi à d'autres options.

'Divers' : cette option permet la configuration du programme ; sur quel Lecteur on va sauver ou charger ,le style du témoin (fleche ou croix), etc...

'Tools' : comme son nom l'indique (Utilitaires); elle contient les différents outils d'utilisations du << Dessin>>, ici il y a deux types : les sous-programmes devant exister avec par l'utilisateur par la dernière commande 'ExtPROC' de la bibliothèque 'Tools', et lance par 'Analyse'.

'Note/watch': cette option permet à l'utilisateur d'être au courant de la date et de l'heure. Elle permet aussi de porter ou de revoir des notes personnelles.

III.7.1.3. Grille d'aide (ou Bibliothèque de dessin):

Cette grille regroupe les fonctions les plus sollicitées par l'utilisateur pour sa saisie . De manière générale, pour faire un dessin pour la M.E.F ou l'Architecture, ou la C.A.O; on a besoin de plusieurs options : définir le type de repère, son origine, l'échelle, les procédures de dessin, effacer et chercher des éléments ou noeuds, etc...

On les groupe par classe d'équivalence de la manière suivante:

a/'Property' : cette option permet à l'utilisateur d'introduire les propriétés fixées par lui auparavant. Il pourra à n'importe quel moment, changer les valeurs et les noms. Cela se passe sur le principe qu'une structure est généralement composée d'éléments différents; d'où la nécessité pour chaque élément d'avoir ses propres propriétés.

b/'Unity/mes': cette option permet à l'opérateur de choisir entre deux commandes :

'Unity' : permet de fixer l'unité de mesure à utiliser : micron, mètre, ou kilomètre, etc...

'Mesure' : permet de mesurer, dans l'espace, la distance entre deux points successifs.

c/'Mark': cette options permet à l'utilisateur de définir le type de repère qu'il veut utiliser pour sa saisie. Il peut être soit Cartésien ($O \times y z$), soit Cylindrique ($O r \varnothing z$) ou soit sphérique ($\rho \varnothing \theta$).

d/'Origin': cette option permet à l'utilisateur de changer l'origine.

e/'Scale': cette option permet à l'utilisateur de choisir le type d'échelle pour une meilleure représentation ou saisie.

f/'Draw': cette option permet à l'utilisateur d'accéder à une bibliothèque contenant un ensemble de procédures de dessin telles que :

'Point' : dessine un point.

'Line' : dessine une ligne.

'Polygone' : dessine un polygone fermé ou ouvert.

'Box' : dessine un rectangle régulier.

'Box3D' : dessine un polyèdre, pouvant être:

(Hexaédrique, prismatique et tétraédrique, etc...) Avec la remarque, que tous ces procédures sont tracées avec L'ELASTIQUE ; c'est dessiner des lignes en mode XOR (inversion vidéo sans effacer la zone balayée).

g/'Refresh' : cette option permet à l'utilisateur de faire place nette ou d'effacer l'écran et refaire le dessin ; pour généralement enlever les taches et erreurs de dessin.

h/'Setting' : c'est ici une nouveauté et ainsi la plus intéressante option, du moment qu'elle permet à l'utilisateur de << configurer >> ou modéliser ses dessins en leur donnant une signification en M.E.F ou autres; cette option a

initialiser tous les procédures de dessin à un seul type d'élément, qui est la BARRE et un nombre de discrétisation égale à un (1) sur chaque direction de maillage (voir chap Maillage), il revient donc à l'utilisateur de varier comme il lui semble.

i/'Tools' : cette option a le même principe que celle déjà citée dans le menu principal, mais elle s'occupe de la gestion et de la manipulation des données (coordonnées):

'Delete' : ecraser un Noeud ou element dont on precise ou pas le numéro de désignation (pour les noeuds).

'Modify' : change la position d'un noeud.

'SearchNOD' : cherche le noeud numero N (introduit par l'utilisateur).

'SerchELT' : cherche l'élément numero N (introduit par l'utilisateur).

...etc.

j/'Maniple' : comme son nom l'indique, cette option permet de manipuler l'objet dessiné ,comme:

'Transf' : permet l'accée au differentes transformations disponibles (Tranelation , Homothetie , rotation, etc...).

'NumerotNOD' : numerote les differentes noeuds se trouvant dans le dessin.

'NumerotELT' : numerote les differentes éléments se trouvant dans le dessin.

'moveORjet' : fait tourner l'objet dessiner .

l/'Zoom' : cette option possède deux autres options :

'Zoom+' : agrandir la zone selectionnée grâce au rectangle élastique .

'Zoom-' : revient à l'etat initial .

m/'Text' : cette option permet à l'utilisateur de porter des commentaires sur son dessin à un endroit selectionné par le rectangle elastique.

n/'ClearWind' : permet à l'utilisateur d'eliminer une zone selectionnée par le parallépipéde elastique du dessin et de la memoire.

d'autres options sont accessibles à l'utilisateur par le clavier alpha-numerique; pour raison de rapidité d'execution telque:

-touche fonction 'F1' : selectionne la vue de face pour la saisie de la souris .

