

# THESE

Présentée à

L'UNIVERSITE DES SCIENCES ET TECHNIQUES  
DE LILLE

Pour obtenir le diplôme de  
DOCTORAT de 3ème CYCLE en  
ELECTRONIQUE

par

**Rachid HAJ NASSAR**

CONTRIBUTION A LA DETERMINATION DES  
FONCTIONS SPLINES D'INTERPOLATION  
NOTION DE PROCESSUS INTERPOLATEUR



Soutenu le 20 Septembre 1983 devant le Jury d'examen

Président	M. Le Professeur R. GABILLARD
Rapporteurs	M. Le Professeur J.C. GENTINA M. D. MEIZEL, Assistant
Examineurs	M. Le Professeur P. BORNE M. P. SABLONNIERE, Maitre-Assistant
Invités	M. J.M. BRUN, Société MATRA DATAVISION M. HEINRICH, P.S.A. M. LEVAUX, P.S.A.

AVANT-PROPOS

La présente étude a été réalisée conjointement aux Laboratoires de Systématique de l'UNIVERSITE DES SCIENCES ET TECHNIQUES de LILLE et d'Informatique Industrielle de l'INSTITUT INDUSTRIEL DU NORD. •

Nous tenons à exprimer toute notre reconnaissance à Monsieur le Professeur GABILLARD, pour l'honneur qu'il nous fait en présidant notre Jury de Thèse. Qu'il trouve ici l'expression de notre respectueuse gratitude.

Nous sommes redevables à Monsieur le Professeur GENTINA pour nous avoir accueilli dans son équipe et avoir mis à notre disposition tous les moyens nécessaires pour mener à bien cette étude. Nous tenons à lui exprimer nos sincères remerciements.

Nous remercions très chaleureusement Monsieur MEIZEL qui a dirigé ce travail. La valeur de ses conseils, sa collaboration compétente et ses encouragements bienveillants ont été un guide précieux pour l'aboutissement de ce travail. Nous tenons à lui exprimer tout particulièrement notre reconnaissance.

Que Monsieur le Professeur BORNE veuille bien trouver l'expression de notre gratitude pour nous avoir initié à la théorie de la commande optimale. Nous le remercions très sincèrement pour sa participation à notre Jury.

Nous sommes particulièrement reconnaissant à Monsieur SABLONNIERE pour l'intérêt qu'il a manifesté pour notre travail. Sa compétence en la matière et les conseils qu'il nous a apportés nous ont été infiniment profitables.

Nous sommes très honorés de l'intérêt que Monsieur BRUN, Concepteur du Logiciel EUCLID, a bien voulu porter à ce travail. Qu'il reçoive ici le témoignage de notre très respectueuse reconnaissance.

Nous sommes très sensibles à l'honneur que nous font Messieurs HEINRICH et LEVAUX, en acceptant de participer à notre Jury de Thèse. Nous sommes heureux de leur exprimer notre profonde gratitude.

Nous tenons également à remercier les membres du Laboratoire et plus particulièrement Monsieur MAYET, pour avoir mis à notre disposition, à tout moment, le matériel technique, ainsi que Madame PETIT et Mademoiselle Marie-Joséqui, avec compétence, ont assuré la dactylographie de ce mémoire, malgré des conditions difficiles.

SOMMAIRE

## INTRODUCTION

- \* Position du problème de l'usinage
- \* Conception / Interpolation / Agrégation des données

## CHAPITRE I : RAPPELS SUR QUELQUES ALGORITHMES CONNUS DE CONCEPTION ET D'INTERPOLATION

### I.1 - Introduction

### I.2 - Interpolation de données ponctuelles par des polynômes de Lagrange

- \* Problème de convergence
- \* Problème d'instabilité

### I.3 - Algorithme des fonctions de Bézier : "Système UNISURF"

- \* Application à la génération des courbes
- \* Lien avec la notion d'observabilité
- \* Application à la génération des surfaces
- \* Conclusion

### I.4 - Algorithme des fonctions "splines"

#### I.4.1 - Propriétés des fonctions Splines d'interpolation d'ordre $q$

#### I.4.2 - Méthode d'intégration pour des fonctions Splines d'ordre 2

#### I.4.3 - Méthode de transport des relations

- \* Etape I
- \* Etape II

#### I.4.4 - Méthode utilisant une base de fonctions : B-Spline

#### I.4.5 - Conclusion

## CHAPITRE II : ALGORITHME DU PROCESSUS INTERPOLATEUR

### II.1 - Introduction

### II.2 - Détermination des fonctions splines cubiques d'interpolation d'ordre $q = 2$

#### II.2.1 - Position du problème et propriétés. Notion de processus interpolateur

#### II.2.2 - Commande optimale des processus : rappels

#### II.2.3 - Structure du processus interpolateur

#### II.2.4 - Détermination du bloc de calcul

\* Relations entre  $v(x_{i+1}^+)$  et  $v(x_i^+)$

\* Calcul du vecteur  $v(x_1^+)$

#### II.2.5 - Réalisation du processus interpolateur

### II.3 - Détermination des fonctions splines sous tension d'ordre 2

#### II.3.1 - Propriétés des fonctions splines sous tension d'ordre 2

#### II.3.2 - Equations de la commande optimale

#### II.3.3 - Structure du processus interpolateur

#### II.3.4 - Calcul explicite du processus interpolateur

\* Relation entre  $v(x_{i+1}^+)$  et  $v(x_i^+)$

\* Calcul de  $v(x_1^+)$

#### II.3.5 - Réalisation du processus interpolateur

### II.4 - Détermination des fonctions Splines sous tension d'ordre $q$ généralisées

#### II.4.1 - Propriétés des fonctions splines sous tension d'ordre $q$

II.4.2 - Structure du processus interpolateur

II.4.3 - Détermination des conditions initiales  $v(x_i^+)$

\* Relation entre  $v(x_{i+1}^+)$  et  $v(x_i^+)$

\* Calcul de  $v(x_1^+)$

II.4.4 - Réalisation du processus interpolateur

II.5 - Application au calcul des fonctions splines d'ajustement d'ordre q

II.5.1 - Introduction

II.5.2 - Quelques propriétés des fonctions splines d'ajustement d'ordre q

II.5.3 - Commande optimale des processus

\* Equations théoriques d'optimalité

\* Conditions de transversalité

II.5.4 - Structure du processus d'ajustement

II.5.5 - Calcul explicite du processus d'ajustement

\* Relation entre  $v(x_i^+)$  et  $v(x_{i+1}^+)$

\* Calcul du vecteur  $v(x_1^+)$

II.5.6 - Réalisation du processus d'ajustement

II.6 - Instabilité numérique

II.6.1 - Stratégie en "boucle fermée"

II.6.2 - Structure du processus interpolateur en boucle fermée

\* Réalisation du processus interpolateur en boucle fermée

II.6.3 - Méthode de décomposition en sous réseaux



II.7 - Comparaison des algorithmes

II.7.1 - Comparaison avec la méthode d'intégration

II.7.2 - Comparaison avec la méthode de transport

II.7.3 - Comparaison avec la méthode des B-Splines

II.7.4 - Comparaison avec la méthode de Bézier

II.8 - Conclusion

CHAPITRE III : MODELISATION DES COURBES NON UNIMODALES. APPLICATION  
A LA SYNTHÈSE DES COURBES GAUCHES

III.1 - Courbes planes  $y = f(x)$

III.1.1 - Fonctions splines paramétrées d'ordre 2

III.1.2 - Propriétés

III.2 - Application à la synthèse des courbes gauches

III.3 - Conclusion

CHAPITRE IV : MODELISATION DES SURFACES

IV.1 - Introduction

IV.2 - Etude des surfaces définies sur des réseaux rectangulaires

IV.2.1 - Analyse de l'étape I

IV.2.2 - Analyse de l'étape II

ANNEXE I : POSITION DU PROBLEME DE LA COMMANDE D'UN SYSTEME LINEAIRE  
DYNAMIQUE

## INTRODUCTION

La conception et la fabrication automatisée d'objets mécaniques a créé le besoin d'algorithmes de définition, de formes géométriques en deux ou trois dimensions.

Indépendamment des primitives simples telles que droites, plans et cercles, la conception de formes plus complexes telles que carrosserie de voiture ou autres fuselages d'avions, met en évidence la nécessité de concevoir de manière automatique des courbes et surfaces gauches issues de considérations esthétiques et/ou aérodynamiques.

Dans cette optique, les algorithmes d'interpolation de données ponctuelles fournissent un outil puissant permettant de passer d'une "esquisse" (définie par exemple par quelques points ainsi que quelques conditions de tangence ou de courbure) à la matérialisation de la forme continue engendrée par les données discrètes de cette esquisse.

Par ailleurs, l'ergonomie d'un poste de travail de C.A.O. ("Conception Assistée par Ordinateur") requiert un intervalle de temps le plus bref possible entre la proposition d'une esquisse et la matérialisation sur écran graphique de la surface engendrée.

Ce point met en évidence le caractère rapide exigé pour les algorithmes de définition de formes géométriques auxquels nous nous intéressons.

Avant de préciser l'objet du travail proposé, nous allons situer tout d'abord le contexte de notre étude à l'intérieur du processus de production dans l'industrie mécanique.

Le problème de l'usinage d'une pièce mécanique consiste d'abord à choisir le type de fraise nécessaire pour réaliser une entaille de section donnée dans un matériau donné, puis à déterminer la vitesse de coupe ainsi que la vitesse maximale d'avance nécessaire à l'obtention d'un état de surface de qualité requise. Ces données sont fournies par le Bureau d'Etudes de Fabrications Mécaniques.

Plus près de la production, un autre aspect de l'usinage consiste à commander le déplacement de l'outil par rapport à la pièce, le long d'une courbe en deux ou trois dimensions, de manière à générer des surfaces par passes successives.

C'est cette dernière phase que nous nous proposons d'aborder.

Nous pouvons maintenant préciser l'objet de ce travail à l'intérieur du cadre général énoncé précédemment.

Il s'agit de fournir à la fraiseuse les consignes de déplacements en vue d'exécuter une courbe ou une surface donnée. Ces consignes sont générées à partir d'algorithmes mathématiques de définition de formes complexes.

Quelle est maintenant la place de ces algorithmes dans la conception d'un objet ?

CONCEPTION / INTERPOLATION / AGREGATION DES DONNEES

Envisageons dans un premier temps, le problème de conception d'une pièce de carrosserie de voiture par exemple.

La forme de cette pièce est contrainte par le raccordement sans angles vifs à l'ensemble des pièces déjà conçues. En dehors de ces points frontières imposés, la forme est laissée libre et sera choisie arbitrairement à partir de données fournies par le service Marketing, c'est-à-dire issues des considérations esthétiques.

Dans ces sens, tout algorithme permettant de définir des courbes à partir de données restreintes (points de raccordement à l'ensemble), peut être un "ALGORITHME DE CONCEPTION" géométrique. On souhaite néanmoins, en général, pouvoir maîtriser de manière aisée les modifications de la forme libre que cet algorithme génère.

Le système "UNISURF"\* présenté dans le Chapitre I est un représentant typique de cette classe de procédés.

Une autre approche dans la conception de dessins libres est celle de l'esquisse.

\* UNISURF : Unification des Surfaces

Dans ce cas, on se fixe, en dehors des conditions de raccordement, des points par lesquels on souhaiterait que la courbe passe (la forme est alors imaginée). Un "ALGORITHME D'INTERPOLATION" permet de relier tous les points de l'esquisse et donc de matérialiser la forme imaginée.

Cette forme, si elle n'est pas validée par le concepteur (pour son aspect esthétique) peut être déformée simplement par déplacement d'un ou plusieurs points de passage. Une illustration de ce procédé est représentée sur les figures suivantes (Fig - 1).

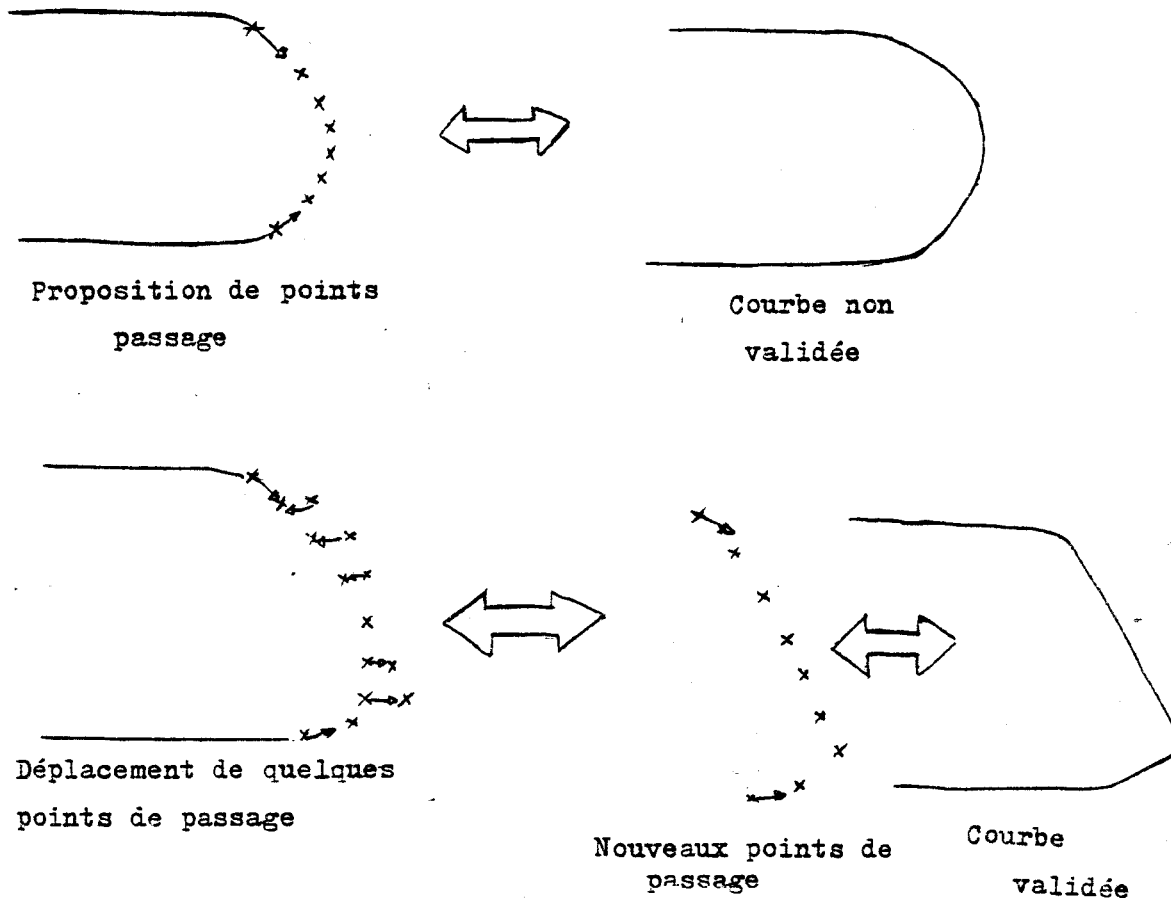


Fig - 1 : VALIDATION D'UNE COURBE PAR DEPLACEMENTS DE POINTS

(INTERPOLATION)

En dehors du cadre de la conception, les algorithmes de définition de courbes permettent, en outre, de restreindre le nombre de données nécessaires pour modéliser une image, c'est-à-dire un ensemble de points dans l'espace à trois dimensions.

Considérons la courbe suivante qu'on souhaite faire réaliser à une machine outil (Fig - 2).

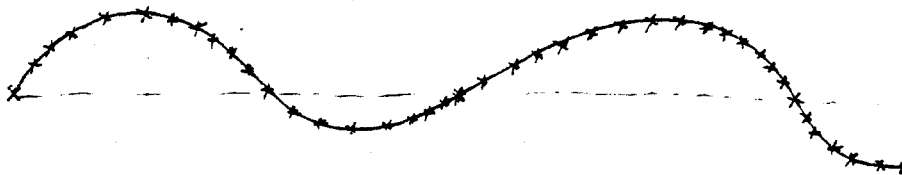


Fig - 2

A priori, il suffirait de mémoriser tous les points de consigne en séquence et de les fournir ensuite à la machine. Cette solution est à proscrire dans la mesure où une représentation graphique comprend énormément de données, ce qui entraînerait alors l'utilisation d'une capacité phénoménale de mémoire.

Il est plus intéressant pratiquement de reconstruire cette courbe à partir de quelques points qui constitueront les données devant être transférées, et "apprendre" à la machine outil la manière de relier ces points ; (algorithme de reconstitution) (cf Fig - 3).

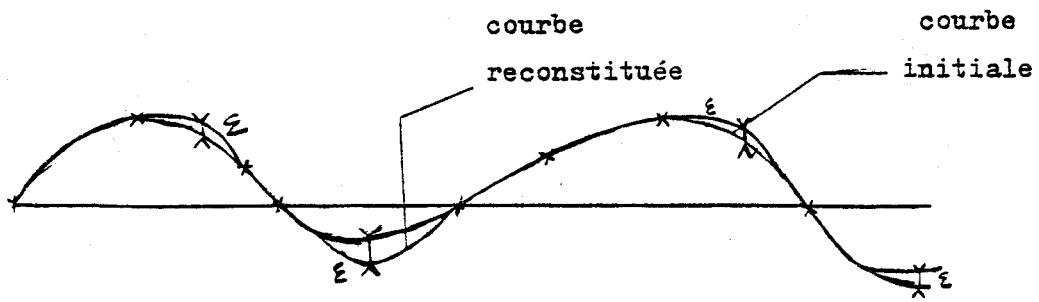


Fig - 3 : RECONSTITUTION D'UNE COURBE A PARTIR DE QUELQUES POINTS

Si l'écart  $\epsilon$  entre la courbe réelle et la courbe reconstruite est trop important, on peut toujours rajouter des données ou bien déplacer un certain nombre de points initialement choisis (cf Fig - 4).

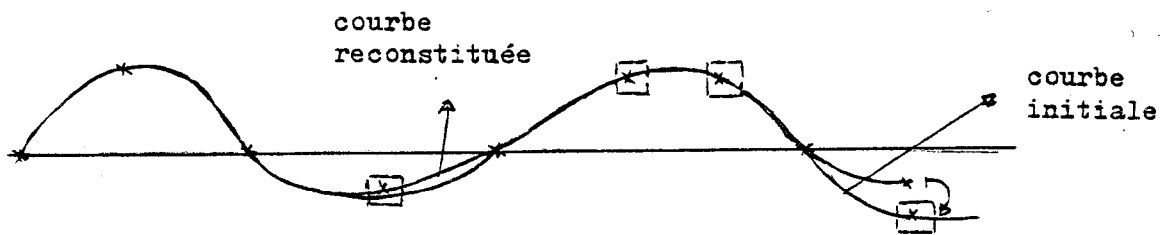


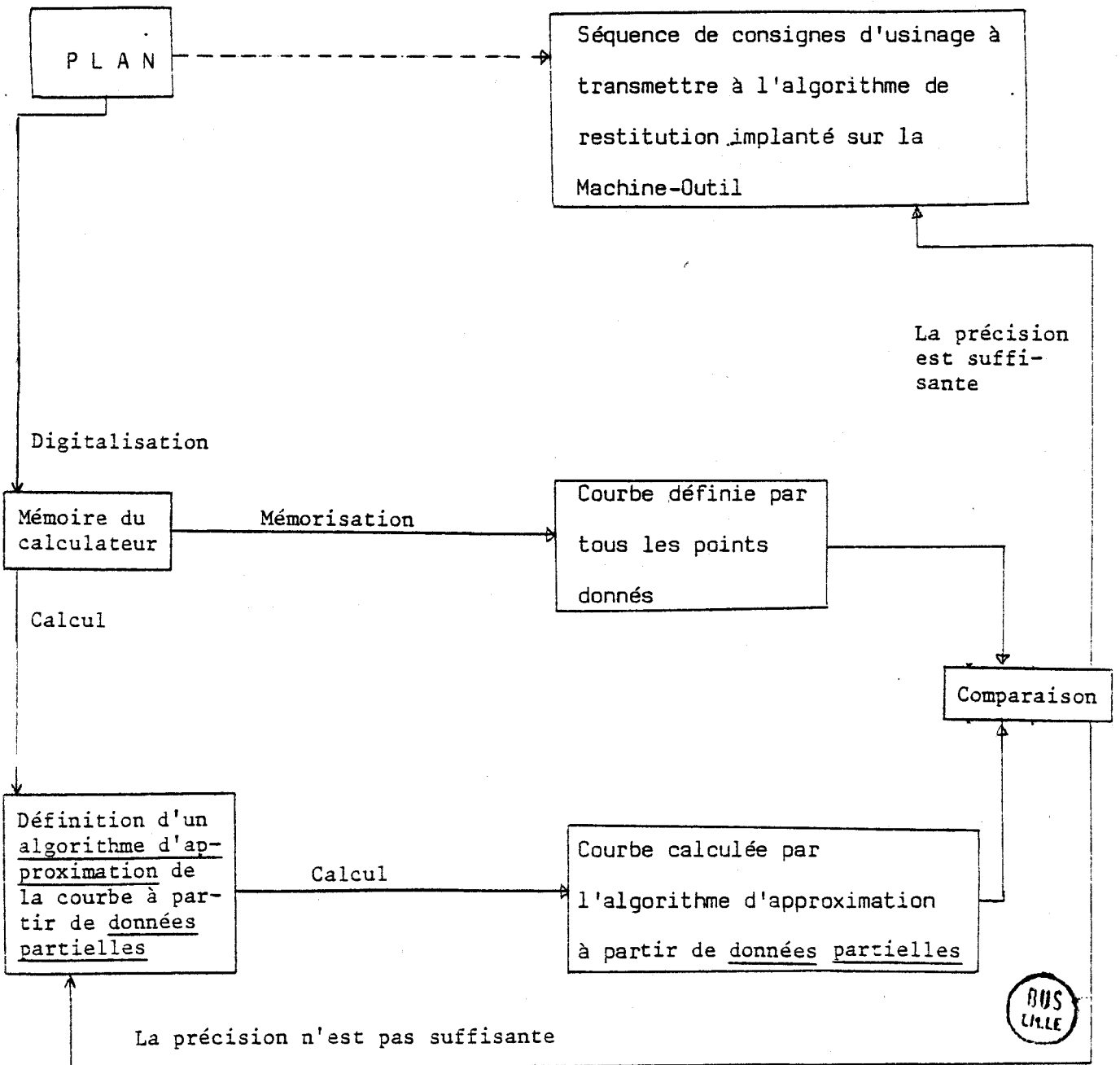
Fig - 4 : RECONSTITUTION DE LA COURBE PAR ADJONCTION DE 3 POINTS SUPPLEMENTAIRES  $\boxed{x}$  ET DEPLACEMENT DU DERNIER POINT



La démarche générale de ce "PROCEDE D'AGREGATION DE DONNEES" est décrite sur le synoptique suivant (cf Fig - 5).

Fig-5

Représentation graphique



Il est à noter que cette phase d'agrégation de données n'est utile que lorsque le plan à réaliser n'a pas été conçu sur un ordinateur numérique. Dans le cas contraire, on a de manière quasi-directe des données agrégées.

A l'intérieur du cadre général énoncé précédemment, le présent travail propose d'apporter une contribution au problème de la modélisation et/ou de la conception des courbes et surfaces gauches définies ponctuellement, en vue de leur usinage.

Dans cet esprit, une surface est définie par une famille de courbes gauches représentant les trajectoires suivies par un pointeur de console graphique ou par un outil. Nous proposons une méthode originale de détermination de ces trajectoires en les interprétant comme les sorties générées par un système dynamique linéaire stationnaire, matérialisé par un schéma ou un programme de simulation.

Ces trajectoires sont décrites par des fonctions "splines" qui constituent un outil puissant d'interpolation et sont générées, dans l'interprétation proposée de la théorie de la commande optimale des processus linéaires continus et/ou de la théorie du calcul des variations.

Nous nous intéressons particulièrement à la mise en oeuvre d'un algorithme rapide de génération de courbes et surfaces splines qui minimisent un critère de forme quadratique. Ces surfaces sont

conçues pour être exécutées par passes successives par une fraiseuse à commande numérique, permettant de "réaliser" les algorithmes développés. Dans ce contexte, la représentation dans l'espace d'état du problème d'interpolation d'un réseau de points par une fonction "spline" d'ordre  $q$  aboutit à la définition nouvelle d'un système dynamique noté "Processus interpolateur".

Dans un premier chapitre, nous introduisons l'étude de la génération des courbes gauches "non mathématiques" (c'est-à-dire dont l'expression analytique n'est pas connue de manière notoire).

Nous présentons quelques rappels concernant la définition de ces fonctions complexes par la méthode "UNISURF" due à P. BEZ IER ainsi que par l'utilisation de fonctions "splines". Ces deux procédés sont à l'heure actuelle, très largement utilisés dans la pratique, et nous en analysons les propriétés.

Dans un second volet, après avoir identifié le problème des fonctions splines, à un problème de commande optimale de processus linéaires, nous définissons au chapitre II les fondements de l'algorithme proposé en introduisant la notion de "processus interpolateur" au cas des fonctions splines cubiques. La même démarche est alors effectuée pour étudier d'autres types de

fonctions splines (exponentielle, trigonométrique, sous-tension, d'ajustement), et permet de rétablir de manière simple les propriétés classiques introduites par l'étude de ces fonctions par l'analyse numérique.

Le problème de l'instabilité numérique possible lorsque le nombre de points à interpoler est élevé, est alors envisagé et une solution proposée.

Le chapitre III définit la modélisation des courbes gauches unimodales et non unimodales dans le plan et dans l'espace de la géométrie à partir d'une décomposition de la courbe en morceaux et/ou du choix d'un paramétrage judicieux.

Le chapitre IV introduit le problème de la génération des surfaces définies ponctuellement sur des réseaux rectangulaires. Celles-ci sont envisagées comme un produit tensoriel de fonctions "splines sous-tension" unidimensionnelles définies sur des sous-réseaux plans.

REMARQUE :

Dans un but de clarté et en raison de l'abondante bibliographie, les résultats fondamentaux de la théorie de la commande optimale utilisés dans ce travail ne font l'objet d'aucune démonstration rigoureuse.

Néanmoins, nous présentons sous forme condensée, en annexe, le principe du maximum de PONTRYAGUINE ainsi que quelques résultats issus du calcul des variations.

# CHAPITRE I

## CHAPITRE I : RAPPELS SUR QUELQUES ALGORITHMES

### CONNUS DE CONCEPTION ET D'INTERPOLATION

---

#### I - 1 : INTRODUCTION :

Le problème fondamental dans la recherche de la solution au problème de conception et/ou de reconstitution de formes géométriques complexes, réside dans le choix d'un modèle mathématique qui réponde à la fois aux spécifications relatives à la géométrie des formes à générer et à des critères de performance (traitement en temps réel).

En général, les conditions présentées dans le cahier des charges et imposées à une courbe sont des conditions de continuité, des points de passage, et éventuellement des tangentes en ces points.

Compte-tenu de ces hypothèses, le problème admet une infinité de solutions, conformément à la figure suivante (cf Fig - 6).

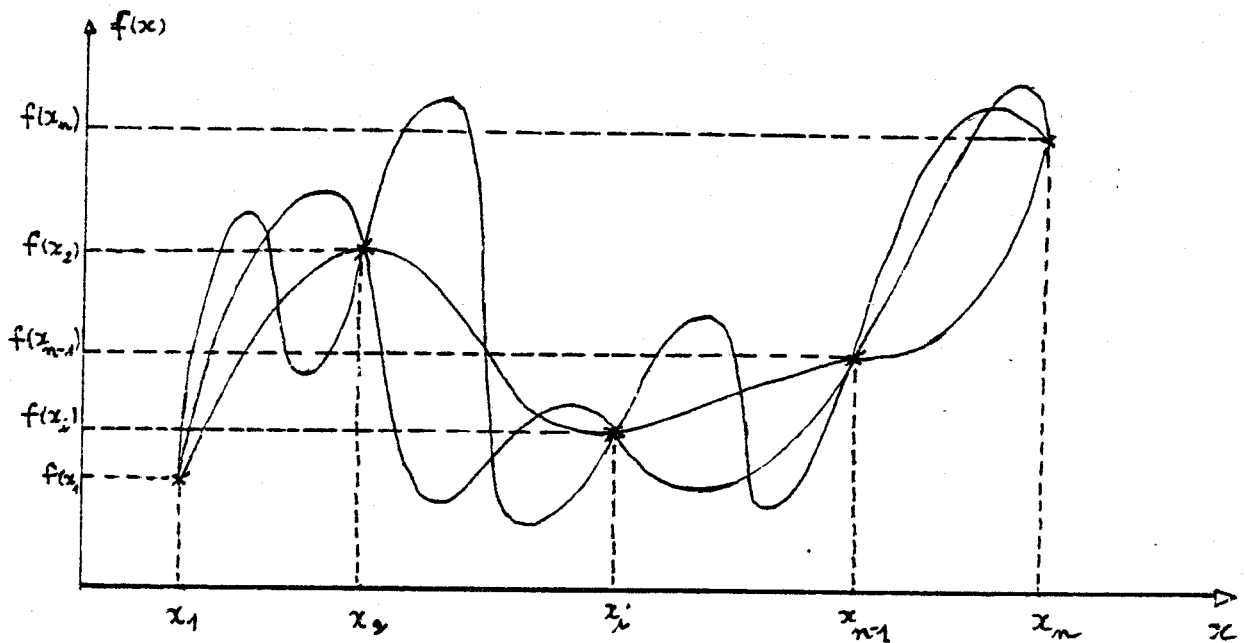


Fig - 6 : QUELQUES SOLUTIONS DU PROBLEME D'INTERPOLATION  
DE DONNEES PONCTUELLES

De manière à obtenir les formes désirées, on utilise en général des algorithmes d'interpolation polynomiale, polynomiale par morceaux, où faisant intervenir des bases de fonctions, dont nous proposons de présenter quelques résultats. Les premiers conduisent à des polynomes d'approximation de "LAGRANGE", alors que l'interpolation polynomiale par morceaux aboutit aux méthodes "d'intégration" et de "transport" (MARCHOUK, 1981 - LAURENT, 1972) pour des fonctions splines d'interpolation. Deux algorithmes définissent l'interpolation à partir de bases de fonctions : ce sont respectivement les algorithmes de BEZIER qui conduit au système U.N.I.S.U.R.F (défini à partir d'une base de Bernstein) (BEZIER, 1968), et de Schoenberg qui aboutit aux fonctions B-splines (utilisant une base de GREEN), (de BOOR, 1978)

I - 2 : INTERPOLATION DE DONNEES PONCTUELLES PAR DES POLYNOMES

D'APPROXIMATION DE LAGRANGE :

Le problème de rechercher une interpolation polynomiale d'une fonction peut s'énoncer en dimension un de la façon suivante :

Soient  $n$  abscisses distinctes  $x_i \in \mathbb{R}$  appartenant à un intervalle  $[x_1, x_n]$  choisi arbitrairement et telles que :

$$x_1 < x_2 < \dots < x_{n-1} < x_n \quad (\text{I - 2 - 1})$$

Soient  $n$  nombres réels  $y_i$  ( $i=1, \dots, n$ ) représentant en général les valeurs en  $x_i$  d'une certaine fonction  $f$  réelle, continue sur  $[x_1, x_n]$ . L'interpolation polynomiale consiste à trouver un polynôme  $P_n$  de degré  $\leq n$  tel que soit vérifiée la relation (I - 2 - 2).

$$P_n(x_i) = f(x_i), \quad \forall i = 1, \dots, n \quad (\text{I - 2 - 2})$$

Si on pose  $P_n(x) = \sum_{j=1}^n a_j \cdot x^j$  ;  $\forall j=1, \dots, n$  ; les  $a_j$  satisfont au système linéaire suivant :



$$\sum_{j=1}^n a_j \cdot x_i^j = f(x_i) \quad ; \quad \forall i=1, \dots, n \quad (I - 2 - 3)$$

Les polynomes d'interpolation de LAGRANGE permettent d'apporter une solution au problème posé précédemment ; si  $L_i^n(x)$  désigne la suite des polynomes de LAGRANGE d'expression (I - 2 - 4)

$$L_i^n(x) = \frac{\prod_{\substack{j=1 \\ i \neq j}}^n (x-x_j)}{\prod_{\substack{j=1 \\ j \neq i}}^n (x_i-x_j)}$$

avec  $L_i^n(x_k) = 0$  si  $i \neq k$  ,  $L_i^n(x_k) = 1$  ,  $P_n(x_k) = f(x_k)$

(I - 2 - 4)

alors le polynome  $P_n(x)$  qui interpole le réseau  $(x_i, f(x_i))$  ;

$\forall i = 1, \dots, n$  s'écrit sous la forme (I - 2 - 5) :

$$P_n(x) = \sum_{i=1}^n f(x_i) \cdot L_i^n(x) \quad (I - 2 - 5)$$

En général, dans la conception assistée par ordinateur de courbes définies par interpolation, le nombre de points à interpoler peut être élevé. On sait dans ce cas, que l'utilisation des polynômes d'approximation de LAGRANGE conduit à une instabilité numérique (P.J. LAURENT, 1972):

Par ailleurs, si  $L_i(x)$  désigne la suite des polynômes de LAGRANGE utilisés dans l'approximation de  $f(x)$  définie sur un intervalle fermé borné, on montre (P.J. LAURENT, 1972) qu'on ne peut assurer la convergence uniforme de  $P_n(x)$  (cf I - 2 - 2) vers la fonction  $f$ .