-touche fonction 'F2' : selectionne la face de cote droit pour la saisie par la souris.

-touche fonction 'F3' : selectionne la face de dessus pour la saisie par la souris.

-touche fonction 'Shift-F1' : change le pas de deplassement par clavier à un (1) en pixel.

-touche fonction 'shift-F2' : change le pas à deux (2).

-touche fonction 'Shift-F3' : change le pas à vingt(20).

-touche fonction 'shift-F4' : change le pas à cinquante (50).

N.B : Comme vous avez remarqué toutes les options sont formulées en Anglais. Ce choix relève du fait que la majeure partie de ces termes sont standardisés en Anglais. Cela n'empêche pas que notre logiciel permet de changer ces termes et même d'ajouter d'autres fonctions dans la langue qu'il désire. Notre choix n'est en aucune façon absolu.

III.7.1.4 Saisie de la 3 eme coordonnee :

C'est un des premiers problèmes qui s'est posé à nous. Comment saisir La Troisième donnée représentant la profondeur? La solution, dans un premier temps, est l'utilisation du clavier, c'est à dire les touches ('↑', '↓', '←', '→', 'Ins' et enfin 'Del') comme respectivement : touches de déplacement à droite, à gauche, en haut, en bas, en profondeur dans le sens positif et en profondeur dans le sens négatif. Cependant le clavier n'est un outil de saisie recommandable du faite de sa lenteur et sa non-maniabilité. Mais il remplace avantageusement l'absence de la 'SOURIS'. Avec la souris qui est un outil de saisie plane, le problème a été résolu de la manière suivante : l'appui sur le bouton gauche simule le déplacement en profondeur dans le sens positif et le bouton droit simule le déplacement dans le sens négatif.

Conclusion

Le Logiciel réalisé présente quelques caractéristiques semblables à celles des grands logiciels du commerce.

Son format standard permet une manipulation très conviviale tout en étant capable de représenter des structures très complexes.

Il permet de faire une saisie très souple aussi bien dans un plan que dans l'espace. Le logiciel permet aussi de traiter une multitude de formes en atteignant un nombre appréciable d'éléments et de noeuds.

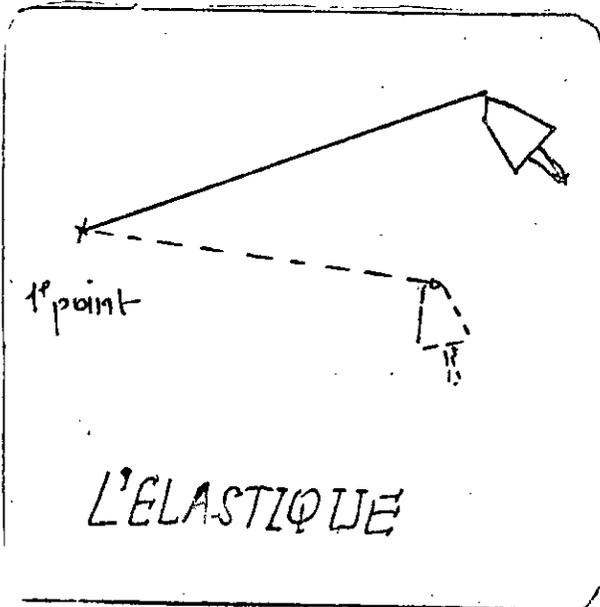
Les points forts du logiciel sont:

- une saisie très souple,
- une représentation très simple,
- un maillage automatique d'une structure,
- une gestion dynamique de la mémoire,

Pour être complet, il reste à compléter ce travail par un module de calcul des structures. D'ailleurs sa conception permet facilement l'adjonction de tel module.

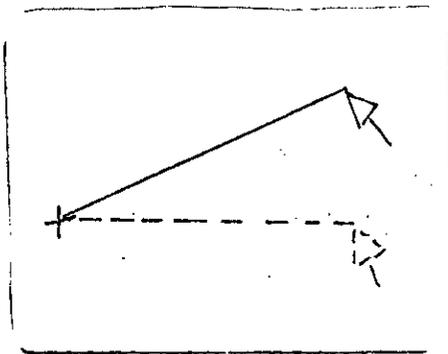
ANNEXE 1

L'ELASTIQUE

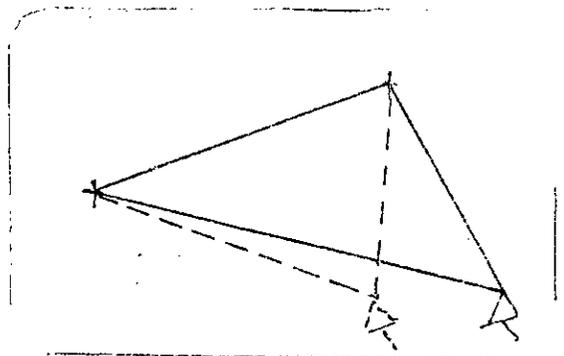


Le première trace non validé par l'utilisateur est immédiatement effacé apres changement de position du témoin sur l'écran.

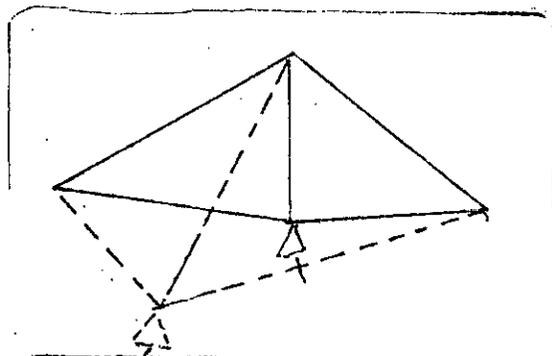
(Fig: 1)



(Fig: 2, a)
"Saisie 1^{er} arête"



(Fig: 2, b)
"Saisie 2^e arête"

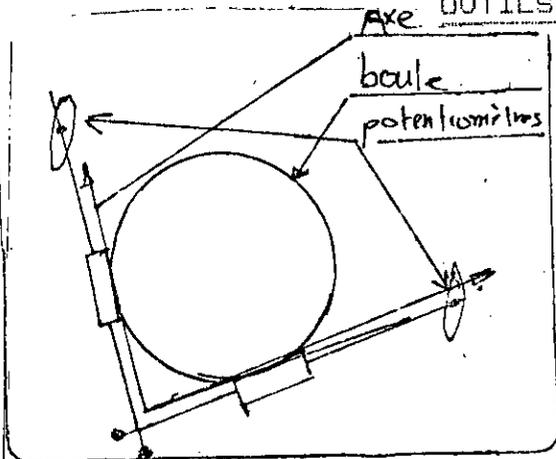


(Fig: 2, c)
"Saisie Finale
du TETRAEDRE"

ANNEXE 2

OUTILS DE DIALOGUES

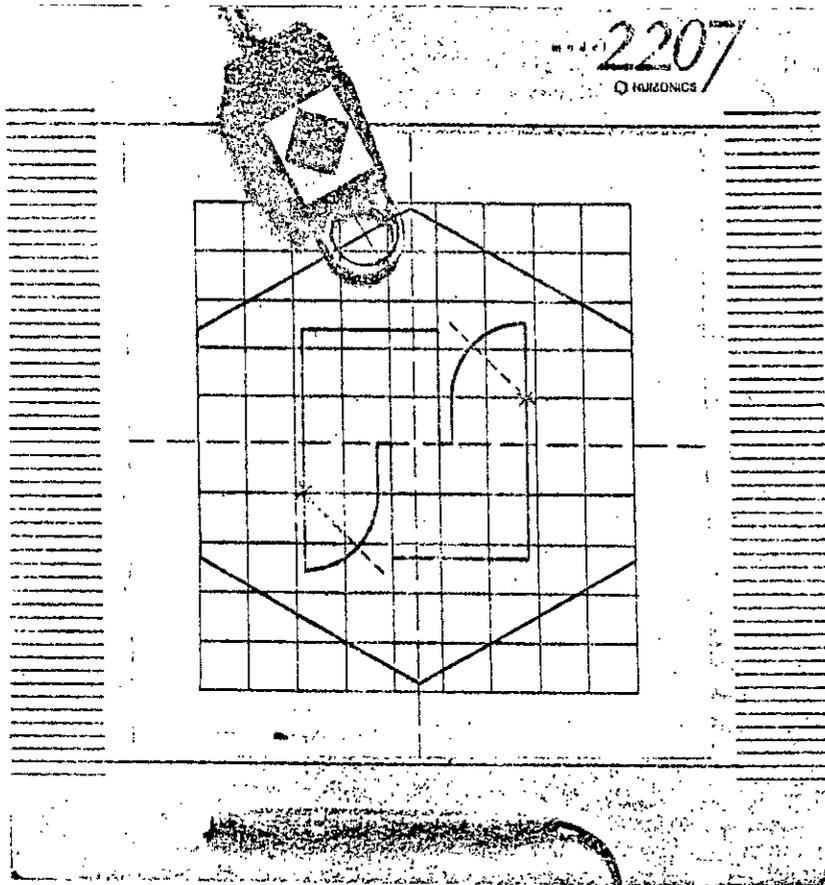
1. SOURIS [14]



(Fig: 1)

Son principe : le déplacement de la souris sur la table entraîne la rotation d'une boule qui grâce à des potentiomètres et un système adéquat, converties ces informations en valeurs significatifs (les coordonnées x et y).