Nous proposons d'explicitier ces deux points :

- PROBLEME DE CONVERGENCE POUR DES POLYNOMES DE LAGRANGE .

Considérons une suite  $L_n$  d'opérateurs linéaires continus d'un espace de BANACH  $E_1$  dans un espace vectoriel nommé  $E_2$  et  $L$  un opérateur linéaire continu de  $E_1$  dans  $E_2$ . On sait, d'après le théorème de BANACH-STEINHAUS (P.J. LAURENT, 1972) qu'une condition nécessaire et suffisante pour que  $L_n$  converge uniformément vers  $L(f)$  pour tout  $f \in E_1$  est que soient vérifiées les deux conditions suivantes (I - 2 - 6) :

$$- \lim_{n \rightarrow \infty} L_n(f) = L(f) \quad \forall f \in F$$

(I - 2 - 6)

$$- \|L_n\| \text{ soit borné en } n$$

Appelons  $C_\infty$  l'espace des fonctions  $f$  continues sur une intervalle  $[a, b] = [x_1, x_n]$  muni de la norme :

$$\|f\|_\infty = \max_{x \in [a, b]} |f(x)|$$

Soit  $P_n$  l'ensemble des polynomes de degré  $n$  et  $P_n$  le polynome de degré  $n$  qui coïncide avec  $f$  en  $n$  abscisses  $x_i$  de  $[a, b]$

$$P_n(x_i) = f(x_i) ; \forall i = 1, \dots, n$$

On sait que l'on peut écrire la relation suivante :

$$\forall x \in [a, b] ; P_n(x) = \sum_{i=1}^n f(x_i) \cdot L_i^n(x)$$

(I - 2 - 7)

$L_i^n(x)$  sont les polynomes d'approximation de LAGRANGE de degré  $n$ .

L'opérateur  $L_n$  qui à toute fonction  $f \in C_\infty$  associe le polynome d'interpolation  $P_n$  est un opérateur linéaire et continu dont la norme est définie par (I - 2 - 8) :

$$\|L_n\| = \max_{x \in [a,b]} \sum_{i=1}^n |L_i^n(x)|$$

(I - 2 - 8)

On sait d'après le théorème de BERNSTEIN-FABER (P.J. LAURENT , 1972) que :

$$\|L_n\| \gg \frac{2}{\pi^2} \cdot \text{Log}(n) + b(n) \quad (\text{I} - 2 - 9)$$

où  $b$  est une fonction bornée.

Par conséquent, quelles que soient les abscisses  $x_i$ ,  $\forall i = 1, \dots, n$  et  $n = 0, 1, 2, \dots$ , il existe au moins une fonction  $f \in C_\infty$  telle que  $L_n(f)$  ne converge pas vers  $f$  et ne soit

pas bornée, de sorte qu'on ne peut affirmer la convergence uniforme de  $P_n$  vers  $f$ .

- PROBLEME D'INSTABILITE

En général les valeurs  $f(x_i)$  aux noeuds  $x_i$  de l'intervalle  $[a, b]$  sont des valeurs connues avec incertitude.

Supposons  $f(x)$  donnée aux noeuds  $x_i$  du réseau par la relation :

$$f(x_i) = \tilde{y}_i + \varepsilon_i \quad \text{avec} \quad \|\varepsilon_i\| \leq \varepsilon \quad (I - 2 - 10)$$

où  $\varepsilon$  désigne l'erreur d'incertitude.

Si pour un  $x$  donné, appartenant à  $[x_1, x_n]$ , (par exemple :  $x = \sqrt{7}$ , qui ne possède pas d'expression décimale ou binaire exacte), on souhaite calculer la quantité :

$$L_n[f](x) = \sum_{i=1}^n f(x_i) \cdot L_i^n(x)$$

où les  $L_i^n(x)$ ,  $\forall i = 1, \dots, n$  sont les polynomes d'approximation de LAGRANGE, alors on montre (Encyclopédia Universalis) que la seule majoration utilisable est :

$$\left| f(x) - \sum_{i=1}^n \tilde{y}_i L_i^n(\tilde{x}) \right| \leq |f(x) - f(\tilde{x})|$$
$$+ \left| f(\tilde{x}) - \sum_{i=1}^n f(x_i) L_i^n(\tilde{x}) \right| + \sum_{i=1}^n |f(x_i) - \tilde{y}_i| \cdot |L_i^n(\tilde{x})|$$

(I - 2 - 11)

où  $\tilde{x}$  est la représentation de  $x$  dans la mémoire du calculateur.

Compte-tenu des relations suivantes (I - 2 - 12) :

$$\|L_n\| = \max_{x \in [a, b]} \sum_{i=1}^n |L_i^n(x)|$$

$$|f(x_i) - \tilde{y}_i| = |\varepsilon_i| \leq \varepsilon$$

(I - 2 - 12)

Le dernier terme du deuxième membre de (I - 2 - 11)

peut s'écrire :

$$\sum_{i=1}^n |f(x_i) - \tilde{y}_i| \cdot |L_i^n(x)| = \sum_{i=1}^n |\varepsilon_i| \cdot |L_i^n(x)|$$

$$\leq \varepsilon \cdot \|L_n\|$$

De sorte que si  $\|L_n\|$  n'est pas majorée (ce qui est le cas pour les polynômes de LAGRANGE), l'instabilité de l'interpolation résulte de la propagation des erreurs d'incertitude

$\varepsilon_i$  (I - 2 - 12). Il résulte alors du théorème de BERNSTEIN-FABER que la suite d'opérateurs de LAGRANGE  $L_n$  est instable lorsque le nombre  $n$  de points d'interpolation augmente.

Tous ces éléments nous amènent à chercher d'autres modèles mathématiques permettant de définir des formes complexes à partir d'un nombre important de données.

Deux algorithmes répondent aux spécifications exigées précédemment, et sont déjà utilisés dans la pratique.

Ce sont les techniques qui utilisent les fonctions "splines" et les fonctions de "BEZIER" d'allure polynomiale.

Ces fonctions sont en particulier, incorporées dans des logiciels de C.A.O. tels qu'EUCLID (Société Matra Datavision) UNISURF (R.N.U RENAULT) pour les fonctions de BEZIER et CADAM (Société LOOKEED) pour les fonctions splines.

De manière à montrer l'analogie entre la méthode proposée et les algorithmes précédents déjà constitués et utilisés, nous proposons de rappeler leurs principes de base, ainsi que les principaux résultats auxquels ils aboutissent. Cette revue nous permettra de situer la méthode que nous proposons par rapport aux procédés existants.

I - 3 : ALGORITHME DES FONCTIONS DE "BEZIER": SYSTEME "UNISURF" :

(BEZIER , 1968 - RIAUX , 1980)

Il s'agit de concevoir des formes géométriques complexes issues de considérations esthétiques, à partir de données telles que des points de passage et de forme à réaliser, ainsi que de conditions de tangence en certains de ces points.



Ces conditions géométriques sont par exemple celles qui permettent à la pièce à concevoir (aile de voiture par exemple) de s'insérer sans discontinuité ni rupture de profil à l'intérieur d'un ensemble. Par ailleurs, le concepteur utilisant le système "UNISURF" est libre de donner la forme qui lui plait entre les conditions aux limites précédemment imposées à la courbe.

Cette méthode est élaborée à partir de la définition d'une relation bi-univoque entre une courbe dite "étalon" et son "polygone" correspondant. Les courbes générées sont engendrées par un point  $P(u)$  de l'espace géométrique à deux ou trois dimensions décrivant la trajectoire définie par la relation suivante (I - 3 - 1) :

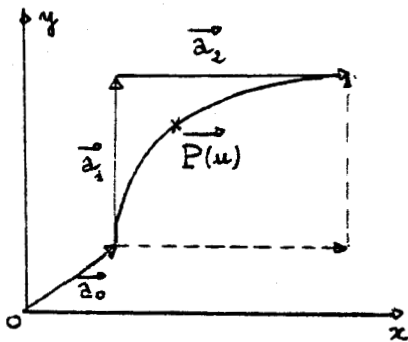
$$\vec{P}(u) = \vec{a}_0 + \sum_{i=1}^n f_n^i(u) \cdot \vec{a}_i \quad ; \quad u \in [0,1]$$

(I - 3 - 1)

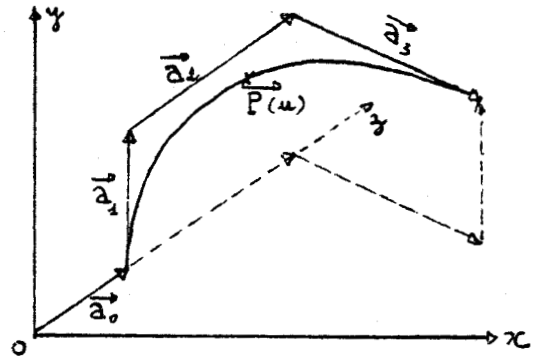
où  $\vec{a}_i$  sont les côtés du polygone associé à l'arc de courbe (défini dans l'espace géométrique à 2 ou 3 dimensions),  $u$  est un paramètre qui définit la position d'un point de l'arc de courbe.

$f_n^i(u)$  sont des fonctions paramétrées par  $u$ ,  
et construites à partir des conditions géométriques suivantes ,  
imposées à la courbe étalon (cf I - 3 - 2, I - 3 - 3, I - 3 - 4).

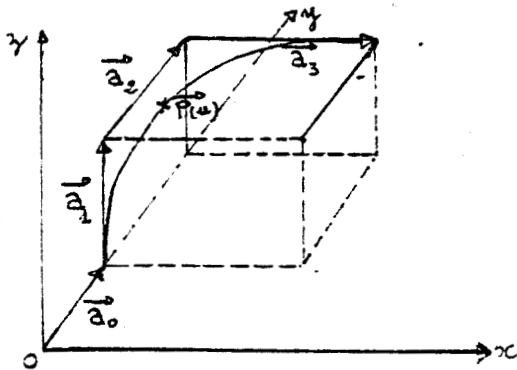
1 - La courbe étalon est inscrite à l'intérieur  
d'une figure géométrique bâtie sur les axes d'un système orthogonal  
de référence. Elle est définie par son polygone correspondant à  
 $n$  côtés, et réciproquement (cf Fig - 7) :



POLYgone A 2 COTES  
DANS  $\mathbb{R}^2$ .



POLYgone A 3 COTES  
DANS  $\mathbb{R}^3$



POLYgone A 3 COTES

Fig-7 : Exemple de courbes  
étalons et de  
polygones associés  
à 2 et 3 côtés dans  
le plan et l'espace.

2 - Le point initial et le point final de l'arc de courbe , correspondent respectivement à la valeur 0 et 1 du paramètre u, c'est-à-dire :

$$\vec{P}(0) = \vec{a}_0 \longrightarrow f_n^i(0) = 0 ; \forall i=1, \dots, n$$

$$\vec{P}(1) = \sum_{i=0}^n \vec{a}_i \cdot f_n^i(1) ; \forall i=1, \dots, n$$

(I - 3 - 2)

3 - Les tangentes aux points initial et final sont respectivement parallèles aux vecteurs  $\vec{a}_1$  et  $\vec{a}_n$ .

Il vient :

$$f_n'(0) \neq 0 ; f_n^i(0) = 0 ; \forall i=2, \dots, n$$

$$f_n^n(1) \neq 0 ; f_n^i(1) = 0 ; \forall i=1, \dots, n-1$$

(I - 3 - 3)

4 - Les plans osculateurs aux points initial et final sont respectivement parallèles aux plans engendrés par les vecteurs  $(\vec{a}_1, \vec{a}_2)$  et  $(\vec{a}_{n-1}, \vec{a}_n)$ . Il s'en suit que :

$$f_n^{i1}(0) \neq 0 \quad ; \quad f_n^{i2}(0) \neq 0 \quad ; \quad f_n^{ii}(0) = 0 \quad ; \quad \forall i=3, \dots, n$$

$$f_n^{in}(1) \neq 0 \quad ; \quad f_n^{in}(1) \neq 0 \quad ; \quad f_n^{ii}(1) = 0 \quad ; \quad \forall i= \dots, n-2$$

(I - 3 - 4)

Il est alors aisé d'extraire du système d'équations précédent une famille de polynomes satisfaisant les conditions (I - 3 - 2 , I - 3 - 3 , I - 3 - 4).

Une suite de polynomes peut être par exemple :

$$f_n^i(u) = \frac{-(-u)^i}{(i-)} \cdot \frac{d^{(i-1)}}{du^{(i-1)}} (\Phi_n)$$

(I - 3 - 5)

avec  $u \in [0, 1]$  et  $\Phi_n = \frac{1 - (1-u)^n}{u}$

Ces fonctions sont dites fonctions de "BEZIER".  
Elles constituent une base pour l'espace des fonctions  $f$  définies sur l'intervalle  $[0, 1]$  (cf LAURENT, 1972) et par extension sur tout intervalle fermé borné.

EXEMPLE DE FONCTIONS DE DURAND :

$n = 2$

$$f_2^1(u) = 2u - u^2$$

$$f_2^2(u) = u^2$$

$n = 3$

$$f_3^1(u) = 3u - 3u^2 + u^3$$

$$f_3^2(u) = 3u^2 - 2u^3$$

$$f_3^3(u) = u^3$$

Exemple de polygones et d'arcs de courbes associés (cf Fig - 8).

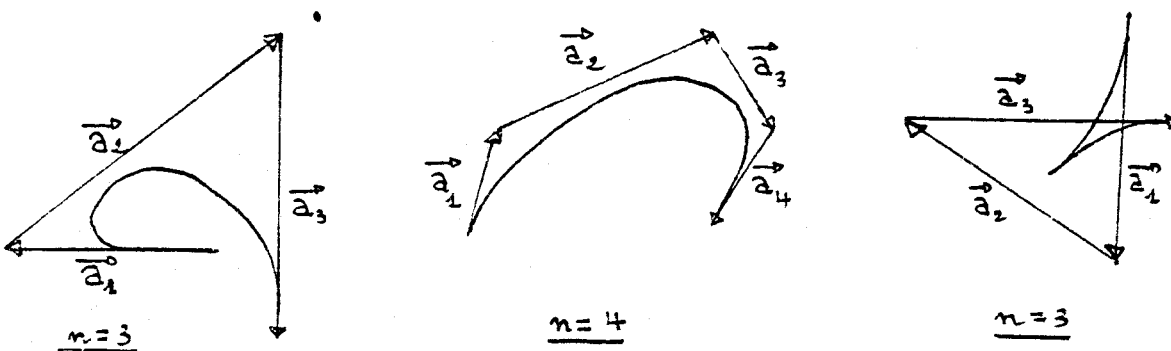


Fig - 8

Ces fonctions permettent la construction de formes géométriques complexes, de la manière suivante .

puisque à toute suite de vecteurs  $\{a_i\}_{i=1}^n$ , il correspond un arc de courbe défini par la relation :

$$\vec{P}(u) = \vec{a}_0 + \sum_{i=1}^n f_n^i(u) \cdot \vec{a}_i$$

et réciproquement, si on déforme le polygone constitué par les vecteurs  $\vec{a}_i$  (par déplacement d'un ou plusieurs sommets), il résulte alors une déformation linéaire de l'arc de courbe correspondant (Fig - 9).

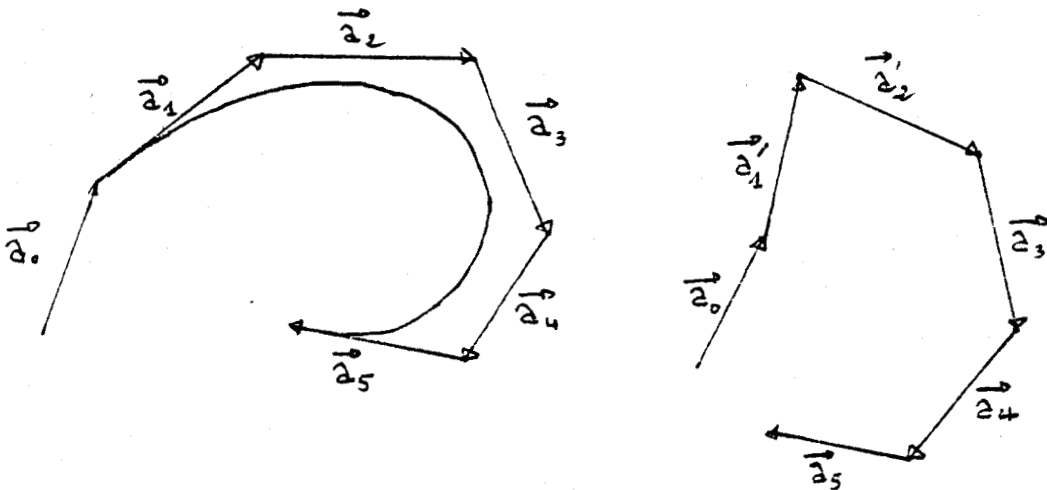


Fig - 9 : DEFORMATION D'UN POLYGONE PAR DEPLACEMENT DES SOMMETS

La conception de formes géométriques plus ou moins complexes est alors réalisée à partir d'un choix initial de sommets d'un polygone puis par déplacement d'un certain nombre d'entre eux, jusqu'à l'obtention de la forme désirée.

Cette procédure se généralise aisément au cas des forme géométriques définies à partir de plusieurs arcs de courbes. Dans ce cas, chaque arc est défini par la donnée des sommets du polygone générateur associé, vérifiant les conditions de raccordement aux extrémités. Par exemple, si la forme à concevoir est continue, il suffit de mémoriser successivement les différents arcs de courbe (cf Fig - 10), puis de choisir judicieusement les extrémités du polygone suivant (contraintes de tangence).

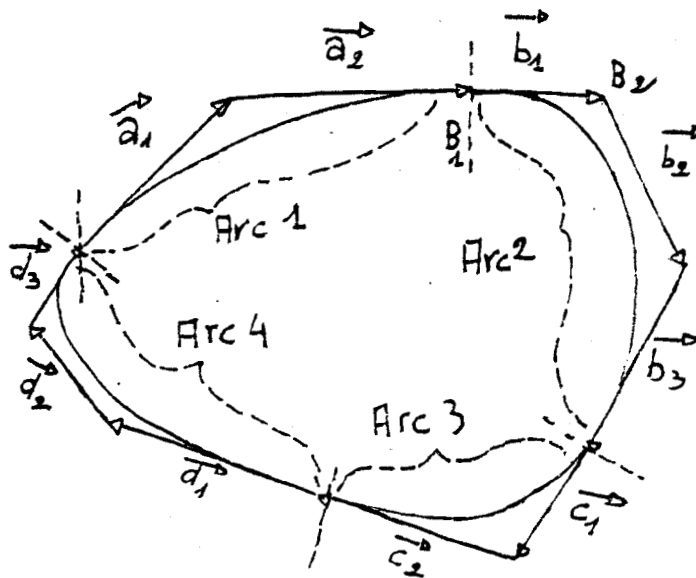


Fig - 10

La continuité jusqu'à l'ordre 1 de la courbe est assurée si le premier polygone intermédiaire est co-linéaire au dernier côté du polygone précédent. Ceci est équivalent à choisir des tangentes continues aux points de raccordement (cf Fig - 10 : Position du point B par rapport à B1).

APPLICATION DE LA METHODE DE BEZIER A LA  
GENERATION DES COURBES :

Avant d'explicitier analytiquement les équations de génération de courbes et surfaces par cette méthode, nous pouvons constater qu'au niveau de la fabrication, et de manière à rendre l'algorithme de BEZIER rapidement exécutable, on substitue aux vecteurs  $\vec{a}_i$  d'autres vecteurs  $\vec{b}_i$ . Ces derniers permettent de minimiser le volume de calcul à effectuer par le processeur engendrant la courbe grâce à l'utilisation de techniques rapides de programmation.

Ces vecteurs  $\vec{b}_j$  s'expriment en fonction des vecteurs  $\vec{a}_j$  par la relation (I - 3 - 6).

$$\vec{b}_j = C_n^j \cdot \left[ C_{j-1}^{j-1} \cdot \vec{a}_j - C_{j-1}^{j-2} \cdot \vec{a}_{j-1} + \dots + C_{j-1}^0 (-1)^{j+1} \cdot \vec{a}_1 \right]$$

$$\forall j = 1, \dots, n$$

(I - 3 - 6)

$$\text{avec } \vec{b}_0 = \vec{a}_0 \quad \text{et } C_n^j = \frac{n!}{j! (n-j)!}$$



Cette expression (I - 3 - 6) peut s'écrire sous la forme condensée suivante (I - 3 - 7) :

$$\begin{aligned} \vec{b}_j &= c_n^j \cdot \vec{\beta}_j \quad ; \quad \forall j=0, \dots, n \\ \vec{b}_0 &= \vec{a}_0 \end{aligned} \quad (\text{I - 3 - 7})$$

En identifiant ces deux relations (I - 3 - 6 , I - 3 - 7) on obtient les vecteurs  $\vec{b}_j$  en fonction des vecteurs  $\vec{a}_i$ .

$$\begin{aligned} \vec{\beta}_1 &= \vec{a}_1 \\ \vec{\beta}_2 &= \vec{a}_2 - \vec{a}_1 \\ \vec{\beta}_3 &= \vec{a}_3 - 2\vec{a}_2 + \vec{a}_1 \\ \vec{\beta}_n &= \vec{a}_n - c_{n-1}^{n-2} \vec{a}_{n-1} + \dots + c_{n-1}^0 (-1)^{n+1} \vec{a}_1 \end{aligned} \quad (\text{I - 3 - 8})$$

Le point courant s'exprime alors de manière immédiate, en fonction de ces vecteurs, par la relation suivante dont le calcul se ramène à un schéma de HORNER.

$$\vec{P}(u) = \vec{b}_0 + \sum_{j=1}^n u^j \cdot \vec{b}_j \quad (\text{I - 3 - 9})$$

De même la dérivée du point courant s'exprime par (I - 3 - 10) :

$$\vec{P}'(u) = \sum_{j=1}^n j \cdot u^{j-1} \cdot \vec{b}_j \quad (\text{I - 3 - 10})$$

Ces deux dernières relations (I - 3 - 9 , I - 3 - 10) vont permettre de résoudre le problème initialement posé.

En effet, la conception d'une courbe gauche peut aboutir partiellement à traiter un problème d'interpolation et il est fréquent d'assujettir un arc de courbe à passer par un certain nombre de points, avec, éventuellement, des conditions de tangences.

Dans ce cas, il s'agit de déterminer une courbe UNISURF, satisfaisant les contraintes précédents ou, ce qui est équivalent, à rechercher les vecteurs du polygone correspondant.

Les points de passage étant définis dans un repère cartésien par la relation (I - 3 - 11) :

$$P_i = \left\{ (x_i, y_i) ; \forall i=1, \dots, n+1 \right\} \quad (\text{I - 3 - 11})$$

on convient, pour simplifier le problème, de leur attribuer les valeurs  $u_k$  du paramètre  $u$  de la manière suivante (I - 3 - 12)

$$\forall k = 0, \dots, n \quad ; \quad u_k = \left\{ \frac{k}{n} \right\} \quad (\text{I - 3 - 12})$$

où  $u_0 = 0$  et  $u_n = 1$  correspondant aux extrémités du réseau  $\{x_i, y_i\}$  défini précédemment.

De manière à étudier conjointement les problèmes de conception et d'interpolation, nous convenons de restreindre l'exposé au cas où les contraintes de tangentes ne concernent que les extrémités du réseau.

Les vecteurs  $\vec{b}_j$  sont alors solution d'un système d'équations linéaires dont l'ordre dépend du nombre de contraintes du type précédemment évoqué, et que nous nous proposons de déterminer.

Soit  $(n + 1)$  points de passage incluant les extrémités du réseau tel que à tout point de passage de coordonnées

$$\left\{ x_i, y_i \right\}_{i=1}^n \quad \text{on associe le paramètre } u_k = \frac{k}{n} \quad ; \quad \forall k=0, \dots, n.$$

Supposons en outre, que les extrémités  $(x_0, y_0)$  et  $(x_n, y_n)$  soient assujetties à des contraintes de tangence, si  $(m+1)$  est le nombre de sommets du polygone à  $m$  côtés associé à la

courbe à générer, alors  $m$  peut être déterminé de la manière suivante, Compte-tenu des données, le système (I-3-9, I-3-10) s'écrit :

$$\begin{aligned} \vec{P}(u_k) &= \vec{b}_0 + \sum_{j=1}^m u_k^j \cdot \vec{b}_j \quad ; \quad \forall k=0, \dots, n \\ \vec{P}'(u_1) &= \sum_{j=1}^m j \cdot u_1^{j-1} \vec{b}_j \quad ; \quad \forall l=1, \dots, n \end{aligned} \quad (I - 3 - 13)$$

$$(I - 3 - 14)$$

Il s'en suit les relations suivantes :

\* (n-1) relations du type

$$\begin{aligned} \vec{P}(u_k) &= \vec{P}_{u_k} \quad ; \quad \forall k=1, \dots, n-1 \quad ; \\ u_k &= \left\{ \begin{array}{c} k \\ n \end{array} \right\} \quad ; \quad (I - 3 - 15) \end{aligned}$$

entre les  $(m + 1)$  inconnues  $\vec{b}_0, \vec{b}_1, \dots, \vec{b}_m$ . Ces relations sont issues de (I - 3 - 13) et sont associées aux contraintes de points de passage intermédiaires.

\* 2 relations du type

$$\vec{P}(0) = \vec{P}_0 = \vec{b}_0$$

$$\vec{P}'(1) = \vec{P}'_1 = \vec{b}_0 + \vec{b}_1 + \dots + \vec{b}_m$$

issues de (I - 3 - 13) et sont associées aux contraintes de points de passage initial et final.

\* 2 relations du type

$$\vec{P}'(0) = \vec{P}'_0 = \vec{b}_1$$

$$\vec{P}'(1) = \vec{P}'_1 = \vec{b}_1 + 2\vec{b}_2 + \dots + m \cdot \vec{b}_m$$

(I - 3 - 17)

De ces  $(n-1 + 2 + 2) = (n + 3)$  équations, on peut extraire  $(n + 1)$  équations entre les  $(m-1)$  inconnues  $\vec{b}_2, \vec{b}_3, \dots, \vec{b}_m$ .

Pour que ce système admette une solution unique, il faut que le nombre de côtés du polygone générateur satisfasse la relation (I - 3 - 18) suivante :

$$m = n + 2$$

(I - 3 - 18)

EXEMPLE D'APPLICATION

$n + 1 = 6 \implies m = 7$  et le système linéaire est à 6 équations à 6 inconnues  $\vec{b}_2, \vec{b}_3, \dots, \vec{b}_7$  : (I - 3 - 19)

$$\begin{bmatrix}
 1 & 1 & 1 & 1 & 1 & 1 \\
 2 & 3 & 4 & 5 & 6 & 7 \\
 u_1^2 & u_1^3 & u_1^4 & u_1^5 & u_1^6 & u_1^7 \\
 u_2^2 & u_2^3 & u_2^4 & u_2^5 & u_2^6 & u_2^7 \\
 u_3^2 & u_3^3 & u_3^4 & u_3^5 & u_3^6 & u_3^7 \\
 u_4^2 & u_4^3 & u_4^4 & u_4^5 & u_4^6 & u_4^7
 \end{bmatrix}
 \begin{bmatrix}
 \vec{b}_2 \\
 \vec{b}_3 \\
 \vec{b}_4 \\
 \vec{b}_5 \\
 \vec{b}_6 \\
 \vec{b}_7
 \end{bmatrix}$$

$$= \begin{bmatrix}
 \vec{P}_1 & - & \vec{P}_0 & - & \vec{P}'_0 \\
 \vec{P}'_1 & - & \vec{P}'_0 & & \\
 \vec{P}_{u_1} & - & \vec{P}_0 & - & u_1 \cdot \vec{P}'_0 \\
 \vec{P}_{u_2} & - & \vec{P}_0 & - & u_2 \cdot \vec{P}'_0 \\
 \vec{P}_{u_3} & - & \vec{P}_0 & - & u_3 \cdot \vec{P}'_0 \\
 \vec{P}_{u_4} & - & \vec{P}_0 & - & u_4 \cdot \vec{P}'_0
 \end{bmatrix}$$

$$\text{avec : } \vec{b}_0 = \vec{P}_0 \text{ et } \vec{b}_1 = \vec{P}_1 ; u_1 = \frac{1}{5}, u_2 = \frac{2}{5}, u_3 = \frac{3}{5},$$

$$u_4 = \frac{4}{5}$$

#### REMARQUE

Il est possible d'influer sur la forme de la courbe en attribuant à certains points de passage intermédiaires des valeurs de paramètre différentes de celles qui constituent la suite  $u_k = \left\{ \frac{k}{n} \right\}$  ;  $k = 1, \dots, n-1$ .

#### LIEN AVEC LA NOTION D'OBSERVABILITE

Inversement, pour  $n$  donné, il est possible de déterminer le nombre de contraintes de points de passage et/ou de tangence nécessaire pour que le système précédent (I - 3 - 9 , I - 3 - 10) admette une solution unique.

Ce cas particulier permet une liaison directe avec la notion d'observabilité introduite dans l'étude des systèmes dynamiques (GILLES , 1975).

Nous proposons d'expliciter la solution de ce problème sur un exemple simple ( $n = 3$ ) en lui substituant la recherche de conditions d'observabilité du système dynamique correspondant au système d'équations (I - 3 - 20) :

$$\vec{P}(u) = \vec{a}_0 + \sum_{i=1}^n f_n^i(u) \cdot \vec{a}_i \quad (\text{I - 3 - 20})$$

En développant les expressions de  $f_n^i(u)$  pour  $n = 3$ , (I - 3 - 20) s'écrit (I - 3 - 21) :

$$x(u) = a_0^1 + a_1^1 (u^3 - 3u^2 + 3u) + a_2^1 (-2u^3 + 3u^2) + a_3^1 u^3$$

$$y(u) = a_0^2 + a_1^2 (u^3 - 3u^2 + 3u) + a_2^2 (-2u^3 + 3u^2) + a_3^2 u^3$$

(I - 3 - 21)

$$\text{avec } \vec{P}(u) = \begin{cases} x(u) \\ y(u) \end{cases} ; \quad \vec{a}_i = \begin{cases} a_i^1 \\ a_i^2 \end{cases}$$

En ordonnant suivant les puissances décroissantes de  $u$ , il vient les expressions suivantes (I - 3 - 22) :



$$x(u) = (a_1^1 - 2 a_2^1 + a_3^1) u^3 + (-3a_1^1 + 3a_2^1) u^2 + 3a_1^1 u + a_0^1$$

$$y(u) = (a_1^2 - 2a_2^2 + a_3^2) u^3 + (-3a_1^2 + 3a_2^2) u^2 + 3 a_1^2 u + a_0^2 \quad (I-3-22)$$

Les inconnues étant les vecteurs  $a_i^k$  ;  
 $\forall i=0, \dots, 3$  ;  $\forall k=1, 2$  ; si on pose ( I - 3 - 23)

$$b_0^k = a_0^k \quad ; \quad b_1^k = 3 a_1^k \quad ;$$

(I - 3 - 23)

$$b_2^k = -3a_1^k + 3 a_2^k \quad ; \quad b_3^k = a_1^k - 2 a_2^k + a_3^k$$

et  $z(u) \begin{cases} x(u) & \text{pour } k = 1 \\ y(u) & \text{pour } k = 2 \end{cases}$  alors le système (I - 3 - 23) peut

se mettre sous la forme du système dynamique suivant :

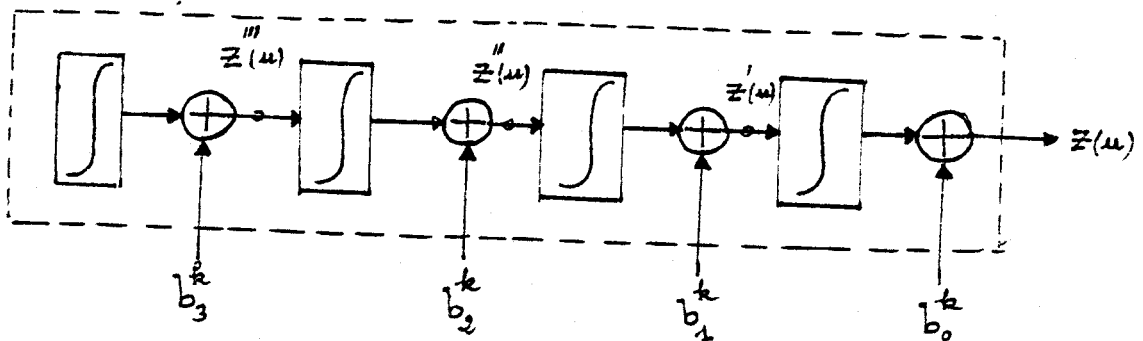


Fig-II

Ce système est complètement observable (GILLES , 1975) par sa sortie  $z(u)$ .

En d'autres termes la donnée de quatre points de la trajectoire  $z(u)$ ,  $u$  appartenant à l'intervalle  $[0,1]$  , permet de calculer les conditions initiales (I - 3 - 24) :

$$\left[ z(0) , z'(0) , z''(0) , z'''(0) \right]^T = \left[ b_0^k , b_1^k , b_2^k , b_3^k \right]^T$$

(I - 3 - 24)

par la résolution d'un système d'équations linéaires.

Cette remarque concernant l'analogie entre les fonctions d'interpolation de BEZIER (système UNISURF) et sa représentation par un système dynamique complètement observable, a déjà été utilisée pour l'élaboration d'un système analogique générateur de trajectoires polynomiales (T. BENNANI , 1975). Cette interprétation d'une courbe d'interpolation comme trajectoire générée par un système dynamique est mise à profit au Chapitre II, par l'introduction de la notion de processus interpolateur pour générer les fonctions splines d'interpolation d'ordre  $q$ .

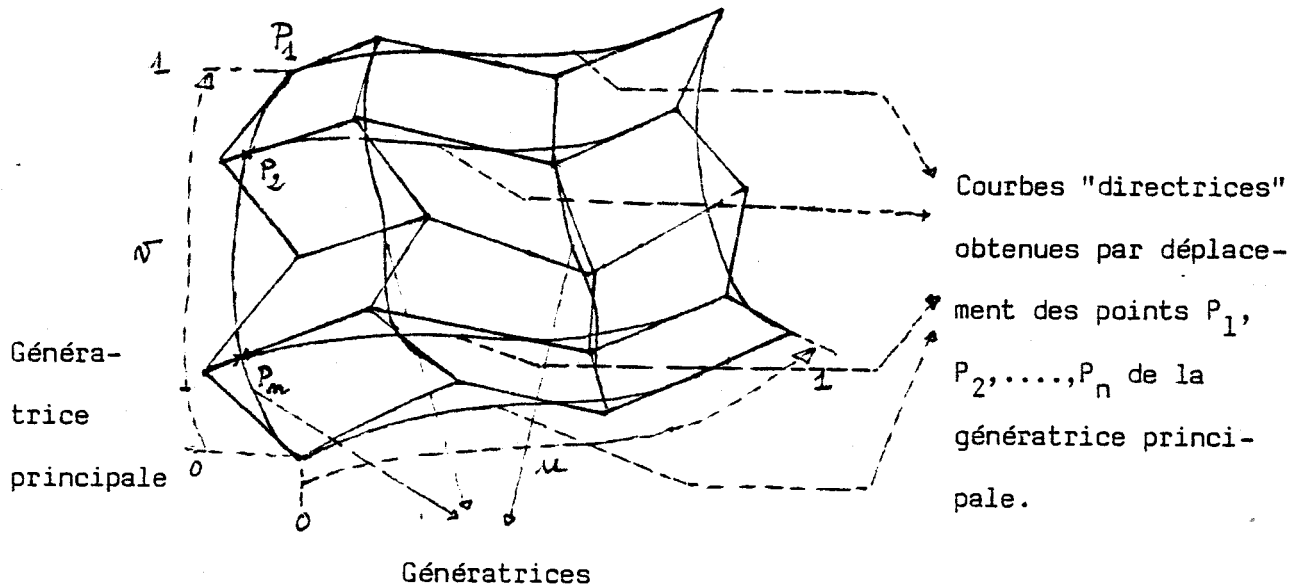
#### GENERALISATION AUX SURFACES :

Les courbes gauches étant définies par leur polygone caractéristique, on généralise la procédure précédente

aux surfaces en concevant celles-ci à partir de passes successives.

Chaque passe est une courbe UNISURF notée "génératrice" définie par un polygone. Pour chaque valeur du paramètre  $u$ , le point correspondant des génératrices se déplace suivant une autre courbe UNISURF dite "directrice", elle même définie par un polygone caractéristique. (RIAUX , 1981) (Fig - 12).

Fig-I2



On conçoit alors la nécessité de deux paramètres  $u$  et  $v$  pour décrire une surface. Ces paramètres sont assignés respectivement aux directrices et génératrices, de sorte qu'une surface est conçue comme étant engendrée par le point  $\vec{P}(u, v)$  défini par la relation (I - 3 - 25) :

$$\vec{P}(u,v) = \vec{a}_0 + \sum_{i=1}^n \vec{a}_{i1} f_n^i(u) + \sum_{j=1}^m \vec{b}_{j1} f_m^j(v)$$

$$+ \sum_{i,j=1}^{n,m} (\vec{a}_{i,j+1} - \vec{a}_{i,j}) \cdot f_n^i(u) \cdot f_m^j(v) \quad (I - 3 - 25)$$

où  $f_n^i(u)$  et  $f_m^j(v)$  sont les fonctions de Durand, relatives aux paramètres  $u$  et  $v$ .

Les vecteurs  $\vec{a}_i$  et  $\vec{b}_j$  correspondent respectivement aux polygones directeurs et générateurs ( cf Fig - 13)

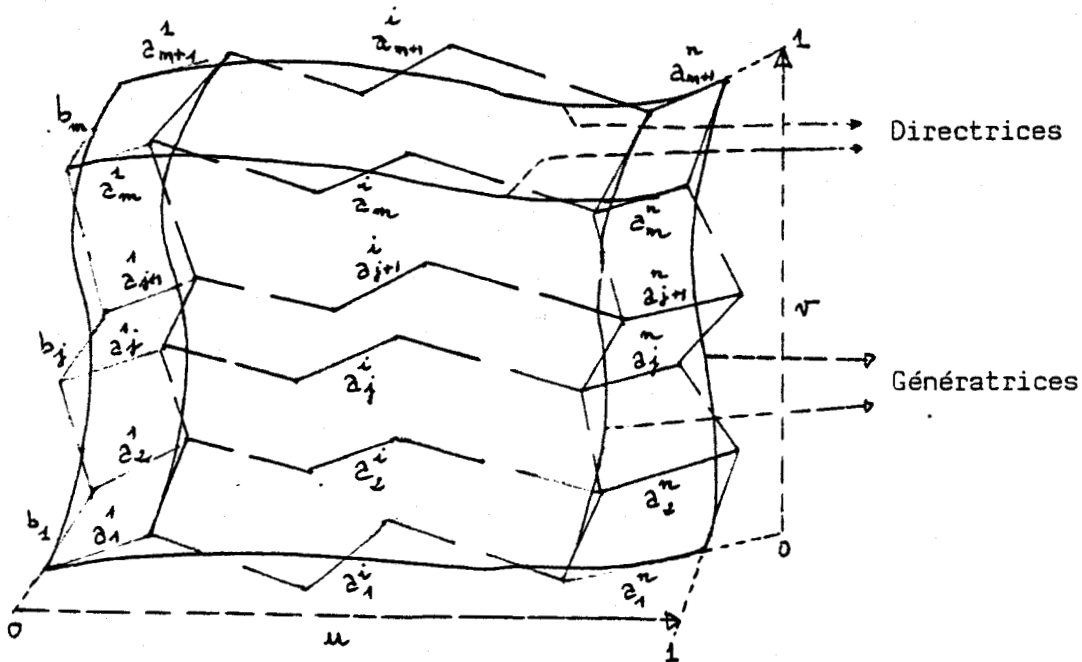


Fig - 13

Le calcul du point courant s'effectue alors en deux temps.

1 - On détermine un point de paramètre  $u$  sur chaque directrice,

2 - A partir du polygone de la génératrice  $u = \text{cste}$  ainsi constituée, on détermine la génératrice de paramètre  $v$ .

Comme pour les courbes unidimensionnelles, on introduit des vecteurs  $\vec{b}_{\text{dir}}$  et  $\vec{b}_{\text{gén.}}$ .

Pour une surface le point courant  $\vec{P}(u,v)$  peut être défini par la relation suivante (I - 3 - 26) obtenue à partir de (I - 3 - 25) dans laquelle on a développé les fonctions  $f_n^i(u)$ ,  $f_m^j(v)$  et groupé les termes  $u^i v^j$  :

$$\vec{P}(u,v) = \sum_{i,j=1}^{n,m} \vec{C}_{ij} \cdot u^{i-1} \cdot v^{j-1} \quad (\text{I} - 3 - 26)$$

Cette dernière relation peut s'écrire sous une forme plus simple, permettant de restreindre le volume de calcul nécessaire pour calculer les vecteurs  $\vec{b}_{\text{dir}}$  et  $\vec{b}_{\text{gén.}}$  (I - 3 - 27) :

$$\vec{P}(u,v) = \left[ 1, u, u^2, \dots, u^n \right] \cdot \left[ \vec{C}_{ij} \right] \cdot \left[ 1, v, v^2, \dots, v^m \right]^T$$

(I - 3 - 27)

Pour une génératrice ( $u=cste$ ) il y a lieu d'effectuer préalablement le calcul suivant : [I - 3 - 28] avant de calculer le point courant.

$$\vec{b}_{\text{géné } j} = \left[ 1, u, u^2, \dots, u^n \right] \cdot \left[ \vec{C}_{ij} \right] \quad (\text{I} - 3 - 28)$$

Ce dernier s'obtient à partir de la relation  
(I - 3 - 29)

$$\vec{P}(u=cste, v) = \left[ \vec{b}_{\text{géné } j} \right] \cdot \left[ 1, v, v^2, \dots, v^m \right]^T$$

$\forall v$  variant de 0 à 1

La détermination de la matrice  $\left[ \vec{C}_{ij} \right]$  utilise  $(n * m)$  fois l'algorithme utilisé pour les courbes en calculant chaque fois les points intermédiaires qui constituent les polygones des génératrices  $u=cste$  et des directrices  $v=cste$ .

CONCLUSION :

La méthode de BEZIER se justifie bien comme méthode de conception de formes géométriques et permet le passage aisé de l'esquisse au dessin. Le caractère pratique de cet algorithme est lié à la rapidité du traitement et la simplicité de la manipulation des formes engendrées. Cet aspect est mis en relief par les remarques suivantes (a) - (b) - (c) :

a) Les relations entre la modification des points du polygone et la modification induite de la courbe sont relativement faciles à apprécier avec un peu d'expérience.

b) La courbe est indépendante du système de coordonnées utilisé pour décrire le polynôme.

c) La rapidité de calcul résulte de l'utilisation d'une base de fonctions polynomiales.

Alors que cette méthode, typiquement adaptée aux problèmes de conception peut être étendue pour traiter les problèmes d'interpolation, il apparaît difficile de l'utiliser dans le problème d'agrégation de données (cf. Introduction Générale). La recherche des sommets du/des polygone(s) associé(s) à la forme géométrique à reconstituer, nécessiterait la résolution d'un système d'équations linéaires d'ordre élevé.

I - 4 : ALGORITHME DES FONCTIONS SPLINES :

L'interpolation par des fonctions splines est une méthode généralisant un procédé graphique utilisé par les dessinateurs.

Etant donné un réseau de points  $(x_i, y_i, i=1, n)$  dans le plan, on utilise une tige flexible qui passe par ces points. La courbe obtenue est continue jusqu'à un ordre élevé et sa forme est nécessairement régulière (non oscillante entre les points d'interpolation) étant donné la plasticité de cette "règle molle".

L'utilisation de ce type de règle molle étant délicate dans les unités centrales des ordinateurs, on a cherché à reconstruire ce type de courbe mathématiquement et on a aboutit ainsi aux fonctions splines qui minimisent des fonctionnelles de type quadratique des dérivées successives de la courbe. Nous proposons d'abord de définir de manière axiomatique les fonctions splines avant d'explicitier trois méthodes usuelles permettant leur calcul.

I - 4 - 1 : PROPRIETES DES FONCTIONS SPLINES

D'INTERPOLATION D'ORDRE  $q$  :

Soit un réseau de points  $(x_i, y_i, i=1, \dots, n)$  de  $\mathbb{R}$ . Les abscisses des points successifs sont ordonnées par la relation suivante :



$$x_1 < x_2 < \dots < x_n$$

Une fonction spline d'ordre  $q$  est une fonction  $f(x)$  ;  $x \in [x_1, x_n]$  satisfaisant les propriétés énoncées ci-dessous.

P-1)  $f(x)$  est de classe  $C^{2q-2}$   $[x_1, x_n]$

P-2) Dans chaque intervalle  $[x_i, x_{i+1}]$ ,  $f(x)$  est un polynôme de degré  $(2q-1)$  d'expression :

$$\forall x \in [x_i, x_{i+1}] ; f(x) = \sum_{l=0}^{2q-1} a_l \cdot (x_{i+1}-x)^l$$

P-3) En tout point  $x_i$  ;  $i=1, \dots, n$  ; on a la propriété d'interpolation :

$$f(x_i) = y_i ; \forall i=1, \dots, n$$

P-4)  $f(x)$  vérifie les conditions aux limites

$$\left\{ f^{(j)}(x_1^-) = f^{(j)}(x_n^+) = 0 ; \forall j = q+1, \dots, 2q-1 \right\}$$

P-5)  $f(x)$  minimise la fonctionnelle quadratique :

$$(f(x)) = \int_{x_1}^{x_n} \{ f^{(q)}(x) \}^2 \cdot dx ; \forall x \in [x_1, x_n]$$

L'existence et l'unicité de la solution sont assurées de manière constructive pour les méthodes connues de calcul de  $f$  dont nous proposons quelques brefs rappels.

I - 4 - 2 : METHODE D'INTEGRATION POUR LES  
FONCTIONS SPLINES D'ORDRE 2 :

(MARCHOUX , 1980 - ATTEIA , 1966)

Compte-tenu de  $(P_2)$ , dans chaque intervalle  $(x_i, x_{i+1})$ , le polynôme d'interpolation s'écrit :

$$f_{i+1}(x) = a_0 + a_1^i \cdot (x_{i+1} - x) + a_2^i \cdot (x_{i+1} - x)^2 + a_3^i \cdot (x_{i+1} - x)^3 \quad (I - 4 - 1)$$

En dérivant deux fois les deux membres de cette relation et en posant :

$$h_{i+1} = x_{i+1} - x_i$$

$$m_i = f''_i(x_i)$$

(I - 4 - 2)

$$m_{i+1} = f''_i(x_{i+1})$$

Il vient les relations suivantes (I - 4 - 3) :

$$2a_2^i = m_{i+1}$$

(I - 4 - 3)

$$6a_3^i = \frac{m_i - m_{i+1}}{h_{i+1}}$$

$$f_i''(x) = m_i \frac{x_{i+1} - x}{b_{i+1}} + m_{i+1} \frac{x - x_i}{h_{i+1}}$$

En intégrant deux fois les deux membres de la dernière expression de (I - 4 - 3) et en calculant les constantes d'intégration à partir de  $(P_3)$ , il vient la relation de  $f(x)$  pour tout  $x \in [x_i, x_{i+1}]$  :

$$f(x) = \frac{m_i (x_{i+1} - x)^3}{6 \cdot h_{i+1}} + m_{i+1} \cdot \frac{(x - x_i)^3}{6 \cdot h_{i+1}} + \left[ y_i - \frac{m_i \cdot h_{i+1}^2}{6} \right] \cdot \frac{(x_{i+1} - x)}{h_{i+1}} + \left[ y_{i+1} - \frac{m_{i+1} \cdot h_{i+1}^2}{6} \right] \cdot \frac{(x - x_i)}{h_{i+1}} \quad (\text{I - 4 - 4})$$

La propriété de continuité  $(P1)$  nous fournit une relation de récurrence à partir de laquelle sont issues les inconnues  $m_i$  et  $m_{i+1}$  du problème ; il vient :

$$\forall i=1, \dots, n-2$$

$$\frac{h_{i+1}}{6} m_i + \frac{h_{i+1} + h_{i+2}}{3} m_{i+1} + \frac{h_{i+2}}{6} = \frac{y_{i+2} - y_{i+1}}{h_{i+2}} - \frac{y_{i+1} - y_i}{h_{i+1}}$$

(I - 4 - 5)

Cette dernière relation peut se mettre sous la forme matricielle suivante :

$$A \cdot M = H \cdot Y \quad (I - 4 - 6)$$

Où :

$$A = \begin{bmatrix} \frac{h_2 + h_3}{3} & \frac{h_3}{6} & 0 & \dots & 0 \\ \frac{h_3}{6} & \frac{h_3 + h_4}{3} & \frac{h_4}{6} & \dots & 0 \\ 0 & \frac{h_4}{6} & \frac{h_4 + h_5}{3} & \frac{h_5}{6} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & \dots & \dots & \frac{h_{n-1}}{6} & \frac{h_n + h_{n-1}}{3} \end{bmatrix} \quad M = \begin{bmatrix} m_2 \\ m_3 \\ \vdots \\ m_{n-1} \end{bmatrix}$$

(I - 4 - 7)

$$H = \begin{bmatrix} \frac{1}{h_2} & \left(-\frac{1}{h_2} - \frac{1}{h_3}\right) & \frac{1}{h_3} & 0 & \dots & 0 \\ 0 & \frac{1}{h_3} & \left(-\frac{1}{h_3} - \frac{1}{h_4}\right) & \frac{1}{h_4} & 0 & \dots & 0 \\ 0 & 0 & \frac{1}{h_4} & \left(-\frac{1}{h_4} - \frac{1}{h_5}\right) & \frac{1}{h_5} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \dots & \left(-\frac{1}{h_{n-1}} - \frac{1}{h_n}\right) & \frac{1}{h_n} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ \vdots \\ y_{n-1} \\ y_n \end{bmatrix}$$

(I - 4 - 8)

A est une matrice symétrique définie positive et non singulière, de sorte que le système (I - 4 - 6) admet une solution unique.

Cette méthode donne de très bons résultats et nécessite la résolution d'un système d'ordre (n-2), qui pour des valeurs élevées de n peut conduire à une instabilité numérique.

Par ailleurs, cette méthode s'avère lourde pour la génération des surfaces définies sur des réseaux rectangulaires. Dans ce contexte une surface est décrite par une fonction spline bicubique  $f(x,y)$  satisfaisant notamment la propriété d'interpolation suivante :

$$f(x_i, y_i) = Z_{ij} \quad ; \quad \begin{array}{l} \forall i=1, \dots, n \\ \forall j=1, \dots, m \end{array}$$

Cette fonction est construite à partir des fonctions splines cubiques unidimensionnelles  $f(x_i = \text{cste}, y)$  et  $f(x, y_j = \text{cste})$  respectivement définies sur les sous-réseaux de rang  $i$  suivants :  $\{ (y_j, Z_{ij}), j=1, \dots, m \text{ et } i \text{ fixé} \}$  et les sous-réseaux de rang  $j$  suivants :  $\{ (y_j, Z_{ij}), i=1, \dots, n \text{ et } j \text{ fixé} \}$ .

On montre [ MARCHOUK, 1981 ] , que les inconnues nécessaires pour déterminer la valeur de la fonction  $f$  en un point  $(x, y)$  du domaine où elle est définie, exigent qu'on résolve cinq fois un système matriciel équivalent à celui utilisé en dimension un.

$$A \cdot M = H \cdot Y \quad (I - 4 - 6)$$

D'où la nécessité d'utiliser une capacité mémoire élevée ( $4.n.m$  éléments de la mémoire interne d'un calculateur).

I - 4 - 3 : METHODE DE TRANSPORT DES RELATIONS :

( M. PETIT , 1971)

Cette méthode permet de calculer l'expression

analytique de la fonction spline d'interpolation  $f(x)$  d'ordre  $q$  sur un réseau  $\{x_i, y_i\} \forall i=1, \dots, n$  et satisfaisant les propriétés (cf § I - 4 - 1), à partir du vecteur  $Z(x)$  défini de la manière suivante (I - 4 - 7) :

$$Z^T(x) = [f(x), f'(x), \dots, f^{2q-2}(x), f^{2q-1}(x)]$$

$$Z(x) \in \mathbb{R}^{2q} \qquad \qquad \qquad (I - 4 - 7)$$

$f(x)$  est complètement déterminée sur  $[x_1, x_n]$ , si on connaît  $Z^T(x)$  en tout point  $x_i^+$  ;  $\forall i=1, \dots, n$ .

La méthode de calcul repose sur l'établissement en tout point  $x_i^+$  de  $2q$  équations linéaires satisfaites par les  $2q$  composantes de  $Z(x_i^+)$ . Ces relations sont issues du transport en  $x_i^-$  ;

- des  $q$  relations des propriétés  $P_3$  et  $P_4$  vérifiées par les composantes de  $Z(x_1^+)$ ,

- des  $q$  relations identiques satisfaites par  $Z(x_n^-)$ .

a) Première étape : Passage de  $Z(x_i^+)$  à  $Z(x_{i+1}^-)$

Les propriétés ( $P_3$ ) et ( $P_4$ ) (cf § I - 4 - 1)

appliquées au point  $x_1^+$  peuvent se mettre sous la forme matricielle

(I - 4 - 8) :

$$M(x_1^+) \cdot Z(x_1^+) = b(x_1^+) \quad (I-4-8)$$

$$\text{avec } Z^T(x_1^+) = [f(x_1^+), f'(x_1^+), \dots, f^{2q-2}(x_1^+), f^{2q-1}(x_1^+)]$$

M est une matrice de dimension (q, 2q)

b est un vecteur de dimension (q)

$$M(x_1^+) = \begin{bmatrix} 0 & \dots & \dots & \dots & 0 & 1 & 0 & \dots & \dots & \dots & 0 \\ \cdot & & & & \cdot & \cdot & 1 & & & & \cdot \\ \cdot & & & & \cdot & \cdot & \cdot & & & & \cdot \\ \cdot & & & & \cdot & \cdot & \cdot & & & & \cdot \\ \cdot & & & & \cdot & \cdot & \cdot & & & & \cdot \\ \cdot & & & & \cdot & \cdot & \cdot & & & & \cdot \\ 0 & & & & \cdot & \cdot & \cdot & & & & 1 \\ 1 & \dots & \dots & \dots & 0 & 0 & \dots & \dots & \dots & \dots & 0 \end{bmatrix} \quad b(x_1^+) = \begin{bmatrix} 0 \\ \cdot \\ \cdot \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ y_1 = f(x_1^+) \end{bmatrix}$$

EXEMPLE : q=2

$$\begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} f(x_1^+) \\ f'(x_1^+) \\ f''(x_1^+) \\ f'''(x_1^+) \end{bmatrix} = \begin{bmatrix} 0 \\ y_1 \end{bmatrix} \Leftrightarrow \begin{cases} f''(x_1^+) = 0 \leftarrow (P_4) \\ f(x_1^+) = y \leftarrow (P_3) \end{cases}$$



L'expression de  $Z(x_2^-)$  s'obtient à partir de l'expression suivante permettant le passage de  $Z(x_1^+)$  à  $Z(x_2^-)$  :

$$Z(x_2^-) = T(h_1).Z(x_1^+) \quad (I - 4 - 9)$$

où  $T(h_1)$  est obtenue en écrivant le développement de Taylor de  $f(x)$  en  $x_2^-$ , avec  $h_1 = x_2 - x_1$  :

$$T(h_1) = \begin{vmatrix} 1 & h_1 & \frac{h_1^2}{2!} & \dots & \frac{h_1^{2q-1}}{(2q-1)!} \\ 0 & 1 & h_1 & \dots & \frac{h_1^{2q-2}}{(2q-2)!} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & h_1 \\ & & & & 1 \end{vmatrix}$$

En injectant (I - 4 - 9) dans (I - 4 - 8), on obtient la relation suivante (I - 4 - 10) :

$$M(x_1^+).T(-h_1).Z(x_2^-) = b(x_1^+) \quad (I - 4 - 10)$$

soit  $M(x_2^-).Z(x_2^-) = b(x_2^-) \quad (I - 4 - 11)$

avec  $b(x_2^-) = b(x_1^+)$  et  $M(x_2^-) = M(x_1^+)$

Pour  $q=2$ , on obtient en  $x_2^-$  la relation suivante :

$$\begin{bmatrix} 0 & 0 & 1 & -h_1 \\ 1 & -h_1 & \frac{h_1^2}{2!} & -\frac{h_1^3}{3!} \end{bmatrix} \cdot \begin{bmatrix} f(x_2^-) \\ f'(x_2^-) \\ f''(x_2^-) \\ f'''(x_2^-) \end{bmatrix} = \begin{bmatrix} 0 \\ y_1 \end{bmatrix}$$

( I - 4 -I2 )

Cette première étape peut se généraliser en énonçant le lemme suivant :

Lemme 1 : A partir de la connaissance, en  $x_i^+$ , de  $q$  relations linéaires satisfaites par  $Z(x_i^+)$  :

$$M(x_i^+) \cdot Z(x_i^+) = b(x_i^+) \quad (I - 4 -I3)$$

on peut déterminer en  $x_{i+1}^-$ ,  $q$  relations linéaires analogues.

$$M(x_{i+1}^-) \cdot Z(x_{i+1}^-) = b(x_{i+1}^-) \quad (I - 4 -I4)$$

b) Deuxième étape : Passage de  $Z(x_{i+1}^-)$  à  $Z(x_{i+1}^+)$  :

Compte-tenu de  $(P_1)$ , la relation (I - 4 - I2) n'est valide en  $x_2^+$  que pour les  $(2q-1)$  premières composantes de  $Z(x_2)$ . En éliminant donc cette composante  $f^{2q-1}(x_2^-)$  entre les  $q$  équations de cette relation, on obtient  $q-1$  nouvelles relations qui sont vérifiées en  $x_2^+$  ( $P_1$ ). On prend comme  $q^{\text{ième}}$  composante

$$f(x_2) = y_2$$

L'exemple précédent donne, en éliminant  $f'''(x_2^-)$  entre les deux relations de (I - 4 - I2) :

$$f(x_2^+) - h_1 \cdot f'(x_2^+) + \frac{h_1^2}{3} \cdot f''(x_2^+) = y_1$$

$$f(x_2^+) = y_2$$

soit :

$$\begin{bmatrix} 1 & -h_1 & \frac{h_1^2}{3} & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \cdot Z(x_2^+) = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

(I - 4 - I5)

Cette deuxième étape se généralise aisément par le lemme suivant :

Lemme 2 : A partir de la connaissance de q relations linéaires indépendantes de la relation  $f(x_i) = y_i$ , vérifiées par  $Z(x_{i+1}^-)$  :

$$M(x_{i+1}^-) \cdot Z(x_{i+1}^-) = b(x_{i+1}^-) \quad \forall i=1, \dots, n-1$$

On peut déterminer q relations linéaires du même type vérifiées par  $Z(x_{i+1}^+)$  :

$$M(x_{i+1}^+) \cdot Z(x_{i+1}^+) = b(x_{i+1}^+).$$

En appliquant, alternativement les lemmes 1 et 2 sur l'intervalle  $[x_1, x_n]$  parcouru de gauche à droite on obtient en tout point :

$$x_i \quad (x_i^- \text{ et/ou } x_i^+) \quad \forall i=1, \dots, n$$

q relations linéaires entre les 2q inconnues. Les q relations manquantes sont obtenues en appliquant une stratégie identique sur l'intervalle

$[x_1, x_n]$  parcouru de droite à gauche en initialisant la relation (I - 4 - 8) au point  $x_n^-$ .

En tout point  $x_i$ , on dispose alors de  $2q$  équations permettant de déterminer les  $2q$  composantes de  $Z(x_i^+)$ . Il convient de remarquer qu'en tout point intermédiaire  $(x_2, \dots, x_{n-1})$  la première composante de  $Z(x_i^+)$  est immédiatement donnée ( $Z_1(x_i^+) = y_i$ ).

Aux points extrêmes  $(x_1, x_n)$ ,  $q-1$  composantes de  $Z(x)$  sont immédiatement issues de la propriété  $P_4$ .

On déduit l'expression du polynôme d'interpolation à partir de la connaissance des  $q$  premières composantes du vecteur  $Z(x_i)$  ( $\forall i = 1, \dots, n$ ) en utilisant la méthode de LAGRANGE-HERMITE à arguments répétés  $x_i$  et  $x_{i+1}$  (schéma de AITKEN-NEVILLE). [TODD, 1962]

#### I - 4 - 4 : METHODE UTILISANT LA BASE DES FONCTIONS

##### B-SPLINES :

(DEBOOR , 1977 - SCHUMAKER , 1981)

Cette méthode consiste à choisir une base de fonctions splines pour l'espace vectoriel des fonctions splines polynomiales par morceaux de degré  $k-1$ , définies sur un réseau  $\mathbb{R}$  de points d'abscisses  $x_1, x_2, \dots, x_n$  tel que :

$$x_1 < x_2 < \dots < x_{n-1} < x_n$$

Ces fonctions sont notées B-splines et sont définies à partir du réseau  $R$  et de  $\forall_i \leq k$  conditions de continuité vérifiées par la fonction spline polynomiale par morceaux en tout point  $x_i$  de  $R$ .

De manière à construire les fonctions B-splines précédemment définies, on introduit des abscisses supplémentaires à celles du réseau  $R$  en "transformant" les conditions de continuité aux noeuds de  $R$  en points d'un nouveau réseau.

$$\text{Soit : } l = k + \sum_{i=2}^{n-1} (k - v_i), \quad \mathcal{C} = \{t_i\}_{i=1}^l$$

est le nouveau réseau sur lequel sont définies les B-splines, on montre (Théorème de SCHOENBERG & CURRY) qu'il est construit à partir du réseau par les règles suivantes (I - 4 - 16) :

1)  $t_1 \leq t_2 \leq \dots \leq t_k \leq x_1$  (I - 4 - 16)

2)  $t_{l+k} \gg \dots \gg t_{l+1} \gg x_n$

3)  $i=2, \dots, n-1$  ; alors  $(k - v_i)$  abscisses de  $\mathcal{C}$  sont égales à  $x_i$

$$4) t_1 \leq t_2 \leq \dots \leq t_i \leq t_{i+1} \dots \leq t_{l+k}$$

Compte-tenu de ces relations, on définit une base de fonctions définies sur le réseau  $\mathcal{E}$ , de la manière suivante :

$$B_{i,k,\mathcal{E}}(x) = (t_{i+k} - t_i) \cdot [t_i, \dots, t_{i+k}] \cdot (t-x)_+^{k-1}; \quad (I-4-17)$$

$$\forall i=1, \dots, l$$

où  $B_{i,k,\mathcal{E}}$  sont les B-splines d'ordre  $k$ , définies sur  $\mathcal{E}$ .