2. DIGITALISEUR



(Fig: 2)

"Table de dessin plus une souris et un stylo spéciaux"

ANNEXE 3

LA SAISIE 3D-3D [11]

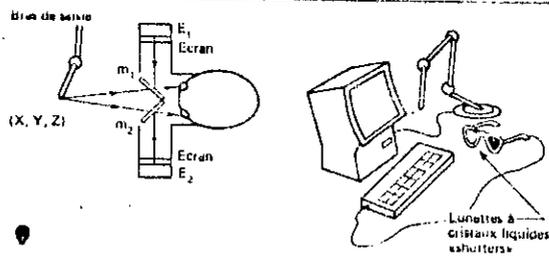


fig 1: CAO en stéréoscopie

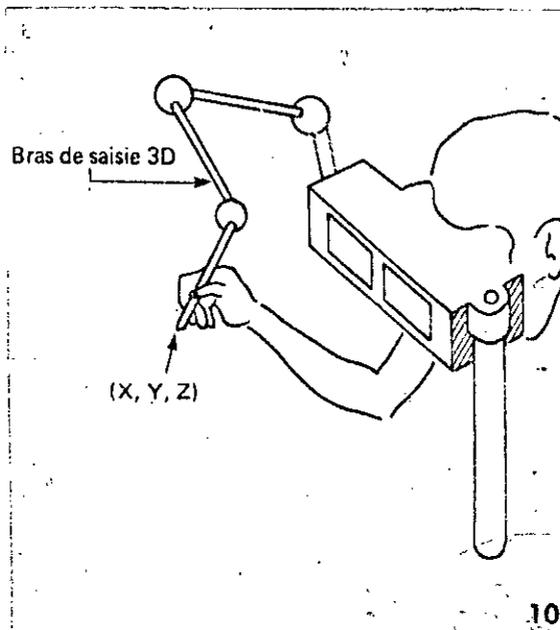


fig 2 : chronostéréoscopie

La magie des courbes de BESIER et B.SPLINE [3]

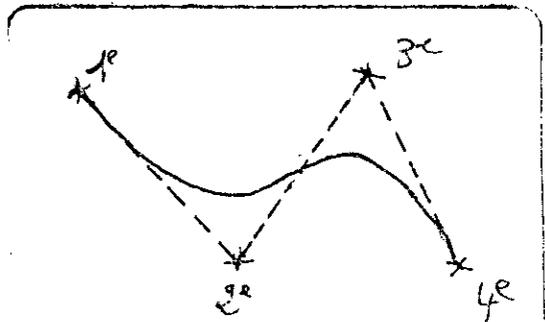
Dans la CAO professionnelle, la grande majorité des objets réels ne sont pas nécessairement des formes remarquables, ils comprennent aussi des contours, courbes gauches et des surfaces courbés, dont la représentation nécessite l'introduction des méthodes tel que :

- méthode de BESIER
- méthode de B.SPLINE

1. La méthode de B.SPLINE

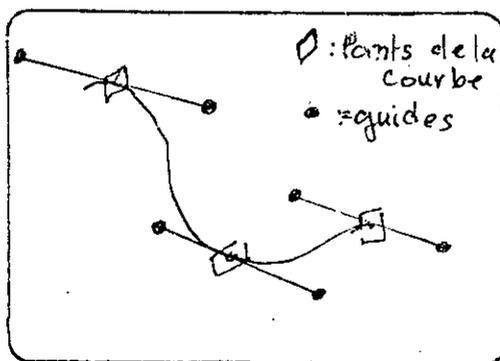
Son principe est relativement simple, une fois que les points approximants du tracé ont été introduit, on considère la courbe à tracer comme une élastique dont les point sont des attracteurs, une fois l'équilibre étant fait la courbe prend sa forme finale, voir figure.

(Fig: 1)



2. La méthode de BESIER

Comme on sait un polygone est un ensemble de points dont les sommets sont reliés par des lignes (segments), une courbe de BESIER est également définie par des points mais à chaque << sommet >>, on associe trois points qui permettent de définir la forme de la courbe, c'est des tangentes en ces points, voir figure.



(Fig: 2)

BBIBLIOGRAPHIE

Informatique

[1] : THESE : "ETUDE ET CONCEPTION D'UN ANALYSEUR SIMULATEUR
POUR STRUCTURE MECANIQUE "
Benterkia Med Fouad & Alikacem El Hachemi
I.N.F.I 1(ex CERI) 1987

[2] : TURBO-PASCAL DANS LA MACHINE
C.DELANNOY

D.A.O et C.A.O

[3] : "Pour passer d'un dessin quickdraw a une impression
postscript"
Revue DECISION Informatique
Juil 1988
pages : 35-36

[4] : serie d'article sur la 3D
Jean-Pierre PETIT
revue "L'Ordinateur Individuel"
n° 80, 81, 89,,