$(t-x)_+^{k-1}$  est la fonction de GREEN définie par

$$(t-x)_+^{k-1} = \max \{ 0, (t-x)^{k-1} \} \quad (I-4-18)$$

$[t_i, \dots, t_{i+k}] \cdot (t-x)_+^{k-1}$  est la différence divisée d'ordre  $k$  de la fonction de GREEN prise en fixant  $x$  et en considérant  $(t-x)_+^{k-1}$  comme une fonction de  $t$ .

si  $t_i \neq t_{i+1}$ , cette expression s'écrit :

$$[t_i, \dots, t_{i+k}] \cdot (t-x)_+^{k-1} = \sum_{j=i}^{i+k} \frac{(t_j - x)_+^{k-1}}{\prod_{\substack{l=i \\ l \neq j}}^{i+k} (t_j - t_l)}$$

Les fonctions  $B_{i,k}^{\mathcal{C}}(x)$  vérifient les propriétés suivantes : (I - 4 - 20)

$$- B_{i,k}^{\mathcal{C}}(x) = 0 ; \text{ si } x \notin [t_i, t_{i+k}] ; \forall i=1, \dots, l$$

$$- B_{i,k}^{\mathcal{C}}(x) > 0 ; \forall x \in ]t_i, t_{i+k}[ ; \forall i=1, \dots, l$$

$$- \sum_{i=1}^l B_{i,k}^{\mathcal{C}}(x) = 1 ; \forall x \in ]t_j, t_{j+1}[ \subset [t_1, t_{l+k}]$$

(I-4-20)

Nous proposons d'illustrer la base de fonctions définies précédemment par l'exemple des B-splines d'ordre 4 suivant :

$$\mathcal{C} = \{0, 1, 2, 3, 4, 5, 6, 7\} = [t_1, t_9] \text{ et } k=4$$

$$B_{1,4}^{\mathcal{C}}(x) = \frac{(0-x)_+^3}{6} - \frac{2}{3}(1-x)_+^3 + (2-x)_+^3 - \frac{2}{3}(3-x)_+^3$$

$$+ \frac{1}{6}(4-x)_+^3 ; \quad \forall x \in [0,4]$$

$$B_{2,4}^{\mathcal{C}}(x) = \frac{1}{6}(1-x)_+^3 - \frac{2}{3}(2-x)_+^3 + (3-x)_+^3 - \frac{2}{3}(4-x)_+^3$$

$$+ \frac{1}{6}(5-x)_+^3 ; \quad \forall x \in [1,5]$$



$$B_{3,4}(x) = \frac{1}{6} (2-x)_+^3 - \frac{2}{3} (3-x)_+^3 + (4-x)_+^3 - \frac{1}{6} (6-x)_+^3$$

$$\forall x \in [2,6]$$

$$B_{4,4}(x) = \frac{1}{6} (3-x)_+^3 - \frac{3}{4} (4-x)_+^3 + \frac{3}{2} (5-x)_+^3 - \frac{11}{12} (6-x)_+^3$$

$$+ \frac{3}{2} (6-x)_+^2$$

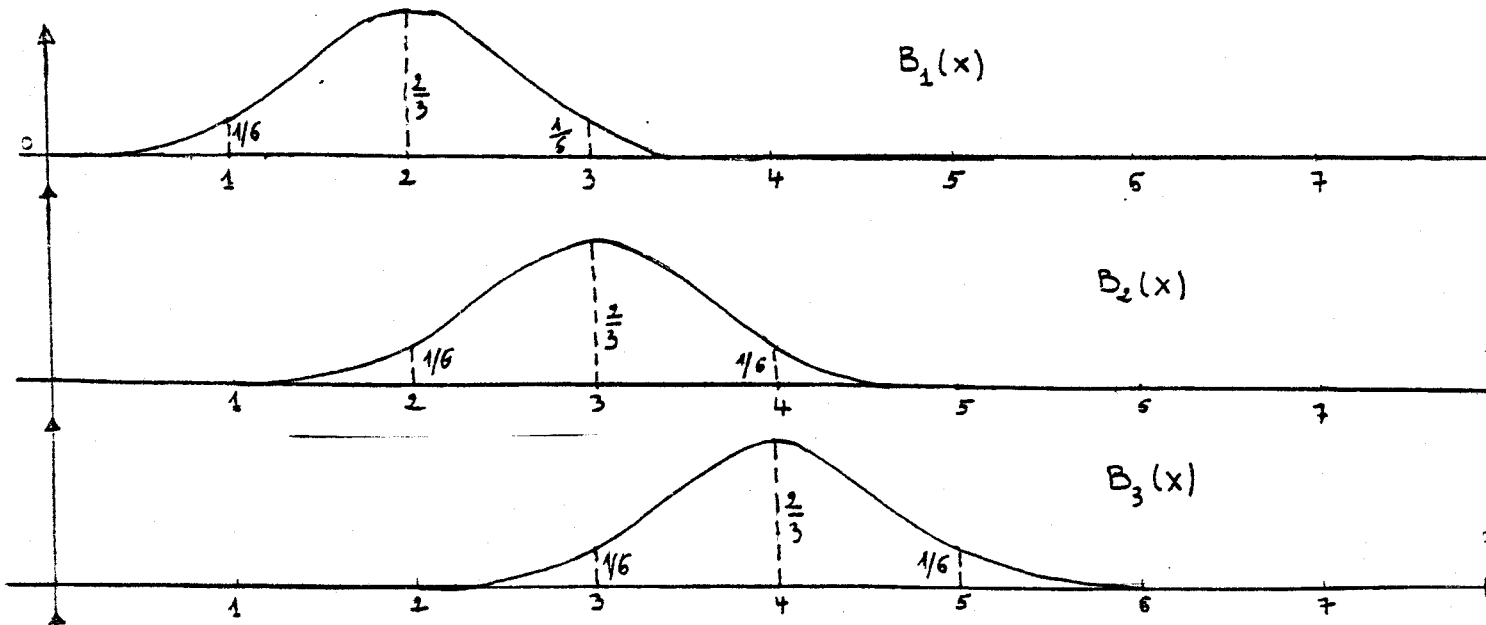
$$\forall x \in [3,6]$$

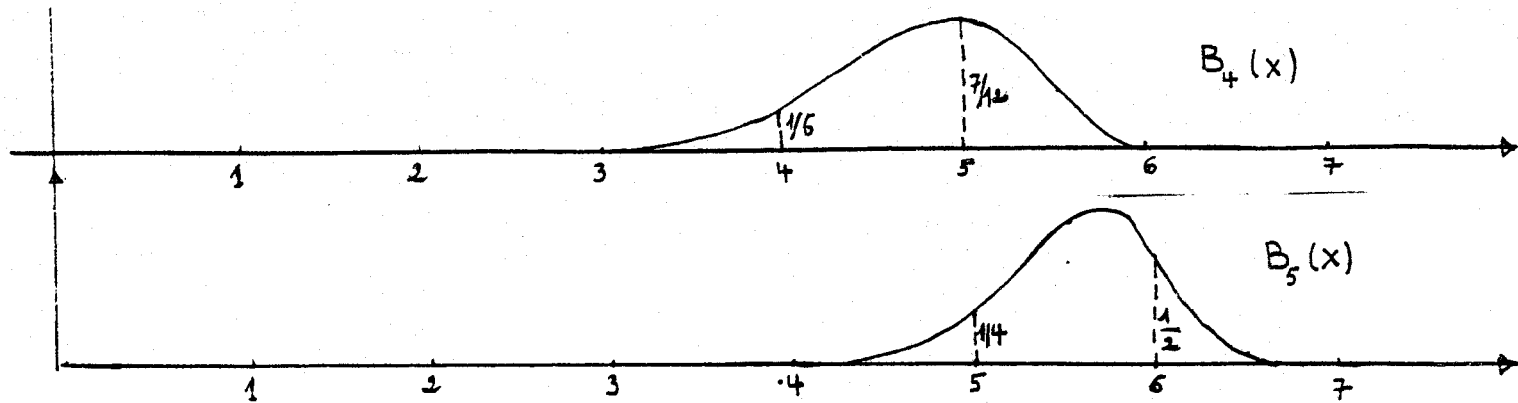
$$B_{5,4}(x) = \frac{1}{4} (4-x)_+^3 - \frac{3}{2} (5-x)_+^3 + \frac{3}{4} (6-x)_+^3 - \frac{9}{2} (6-x)_+^2$$

$$+ \frac{1}{2} (7-x)_+^3$$

$$\forall x \in [4,7]$$

Fig - 14 : EXEMPLE DE B-SPLINES





Dans la pratique le calcul des B-splines d'ordre  $k$  utilise de manière répétitive l'équation de récurrence suivante (I - 4 - 2I) :

$$B_{i,k}(x) = \frac{x-t_i}{t_{i+k}-t_i} \cdot B_{i,k-1}(x) + \frac{t_{i+k}-x}{t_{i+k}-t_{i+1}} \cdot B_{i+1,k-1}(x)$$

$$B_{i+1,k-1}(x)$$

$$\forall x \in [t_i, t_{i+1}[ \quad ; \quad \forall i = 1, \dots, l+k-1$$

(I - 4 - 2I)

que l'on initialise pour :

$$B_{j,i}(x) = \begin{cases} 1 & t_j \leq x < t_{j+1} \\ 0 & \text{ailleurs} \end{cases}$$

(I-4-22)

Les B-splines étant définies (I - 4 - 17) et calculées comme précédemment (I - 4 - 21, I - 4 - 22) toute combinaison de la forme (I - 4 - 23) :

$$f(x) = \sum_{i=1}^l \alpha_i \cdot B_{i,k,\mathcal{C}}(x) \quad (\text{I - 4 - 23})$$

est notée fonction spline d'ordre  $k$  pour la séquence  $\mathcal{C} = \left\{ t_i \right\}_{i=1}^{l+k}$

Cette dernière relation permet de calculer la représentation polynomiale par morceaux d'une fonction  $f$  à partir de sa représentation B-spline.  $\mathcal{C}$  ainsi que les coefficients B-splines sont alors donnés.

Un tel passage se réalise aisément dans la mesure où on peut calculer et différencier la relation (I - 4 - 23). La détermination des dérivées  $f^{(j-1)}(x_i)$  ;  $\forall j=1, \dots, k$  définissant complètement la représentation polynomiale par morceaux de  $f$ , s'effectue de la manière suivante.

On construit le réseau  $\left\{ x_i \right\}_{i=1}^n$  à partir des données du problème :

(  $\mathcal{C} = \left\{ t_i \right\}_{i=1}^{l+k}$  ) en respectant les différentes règles (I - 4 - 10) définies précédemment, puis on calcule les valeurs des dérivées d'ordre  $k-1$  de la fonction spline en tout point  $x_i^+$ .

Cette dernière étape nécessite la connaissance des B- coefficients des dérivées d'ordre  $k-1$  en tout  $x_i$ .

On montre (DEBOOR , 1978) que ces derniers peuvent être obtenus en différenciant répétitivement la fonction spline  $f(x)$ , (I - 4 - 17), obtenant par la même occasion des fonctions splines d'ordre inférieur.

Sur l'intervalle  $[t_r, t_s]$  il vient les relations suivantes (I - 4 - 24, I - 4 - 25) :

$$\begin{aligned} D \sum_{i=r}^s \alpha_i \cdot B_{i,k,\mathcal{C}}(x) &= \frac{d^j}{dx^j} \left\{ \sum_{i=r}^s \alpha_i \cdot B_{i,k,\mathcal{C}}(x) \right\} \\ &= \sum_{i=r}^{s+} \alpha_i^{(j+1)} \cdot B_{i,k,\mathcal{C}}(x) \end{aligned}$$

(I - 4 - 24)

avec :

$$\alpha_i^{(j+1)} = \begin{cases} \alpha_i & ; \text{ si } j = 0 \\ \frac{\alpha_i^{(j)} - \alpha_{i-1}^{(j)}}{(t_{i+k-j} - t_i) / (k-j)} & ; \text{ si } j > 0 \end{cases}$$

(I - 4 - 25)

Cette dernière formulation conjointement à l'équation (I - 4 - 21.) donnant les valeurs des B-splines en tout point  $x_i$  ;  $\forall i=1, \dots, n$ , permet de calculer les  $f^{(j-1)}(x_i^+)$  ;  $j=1, \dots, k$  ; puis de déduire la fonction spline  $f(x)$ .

Par ailleurs, si sont respectivement données une suite  $\mathcal{C} = \{t_i\}_{i=1}^{l+k}$ , une séquence de points d'abscisses  $\{\tau_i\}_{i=1}^l$  et une fonction  $g$  prenant les valeurs :  $g(\tau_i)$  en  $\tau_i$ , alors  $g$  peut être approximée par une fonction spline  $f(x)$  d'ordre  $k$  admettant les points  $\{\tau_i, g(\tau_i) ; i=1, \dots, l\}$  comme noeuds.

C'est le problème "d'agrégation de données".

Il suffit pour cela de rechercher la suite  $\{\tau_i\}_{i=1}^l$  qui vérifie la système d'équations linéaires d'ordre  $l$  suivant (I - 4 - 26) :

$$\sum_{j=1}^{\ell} \alpha_j \cdot B_{j,k,\ell}(\tau_i) = g(\tau_i) ; \forall i=1, \dots, \ell$$

(I - 4 - 26)

Ce système admet une solution si et seulement si la matrice (I - 4 - 27) :

$$A = \begin{bmatrix} B_1(\tau_1) & B_2(\tau_1) \dots B_1(\tau_1) \\ \cdot & \cdot \\ \cdot & B_j(\tau_i) \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ B_1(\tau_1) \dots \dots \dots B_i(\tau_1) \end{bmatrix} \quad (\text{I - 4 - 27})$$

est inversible.

Cette dernière condition se traduit par le choix suivant des points  $\tau_i$  (I - 4 - 22). (Théorème de SCHOENBERG-WHITNEY) (cf DEBOOR , 1978)

$$t_i < \tau_i < t_{i+k} \quad \forall i=1, \dots, \ell \quad (\text{I - 4 - 28})$$

Un choix de points  $\tau_i$  recommandé (lorsqu'il n'est pas imposé) est le suivant :

$$\tau_i = (t_{i+1} + t_{i+2} + \dots + t_{i+k-1}) / (k-1)$$

$$\forall i=1, \dots, \ell \quad (\text{I - 4 - 29})$$

Dans la pratique, le problème de l'interpolation d'un réseau  $\{\tau_i\}_{i=1}^l$  par une fonction spline  $f(x)$  prenant les valeurs  $g(\tau_i)$  en ces points consiste à déterminer la séquence  $\mathcal{C}$  sur laquelle sont définies les B-splines, de telle sorte que soit vérifiée la relation (I - 4 - 28).

En général, il est impossible de réaliser un tel choix et on substitue le problème en recherchant la fonction spline d'interpolation  $f(x)$  telle que, si l'estimation de l'erreur d'interpolation est définie par la relation suivante (I - 4 - 30):

$$\|g - f\| \leq \varepsilon_f \cdot \|g^{(k)}\| ; \quad (I - 4 - 30)$$

où ( $\varepsilon_f =$  constante dépendant de  $f$ ), alors  $\varepsilon_f$  est la plus faible possible.

Sous ces hypothèses, une telle fonction requiert le choix du réseau  $\mathcal{C}$  suivant (I - 4 - 3I):

$$t_{i+k} = (\tau_{i+1} + \tau_{i+2} + \dots + \tau_{i+k-1}) / (k-1) \quad ; \quad i=1, \dots, l-k$$

$$t_1 = t_2 = \dots = t_k = \tau_1 \quad (I - 4 - 3I)$$

$$t_{l+1} = t_{l+2} = \dots = t_{l+k} = \tau_l$$

La méthode des B-splines décrite précédemment donne de très bons résultats (minimisation de volume de calcul, obtention de formes douces, déplacements de points sans altérer le reste de la courbe) et résoud aisément le problème d'agrégation de données lorsque  $\mathcal{C}$  est donné.

Néanmoins, le problème classique de l'interpolation (recherche de la séquence  $\mathcal{C}$  à partir des points de passage) demeure en général impossible à résoudre de manière optimale.

D'autres situations nécessitent le passage à une représentation polynomiale par morceaux. C'est la cas où le nombre de points à interpoler est élevé, et où les abscisses sont équidistantes (SHUMAKER, 1981)

#### I - 4 - 5 : CONCLUSION :

Indépendamment des problèmes soulevés aux paragraphes précédents et qui, dans certains cas, ne permettent pas d'élaborer des formes géométriques complexes, la méthode de BEZIER constitue néanmoins une excellente méthode de conception.

Elle permet une représentation directe de courbes gauches dans  $\mathbb{R}^3$  et introduit de par son paramétrage, la notion de système dynamique ; une courbe est considérée comme la trajectoire d'un point.



Alors que la méthode des B-splines semble être un cas typique pour résoudre le problème d'agrégation de données, (E connue la méthode du transport des relations (représentation polynomiale par morceaux - LAURENT , 1971-) semble être la plus rapide, notamment lorsque le nombre de points à interpoler est élevé (représentation graphique ). (SCHUMAKER , 1981)

Dans la méthode que nous proposons comme dans celle de BEZIER, la courbe à générer est envisagée comme la trajectoire d'un point. Ce point est considéré comme la sortie d'un processus (au sens de la théorie de l'Automatique) dont le vecteur d'état (P. NASLIN , 1968) est le vecteur  $Z(x)$  utilisé dans la méthode de transport (cf § I - 4 - 3).

La structure de ce processus est définie par sa représentation dans l'espace d'état à partir des propriétés de base des fonctions splines d'ordre  $q$ , et par l'application des résultats fondamentaux de la théorie de la commande optimale (PONTRYAGUINE , 1968).

## CHAPITRE II

## CHAPITRE II : ALGORITHME DU PROCESSUS INTERPOLATEUR

---

### II - 1. INTRODUCTION :

Dans ce chapitre, nous considérons une fonction Spline d'interpolation comme la trajectoire issue d'un système dynamique linéaire stationnaire complètement observable (GILLES , 1978), soumis à une loi de commande.

Ce système dynamique muni de sa commande est noté "PROCESSUS INTERPOLATEUR". Il est construit à partir des propriétés des fonctions Splines d'interpolation d'ordre  $q$  définies au chapitre précédent.

L'application des résultats de la théorie de la commande optimale (BOUDAREL , 1968 - PONTRYAGUINE , 1974) des processus permet d'expliciter la commande cherchée, à partir d'une représentation du système dynamique dans l'espace d'état (NASLIN , 1968).

L'interprétation du problème comme la détermination d'une commande optimale permet, entre autre, de restreindre le nombre des propriétés définissant les fonctions Splines d'interpolation et d'en rendre l'utilisation plus souple, en particulier lorsqu'on souhaite réaliser une courbe à partir de la concaténation de plusieurs arcs.

Cette procédure, est au sens propre du terme, analogue au système dynamique de génération des fonctions de Bézier (T. BENNANI , 1975), et s'étend aisément aux fonctions Splines

généralisées (M. ATTEIA , 1968) d'ordre  $q$  (interpolation, sous-tension, ajustement).

Dans un souci pédagogique, nous proposons d'introduire notre méthodologie à partir de la construction de fonctions Splines d'interpolation cubique ( $q = 2$ ), en introduisant les propriétés de base qui permettent de définir le "processus interpolateur", dont la sortie décrit la courbe souhaitée.

Après avoir défini la structure du processus interpolateur, nous explicitons le calcul de ses paramètres à partir de la représentation dans l'espace d'état.

Cette démarche est alors généralisée pour les autres types de fonctions Splines.

## II- 2. DETERMINATION DES FONCTIONS SPLINES CUBIQUES D'INTERPOLATION

### D'ORDRE $q = 2$

#### II - 2 - 1 : POSITION DU PROBLEME ET PROPRIETES :

Pour l'essentiel, il s'agit de résoudre de manière plus aisée un problème pour lequel nous avons explicité une solution au chapitre précédent. Nous comparerons en conclusion la complexité des différents algorithmes.

Soit un réseau de points  $\{x_i, y_i \in \mathbb{R} ; i = 1, \dots, n\}$  ordonné par la relation suivante :

$$x_1 < x_2 < \dots < x_{n-1} < x_n$$

Le problème de l'interpolation de ce réseau par une fonction Spline d'ordre 2 consiste à rechercher la fonction  $f(x)$

satisfaisant les propriétés identiques à celles exposées au précédent chapitre et que nous proposons de rappeler :

P 1)  $f(x)$  est de classe  $\mathcal{C}^2 [x_1, x_n]$  .

(En d'autres termes,  $f, f'$  et  $f''$  sont continues sur  $[x_1, x_n]$  ).

P 2)  $f(x)$  interpole le réseau  $\{x_i, y_i\}$  :

$$f(x_i) = y_i ; \forall i = 1, \dots, n.$$

P 3)  $f(x)$  minimise dans l'espace des fonctions  $\mathcal{P}(x)$  satisfaisant les propriétés (P1) et (P2), la fonctionnelle de forme :

$$\Phi(f(x)) = \int_{x_1}^{x_n} (\mathcal{P}''(x))^2 dx ;$$

P 4) Cette propriété constitue, comme nous le montrons par la suite, un corollaire des propriétés (P1), (P2), (P3).

Dans tout intervalle  $[x_i, x_{i+1}]$  ;  $i = 1, \dots, n - 1$  ;

$f(x)$  est un polynôme de degré 3 qui s'écrit :

$$f_i(x) = \sum_{k=0}^3 a_k^i \cdot (x_{i+1} - x)^k ; \forall i = 1, \dots, n - 1 ;$$

Géométriquement, la propriété (P3) spécifie la minimisation des variations de pentes, ce qui équivaut à rechercher des courbes qui n'oscillent pas entre deux points d'échantillonnage.

Si nous définissons le schéma suivant : (Fig - 15)

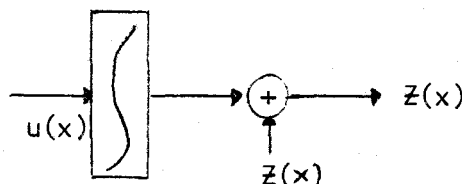


Fig - 15

comme l'intégrale :

$$Z(x) = Z(x) + \int_{x(0)}^x u(\alpha) \cdot d\alpha$$

La fonction  $f(x)$  recherchée apparait comme la sortie du système dynamique en boucle ouverte suivant : (Fig - 16)

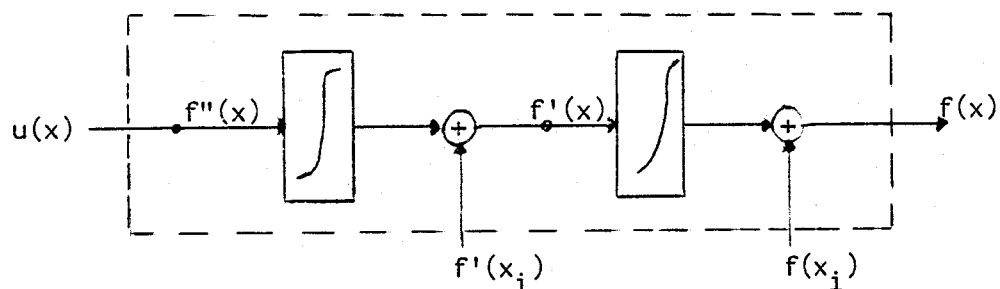


Fig - 16

Les éléments représentés dans cette figure intègrent par rapport à la variable  $x \in [x_1, x_n]$  .

Mis sous cette forme, le problème précédent peut-être explicité en terme de commande optimale des processus de la manière suivante :

Il s'agit de déterminer la commande  $u(x)$  qui engendre la trajectoire  $f(x)$  ;  $x \in [x_1, x_n]$  passant par tous les points d'échantillonnage  $(x_i, y_i)$  et qui minimise la fonctionnelle

$$\Phi(u(x)) = \int_{x_1}^{x_n} u^2(x).dx ; \quad (\text{II-2-1})$$

Il est manifeste qu'il existe des commandes telles que la trajectoire  $f(x)$  satisfasse la propriété d'interpolation et que parmi celles-ci, on cherche la/les commandes(s) qui minimise(nt). (II-2-1)

Une représentation du système (Fig - 16) sous forme d'équation d'état est la suivante :

$$\frac{d}{dx}(\bar{Z}(x)) = A \cdot \bar{Z}(x) + b \cdot u(x)$$
$$Z(x) = C \cdot \bar{Z}(x)$$
$$; \quad \begin{matrix} b \in \mathbb{R}^{2 \times 1} \\ A \in \mathbb{R}^{2 \times 2} \end{matrix}; \quad C \in \mathbb{R}^{2 \times 1} \quad (\text{II-2-2})$$

où :  $\bar{Z}(x)^T = [Z(x), Z'(x)] = [f(x), f'(x)] \in \mathbb{R}^{2 \times 1}$

$$b^T = \begin{bmatrix} 0 & 1 \end{bmatrix} ; \quad A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$$
$$c = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

Nous proposons de noter l'ensemble formé du système dynamique (Fig - 16) et de sa commande  $u(x)$ , "PROCESSUS INTERPOLATEUR".

Cette dernière formulation permet de résoudre le problème posé, par application de la théorie de la commande optimale dont nous proposons de rappeler les quelques résultats fondamentaux que nous utilisons immédiatement. Des rappels plus détaillés sont explicités en annexe N°1.

## II - 2 - 2 : COMMANDE OPTIMALE DES PROCESSUS : RAPPELS :

La théorie de la commande optimale appliquée au problème particulier des fonctions Splines d'ordre 2 consiste à déterminer la commande  $u(x)$  ;  $x \in [x_1, x_n]$  qu'il convient d'appliquer au système dynamique (Fig - 16) générant la trajectoire  $Z(x)$  pour que celle-ci satisfasse aux conditions de passage

et de continuité P1), P2) (cf. § II - 2 - 1) tout en minimisant la fonctionnelle quadratique suivante :

$$J ( u(.) ) = \int_{x_1}^{x_n} u^2(x).dx \quad (\text{II} - 2 - 3)$$

J étant une fonctionnelle additive, sa minimisation implique celle de toutes les fonctionnelles.

$$J ( i, u(.) ) = \int_{x_i}^{x_{i+1}} u^2(x). dx \quad ; \quad \forall i = 1, \dots, n-1.$$

Dans tout intervalle  $]x_i, x_{i+1}[$ , pour lequel il n'existe pas de contrainte, la commande  $u_i(x)$  qui transfère le système d'un état  $(Z(x_i), Z'(x_i))$  à un état  $(Z(x_{i+1}), Z'(x_{i+1}))$  est obtenue par application du principe du maximum de PONTRYAGUINE (cf. PONTRYAGUINE 1974).

On montre (cf. Annexe 1) que cette dernière formulation ramène le problème d'optimisation d'une fonctionnelle (II - 2 - 3 par exemple) à celui de la maximisation en tout point  $x \in ]x_1, x_n[$ , d'une fonction scalaire notée HAMILTONIEN et définie par la relation :

$$\forall x \in ]x_i, x_{i+1}[ \quad ; \quad \forall i=1, \dots, n-1 \quad (\text{II} - 2 - 4)$$
$$H (\bar{Z}(x), u(x), \Psi(x)) = -\frac{1}{2} u^2(x) + \Psi^T (A \cdot \bar{Z}(x) + b \cdot u(x) )$$

Pour des variations infiniment petites  $\delta \bar{Z}(x)$  ;  
 $x \in ]x_1, x_n[$  et  $\delta u(x)$ , il s'en suit que :

$$\delta J ( . ) = \int_{x_1}^{x_n} \delta H. dx \geq 0 \quad (\text{II} - 2 - 5)$$



Sans contraintes sur la commande, il vient :

$$\forall x \in ]x_i, x_{i+1}[ ; \quad \forall i = 1, \dots, n-1 \quad (\text{II} - 2 - 6)$$

$$u(x) = b^T \cdot \Psi(x)$$

L'évolution du vecteur  $\Psi(x)$  en tout point  $x \in ]x_i, x_{i+1}[$  est régie par le système d'équations différentielles suivant :

$$\frac{d}{dx}(\Psi(x)) = -A^T \cdot \Psi(x) ; \quad \forall x \in ]x_i, x_{i+1}[ ; \quad \forall i = 1, \dots, n-1$$

(II - 2 - 7)

De même, pour des variations infinitésimales admissibles  $\delta \bar{z}(x_1)$  et  $\delta \bar{z}(x_n)$  du vecteur d'état aux extrémités du réseau  $[x_1, x_n]$ , la variation  $\delta J$  de la fonctionnelle (II-2-5) conduit aux conditions aux limites suivantes qui sont notées conditions de transversalité.

$$\left\{ \begin{array}{l} \Psi^T(x_1) \cdot \delta \bar{z}(x_1) = 0 \\ \Psi^T(x_n) \cdot \delta \bar{z}(x_n) = 0 \end{array} \right. \quad (\text{II} - 2 - 8)$$

Dans ces expressions,  $\delta \bar{z}(x)$  note un accroissement infinitésimal du vecteur  $\bar{z}(x)$  compatible avec les données du problème.

En particulier  $\bar{z}_1(x_1) = Z(x_1)$  étant fixé, son accroissement admissible  $\delta \bar{z}_1(x_1)$  est identiquement nul et on n'obtient alors aucune relation caractérisant  $\Psi_1(x_1)$  d'après les conditions de transversalité. (II - 2 - 8)

Ces relations définissent complètement la commande optimale  $u(x)$  cherchée dans tout intervalle  $[x_i, x_{i+1}]$  ;  $\forall i = 1, \dots, n-1$ ; à partir de schémas différentiels que l'on peut intégrer à partir de conditions initiales issues des conditions de transversalité (II - 2 - 8) et des données imposées (cf. § II-2-1) au processus.

Nous proposons d'expliciter la structure du processus interpolateur, puis de le calculer à partir des données du problème.

### II - 2 - 3 : STRUCTURE DU PROCESSUS INTERPOLATEUR :

Les résultats rappelés au paragraphe précédent dans le cadre de la formulation dans l'espace d'état (II - 2 - 2) du problème d'interpolation posé initialement, aboutissent à définir la structure suivante du processus interpolateur générant la trajectoire  $Z(x)$  : (II - 2 - 17)

$$\forall i = 1, \dots, n-1 ; \forall x \in [x_i, x_{i+1}[$$

$$\frac{d}{dx} \begin{bmatrix} \bar{Z}(x) \\ \Psi(x) \end{bmatrix} = \begin{bmatrix} A & b \cdot b^T \\ 0 & -A^T \end{bmatrix} \cdot \begin{bmatrix} \bar{Z}(x) \\ \Psi(x) \end{bmatrix} ; \quad (\text{II} - 2 - 17)$$

En posant  $W(x) \in \mathbb{R}^{4 \times 1}$  :  $W^T(x) = (\bar{Z}^T(x), \Psi^T(x))$  , la relation précédente (II - 2 - 17) s'écrit sous forme abrégée

$$\frac{d}{dx} (W(x)) = J \cdot W(x) \text{ avec } J = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{bmatrix} ; \quad (\text{II} - 2 - 18)$$

$\forall i = 1, \dots, n-1, \forall x \in [x_i, x_{i+1}[$

Effectuons la transformation suivante sur les composantes du vecteur d'état  $W(x)$  :

$$\begin{aligned} \Psi_1(x) &\longrightarrow \Psi_2(x) \\ \Psi_2(x) &\longrightarrow -\Psi_1(x) \end{aligned} \quad (\text{II} - 2 - 19)$$

En notant :

-  $v(x)$  le vecteur défini par :

$$v^T(x) = [Z(x), Z'(x), \Psi_2(x), -\Psi_1(x)]$$

-  $\tilde{A}$  la matrice de Jordan définie par (II - 2 - 20), le système (II - 2 - 18) s'écrit après la transformation (II - 2 - 19) sous la forme suivante :

$$\frac{d}{dx} (v(x)) = \tilde{A} \cdot v(x) ; \text{ où } \tilde{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} ; (\text{II} - 2 - 20)$$

Cette dernière relation correspond alors au schéma de simulation suivant : (Fig - 17), valable pour tout  $x \in [x_i, x_{i+1}[$  .

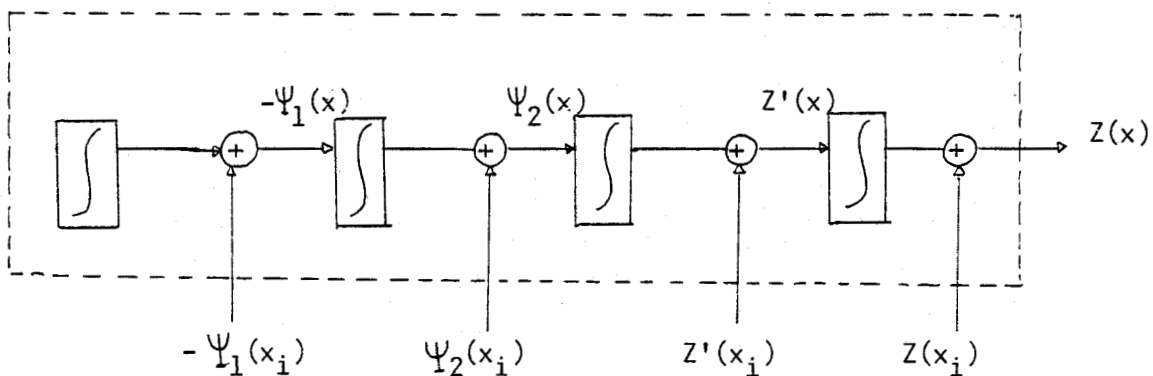


Fig - 17

L'équation de progression définissant numériquement la trajectoire  $Z(x)$  sur tout intervalle  $[x_i, x_{i+1}[$ , est obtenue par intégration de l'équation d'état précédente (II - 2 - 20) à partir des conditions initiales  $v(x_i)$ .

Il vient :

$$\forall i = 1, \dots, n-1 ; \forall h \in [0, x_{i+1} - x_i[$$

(II - 2 - 21)

$$v(x_i + h) = M(h) \cdot v(x_i)$$

avec :  $M(h) = \exp(\tilde{A}h)$

$$M(h) = \begin{bmatrix} 1 & h & \frac{h^2}{2!} & \frac{h^3}{3!} \\ 0 & 1 & h & \frac{h^2}{2!} \\ 0 & 0 & 1 & h \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La fonction  $Z(x)$  générée par le système dynamique (cf. Fig - 17) est ainsi complètement définie à partir de  $(n-1)$  conditions initiales sur le vecteur  $v(x_i) \in \mathbb{R}^4$ . ( $\forall i=1, \dots, n-1$ ).

Il convient donc maintenant de compléter le processus interpolateur en adjoignant au système dynamique un bloc de calcul fournissant les conditions initiales  $v(x_i)$  à partir des données du problème d'interpolation.

De manière à expliciter cette structure, nous proposons de représenter le processus interpolateur sous la forme du

schéma suivant, correspondant à une réalisation hybride de type II (Fig - 18) ; le bloc de calcul fournit successivement les composantes des vecteurs  $v(x_i)$  lorsque la variable d'intégration  $x$  devient égale à  $x_i$  ( $i=1, \dots, n-1$ ).

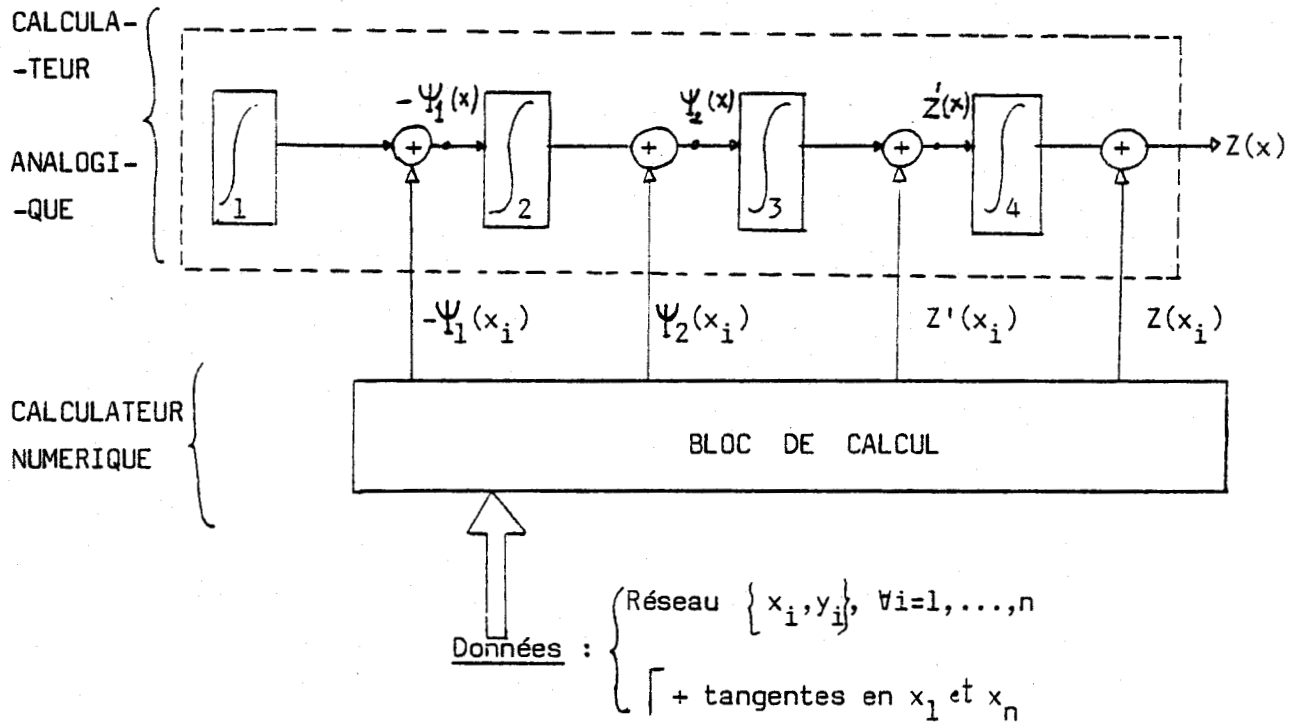


Fig - 18

Nous proposons maintenant d'expliciter le bloc de calcul du processus interpolateur.

II - 2 - 4 : DETERMINATION DU BLOC DE CALCUL :

La technique utilisée est basée sur l'équation de progression (II - 2 - 21) qui permet l'obtention du vecteur  $v(x_{i+1})$  à

partir du vecteur  $v(x_i^+)$ .

En tenant compte de la propriété de continuité(P1), nous proposons de montrer qu'il existe une relation affine entre les vecteurs  $v(x_{i+1}^+)$  et  $v(x_i^+)$ . Il ne restera plus qu'à déterminer les composantes du vecteur initial  $v(x_1^+)$ .

II - 2 - 4 - 1 ) Relation entre  $v(x_{i+1}^+)$  et  $v(x_i^+)$  :

La fonction de progression (II - 2 - 21) permet d'écrire :

$$v(x_{i+1}^-) = M(x_{i+1} - x_i) \cdot v(x_i^+) \quad (\text{II - 2 - 22})$$

$Z(x)$  étant de classe  $c^2$  sur  $[x_1, x_n]$ , les trois premières composantes de  $v(x_{i+1}^+)$  sont égales à celles de  $v(x_{i+1}^-)$ , ce qui se traduit par la relation matricielle (II - 2 - 23):

$\forall i = 1, \dots, n-1 :$

$$\begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & 0 \end{bmatrix} \cdot v(x_{i+1}^+) = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & 0 \end{bmatrix} \cdot v(x_{i+1}^-) \quad (\text{II - 2 - 23})$$

La dernière composante est obtenue à partir de la donnée de  $Z(x_{i+2}) = y_{i+2} = Z_{i+2}$ , par la première ligne de l'équation de progression (II - 2 - 21) sur l'intervalle  $[x_{i+1}, x_{i+2}]$ .

$$\begin{aligned} y_{i+2} = Z(x_{i+2}^+) &= Z(x_{i+2}^-) = [1 \ 0 \ 0 \ 0] \cdot M(h_{i+2}) \cdot v(x_{i+1}^+) \quad (\text{II - 2 - 24}) \\ &= [1; m_{12}(i+2); m_{13}(i+2); m_{14}(i+2)] \cdot v(x_{i+1}^+) \end{aligned}$$

L'ensemble des relations précédentes (II - 2 - 23, II - 2 - 24) s'écrit sous la forme affine suivante (II - 2 - 25) :

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & h_{i+2} & \frac{h_{i+2}^2}{2!} & \frac{h_{i+2}^3}{3!} \end{bmatrix} v(x_{i+1}^+) =$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 & h_{i+1} & \frac{h_{i+1}^2}{2!} & \frac{h_{i+1}^3}{3!} \\ 0 & 1 & h_{i+1} & \frac{h_{i+1}^2}{2!} \\ 0 & 0 & 1 & h_{i+1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot v(x_i^+)$$

$$+ \begin{bmatrix} 0 \\ 0 \\ 0 \\ y_{i+2} \end{bmatrix} \quad (\text{II - 2 - 25})$$

avec :

$$\begin{cases} h_{i+1} = x_{i+1} - x_i \\ h_{i+2} = x_{i+2} - x_{i+1} \end{cases}$$

soit sous forme abrégée :

$$N(h_{i+2}) \cdot v(x_{i+1}^+) = Q.M(h_{i+1}) \cdot v(x_i^+) + Y_1(i+2) \quad (\text{II-2-26})$$

La matrice  $N(h_{i+2})$  est inversible et son inverse est facilement calculable de manière formelle. Il vient donc l'équation de récurrence suivante : (II - 2 - 27)

$$V(x_{i+1}^+) = A(i+2) \cdot v(x_i^+) + B(i+2) \quad (\text{II - 2 - 27})$$

$$\forall i = 1, \dots, n - 2$$

avec :  $A(i+2) = N^{-1}(h_{i+2}) \cdot Q.M(h_{i+1})$

(II - 2 - 28)

$$B(i+2) = N^{-1}(h_{i+2}) \cdot Y_1(i+2)$$

$$N^{-1}(h_{i+2}) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\frac{6}{h_{i+2}^3} & -\frac{6}{h_{i+2}^2} & -\frac{3}{h_{i+2}} & \frac{6}{h_{i+2}^3} \end{bmatrix} ; B(i+2) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{6}{h_{i+2}^3} \cdot y(i+2) \end{bmatrix} \quad (\text{II-2-29})$$



$$A = \begin{bmatrix} 1 & h_{i+1} & \frac{h_{i+1}^2}{2} & h_{i+1}^3/3! \\ 0 & 1 & h_{i+1} & h_{i+1}^2/2! \\ 0 & 0 & 1 & h_{i+1} \\ -\frac{6}{h_{i+2}^3} ; & -\frac{6}{h_{i+2}^2} \left( \frac{h_{i+1}}{h_{i+2}} + 1 \right) ; & -\frac{3h_{i+1}}{h_{i+2}^2} \left( \frac{h_{i+1}}{h_{i+2}} + \frac{h_{i+2}}{h_{i+1}} + 2 \right) ; & -\frac{3h_{i+1}}{h_{i+2}} \left( 1 + \frac{h_{i+1}}{h_{i+2}} + \frac{h_{i+1}^2}{h_{i+2}^2} \right) \end{bmatrix}$$

(II - 2 - 30)

Il ne reste plus qu'à initialiser l'équation récurrente précédente par le calcul du vecteur  $v(x_1^+)$ .

II - 2 - 4-2 ) Calcul du vecteur  $v(x_1^+)$  :

Les données initiales du problème sont :

$$Z(x_1) = y_1$$

$$Z(x_n) = y_n$$

respectivement premières composantes des vecteurs  $v(x_1)$  et  $v(x_n)$ .

Par ailleurs, les valeurs des dérivées  $Z'(x_1)$  ou  $Z'(x_n)$  peuvent indépendamment faire partie des données du problème.

Dans le cas où l'une de ces dérivées n'est pas fixée, on peut déduire une autre relation sur les composantes du vecteur  $v(x)$  à l'extrémité considérée à partir des conditions de transversalité (II - 2 - 8).

Considérons en effet la variation du critère (II-2-3) sous l'effet des seules variations admissibles de la valeur  $\bar{Z}(x_1)$  de  $Z(x)$  à la borne inférieure  $x_1$ . Elle est exprimée par la relation suivante (II - 2 - 31) :

$$\delta J = \left[ \Psi^T(x) \cdot \delta \bar{Z}(x) \right]_{x=x_1} = 0 \quad (\text{II} - 2 - 31)$$

$$\bar{Z}^T(x) = [Z(x), Z'(x)]$$

$$v^T(x) = [Z(x), Z'(x), \Psi_2(x), -\Psi_1(x)]$$

$Z(x_1)$  étant fixée, la variation admissible  $\delta \bar{Z}_1(x)$  est nulle.

$Z'(x_1)$  n'étant pas fixée a priori, la relation précédente s'écrit :

$$\Psi_2(x_1) \cdot \delta Z'(x_1) = 0$$

Les variations  $\delta Z'(x_1)$  n'étant pas nulles, il vient alors :

$$\Psi_2(x_1) = 0 \quad (\text{II} - 2 - 32)$$

A ce stade d'analyse des données aux extrémités  $x_1$  et  $x_n$  ;

- on connaît la valeur de  $Z(x_1)$  (resp  $Z(x_n)$ )

- on connaît la valeur de  $Z'(x_1)$  (resp  $Z'(x_n)$ ) sinon  $\Psi_2(x_1)$  (resp  $\Psi_2(x_n)$ ) est nul.

En d'autres termes, les conditions limites du vecteur  $v(x)$  sont les suivantes :

$\left\{ \begin{array}{l} v_1(x_1) \text{ connu} \\ \text{et } v_2(x_1) \text{ fixé sinon } v_3(x_1) = 0 \end{array} \right.$ <p style="text-align: center;">et</p> $\left\{ \begin{array}{l} v_1(x_n) \text{ connu} \\ \text{et } v_2(x_n) \text{ fixé sinon } v_3(x_n) = 0 \end{array} \right.$
---

(II - 2 - 33)

Deux composantes de  $v(x_1)$  sont ainsi directement connues.

Les deux autres composantes peuvent être obtenues à partir,

1) de l'équation de progression (II - 2 - 21) qui n'a pas été utilisée au pas précédent pour  $x = x_2$  :

$$Z(x_2) = [1, 0, 0, 0] \cdot M(x_2 - x_1) \cdot v(x_1)$$

2) de la valeur de  $v_2(x_n^-)$  ou à défaut de la nullité de  $v_3(x_n^-)$ .

Cette composante est liée à  $v(x_1^+)$  en itérant la relation de récurrence établie précédemment (cf II - 2 - 27). Il vient :

$$v(x_n^-) = M(h_n) \cdot \left\{ \prod_{k=3}^n A(k) \right\} \cdot v(x_1) \tag{II - 2 - 34}$$

$$+ M(h_n) \cdot \left\{ \sum_{k=3}^n \left( \prod_{e=k+1}^n A(e) \right) \cdot B(h) \right\}$$

L'expression de  $M(h_n) = M(x_n - x_{n-1})$  est explicitée par (II - 2 - 21),

L'expression de  $A(k)$  est explicitée par (II - 2 - 30).

L'expression de  $B(k)$  est explicitée par (II - 2 - 29).

Les composantes de  $v(x_1)$  sont donc liées par une deuxième relation linéaire obtenue à partir de l'identité précédente en écrivant que  $v_2(x_n^-)$  est connu où bien que  $v_3(x_n^-)$  est nul.

EXEMPLE

Les données limites d'un problème d'interpolation sont définies par les valeurs de :

$$Z(x_1) , Z(x_n) , Z'(x_n)$$

Les conditions limites sur  $v(x_1)$  sont alors :

$$v_1(x_1) = Z(x_1)$$

(II - 2 - 35)

$$v_3(x_1) = 0$$

Les relations précédentes étant écrites sous forme abrégée par :

$$Z(x_2) = (1, 0, 0, 0) \cdot M \cdot v(x_1)$$

$$v(x_n^-) = P \cdot v(x_1^+) + q \quad (\text{II} - 2 - 36)$$

$$M \in \mathbb{R}^{4 \times 4}, P \in \mathbb{R}^{4 \times 4}; q \in \mathbb{R}^4$$

Les composantes  $v_2(x_1^+)$  et  $v_4(x_1^+)$  sont alors solutions du système suivant :

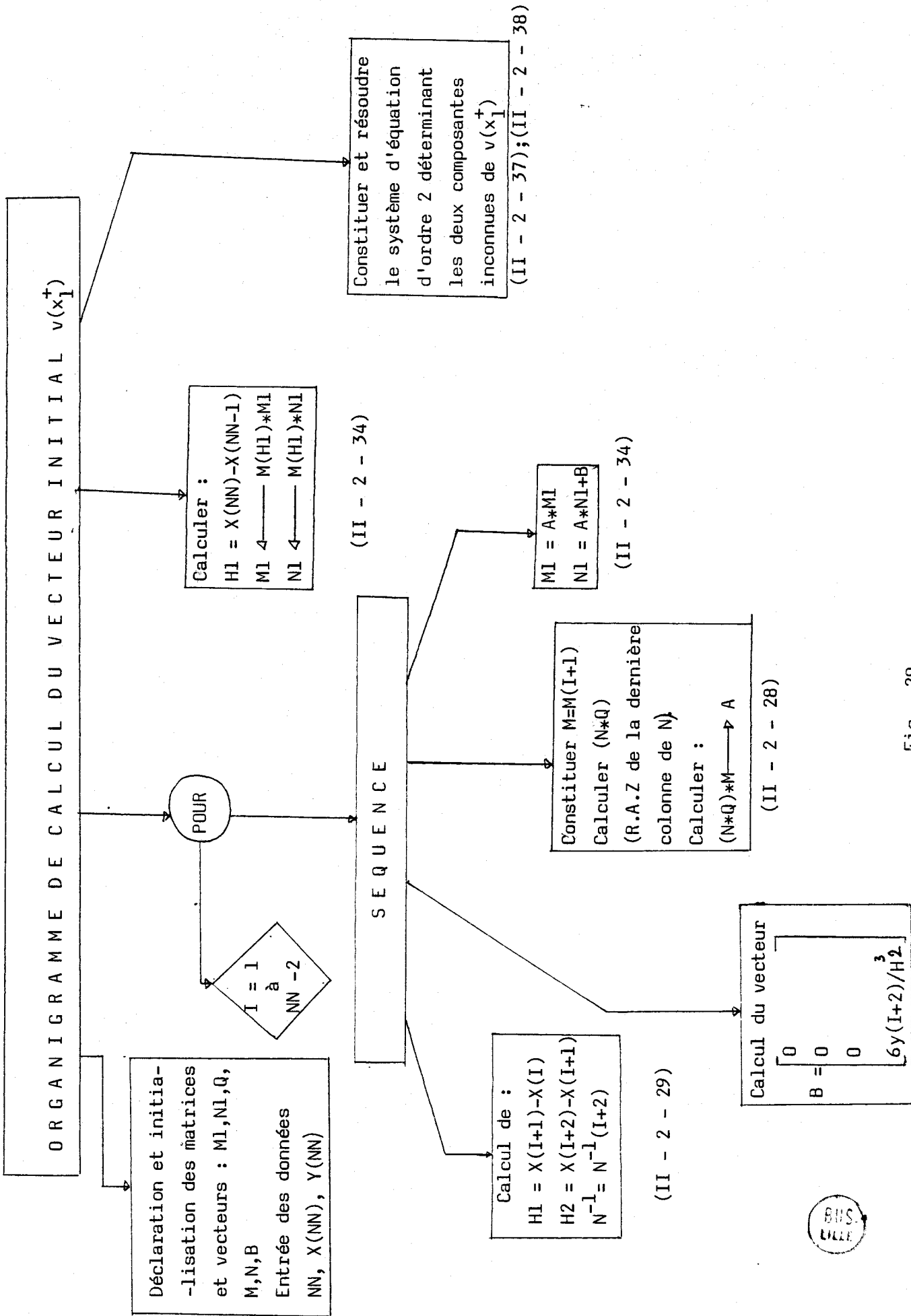
$$\begin{bmatrix} M_{12} & M_{14} \\ P_{22} & P_{24} \end{bmatrix} \cdot \begin{bmatrix} v_2(x_1^+) \\ v_4(x_1^+) \end{bmatrix} = \begin{bmatrix} + Z(x_2) - Z(x_1) \\ Z'(x_n^-) - P_{21} \cdot Z(x_1) - q_2 \end{bmatrix}$$

(II - 2 - 37)

Si les conditions limites en  $x_n$  étaient restreintes à la connaissance de  $Z(x_n)$ , on aurait à résoudre le système suivant :

$$\begin{bmatrix} M_{12} & M_{14} \\ P_{32} & P_{34} \end{bmatrix} \cdot \begin{bmatrix} v_2(x_1^+) \\ v_4(x_1^+) \end{bmatrix} = \begin{bmatrix} Z(x_2) - Z(x_1) \\ - P_{31} \cdot Z(x_1) - q_3 \end{bmatrix}$$

(II - 2 - 38)



BUS.  
1772

Les vecteurs  $v(x_1^+)$  ;  $\forall i=2, \dots, n-1$  se déduisent directement de  $v(x_1^+)$  à partir de l'équation (II - 2 - 22).

L'organigramme de calcul du vecteur  $v(x_1^+)$  est illustré sur la Fig - 20.

II - 2 - 5 : REALISATION DU PROCESSUS INTERPOLATEUR :

Nous proposons d'explicitier la méthodologie de calcul de la loi de commande optimale du processus interpolateur en utilisant un calculateur hybride (analogique et numérique) conformément au schéma de la figure suivante (Fig - 21).

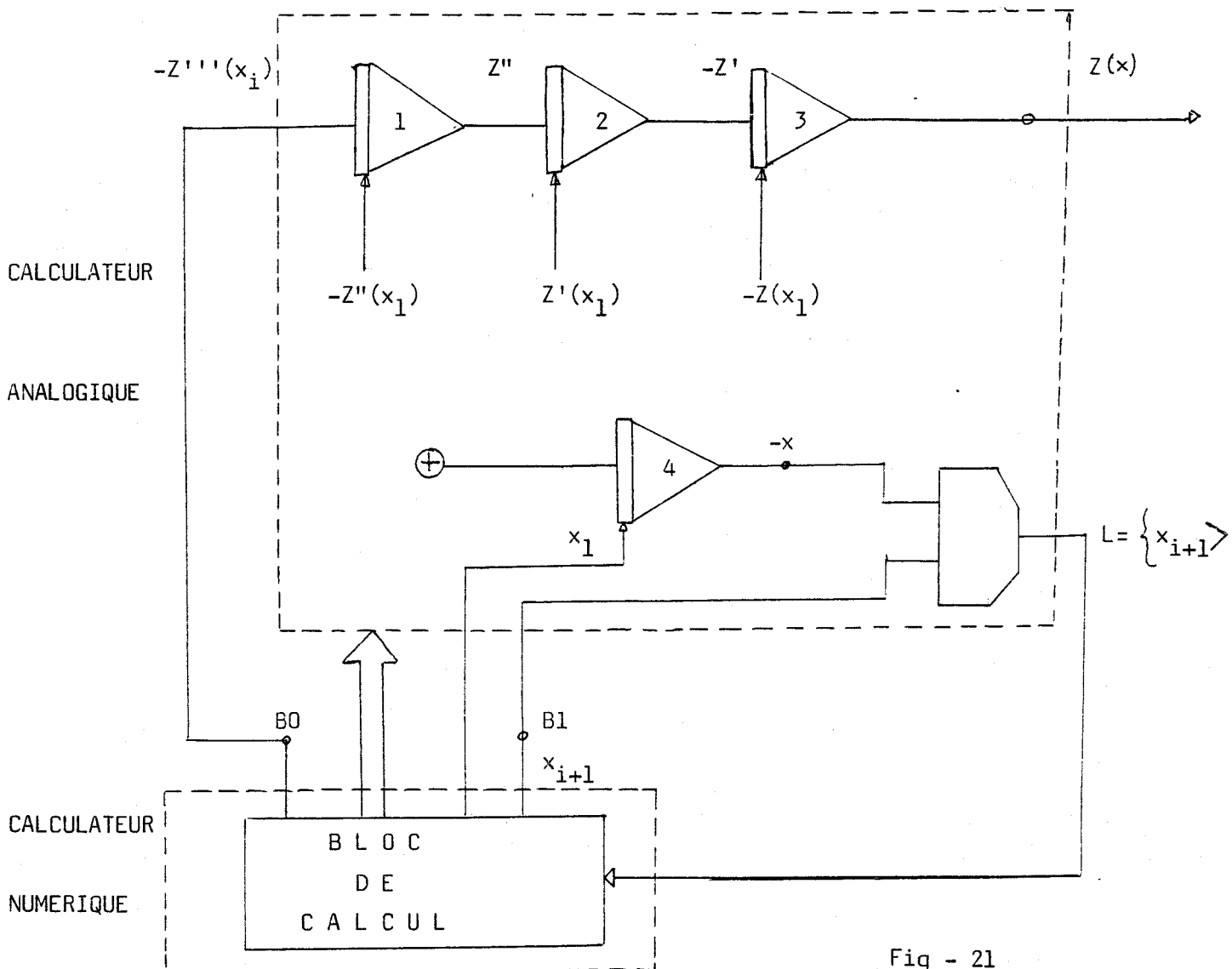


Fig - 21

Le calculateur analogique effectue, par intégration, le calcul de la trajectoire sur tout intervalle  $[x_i, x_{i+1}[$ .

Le calculateur analogique étant en mode initialisation, ce calcul est initialisé au point  $x = x_1$  par la définition, par le calculateur numérique, des composantes initiales des intégrateurs, l'entrée  $-Z'''(x_1)$  de l'intégrateur N°1 et l'entrée  $x_2$  du comparateur (borne B1).

Le calculateur analogique est alors mis en mode opération.

Lorsque la sortie du comparateur commute, cela signifie que  $x$  est sorti de l'intervalle  $[x_1, x_2[$ , le calculateur numérique fournit alors une nouvelle entrée  $-Z'''(x_2)$  sur la borne B0 et la nouvelle borne supérieure  $x_3$  de  $x$  sur la borne B1.

Ce processus est ainsi répété suivant le graphe (Fig - 22) jusqu'à ce que  $x$  devienne égal à  $x_n$ , ce qui déclenche un arrêt de la partie analogique.



\* Mise en conditions initiales du  
calculateur analogique .

\* Calcul de  $v(x_1)$  à partir des  
données du problème.

\* Sortie des valeurs initiales

- $Z(x_1)$  sur intégrateur N°3
- +  $Z'(x_1)$  " " N°2
- $Z''(x_1)$  " " N°1
- +  $x_1$  " " N°4

\* Sortie des consignes

- $Z'''(x_1)$  sur la borne B0
- +  $x_2$  " " " B1

\*  $i := 1$

\* Mise en opération du calculateur  
analogique

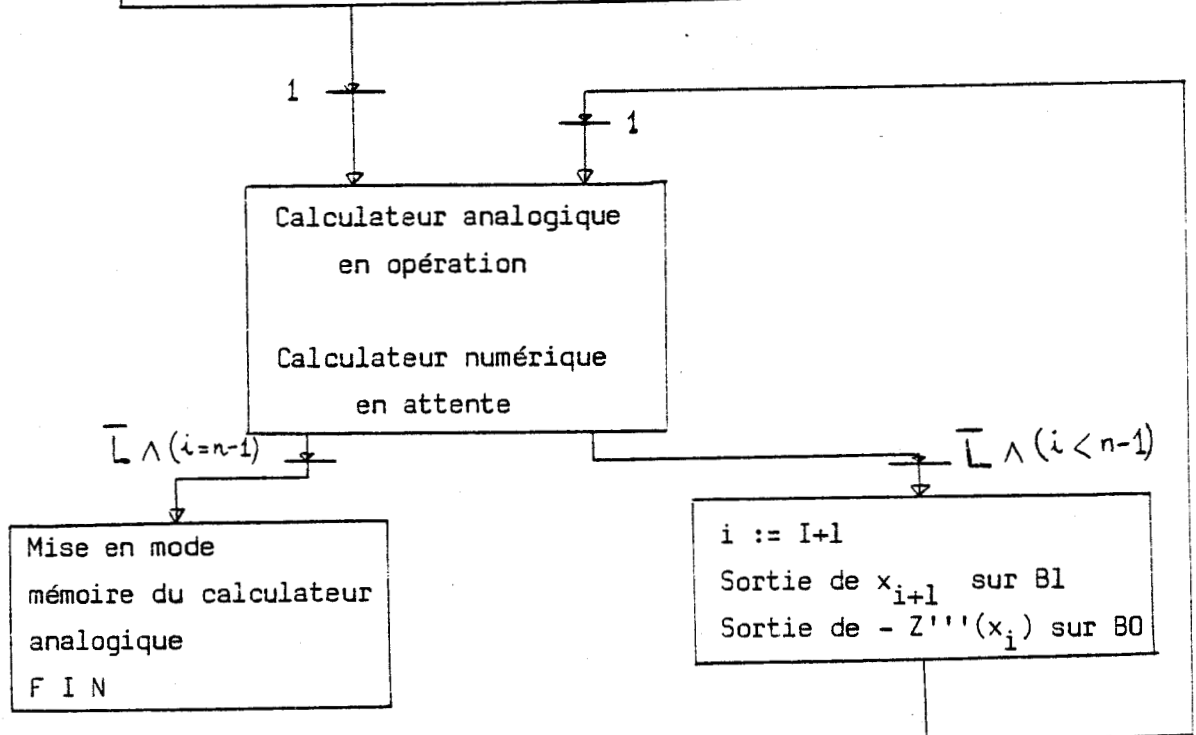
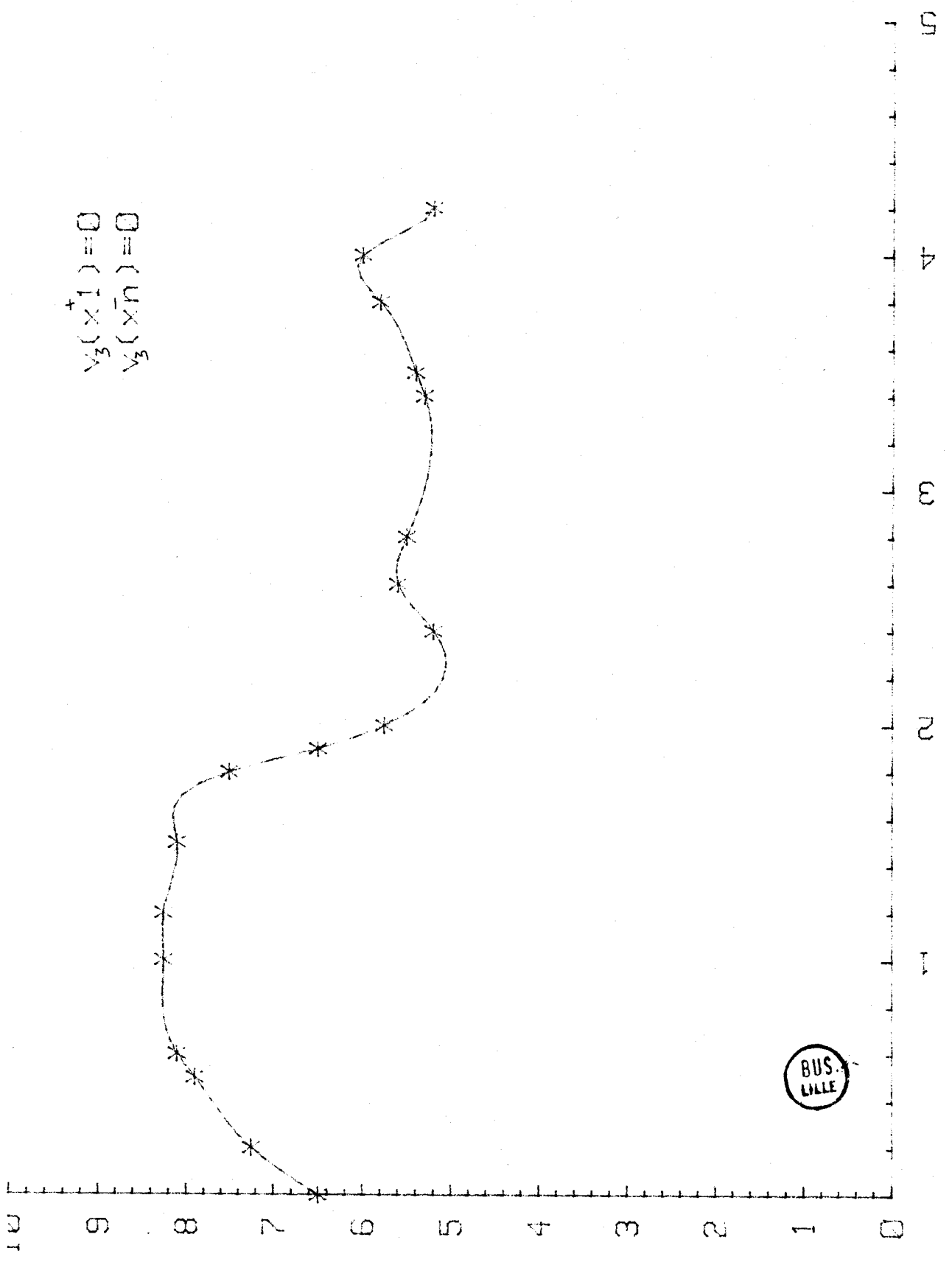
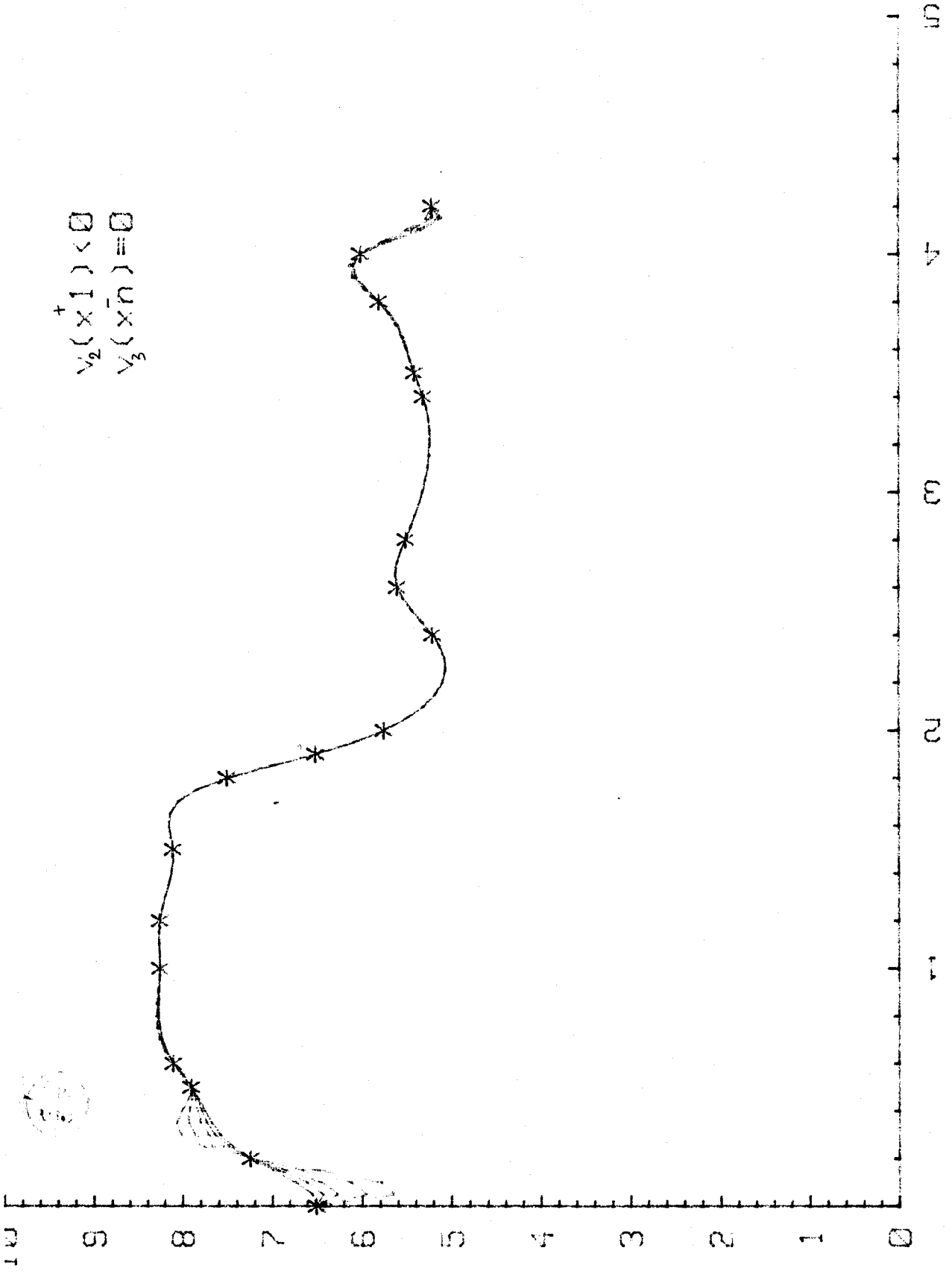