[5] : Computer graphics
S.Park
ALABAMA 1985

[6] : Introduction à la C.A.O
R.Camerero
MONTREAL 1987

[7] : technique Graphics Interactives et C.A.O
Y.Gardan & M.Lucas
PARIS 1983

[8] : Graphisme 3D sur votre Ordinateur
Jean-Louis VULDY
EYROLLES dec 1984

[9] : Mathematique Elements for COMPUTER
Davide.ROGERS & J.ALAIN ADAMS

[10] : "La prochaine révolution Informatique"
Abraham Peled
revue Pour la science n°122
pages : 20-29 Dec 1987

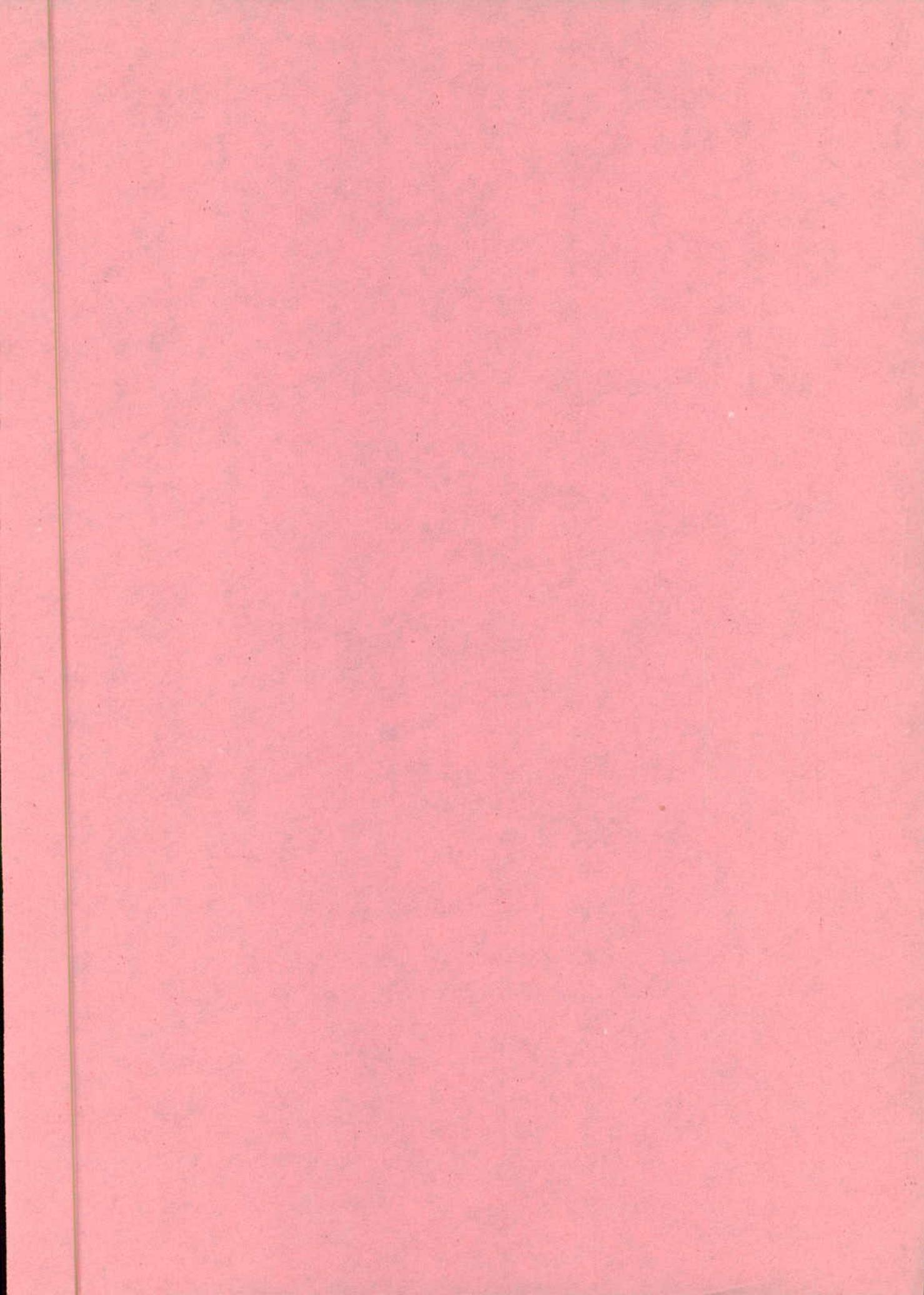
[11] : " La 3D à la portée de micro "
Jean-peirre PETIT