Fig - 22

$$v_3(x_1^+) = 0$$
$$v_3(x_{1n}^-) = 0$$

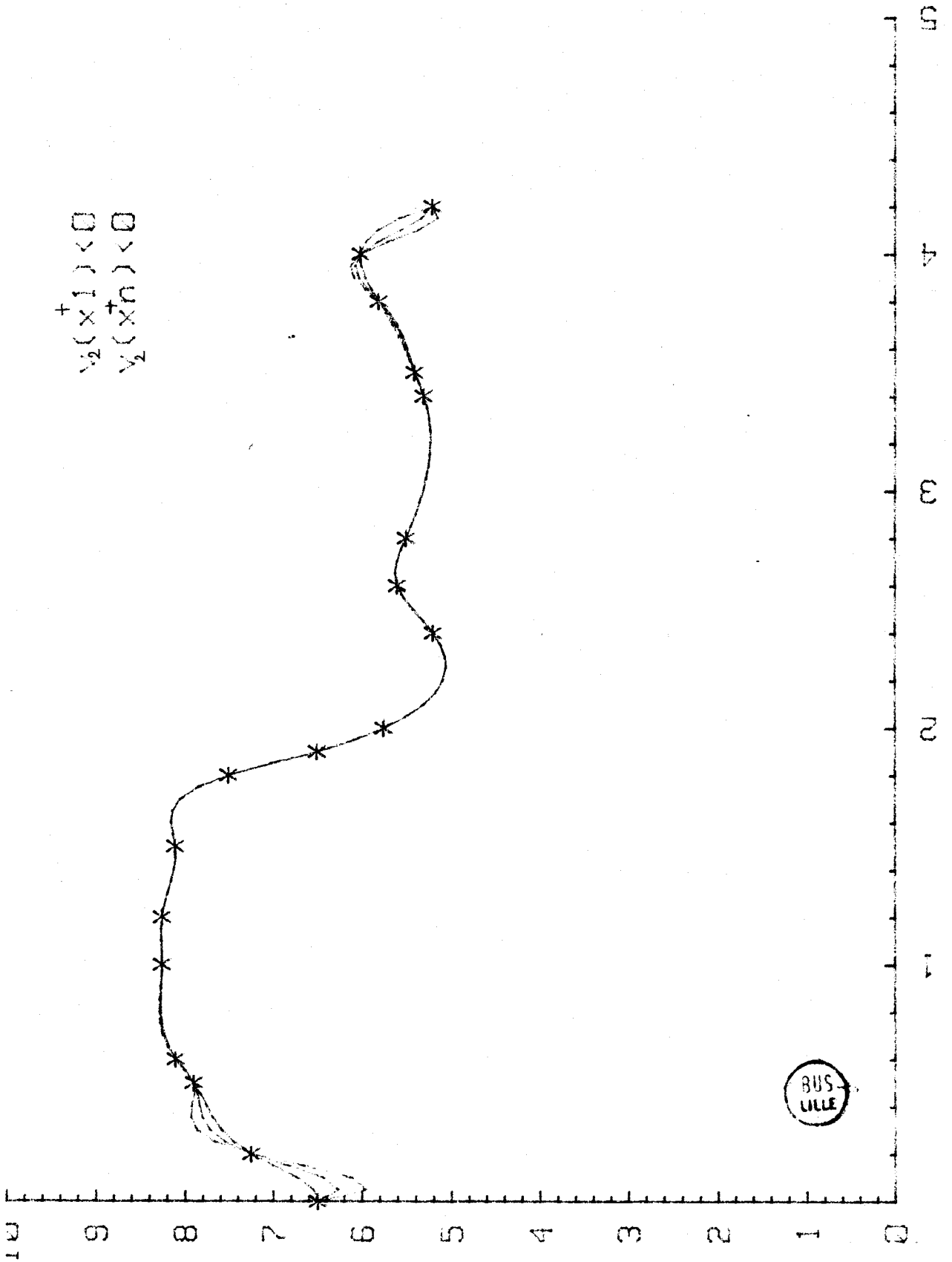


BUS.  
KALLE



$$v_2(x_1) < 0$$

$$v_3(x_1) = 0$$



BUS  
LILLE

II - 3. DETERMINATION DES FONCTIONS SPLINES SOUS TENSION D'ORDRE 2 :

Au paragraphe précédent (§ II - 2), notre intérêt s'est porté sur la recherche et l'obtention de fonctions splines d'interpolation les plus "douces" possibles c'est-à-dire qui minimisent en tout  $x \in [x_1, x_n]$  la fonctionnelle J quadratique de la dérivée seconde de la fonction (cf § II - 2 - 2 ; P4).

De manière à supprimer les points d'inflexion supplémentaires créés sur la courbe d'interpolation cubique précédemment étudiée, on s'intéresse aux fonctions splines qui minimisent une fonctionnelle quadratique des dérivées seconde et première.

Dans ce cas, on réalise un compromis entre la "douceur" et la "longueur" de la courbe, en introduisant dans la fonctionnelle J, un terme supplémentaire représentatif de la longueur.

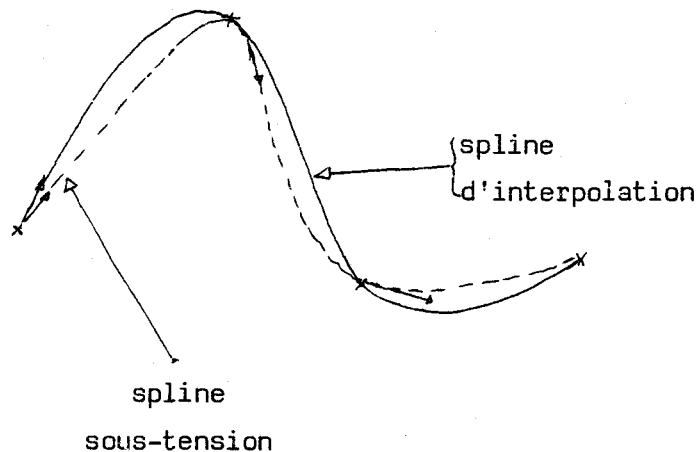


Fig - 23 : Réduction de la longueur d'une courbe par traction sur les extrémités.

Les fonctions caractérisées par cette propriété sont notées fonctions splines sous-tension d'ordre 2. (D. PILCHER , 1973).

Nous proposons, dans un premier volet, de déterminer ces fonctions à partir de l'algorithme du processus interpolateur développé au paragraphe précédent.

Dans un deuxième volet, nous généralisons cette procédure aux fonctions splines sous-tension d'ordre q notées  $L_q$  splines (où L est un opérateur différentiel linéaire invariant). (CORREC , 1980).

II - 3 - 1 : PROPRIETES DES FONCTIONS SPLINES  
SOUS-TENSION D'ORDRE 2 :

Une fonction  $Z(x)$  définie sur un réseau  
•  $\{x_i, y_i ; \forall i = 1, \dots, n\}$  ordonné par la relation :

$$x_1 < x_2 < \dots < x_{n-1} < x_n$$

est dite fonction spline sous-tension d'ordre 2 si elle satisfait les propriétés suivantes :

Q1 )  $Z(x)$  ainsi que ses dérivées à l'ordre 1 et 2 sont continues sur  $[x_1, x_n]$  .

Q2 )  $Z(x)$  interpole le réseau  $\{x_i, y_i\}$  :

$$Z(x_i) = y_i ; \forall i=1, \dots, n.$$

De plus, les dérivées premières aux extrémités du réseau peuvent être imposées.

Q3 )  $Z(x)$  minimise dans l'espace des fonctions définies sur  $[x_1, x_n]$ , satisfaisant les propriétés Q1), Q2), la fonctionnelle de forme suivante :

$$\forall x \in [x_1, x_n] ; \Phi(f(x)) = \int_{x_1}^{x_n} (f''(x)^2 + \sigma^2 f'(x)^2) dx.$$

$\sigma$  est noté coefficient de tension .

REMARQUES :

$\sigma = 0$  correspond au cas particulier des fonctions splines polynomiales par morceaux d'ordre 2 pour lesquelles la méthode proposée a été développée au paragraphe précédent (cf § II - 2).

Le terme introduit avec  $\sigma \neq 0$  réalise un compromis entre les propriétés de longueur minimale et de douceur maximale de l'interpolation. Pour de faibles valeurs de  $\sigma$ , les courbes correspondantes tendent vers la courbe d'interpolation cubique.

Lorsque  $\sigma$  tend vers l'infini, la fonction cherchée tend de manière manifeste vers l'interpolation linéaire par morceaux, qui minimise la longueur des courbes d'interpolation

$$\left( \int_{x_1}^{x_n} (f'(x))^2 dx \right).$$

En interprétant  $f(x)$  comme la trajectoire du système dynamique suivant : (cf Fig - 24)

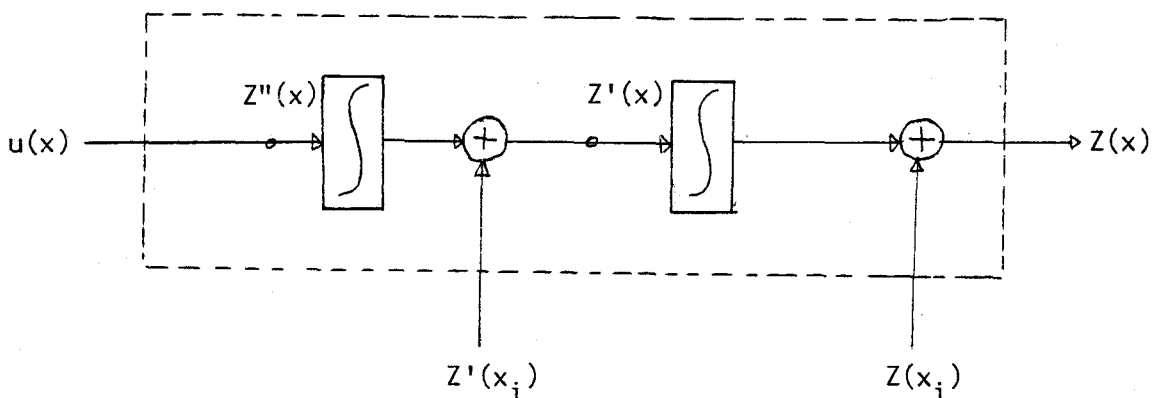


Fig - 24

dont le point représentatif évolue suivant la variable  $x$ , nous proposons de déterminer son expression analytique dans tout intervalle  $[x_i, x_{i+1}]$  où elle est définie, à partir des équations issues de la théorie de la commande optimale des systèmes linéaires stationnaires.



II - 3 - 2 : EQUATIONS DE LA COMMANDE OPTIMALE :

Si  $\bar{Z}^T(x) = [Z(x), Z'(x)]$  est le vecteur état du système (cf Fig - 24), alors une représentation dans l'espace d'état est la suivante : (II - 3 - 1)

$$\begin{aligned} \frac{d}{dx}(Z(x)) &= A \cdot \bar{Z}(x) + b \cdot u(x) \\ Z(x) &= C \cdot \bar{Z}(x) \end{aligned} \quad (\text{II} - 3 - 1)$$

où :  $Z(x)$ ,  $Z'(x)$  représentent respectivement la fonction  $f(x)$  et sa dérivée première  $f'(x)$ .

$A \in \mathbb{R}^{2 \times 2}$  ;  $b \in \mathbb{R}^{2 \times 1}$ ,  $c \in \mathbb{R}^{1 \times 2}$  sont définis par :

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} ; b = \begin{bmatrix} 0 \\ 1 \end{bmatrix} ; c = [1, 0] \quad (\text{II} - 3 - 2)$$

On caractérise une trajectoire particulière dite "trajectoire optimale" par le fait qu'elle minimise la fonctionnelle de l'état et de la commande :

$$J(Z(x), u(x)) = +\frac{1}{2} \int_{x_1}^{x_n} (\bar{Z}^T(x) \cdot P \cdot \bar{Z}(x) + u^2(x)) dx$$

$$(\text{II} - 3 - 3)$$

$$P = \begin{bmatrix} 0 & 0 \\ 0 & \sigma^2 \end{bmatrix} \in \mathbb{R}^{2 \times 2} ; \sigma > 0 ; P > 0$$

En tout point  $x$  pour lequel n'existe pas de contrainte sur la fonction ( $x \neq x_1, x_2, x_3, \dots, x_n$ ), la variation  $\delta J$  de la fonctionnelle  $J$  (cf II - 3 - 3) consécutive à une variation admissible  $\delta u(x)$  de la commande  $u(x)$  est équivalente à la variation  $\delta H$  de l'hamiltonien  $H$  par rapport à  $u(x)$ , en tout point  $x(x \neq x_i ; \forall i=1, \dots, n)$  de la trajectoire optimale :  
(LARMINAT & THOMAS, 1977)

$$\delta J = \int_{x_1}^{x_n} \delta H. dx \quad (\text{II} - 3 - 4)$$

où  $H$  est la fonction scalaire définie par :

$$H(u(x), Z(x), \Psi(x)) = -\frac{1}{2} (u^2(x) + \bar{Z}^T(x) \cdot P \cdot \bar{Z}(x)) + \Psi^T(x) (A \cdot \bar{Z}(x) + b \cdot u(x)) \quad (\text{II} - 3 - 5)$$

La commande  $u(x)$  n'étant soumise à aucune contrainte, il s'en suit :

$$\frac{dH}{du(x)} = 0 \implies u(x) = b^T \cdot \Psi(x) ; \forall x \in ]x_i, x_{i+1}[ \quad (\text{II} - 3 - 6)$$

L'évolution du vecteur  $\Psi(x) \in \mathbb{R}^{2 \times 1}$ , caractéristique de la commande, est régie par le système d'équations différentiel

d'HAMILTON - PONTRYAGUINE suivant :

$$\frac{d\Psi(x)}{dx} = -\frac{dH}{d\bar{Z}(x)} ; \forall x \in ]x_i, x_{i+1}[ ; \forall i=1, \dots, n-1 \quad (\text{II} - 3 - 7)$$

On en déduit la relation (II - 3 - 8) :

$$\frac{d\Psi(x)}{dx} = -A^T \cdot \Psi(x) + P \cdot \bar{Z}(x) ; \forall x \in ]x_i, x_{i+1}[ ; \forall i=1, \dots, n-1$$

(II - 3 - 8)

Ces relations (II - 3 - 6 ; II - 3 - 8) définissent  $u(x)$  à partir d'un système différentiel qu'on peut intégrer à partir des conditions initiales issues des conditions de transversalité sur  $\Psi(x)$ , aux extrémités du réseau.



Ces dernières s'écrivent :

$$\delta J = (\Psi^T(x) \cdot \delta \bar{Z}(x)) = 0 \quad (\text{II} - 3 - 9)$$

$$\begin{matrix} x = x_1 \\ x = x_n \end{matrix}$$

En particulier, comme dans le cas des fonctions splines cubiques, si une composante de  $Z(x_1)$  (resp  $Z(x_n)$ ) n'est pas fixée, la composante correspondante de  $\Psi(x_1)$  (resp  $\Psi(x_n)$ ) est nulle.

Pour déterminer la fonction  $Z(x)$ , on convient de la représenter par son processus interpolateur défini à partir des relations précédentes et dont nous proposons d'expliciter la structure, puis le calcul.

II - 3 - 3 : STRUCTURE DU PROCESSUS INTERPOLATEUR :

Les relations définissant la commande  $u(x)$  et le vecteur  $\Psi(x)$ , ainsi que la formulation du problème initial dans l'espace d'état, peuvent se mettre sous la <sup>forme</sup> matricielle suivante (II - 3 - 10) :

$$\frac{d}{dx} \begin{bmatrix} \bar{Z}(x) \\ \Psi(x) \end{bmatrix} = \begin{bmatrix} A & bb^T \\ P & -A^T \end{bmatrix} * \begin{bmatrix} \bar{Z}(x) \\ \Psi(x) \end{bmatrix} ;$$

$$\forall x \in [x_i, x_{i+1}[ ; i=1, \dots, n-1$$

(II - 3 - 10)

Notons  $J$  la matrice de ce système et  $\bar{z}(x) \in \mathbb{R}^{4 \times 1}$  le vecteur d'état correspondant, définis respectivement par :

$$J = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & \sigma^2 & -1 & 0 \end{bmatrix} \quad \text{et} \quad \bar{z}^T(x) = [Z(x), Z'(x), \Psi_1(x), \Psi_2(x)]$$

(II - 3 - 11)

Effectuons la transformation suivante sur les composantes du vecteur  $\Psi(x)$  :

$$\begin{aligned}\Psi_1(x) &\longrightarrow \Psi_2(x) \\ \Psi_2(x) &\longrightarrow -\Psi_1(x)\end{aligned}\tag{II - 3 - 12}$$

Le système (II - 3 - 10) est alors représenté sous la forme plus simple suivante :

$$\frac{d}{dx} (v(x)) = \tilde{A} \cdot v(x)\tag{II - 3 - 13}$$

avec :

$$v^T(x) = [Z(x), Z'(x), \Psi_2(x), -\Psi_1(x)]$$

$$\tilde{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & \sigma^2 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}\tag{II - 3 - 14}$$

Cette dernière relation correspond au schéma de stimulation dont la structure est la suivante (cf Fig - 25)

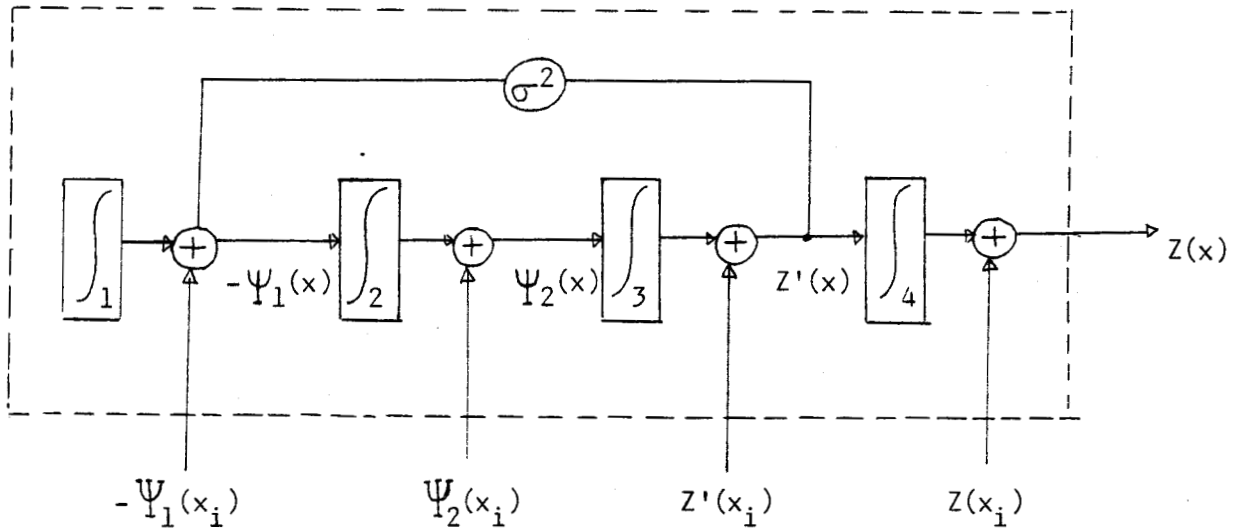


Fig - 25 : Structure du processus interpolateur d'une fonction spline sous-tension  $Z(x)$

L'équation de progression définissant numériquement  $Z(x)$  (ou  $f(x)$ ) sur tout intervalle  $]x_i, x_{i+1}[$  est obtenue par intégration de l'équation d'état (II - 3 - 13) à partir de conditions initiales  $v(x_i^+)$ .

Il vient :

$$\begin{aligned} \forall i = 1, \dots, n-1 \\ \forall h \in [0, x_{i+1} - x_i[ \end{aligned} \tag{II - 3 - 15}$$

$$v(x_i + h) = M(h) \cdot v(x_i^+)$$

où  $M(h) = \exp(\tilde{A} \cdot h)$

$$\sigma \neq 0 \Rightarrow M(h) = \begin{bmatrix} 1 & \frac{sh\sigma h}{\sigma} & \frac{ch\sigma h - 1}{\sigma^2} & \frac{sh\sigma h - \sigma h}{\sigma^3} \\ 0 & ch\sigma h & \frac{sh\sigma h}{\sigma} & \frac{ch\sigma h - 1}{\sigma^2} \\ 0 & \sigma sh\sigma h & ch\sigma h & \frac{sh\sigma h}{\sigma} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (II - 3 - 16)$$

La trajectoire  $Z(x)$  générée par le schéma précédent étant définie par morceaux ( $x \in [x_i, x_{i+1}[$ ,  $i=1, \dots, n-1$ ) on est amené à compléter le système précédent (cf Fig - 26) en lui adjoignant un bloc de calcul fournissant les conditions initiales  $v(x_i^+)$  à partir des données conformément à la Fig - 26 :

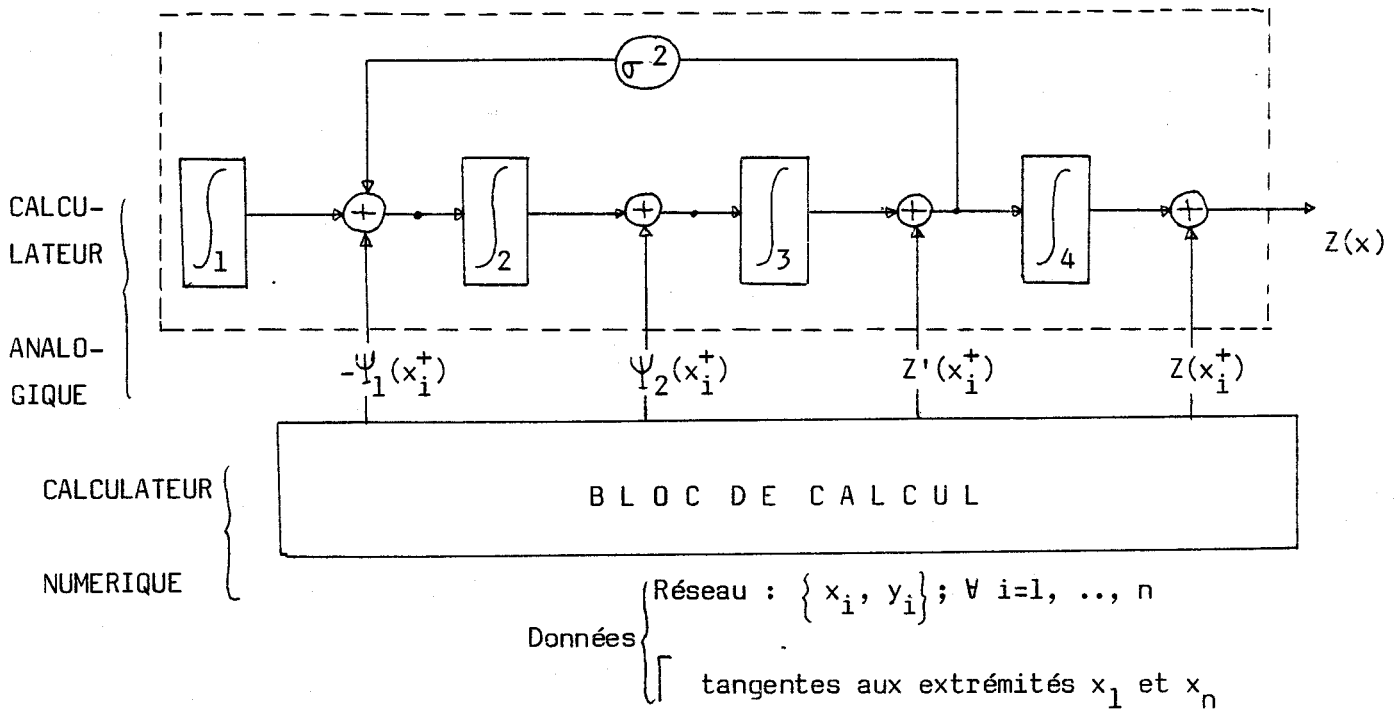


Fig - 26 : Simulation de la trajectoire spline sous-tension  $Z(x)$  sur calculateur hybride.

Le calcul des conditions initiales  $v(x_i^+)$  fait l'objet de la présente étude.

II - 3 - 4 : CALCUL EXPLICITE DU PROCESSUS INTERPOLATEUR :

La procédure de calcul est en tout point parallèle à celle développée précédemment ( § II - 2 - 4 ).

La détermination des vecteurs  $v(x_i^+)$  en tout point  $x_i$  du réseau est basée sur l'existence d'une équation de récurrence linéaire entre les vecteurs  $v(x_{i+1}^+)$  et  $v(x_i^+)$ .

Cette équation est établie de la manière suivante, à partir des propriétés Q1 et Q2 (cf II - 3 - 1) des fonctions splines sous-tension d'ordre 2 et de l'équation de progression définie au paragraphe précédent (cf II - 3 - 15).

II - 3 - 4 - 1 ) Relation entre  $v(x_{i+1}^+)$  et  $v(x_i^+)$  :

$Z(x)$  étant de classe  $C^2$  sur l'intervalle  $[x_1, x_n]$ , l'équation de progression :

$$v(x_{i+1}^-) = M(x_{i+1} - x_i) \cdot v(x_i^+) \quad (\text{II - 3 - 17})$$

est vérifiée en  $x_{i+1}^+$  uniquement pour les trois premières composantes, ce qui se traduit par : (II - 3 - 18)



$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot v(x_{i+1}^+) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot v(x_{i+1}^-)$$

(II - 3 - 18)

La dernière composante est issue de la donnée de  $Z(x)$  au point  $x_{i+2}$ , par la première ligne de l'équation de progression, sur l'intervalle  $[x_{i+1}, x_{i+2}]$ .

$$z(x_{i+2}) = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \cdot M(x_{i+2} - x_{i+1}) \cdot v(x_{i+1}^+)$$

(II - 3 - 19)

Ces relations (II - 3 - 18 et II - 3 - 19) peuvent être groupées sous la forme matricielle suivante : (II - 3 - 20)

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ m_{11} & m_{12} & m_{13} & m_{14} \end{bmatrix} \cdot v(x_{i+1}^+) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} *$$

$$\begin{bmatrix} 1 & \frac{\text{sh}\sigma \cdot h_1}{\sigma} & \frac{\text{ch}\sigma \cdot h_1 - 1}{\sigma^2} & \frac{\text{sh}\sigma h_1 - \sigma h_1}{\sigma^3} \\ 0 & \text{ch}\sigma \cdot h_1 & \frac{\text{sh}\sigma \cdot h_1}{\sigma} & \frac{\text{ch}\sigma \cdot h_1 - 1}{\sigma^2} \\ 0 & \text{sh}\sigma \cdot h_1 & \text{ch}\sigma h_1 & \frac{\text{sh}\sigma \cdot h_1}{\sigma} \\ 0 & 0 & 0 & 1 \end{bmatrix} * v(x_i^+) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ Z(x_{i+2}) \end{bmatrix}$$

$$\text{où } m_{11} = 1 ; m_{12} = \frac{\text{sh}\sigma \cdot h_2}{\sigma} ; m_{13} = \frac{\text{ch}\sigma \cdot h_2 - 1}{\sigma^2} ; m_{14} = \frac{\text{sh}\sigma h_2 - \sigma h_2}{\sigma^3}$$

$$h_1 = h_{i+1} = x_{i+1} - x_i$$

$$h_2 = h_{i+2} = x_{i+2} - x_{i+1}$$

soit sous forme abrégée :

$$N(h_{i+2}) \cdot v(x_{i+1}^+) = Q \cdot M(h_{i+1}) \cdot v(x_i^+) + Y_1(i+2) \quad (\text{II} - 3 - 21)$$

La matrice  $N(h_{i+2})$  est inversible et son inverse est calculée de manière formelle. Il vient la relation :

$$v(x_{i+1}^+) = A(i+2) \cdot v(x_i^+) + B(i+2) \quad (\text{II} - 3 - 22)$$

avec  $A(i+2) = N^{-1}(h_{i+2}) \cdot Q \cdot M(h_{i+1}) \quad (\text{II} - 3 - 23)$

$$B(i+2) = N^{-1}(h_{i+2}) \cdot Y_1(i+2) \quad (\text{II} - 3 - 24)$$

où  $N^{-1}(h_{i+2})$  est donné par :

$$N^{-1}(h_{i+2}) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \frac{-1}{\text{Det } N} & \frac{-\text{sh}\sigma h_{i+2}}{\sigma \cdot \text{Det } N} & \frac{-\text{ch}\sigma h_{i+2} + 1}{\sigma^2 \cdot \text{Det } N} & \frac{1}{\text{Det } N} \end{bmatrix} \quad (\text{II} - 3 - 25)$$

et :

$$\text{Det } N = \frac{\text{sh}\sigma h_{i+2} - \sigma h_{i+2}}{\sigma^3} = \frac{\theta_2}{\sigma^3} ; \sigma > 0$$

En notant  $\theta_j = \text{sh}\sigma \cdot h_{i+j} - \sigma h_{i+j} ; \forall j = 1, 2, A \text{ et } B$

s'écrivent alors : (II - 3 - 26)

$$A(i+2) = \begin{bmatrix} 1 & \frac{\text{sh}\sigma h_1}{\sigma} & \frac{\text{ch}\sigma h_1 - 1}{\sigma^2} & \frac{\theta_1}{\sigma^3} \\ 0 & \text{ch}\sigma h_1 & \frac{\text{sh}\sigma h_1}{\sigma} & \frac{\text{ch}\sigma h_1 - 1}{\sigma^2} \\ 0 & \sigma \text{sh}\sigma h_1 & \text{ch}\sigma h_1 & \frac{\text{sh}\sigma h_1}{\sigma} \\ -\frac{\sigma^3}{\theta_2} & -\frac{\sigma^2}{\theta_2} \text{sh}\sigma(h_1+h_2) & \frac{\sigma}{\theta_2} (1 - \text{ch}\sigma(h_1+h_2)) & -\left[ \frac{\theta_1 + \text{sh}\sigma(h_1+h_2) - (\text{sh}\sigma h_1 + \text{sh}\sigma h_2)}{\theta_2} \right] \end{bmatrix}$$

$$B^T(i+2) = \begin{bmatrix} 0 & 0 & 0 & \frac{\sigma^3}{\theta_2} Z(x_{i+2}) \end{bmatrix} \quad (\text{II} - 3 - 26)$$

L'équation de récurrence (II - 3 - 22) itérée jusqu' à l'extrémité finale permet d'exprimer une relation entre  $v(x_1^+)$  et  $v(x_n^-)$  : (II - 3 - 27)

$$v(x_n^-) = M(h_n) \cdot \left\{ \prod_{k=3}^n A(k) \right\} \cdot v(x_1^+) + M(h_n) \cdot \left\{ \sum_{k=3}^n \left( \prod_{e=k+1}^n A(e) \right) B(k) \right\}$$

(II - 3 - 27)

Cette dernière relation nous permet, comme au paragraphe précédent, d'expliciter le calcul des composantes du vecteur initial  $v(x_1^+)$ .

II - 3 - 4 - 2    Calcul de  $v(x_1^+)$  :

Les données initiales étant  $Z(x_1)$  et  $Z(x_n)$ , on définit deux autres données en calculant la variation de la fonctionnelle (II - 3 - 3) consécutive à des variations  $\delta Z(x_1)$  et  $\delta Z(x_n)$ . Il vient les conditions aux limites suivantes (cf II - 3 - 9)

$$\begin{cases} \delta J = \Psi^T(x_1^+) \cdot \delta Z(x_1^+) = 0 & \text{(II - 3 - 28)} \\ \delta J = \Psi^T(x_n^-) \cdot \delta Z(x_n^-) = 0 & \text{(II - 3 - 29)} \end{cases}$$

Si  $Z'(x_1^+)$  (resp  $Z'(x_n^-)$ ) n'est pas fixé, (II - 3 - 28) (resp. II - 3 - 29) établit la nullité de la troisième composante de  $v(x_1^+)$  (resp  $v(x_n^-)$ ).

$$\Psi_2(x_1^+) = 0 \quad ; \quad (\text{resp } \Psi_2(x_n^-) = 0) \quad \text{(II - 3 - 30)}$$

$v(x)$  étant le vecteur de composantes ;

$$v^T(x) = [Z(x), Z'(x), \Psi_2(x), -\Psi_1(x)]$$

il vient les conditions limites suivantes :

	}	$v_1(x_1)$ connu	
	et	$v_2(x_1)$ fixé sinon $v_3(x_1) = 0$	
et	}	$v_1(x_n)$ connu	
	et	$v_2(x_n)$ fixé sinon $v_3(x_n) = 0$	

(II - 3 - 31)

Les deux composantes manquantes sont obtenues à partir :

\* De l'équation de progression (II - 3 - 17) non utilisée au pas précédent :

$$Z(x_2) = [1, 0, 0, 0] \cdot M(x_2-x_1) \cdot v(x_1^+) \quad (\text{II} - 3 - \dots)$$

\* De la deuxième ligne ou à défaut de la 3ème ligne de l'équation itérée (II - 3 - 27) qui peut s'écrire sous la forme compacte :

$$v(x_n^-) = p \cdot v(x_1^+) + q \quad (\text{II} - 3 - 33)$$

avec  $p \in \mathbb{R}^{4 \times 4}$ ,  $q \in \mathbb{R}^4$

Les composantes inconnues de  $v(x_1^+)$  sont alors solutions du système :

\* Si  $v_2(x_1^+)$  fixé et  $v_3(x_n^-) = 0$  :

$$\begin{bmatrix} M_{13} & M_{14} \\ P_{33} & P_{34} \end{bmatrix} \cdot \begin{bmatrix} v_3(x_1^+) \\ v_4(x_1^+) \end{bmatrix} = \begin{bmatrix} v_1(x_2) - v_1(x_1) - M_{12} \cdot v_2(x_1^+) \\ -P_{31} \cdot v_1(x_1^+) - P_{32} \cdot v_2(x_1^+) - q_3 \end{bmatrix}$$

(II - 3 - 34)

\* Si  $v_2(x_1^+)$  et  $v_2(x_n^-)$  fixés :

$$\begin{bmatrix} M_{13} & M_{14} \\ P_{23} & P_{24} \end{bmatrix} \cdot \begin{bmatrix} v_3(x_1^+) \\ v_4(x_1^+) \end{bmatrix} = \begin{bmatrix} v_1(x_2) - v_1(x_1) - M_{12} \cdot v_2(x_1^+) \\ v_2(x_n^-) - P_{21} \cdot v_1(x_1) - P_{22} \cdot v_2(x_1^+) \end{bmatrix}$$

(II - 3 - 35)

\* Si  $v_3(x_1^+)$  et  $v_3(x_n^-)$  sont fixés à 0 :

$$\begin{bmatrix} M_{12} & M_{14} \\ P_{32} & P_{34} \end{bmatrix} \cdot \begin{bmatrix} v_2(x_1^+) \\ v_4(x_1^+) \end{bmatrix} = \begin{bmatrix} v_1(x_2) - v_1(x_1) \\ -P_{31} \cdot v_1(x_1) - q_3 \end{bmatrix} \quad (\text{II - 3 - 36})$$

\* Si  $v_2(x_n^-)$  fixé et  $v_3(x_1^+) = 0$  :

$$\begin{bmatrix} M_{12} & M_{14} \\ P_{22} & P_{24} \end{bmatrix} \cdot \begin{bmatrix} v_2(x_1^+) \\ v_4(x_1^+) \end{bmatrix} = \begin{bmatrix} v_1(x_2) - v_1(x_1) \\ v_2(x_n^-) - P_{21} \cdot v_1(x_1) - q_2 \end{bmatrix}$$

(II - 3 - 37)

Après résolution de ce système, on déduit les vecteurs  $v(x_i^+)$  ;  $i=2, \dots, n$  à partir de l'équation de progression (II - 3 - 17).

L'expression analytique de la fonction spline sous-tension d'ordre 2 dans tout intervalle  $[x_i, x_{i+1}]$  est :

$$\forall x \in [x_i, x_{i+1}] ; \quad \forall i = 1, \dots, n-1 \quad (\text{II} - 3 - 34)$$

$$\begin{aligned} f(x) = & v_1(x_i^+) + \frac{v_2(x_i^+)}{\sigma} \cdot \text{sh} \sigma \cdot (x-x_i) + \frac{v_3(x_i^+)}{\sigma^2} \cdot (\text{ch} \sigma \cdot (x-x_i) - 1) \\ & + \frac{v_4(x_i^+)}{\sigma^3} \cdot (\text{sh} \sigma \cdot (x-x_i) - \sigma \cdot (x-x_i)) \end{aligned}$$

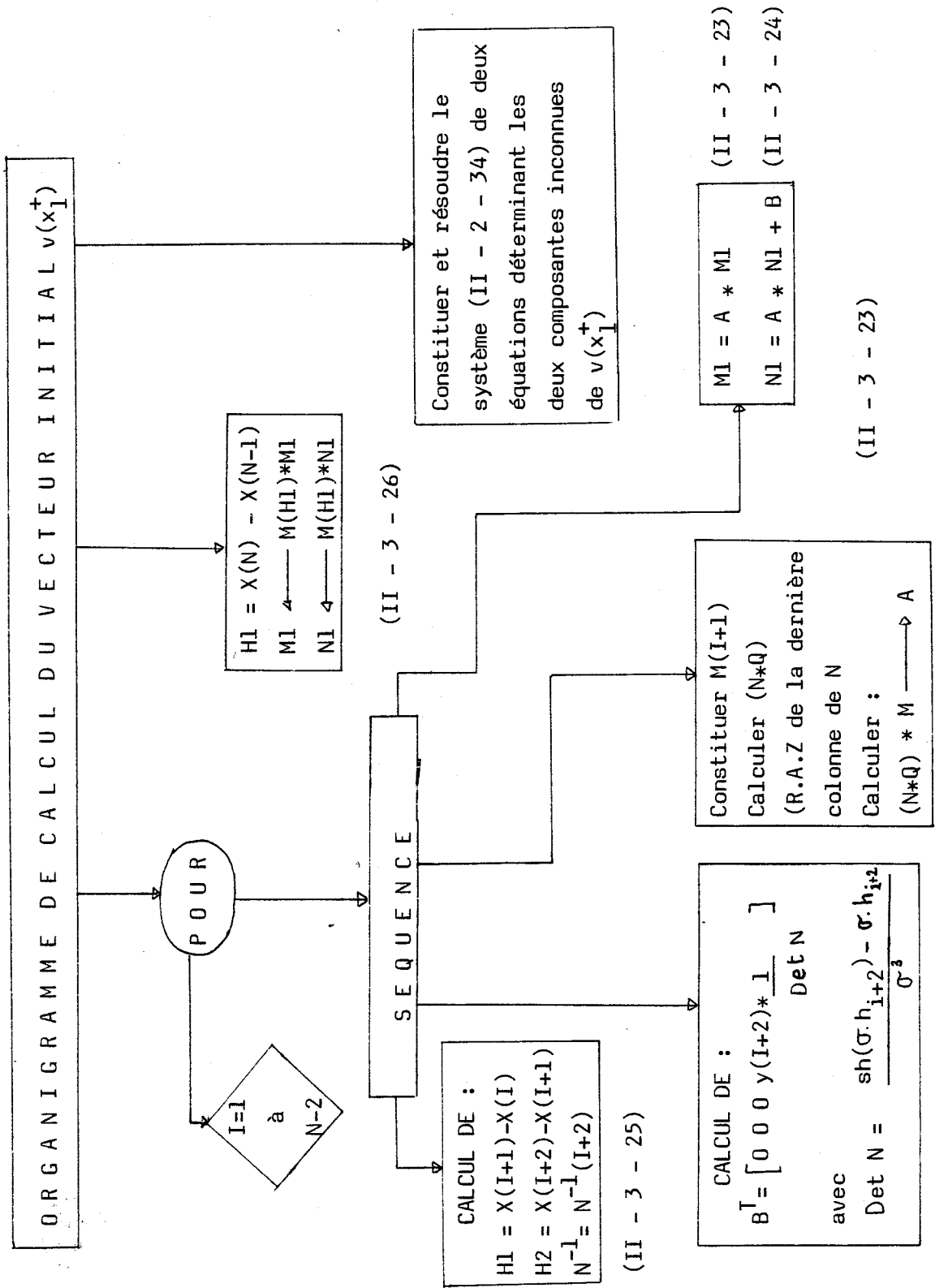


II - 3 - 5 : REALISATION DU PROCESSUS INTERPOLATEUR :

La réalisation est assurée de manière identique à celle exposée pour les fonctions splines d'interpolation d'ordre 2 (cf § II - 2).

Nous présentons ci-joint l'organigramme correspondant à la procédure de calcul du vecteur  $v(x_1^+)$  (Fig - 27).

AUS  
LILL

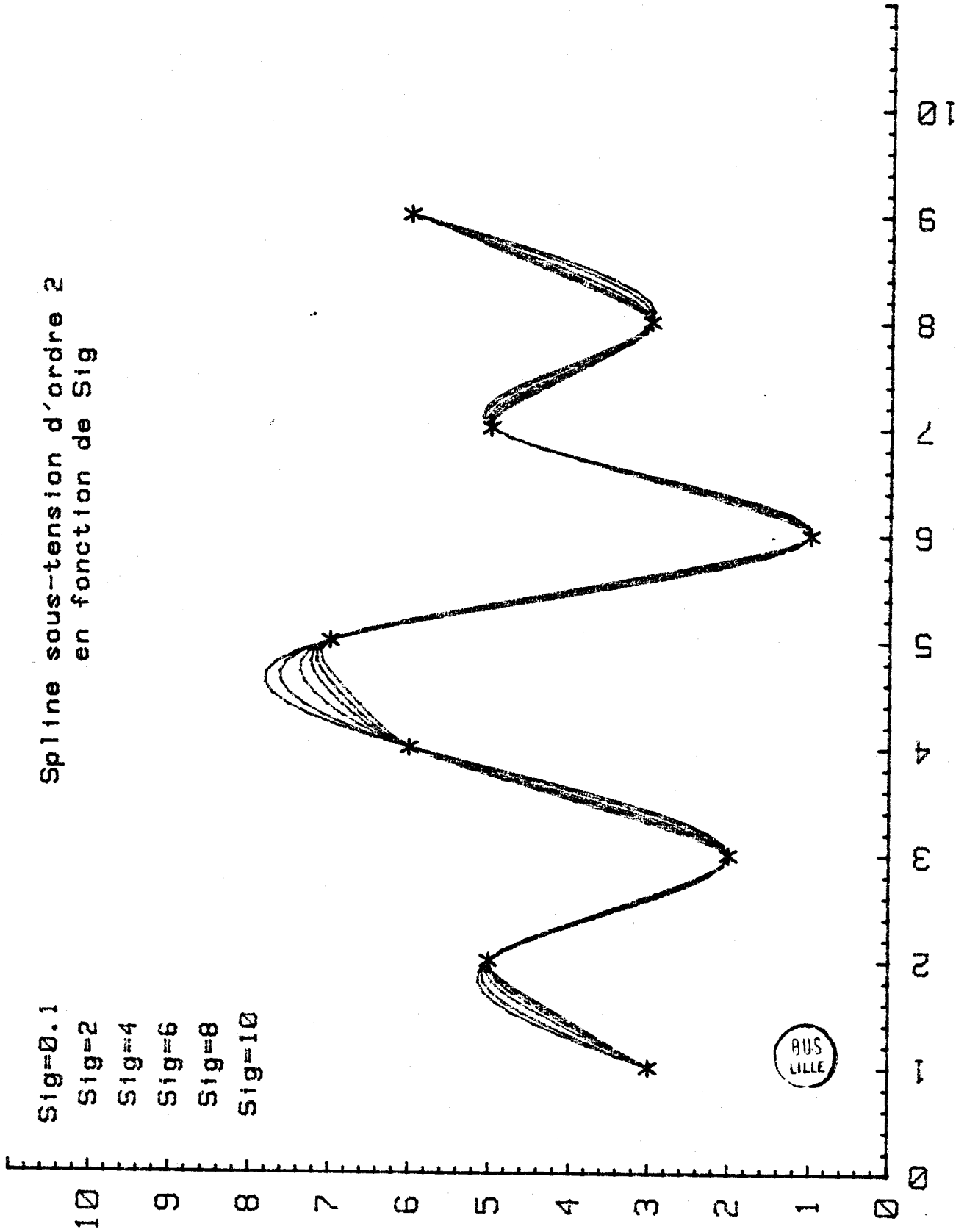


(II - 3 - 24)

Fig - 27

Spline sous-tension d'ordre 2  
en fonction de Sig

- Sig=0.1
- Sig=2
- Sig=4
- Sig=6
- Sig=8
- Sig=10



La méthode décrite dans les deux précédents paragraphes se généralise aisément à un ordre  $q$  supérieur à deux. Dans ce contexte, on peut étudier le cas des fonctions splines d'ordre  $q$  qui minimisent une fonctionnelle dont l'opérateur dérivée seconde peut se généraliser à un opérateur différentiel linéaire.

Ce cas général fait l'objet de la présente étude.

II - 4. DETERMINATION DES FONCTIONS SPLINES SOUS TENSION D'ORDRE  $q$  GENERALISEES:

Le principe de la méthode ne différant pas de celle exposée aux paragraphes (II - 2 et II - 3) précédents, nous convenons de restreindre la présente étude à l'énoncé des fonctions splines sous-tension d'ordre  $q$  et à la détermination de la structure du processus interpolateur associé.

II - 4 - 1 : PROPRIETES DES FONCTIONS SPLINES SOUS-TENSION D'ORDRE  $q$  :

Soit le réseau de points  $\{x_i, y_i\}; \forall i = 1, \dots, n$  ordonné par la relation :

$$x_1 < x_2 < \dots < x_{n-1} < x_n$$

Une fonction  $f(x)$  définie sur l'intervalle  $[x_1, x_n]$  est dite fonction spline sous-tension d'ordre  $q$  si elle satisfait les propriétés suivantes.   
 généralisée

R1)  $f(x)$  ainsi que ses dérivées jusqu'à l'ordre  $2q-2$  sont continues sur  $[x_1, x_n]$ .

$$f^{(j)}(x_i^-) = f^{(j)}(x_i^+) \quad \forall i = 1, \dots, n \\ \forall j = 0, \dots, 2q-2$$

R2)  $f(x)$  interpole le réseau:

$$f(x_i) = y_i \quad \forall i = 1, \dots, n$$

De plus les  $(q-1)$  premières dérivées de  $f(x)$  aux extrémités du réseau peuvent être imposées indépendamment :

$$f^{(k)}(x_1) \text{ et } f^{(k)}(x_n) \text{ fixées : } \forall k = 1, \dots, q-1$$

R3)  $f(x)$  minimise dans l'espace des fonctions de carré sommable satisfaisant les propriétés précédentes, la fonctionnelle de forme :

$$\Phi(\varphi(x)) = \int_{x_1}^{x_n} ((L.\varphi(x))^2 + \sigma^2.(\varphi'(x))^2) dx ; \sigma > 0$$

où  $L$  est l'opérateur différentiel linéaire à coefficients constants défini par :

$$L = \left(\frac{d}{dx}\right)^q + \sum_{i=1}^{q-1} a_i \cdot \left(\frac{d}{dx}\right)^{i-1} ; a_i \in \mathbb{R}$$

Compte tenu de ces propriétés, on interprète  $f(x)$  (notée aussi  $Z(x)$ ) comme la trajectoire d'un système dynamique dont le schéma de simulation est le suivant :  
(cf Fig - 28)

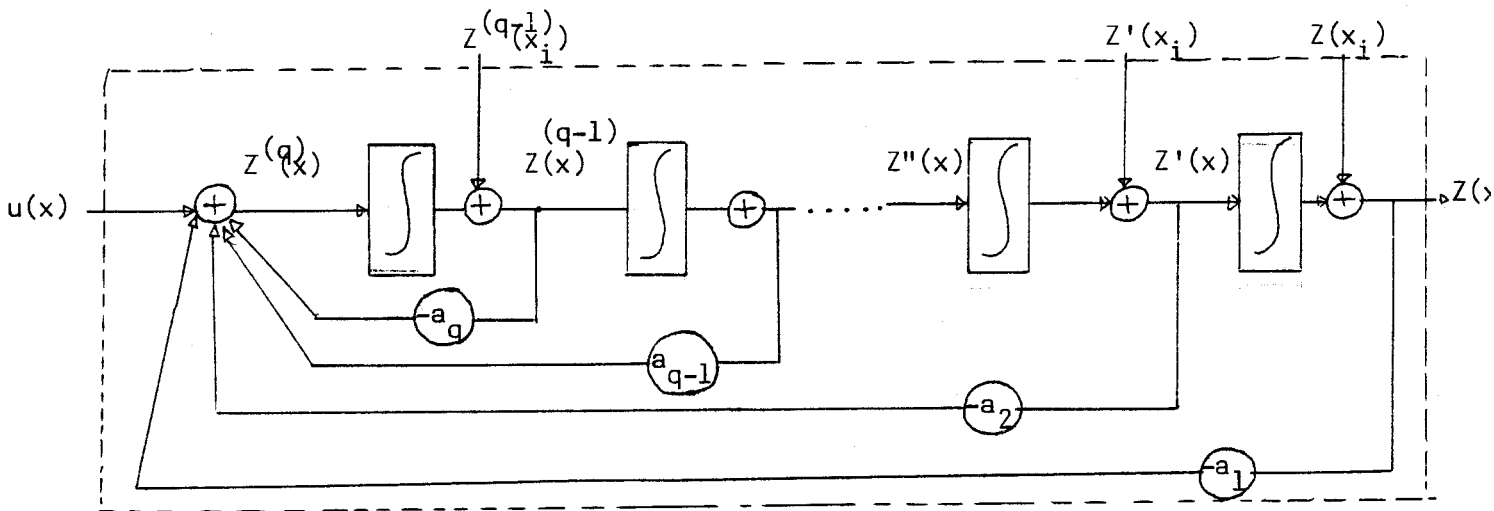


Fig - 28

Une représentation de ce système dans l'espace d'état se déduit directement. En notant  $\bar{Z}(x)$  le vecteur :

$$\bar{Z}^T(x) = [f(x), f(x), \dots, f^{(q-1)}(x)] = [Z(x), Z'(x), \dots, Z^{(q-1)}(x)]$$

(II - 4 - 1)

Il vient :

$$\frac{d}{dx} (\bar{Z}(x)) = A \cdot \bar{Z}(x) + b \cdot u(x)$$

$$\bar{Z}(x) \in \mathbb{R}^{q \times 1}$$

$$A \in \mathbb{R}^{q \times q}$$

$$Z(x) = c \cdot \bar{Z}(x)$$

$$b \in \mathbb{R}^{q \times 1}, c \in \mathbb{R}^{1 \times q}$$

(II - 4 - 2)



$$P \in \mathbb{R}^{q \times q} \quad ; \quad p > 0$$

$$P = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & \sigma^2 & & \vdots \\ \vdots & 0 & 0 & \vdots \\ \vdots & & & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} \quad ; \quad \sigma > 0$$

Ces relations (cf II - 4 - 3, II - 4 - 2) ainsi que la formulation du problème dans l'espace d'état (cf II - 4 - 1, II - 4 - 2) s'écrivent sous la forme matricielle compacte suivante : (II - 4 - 5)

$$\frac{d}{dx} \begin{bmatrix} \bar{Z}(x) \\ \Psi(x) \end{bmatrix} = \begin{bmatrix} A & b \cdot b^T \\ P & -A^T \end{bmatrix} * \begin{bmatrix} \bar{Z}(x) \\ \Psi(x) \end{bmatrix} \quad (\text{II} - 4 - 5)$$

Notons J cette matrice et z(x) le vecteur d'état à 2q composantes, définies respectivement par :

$$J = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & | & 0 & \dots & \dots & 0 \\ & & & & \vdots & & \vdots & & & \vdots \\ 0 & 0 & 1 & & \vdots & & \vdots & & & \vdots \\ 0 & & & & 1 & & 0 & & & \vdots \\ -a_1 & -a_2 & \dots & \dots & -a_q & | & 0 & \dots & \dots & 1 \\ \hline 0 & 0 & & & 0 & | & 0 & 0 & \dots & +a_1 \\ 0 & \sigma^2 & & & 0 & | & 1 & 0 & & +a_2 \\ \vdots & & & & \vdots & & 0 & 1 & & \\ \vdots & & & & \vdots & & \vdots & & & \\ 0 & \dots & \dots & \dots & 0 & | & 0 & 0 & 1 & +a_q \end{bmatrix} ;$$



$$z^T(x) = [Z(x), Z'(x), \dots, Z^{(q-1)}(x), \Psi_1(x), \dots, \Psi_q(x)]$$

(II - 4 - 6)

En définissant un vecteur  $v(x) \in \mathbb{R}^{2q}$  adéquat à partir de  $\bar{Z}(x)$  et  $\Psi(x)$ , le système (II - 4 - 5) devient :

$$\frac{d}{dx}(v(x)) = \tilde{A} \cdot v(x) \quad (\text{II - 4 - 7})$$

avec  $v^T(x) = [Z(x), Z'(x), \dots, Z^{(q-1)}(x), \Psi_q(x), -\Psi_{q-1}(x), \dots$   
 $\dots, (-1)^q \cdot \Psi_2(x), (-1)^{q+1} \Psi_1(x)]$

$$A = \begin{bmatrix} 0 & 1 & \dots & \dots & 0 & 0 & \dots & \dots & 0 \\ \vdots & & & & \vdots & \vdots & & & \vdots \\ \vdots & & & & \vdots & \vdots & & & \vdots \\ \vdots & & & & \vdots & \vdots & & & \vdots \\ \vdots & & & & 1 & 0 & \dots & \dots & 0 \\ -a_1 & -a_2 & \dots & \dots & -a_{q-1} & -a_q & 1 & 0 & \dots & \dots & 0 \\ \hline 0 & 0 & & & 0 & 0 & a_q & 1 & & & \vdots \\ 0 & & & & \vdots & & & & & & \vdots \\ \vdots & & & & \vdots & & & & & & \vdots \\ \vdots & & & & \vdots & & & & & & \vdots \\ 0 & (-)^q \sigma^2 & & & 0 & 0 & (-)^q a_2 & & & & 1 \\ 0 & \cdot & \cdot & \cdot & 0 & 0 & (-)^{q+1} a_1 & \cdot & \cdot & \cdot & 0 \end{bmatrix} \in \mathbb{R}^{2q \times 2q}$$

(II - 4 - 8)



Il vient :

$$\forall h \in [0, x_{i+1} - x_i[ \quad ; \quad \forall i = 1, \dots, n-1$$

$$v(x_i+h) = M(h) \cdot v(x_i^+) \quad (\text{II} - 4 - 9)$$

$$\text{avec } M(h) = \exp (Ah)$$

Le calcul de  $M(h)$  peut s'effectuer par la méthode classique des matrices constituantes, (GANTMACHER - 1975), basée sur le théorème suivant :

THEOREME : Soit  $M$  une matrice carrée d'ordre  $n$  et  $f(\lambda)$  une fonction scalaire de  $\lambda$  admettant des dérivées successives.

Si  $\lambda_1, \lambda_2, \dots, \lambda_k$  sont les valeurs propres de  $M$  d'ordre de multiplicité  $n_1, n_2, \dots, n_k$ , alors une fonction de  $M$  ;  $f(M)$  s'écrit :

$$f(M) = \sum_{i=1}^k \sum_{j=1}^{n_i} z_{ij} \cdot f^{(j-1)}(\lambda_i) \quad (\text{II} - 4 - 10)$$

$z_{ij}$  sont les matrices constituantes, indépendantes de  $f$ . A titre d'exemple, nous donnons la forme de  $M(h)$  associée à une fonction spline hyperbolique (II-4-II) et une fonction spline trigonométrique d'ordre 2 (II - 4 - I4)

EXEMPLES :

1° Splines hyperboliques :  $\sigma=1$  ;  $a_1=0$  ;  $a_2 = 3$  :

Il vient :

$$\frac{d}{dx} (v(x)) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\sqrt{3} & 1 & 0 \\ 0 & 1 & \sqrt{3} & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot v(x) ;$$

avec  $v(x) = \begin{bmatrix} Z(x) \\ Z'(x) \\ \Psi_2(x) \\ -\Psi_1(x) \end{bmatrix}$

Les valeurs propres sont :

$$\begin{array}{l} \lambda_1 = \lambda_2 = 0 \\ \lambda_3 = 2 \\ \lambda_4 = -2 \end{array}$$

et les matrices constituantes s'écrivent :

$$Z_{11} = \frac{4I - A^2}{4} ; Z_{12} = \frac{A \cdot (4I - A^2)}{4} ; Z_{21} = \frac{A^2 \cdot (A + 2 \cdot I)}{16} ;$$

$$Z_{31} = \frac{A^2 \cdot (2 \cdot I - A)}{16}$$

Soit :

(II - 4 - II)

$$M(h) = \begin{bmatrix} 1 & \frac{1}{2} \left( \frac{\sqrt{3}}{2}(1-\text{ch}2h) + \text{sh}2h \right) & \frac{1}{4} (\text{ch}2h - 1) & \frac{1}{4} \left( \frac{\text{sh}2h}{2} - h \right) \\ 0 & - \left( \text{ch}2h - \frac{\sqrt{3}}{2} \text{sh}2.h \right) & \frac{\text{sh}2h}{2} & \frac{1}{4} (\text{ch}2h-1) \\ 0 & \frac{\text{sh}2h}{2} & \left( \text{ch}2h + \frac{\sqrt{3}}{2} \text{sh}2h \right) & \frac{1}{2} \left( \frac{\sqrt{3}}{2}(\text{ch}2h-1) + \text{sh}2h \right) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2° Fonctions splines trigonométriques d'ordre 2

On essaie d'approcher la solution ( $f(x) = \sin x$ ) de l'équation différentielle  $f'' + f = 0$ , en posant

$$L = \left( \frac{d}{dx} \right)^2 + \left( \frac{d}{dx} \right)^0, \text{ soit } \sigma = 0; a_1 = 1 \text{ et } a_2 = 0$$

De manière analogue les équations théoriques d'optimalité (principe du maximum) conduisent au système matriciel suivant :

(II - 4 - 12)

$$\frac{d}{dx} \begin{bmatrix} Z(x) \\ Z'(x) \\ \Psi_2(x) \\ -\Psi_1(x) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} Z(x) \\ Z'(x) \\ \Psi_2(x) \\ -\Psi_1(x) \end{bmatrix} \iff \frac{d}{dx}(v(x)) = \tilde{A} \cdot v(x) \quad (\text{II} - 4 - 12)$$

dont le processus interpolateur est représenté ci-dessous  
(Fig - 30) :

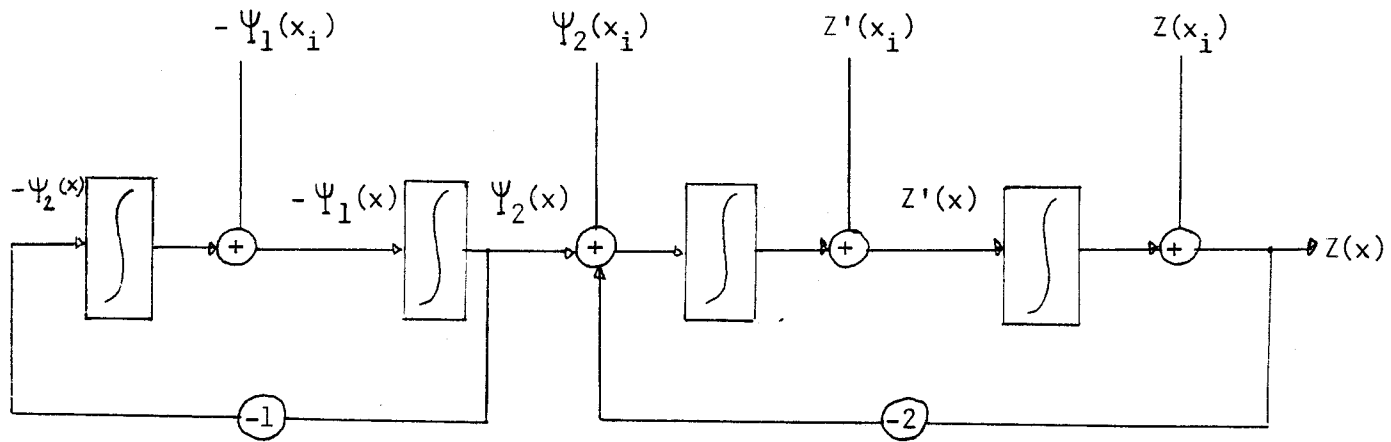


Fig - 30

L'intégration du système par la méthode des matrices  
constituantes aboutit aux équations suivantes.

$$z_1 = \frac{I}{2}; \quad z_3 = \frac{A^2 + I - (A - iI)^2(A + iI)}{4i}; \quad z_2 = z_3 - \frac{A}{2}; \quad z_4 = -\frac{I + A^2}{4}$$

$$\forall i = 1, \dots, n-1$$

$$\forall h \in [0, x_{i+1} - x_i[$$

$$v(x_i + h) = M(h) v(x_i^+) \quad (\text{II} - 4 - 13)$$

où  $M(h) = \exp(\tilde{A}h)$  est la matrice suivante :

$$M(h) = \begin{bmatrix} ((1-h)\sinh - h\cosh) & ((h-1)\cosh - (h+1)\sinh) & ((h/2-1)\sinh + h\cosh) \\ ((1-h)\cosh + (1+h)\sinh) & ((1-h)\sinh - h\cosh) & (-(1+3h/2)\sinh + (h-3/2)\cosh) \\ 0 & 0 & ((1-h)\sinh - h\cosh) \\ 0 & 0 & ((1-h)\cosh + (1+h)\sinh) \end{bmatrix}$$

$$\left[ \begin{array}{l} \frac{1}{2}(\cosh - h\sinh) \\ (h/2-1)\sinh + h\cosh \\ ((h-1)\cosh - (h+1)\sinh) \\ ((1-h)\sinh - h\cosh) \end{array} \right]$$

(II - 4 - 14)

Il ne reste plus alors qu'à expliciter le calcul des conditions initiales  $v(x_i^+)$  en initialisant le vecteur  $v(x_i^+)$ .