revue L'Ordinateur individuel n°89
pages : 106 Fev 1987

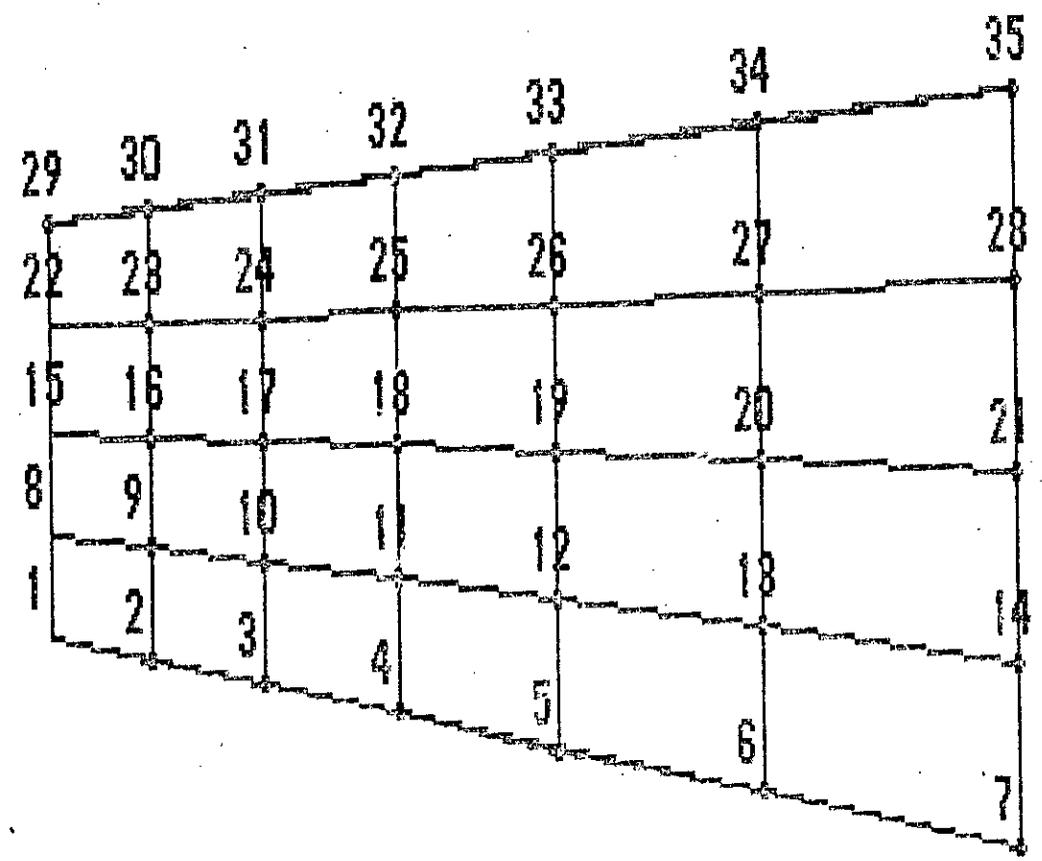
- [12] : " La C.A.O sur micro "
Marc Ferretti
revue L'Ordinateur individuel n°86
pages : 114-120 NOV 1986
- [13] : "L'industrie et les ordinateurs de demain"
Albert Erisman & Kenneth Nevs
revue Pour La science n°122
pages : 144-253 DEC 1987
- [14] : " La souris "
revue de IBM/PC juin 1987

M.E.F

- [15] : Presentation de la Methode des Elements Finis
Gouri DHATT & Gilbert TOUZOT
2eme Edition Maloine
S.A.Editeur PARIS
- [16] : Introduction aux elements Finis
traduit de l'anglais par Jean-louis CLAUDON
Ingenieur Arts et metier
Master of Science
- [17] : Introduction à la methode des elements finis
R. Evans & W.Griffiths & D.Nethercot
PARIS 1979
- [18] : Dynamique of structure
Ray W.Clought & J.penzien
Moscou 1979
- [19] : Calculation of mean areal depth of Precipitation
J.E.AKIN
journal of Hydrology
pages : 363-367 1971



- PROG LOG
- INITIATES
- VIEW
- DETERM
- SCALE
- Draw
- REFRESH
- SET LOG
- MODIS
- RETRIPLE
- ANIMATE
- LOAD
- EXIT
- REAR VIEW



On 22:6:1990 at : 12:47:9 in file:[A:DemoNum.bak]

32 0 0 0 0

1 : table of coordonnees : n° - X - Y - Z [

34	8.43333E-01	8.26813E-01	7.38693E-01
35	1.01000E+00	9.62441E-01	7.38693E-01
28	1.01000E+00	9.62441E-01	5.64070E-01
33	6.76667E-01	6.91184E-01	7.38693E-01
34	8.43333E-01	8.26813E-01	7.38693E-01
27	8.43333E-01	8.26813E-01	5.64070E-01
32	5.10000E-01	5.55556E-01	7.38693E-01
33	6.76667E-01	6.91184E-01	7.38693E-01
26	6.76667E-01	6.91184E-01	5.64070E-01
31	3.43333E-01	4.19927E-01	7.38693E-01
32	5.10000E-01	5.55556E-01	7.38693E-01
25	5.10000E-01	5.55556E-01	5.64070E-01
30	1.76667E-01	2.84298E-01	7.38693E-01
31	3.43333E-01	4.19927E-01	7.38693E-01
24	3.43333E-01	4.19927E-01	5.64070E-01
29	1.00000E-02	1.48670E-01	7.38693E-01
23	1.76667E-01	2.84298E-01	5.64070E-01
22	1.00000E-02	1.48670E-01	5.64070E-01
21	1.01000E+00	9.62441E-01	3.89447E-01
20	8.43333E-01	8.26813E-01	3.89447E-01
19	6.76667E-01	6.91184E-01	3.89447E-01
18	5.10000E-01	5.55556E-01	3.89447E-01
17	3.43333E-01	4.19927E-01	3.89447E-01
16	1.76667E-01	2.84298E-01	3.89447E-01
15	1.00000E-02	1.48670E-01	3.89447E-01
14	1.01000E+00	9.62441E-01	2.14824E-01
13	8.43333E-01	8.26813E-01	2.14824E-01
12	6.76667E-01	6.91184E-01	2.14824E-01
11	5.10000E-01	5.55556E-01	2.14824E-01
10	3.43333E-01	4.19927E-01	2.14824E-01
9	1.76667E-01	2.84298E-01	2.14824E-01
8	1.00000E-02	1.48670E-01	2.14824E-01
7	1.01000E+00	9.62441E-01	4.02010E-02
6	8.43333E-01	8.26813E-01	4.02010E-02
5	6.76667E-01	6.91184E-01	4.02010E-02
4	5.10000E-01	5.55556E-01	4.02010E-02
3	3.43333E-01	4.19927E-01	4.02010E-02
2	1.76667E-01	2.84298E-01	4.02010E-02
1	1.00000E-02	1.48670E-01	4.02010E-02

2 : table of Segment connection : n° - n1 - n2

3 : table Box connection : n° - n1 - n2 - n3 - n4

24	27	28	35	34
23	26	27	34	33
22	25	26	33	32
21	24	25	32	31
20	23	24	31	30
19	22	23	30	29

18	20	21	28	27
17	19	20	27	26
16	18	19	26	25
15	17	18	25	24
14	16	17	24	23
13	15	16	23	22
12	13	14	21	20
11	12	13	20	19
10	11	12	19	18
9	10	11	18	17
8	9	10	17	16
7	8	9	16	15
6	6	7	14	13
5	5	6	13	12
4	4	5	12	11
3	3	4	11	10
2	2	3	10	9
1	1	2	9	8

4 : table Box3D connect: n° - n1 - n2 - n3 - n4 - n5 - n6 - n7 - n8

5 : table of proprietes : for first to last one

0

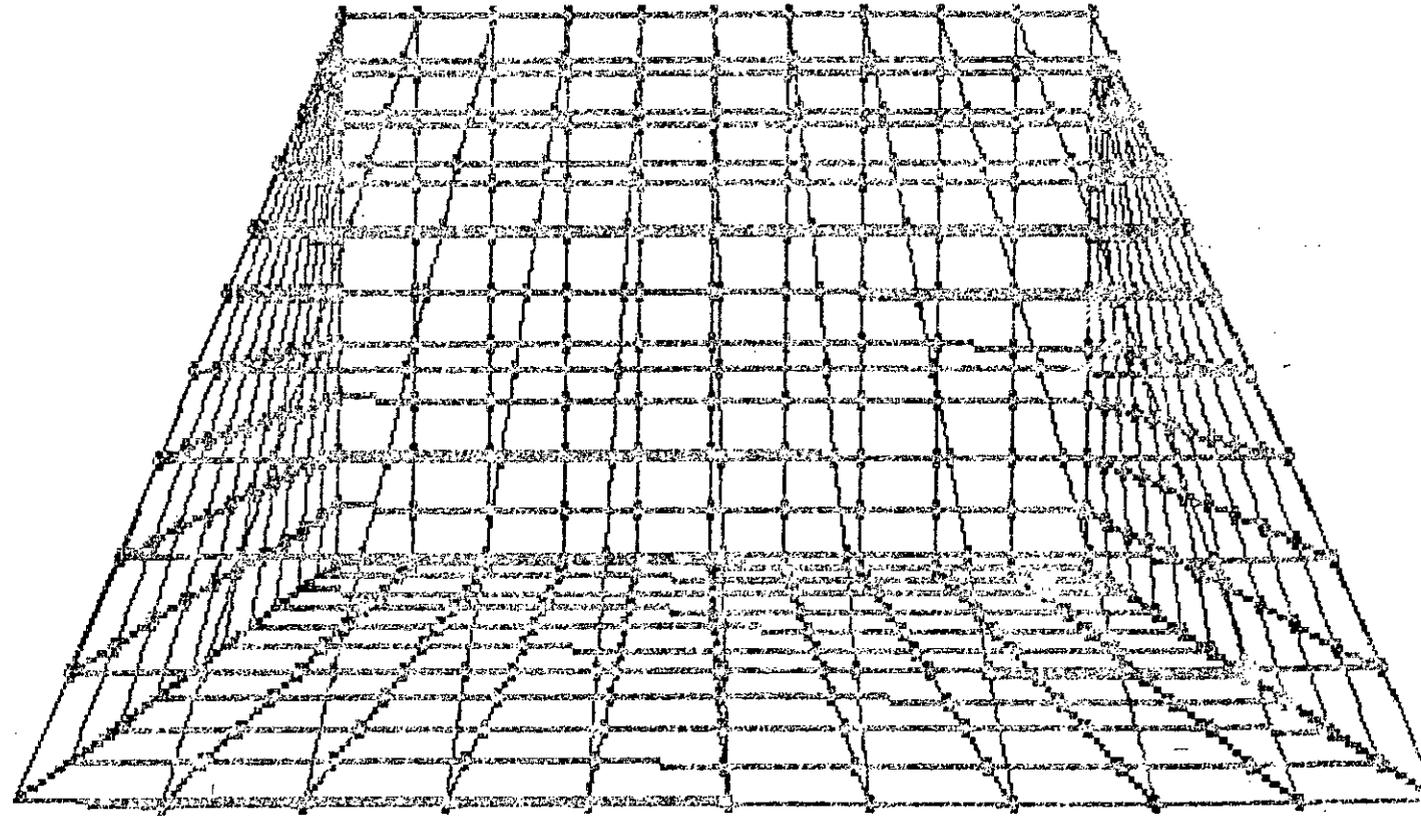
1.00000E+00

6 : table of sollicitation : n° of NOD Fx - Fy - Fz - Mx - My - Mz

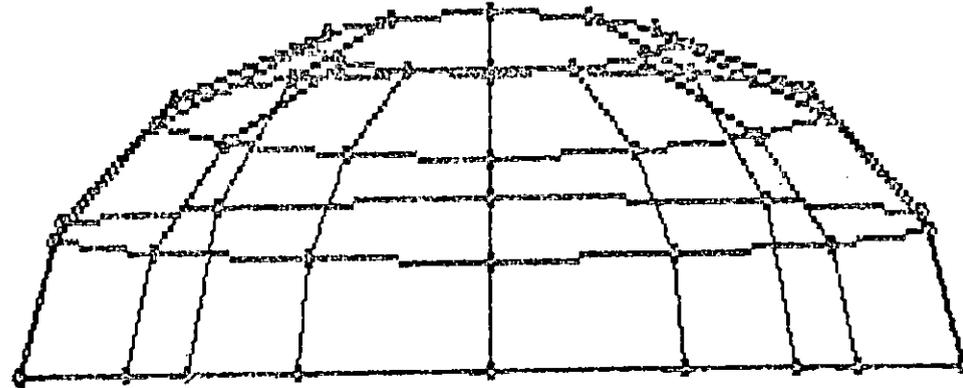
7 : table of Deplacements : n° NOD Dx - Dy - Dz

***** END of FILE *****

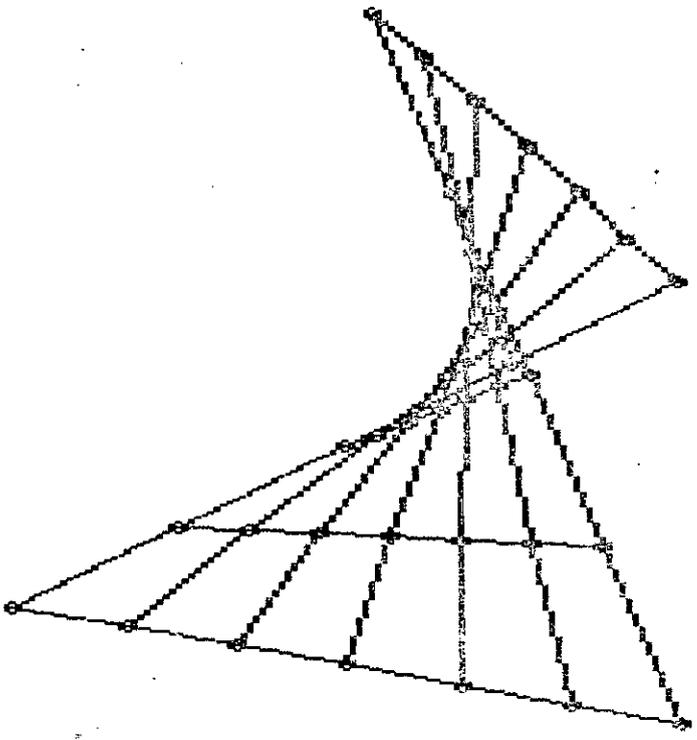
Press Inkey To return to NewCAD A:\NEWCAD\cap.bak



Press Inkey To return to NewCAD A:\Copole



Press Inkey To return to MWD00 A: MWD00E.D00



0

4