II - 4 - 3 : DETERMINATION DES CONDITIONS INITIALES

$v(x_i^+)$  :

II - 4 - 3 - 1 : Relation entre  $v(x_i^+)$  et  $v(x_{i+1}^+)$  :

La détermination des conditions initiales  $v(x_i^+)$  s'étudie par la récurrence désormais classique présentée aux paragraphes (II - 2) et (II-3), à partir d'une équation de récurrence liant les vecteurs  $v(x_i^+)$  et  $v(x_{i+1}^+)$ . Il vient :

$$v(x_{i+1}^+) = [N^{-1}(I+2) \cdot Q \cdot M(x_{i+1}-x_i)] \cdot v(x_i^+) + [N^{-1}(I+2) \cdot Y_1(I+2)]$$

(II - 4 - 15)

$$\forall i=1, \dots, n-2$$

où  $N^{-1} \in \mathbb{R}^{2q \times 2q}$  ;  $Q \in \mathbb{R}^{2q \times 2q}$  ;  $M \in \mathbb{R}^{2q \times 2q}$  ;  $Y_1 \in \mathbb{R}^{2q \times 1}$  ;  $v(x) \in \mathbb{R}^{2q \times 1}$

avec

$$N^{-1} = \begin{bmatrix} 1 & 0 & \dots & \dots & 0 \\ 0 & \dots & 1 & 0 & \\ \circ & & & & \\ \vdots & & & & \\ m_{11} & m_{12} & \dots & \dots & m_{12q} \end{bmatrix} ; \quad Y_1 = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ \vdots \\ y_{(i+2)} \end{bmatrix} ; \quad Q = \begin{bmatrix} 1 & 0 & \dots & \dots & 0 \\ 0 & & & \circ & \vdots \\ \vdots & & & & \vdots \\ \vdots & & & & \vdots \\ 0 & \dots & \dots & 0 & 1 & 0 \\ 0 & \dots & \dots & \dots & \dots & 0 \end{bmatrix}$$

$m_{11}, m_{12}, \dots, m_{12q}$  = 1ère ligne de la matrice  $M(x_{i+2}^+ x_{i+1}^+)$



II - 4 - 3 - 2 : Calcul de  $v(x_1^+)$  :

Soit  $\bar{Z}^T(x)$  le vecteur de composantes  $Z(x), Z'(x), \dots, Z^{(q-1)}(x), z^{(q)}(x)$ , dont la première composante aux extrémités du réseau est imposée.

La variation de la fonctionnelle  $\bar{\Phi}(\bar{Z}(x))$  ;  
(cf § II - 4 - 1 ; (R3)) consécutive à des variations infiniment petites  $\delta \bar{Z}(x_1)$  et  $\delta \bar{Z}(x_n)$  des vecteurs  $\bar{Z}(x_1)$  et  $\bar{Z}(x_n)$  aux extrémités du réseau s'exprime par les conditions de transversalité

$$\delta J = \left\{ \Psi^T(x) \cdot \delta \bar{Z}(x) \right\}_{\substack{x=x_1 \\ x=x_n}} = 0 \quad (\text{II - 4 - 16})$$

Si les dérivées successives  $Z'(x_1), \dots, Z^{(q)}(x_1)$  (resp  $Z'(x_n), \dots, Z^{(q)}(x_n)$ ) ne sont pas fixées, cette dernière relation établit la nullité des composantes de même rang du vecteur  $\Psi(x_1)$  (resp  $\Psi(x_n)$ ).

$$\text{Il vient : } \Psi_j(x_1^+) = 0 ; \quad (\text{resp } \Psi_j(x_n^-) = 0)$$

$$\forall j = 2, \dots, q. \quad (\text{II - 4 - 17})$$

En d'autres termes, les conditions limites du vecteur  $v(x)$  aux extrémités sont les suivantes : (II - 4 - 18)

$$\left\{ \begin{array}{l} v_1(x_1) \text{ connu} \\ \text{et } v_j(x_1) \text{ fixé sinon } v_{q+j-1}(x_1) = 0 \end{array} \right.$$

$$\forall j = 2, \dots, q$$

et

$$\left\{ \begin{array}{l} v_1(x_n) \text{ connu} \\ \text{et } v_j(x_n) \text{ fixé sinon } v_{q+j-1}(x_n) = 0 \end{array} \right.$$

(II - 4 - 18)

q composantes du vecteur  $v(x_1^+)$  (resp  $v(x_n^-)$ ) étant fixées les q autres composantes sont obtenues à partir :

1) de l'équation de progression (II - 4 - 9) où

$$h \text{ est tel que : } h = x_2 - x_1$$

$$Z(x_2) = [1 \ 0 \ \dots \ 0] \cdot M(x_2 - x_1) \cdot v(x_1)$$

2) de (q-1) équations linéaires extraites de la relation (II - 4 - 15) itérée jusqu'au point  $x_n^-$  et qui aboutit à (II - 4 - 19) :

$$v(x_n^-) = M(h_n) \cdot \left\{ \prod_{k=3}^n A(k) \right\} \cdot v(x_1^+) + M(h_n) \cdot \left\{ \sum_{k=3}^n \left( \prod_{e=k+1}^n A(e) \right) \cdot B(k) \right\}$$

(II - 4 - 19)

Pour de faibles valeurs de  $q(2,3)$ , ce système se résoud de manière formelle.

Les vecteurs  $v(x_i^+)$  ;  $i = 2, \dots, n-1$  se déduisent de  $v(x_1^+)$  à partir de l'équation de progression :

$$v(x_{i+1}^-) = M(x_{i+1}-x_i) \cdot v(x_i^+)$$

Pour tout  $x \in [x_i, x_{i+1}]$  , la fonction spline sous tension d'ordre  $q$  s'écrit :

$$f(x) = m_{11} v_1(x_i^+) + m_{12} v_2(x_i^+) + \dots + m_{12q} v_{2q}(x_i^+)$$

où les  $m_{1j}$  sont des fonctions de  $(x-x_i)$  définissant les éléments de la première ligne de la matrice  $M$ .

#### II - 4 - 4 : REALISATION DU PROCESSUS INTERPOLATEUR :

Le principe est en tout point identique à celui exposé pour les fonctions splines polynomiales et sous-tension d'ordre 2 (cf § II - 2 - 5 & § II - 3 - 5).

Par ailleurs, il convient de rajouter dans l'organigramme de calcul de  $v(x_1^+)$ , des boucles de programme qui tiennent compte de l'inversion de la matrice  $N$  et du calcul des matrices constituantes.

Nous présentons ci-dessous le schéma de réalisation hybride des fonctions splines sous tension d'ordre  $q$  généralisées, dont le fonctionnement est identique en tout point à celle exposée pour les fonctions splines d'interpolation d'ordre 2 (cf § II - 2 - 5).

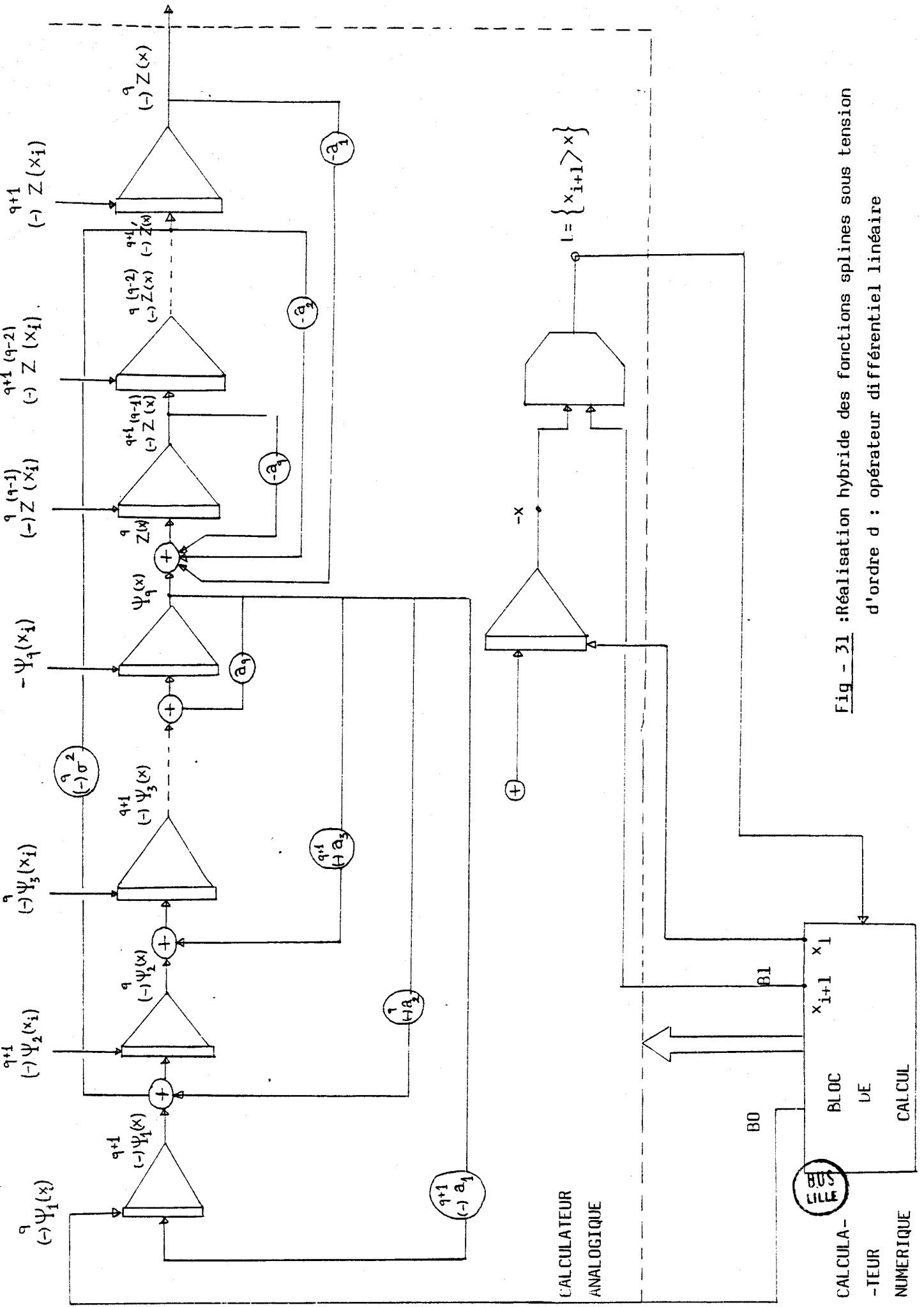
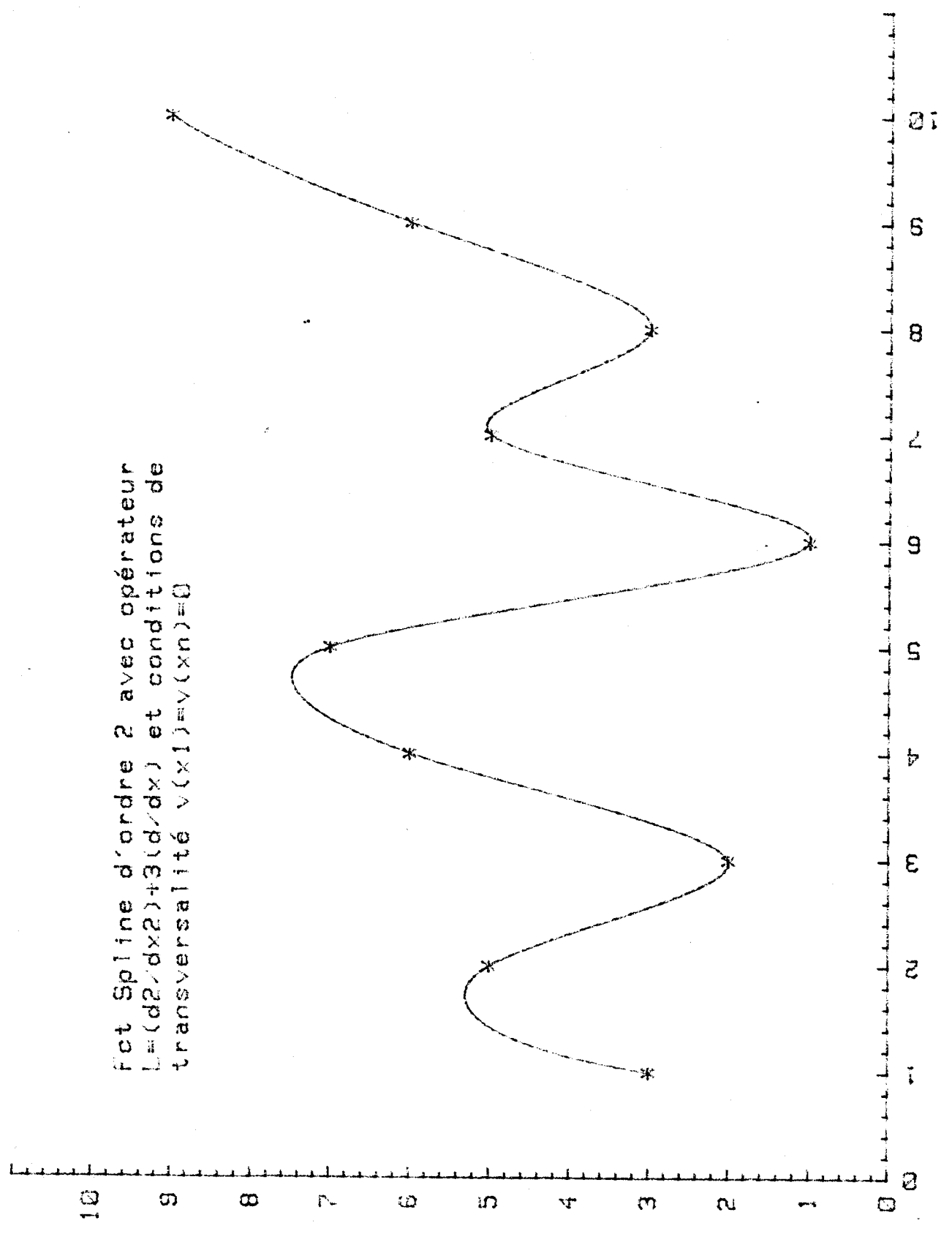


Fig - 31 : Réalisation hybride des fonctions splines sous tension d'ordre d : opérateur différentiel linéaire

Fct Spline d'ordre 2 avec opérateur  
 $L=(d^2/dx^2)+3(d/dx)$  et conditions de  
transversalité  $v(x_1)=v(x_n)=0$



II - 5 : APPLICATION AU CALCUL DES FONCTIONS SPLINES D'AJUSTEMENT

D'ORDRE  $q$  :

II-5-1 : Introduction :

Jusqu'à présent, nous avons étudié le cas des fonctions dont les valeurs aux neuds du réseau étaient connues avec précision.

En général, ces valeurs sont souvent entachées d'erreurs (erreurs de mesures dues à l'expérimentation).

Dans ce cas, la fonction  $f(x)$  n'est plus contrainte à passer par ces points et on recherche un compromis entre l'approximation de  $f$  et l'ajustement des valeurs  $y_i$  (cf Fig - 32)

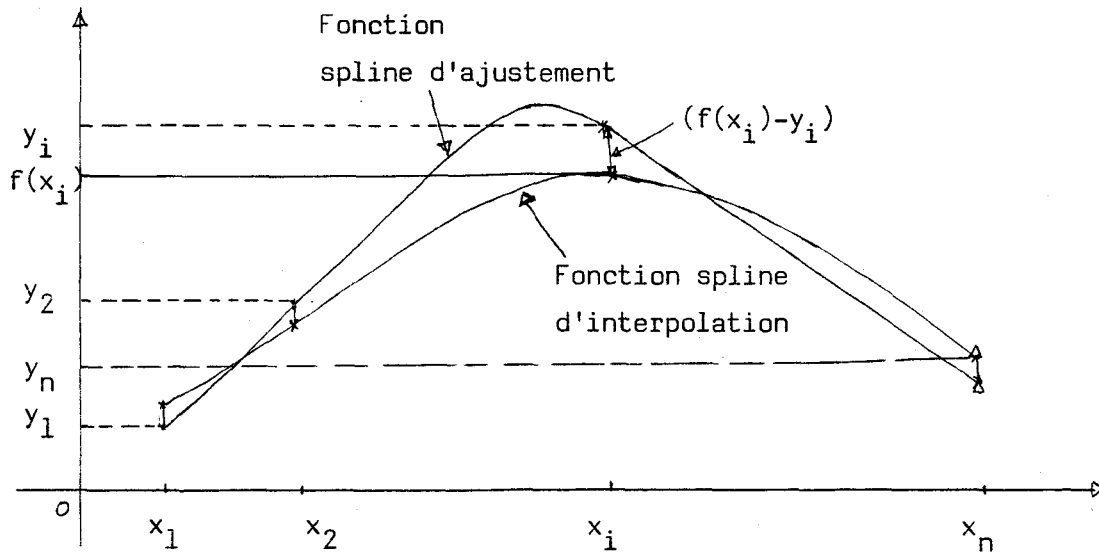


Fig 32 : Ajustement d'un réseau  $\{x_i, y_i\}$  par une fonction spline.

La fonction recherchée doit donc satisfaire deux objectifs :

- a) La courbe correspondante doit être la plus douce possible,
- b) sous réserve de (a) ;  $f(x)$  doit ajuster les valeurs  $y_i$  de telle sorte que l'écart  $(f(x_i) - y_i)$  soit le plus petit possible.

Les fonctions qui réalisent ce compromis sont notées fonctions splines d'ajustement ou de lissage (P.J. LAURENT, 1972).

En utilisant l'algorithme du processus interpolateur, nous proposons de déterminer leur expression analytique dans tout intervalle où elles sont définies, simplement à partir de quelques propriétés de base.

Par ailleurs, l'utilisation de la théorie de la commande optimale des systèmes dynamiques permet de déduire des théorèmes fondamentaux relatifs aux fonctions splines d'ajustement, fréquemment utilisés en analyse numérique (LAURENT - 1971), montrant ainsi le lien étroit existant entre la théorie de l'interpolation et celle de la commande optimale.

## II - 5 - 2 : QUELQUES PROPRIETES DES FONCTIONS SPLINES D'AJUSTEMENT D'ORDRE $q$ :

Les fonctions splines d'ajustement d'ordre  $q$  définies sur le réseau  $\{x_i, z_i \in \mathbb{R} ; \forall i=1, \dots, n\}$  ordonné par la relation

$$x_1 < x_2 < \dots < x_{n-1} < x_n$$



sont caractérisées par les propriétés suivantes :

S1)  $f(x)$  ainsi que ses dérivées jusqu'à l'ordre  $2q-2$  sont continues sur l'intervalle  $[x_1, x_n] \subset \mathbb{R}$

$$z^{(j)}(x_i^+) = z^{(j)}(x_i^-) \quad \forall i = 1, \dots, n$$
$$\forall j = 0, \dots, 2q-2$$

S2)  $f(x)$  minimise dans l'espace des fonctions  $f(x)$  du carré sommable et satisfaisant (S1), la fonctionnelle de forme

$$\Phi(\varphi(x)) = \int_{x_1}^{x_n} (\varphi^{(q)}(x))^2 dx + \sum_{i=1}^n \rho_i \cdot (\varphi(x_i) - z_i)^2$$

$$\forall i = 1, \dots, n, \rho_i > 0$$

S3) Sur tout intervalle  $[x_i, x_{i+1}]$  ;  $\forall i=1, \dots, n-1$

$f(x)$  est une fonction polynomiale de degré  $2q-1$ .

En fait, (S3) constitue un corollaire de (S1) et (S2).

REMARQUE :

L'introduction du terme de pénalisation dans le deuxième membre de la fonctionnelle garde la structure du processus interpolateur initialement défini dans l'espace d'état, mais modifie les conditions de transversalité.

Il s'en suit que dans tout intervalle  $[x_i, x_{i+1}[$ , la fonction  $Z(x)$  apparait comme la sortie du système dynamique suivant, noté processus d'ajustement ( Fig - 33)

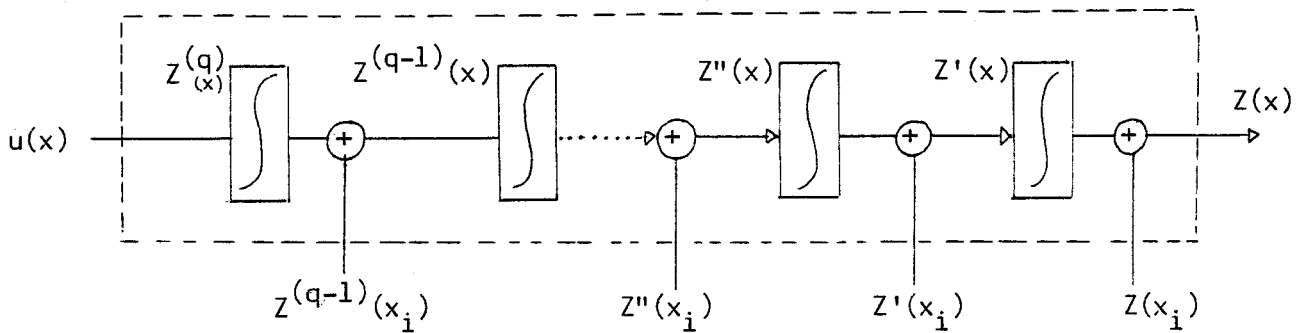


Fig - 33

En définissant le vecteur  $\bar{Z}(x)$  par :

$$\bar{Z}^T(x) = [Z(x), Z'(x), \dots, Z^{q-1}(x)] \in \mathbb{R}^q$$

ce système peut se représenter dans l'espace d'état de la manière suivante :

$$\frac{d}{dx} (\bar{Z}(x)) = A \cdot \bar{Z}(x) + b \cdot u(x)$$

$$Z(x) = C \cdot \bar{Z}(x)$$

$; b \in \mathbb{R}^{q \times 1} ; c \in \mathbb{R}^{1 \times q} ; A \in \mathbb{R}^{q \times q}$

où :  $b^T = [0, 0, \dots, 0, 1]$  ;  $c = [1, 0, \dots, 0]$  (II - 5 - 1)

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \vdots & \circlearrowleft & & \\ 0 & 0 & \dots & 0 \end{bmatrix}$$

Cette dernière formulation permet de résoudre le problème par application des équations théoriques d'optimalité et des conditions de transversalité que nous proposons de déterminer.

II - 5 - 3 : COMMANDE OPTIMALE DES PROCESSUS :

II - 5 - 3 - 1 : Equations théoriques d'optimalité :

Si H est la fonction Hamiltonien du problème défini par les propriétés (S1) et (S2), et par l'équation (II - 5 - 1), alors dans tout intervalle  $]x_i, x_{i+1}[$ , la commande  $u(x)$  cherchée est celle qui maximise H.

$$H = -\frac{1}{2} u^2(x) + \Psi^T \cdot (A \cdot \bar{z}(x) + b \cdot u(x)) \quad (\text{II - 5 - 2})$$

Il vient:

$$u(x) = b^T \cdot \Psi(x) \quad \forall x \in ]x_i, x_{i+1}[ \quad ;$$

$$\forall i = 1, \dots, n-1 \quad (\text{II - 5 - 3})$$

L'évolution du vecteur  $\Psi(x)$  dans tout intervalle  $]x_i, x_{i+1}[$  ;  $\forall i = 1, \dots, n-1$  est régie par :

a) Le système d'équations différentielles (équation de HAMILTON-PONTRYAGUINE) :

$$\frac{d\Psi(x)}{dx} = - \frac{dH}{d\bar{Z}(x)} ; \forall x \in ]x_i, x_{i+1}[ ; \forall i=1, \dots, n-1$$

qui s'écrit :

$$\frac{d}{dx} (\Psi(x)) = -A^T \Psi(x) ; \forall x \in ]x_i, x_{i+1}[ ; \forall i=1, \dots, n-1$$

(II - 5 - 4)

b) Les conditions de transversalité sur le vecteur  $\Psi(x)$  aux extrémités  $x_1$  et  $x_n$  du réseau.

Du fait de l'introduction du terme de pénalisation, ces conditions sont modifiées de la manière suivante.

II - 5 - 3 - 2 : Conditions de transversalité :

Pour déterminer ces conditions, on étudie la variation de la fonctionnelle suivante :

$$J = \int_{x_1^+}^{x_n^-} f(\bar{Z}'(x), \bar{Z}(x), x) dx + \frac{1}{2} \sum_{i=1}^n \beta_i (Z(x_i) - y_i)^2$$

II - 5 - 5)

consécutive à une variation  $\delta \bar{Z}(x)$  de  $\bar{Z}(x)$ ,

$$\bar{Z}'(x) = \frac{d}{dx} (\bar{Z}(x)) ; \bar{Z}^T(x) = [Z(x), Z'(x), \dots, Z^{q-1}(x)]$$

où  $f$  est une fonction de classe  $C^1$  sur  $[x_1, x_n]$ , autrement dit :

$$f \text{ et } \frac{df}{d\bar{Z}(x)} \text{ sont continues}$$

en posant :  $c^T = [1, 0, \dots, 0]$  et  $f_{\bar{Z}'(x)} = \Psi(x)$ .

La variation  $\delta J$  de la fonctionnelle  $J$  sur  $[x_1, x_n]$  s'écrit :

$$\delta J = \int_{x_1^+}^{x_n^-} \left\{ f_{\bar{Z}(x)}^T(x) \cdot \delta \bar{Z}(x) + f_{\bar{Z}'(x)}^T(x) \cdot \frac{d}{dx} (\delta \bar{Z}(x)) \right\} \cdot dx +$$

$$\sum_{i=1}^n \left\{ \rho_i \cdot (Z(x_i) - z_i) \cdot c \right\}^T \cdot \delta \bar{Z}(x_i)$$

(II - 5 - 6)

Or la propriété (S1)  $\implies f_{\bar{Z}'(x)}$  non continue sur  $[x_1, x_n]$

$$\int_{x_1^+}^{x_n^-} f_{\bar{Z}'(x)}^T \cdot \frac{d}{dx} (\delta \bar{Z}(x)) \cdot dx = \sum_{i=1}^{n-1} \int_{x_i^+}^{x_{i+1}^-} f_{\bar{Z}'(x)}^T(x) \cdot \frac{d}{dx} (\delta \bar{Z}(x)) \cdot dx$$

(II - 5 - 7)

$$\int_{x_1^+}^{x_n^-} f_{\bar{Z}(x)}^{I'} \cdot \frac{d}{dx} (\delta \bar{Z}(x)) \cdot dx =$$

$$\sum_{i=1}^{n-1} \left\{ f_{\bar{Z}(x)}^{I'} \cdot \delta \bar{Z}(x) \right\}_{x_i^+}^{x_{i+1}^-} - \int_{x_1^+}^{x_n^-} \frac{d}{dx} (f_{\bar{Z}(x)}^{I'}) \cdot \delta \bar{Z}(x) \cdot dx$$

(II - 5 - 8)

Il s'en suit que :

$$\delta J = \int_{x_1^+}^{x_n^-} (f_{\bar{Z}(x)} - \frac{d}{dx} f_{\bar{Z}(x)}^{I'})^T \cdot \delta \bar{Z}(x) dx + \sum_{i=1}^{n-1} \left\{ f_{\bar{Z}(x)}^{I'} \cdot \delta \bar{Z}(x) \right\}_{x_i^+}^{x_{i+1}^-} +$$

$$\sum_{i=1}^n \left\{ \rho_i \cdot (Z(x_i) - z_i) \cdot C \right\}^T \cdot \delta \bar{Z}(x_i)$$

(II - 5 - 9)

En développant le deuxième membre et en groupant les termes, on obtient finalement :

$$\delta J = \int_{x_1^+}^{x_n^-} \left( f_{\bar{Z}(x)} - \frac{d}{dx} f'_{\bar{Z}(x)} \right)^T \delta \bar{Z}(x) dx + \sum_{i=2}^{n-1} \left\{ \right.$$

$$\left. \left[ f'_{\bar{Z}(x_i^-)} - f'_{\bar{Z}(x_i^+)} \right] + \left[ \beta_i (Z(x_i) - z_i) C \right]^T \right\} \cdot \delta \bar{Z}(x_i) +$$

$$\left( f'_{\bar{Z}(x_1^+)} - \beta_1 (Z(x_1) - z_1) \cdot C \right)^T \cdot \delta \bar{Z}(x_1) + \left( f'_{\bar{Z}(x_n^-)} + \beta_n (Z(x_n) - z_n) \cdot C \right)^T \cdot \delta \bar{Z}(x_n)$$

(II - 5 - 10)

$\delta J = 0$   $\implies$  Les conditions de transversalité sont alors :

$$\begin{cases} \Psi_1(x_1^+) = \beta_1 (Z(x_1) - z_1) \\ \Psi_j(x_1^+) = 0 \quad ; \quad \forall j=2, \dots, q \end{cases} \quad (\text{II - 5 - 11})$$

$$\begin{cases} \Psi_1(x_n^-) = -\beta_n (Z(x_n) - z_n) \\ \Psi_j(x_n^-) = 0 \quad ; \quad \forall j=2, \dots, q \end{cases} \quad (\text{II - 5 - 12})$$

$$\begin{cases} (\Psi_1(x_i^-) - \Psi_1(x_i^+)) + \beta_i (Z(x_i) - z_i) = 0 \quad ; \quad \forall i=2, \dots, n-1 \\ \Psi_j(x_i^+) = \Psi_j(x_i^-) \quad ; \quad \forall j=2, \dots, q \end{cases} \quad (\text{II - 5 - 13})$$

$\forall i=2, \dots, n-1$

$$(-1)^{q+1} \cdot \Psi_1(x_i) = Z^{(2q-1)}(x_i) \quad (\text{II - 5 - 14})$$

et  $f_{\bar{Z}}(x) - \frac{d}{dx} f'_{\bar{Z}} = 0$  n'est autre que l'équation d'EULER.

Compte tenu de (II - 5 - 14), la première équation de (II - 5 - 13) s'écrit :

$$(-1)^q \cdot (Z^{(2q-1)}(x_i^+) - Z^{(2q-1)}(x_i^-)) = \rho_i \cdot (z_i - Z(x_i)) \quad (\text{II} - 5 - 15)$$

$$\forall i = 2, \dots, n-1$$

Les relations (II - 5 - 11  $\longrightarrow$  II - 5 - 15) constituent les conditions aux limites du problème initialement posé, et permettent de calculer le processus d'ajustement associé à la fonction  $Z(x)$ .

#### II - 5 - 4 : STRUCTURE DU PROCESSUS D'AJUSTEMENT :

Les équations théoriques d'optimalité (II - 5 - 3 et II - 5 - 4) ainsi que la formulation du problème dans l'espace d'état (II - 5 - 1) aboutissent au système matriciel suivant : (II - 5 - 16)

$$\forall x \in ]x_i, x_{i+1}[ \quad ; \quad \frac{d}{dx} \begin{pmatrix} \bar{Z}(x) \\ \Psi(x) \end{pmatrix} = \begin{bmatrix} A & | & b \cdot b^T \\ \hline 0 & | & -A^T \end{bmatrix} * \begin{pmatrix} \bar{Z}(x) \\ \Psi(x) \end{pmatrix} \quad ; \quad (\text{II} - 5 - 16)$$



Notons  $J$  la matrice de ce système et  $z(x)$  le vecteur d'état à  $2q$  composantes, respectivement définies par :

$$J = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 & \dots & \dots & 0 \\ & & & & & \vdots & & & \vdots \\ & & & & & \vdots & & & \vdots \\ 0 & \dots & \dots & \dots & 1 & 0 & \dots & \dots & 0 \\ & & & & & \vdots & & & \vdots \\ 0 & \dots & \dots & \dots & 0 & 0 & \dots & \dots & 1 \\ \hline & & & & & 0 & \dots & \dots & 0 \\ & & & & & \vdots & & & \vdots \\ & & & & & 1 & & & \vdots \\ & & & & & \vdots & & & \vdots \\ & & & & & 0 & & & \vdots \\ & & & & & \vdots & & & \vdots \\ & & & & & 0 & & & 1 & 0 \end{bmatrix}$$

et  $\bar{z}^T(x) = [z(x), z'(x), \dots, z^{(q-1)}(x), \Psi_1(x), \dots, \Psi_q(x)]$

(II - 5 - 17)

Si on effectue la transformation suivante sur les composantes de  $\bar{z}^T(x)$

$$z_{i+q} \longrightarrow (-1)^{2q-i+1} \cdot z_{2q-i+1} \quad ; \quad \forall i=1, \dots, q$$

(II - 5 - 18)

le système (II - 5 - 16) devient :

$$\frac{d}{dx}(v(x)) = \tilde{A}.v(x) \quad (\text{II} - 5 - 19)$$

avec  $v^T(x) = [Z(x), Z'(x), \dots, Z^{(q-1)}(x), \Psi_q(x), -\Psi_{q-1}(x), \dots$   
 $\dots, (-1)^q \Psi_2(x), (-1)^{q+1} \Psi_1(x)]$

$$\tilde{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 \\ 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 \end{bmatrix}$$

Ce qui correspond au schéma de simulation suivant :

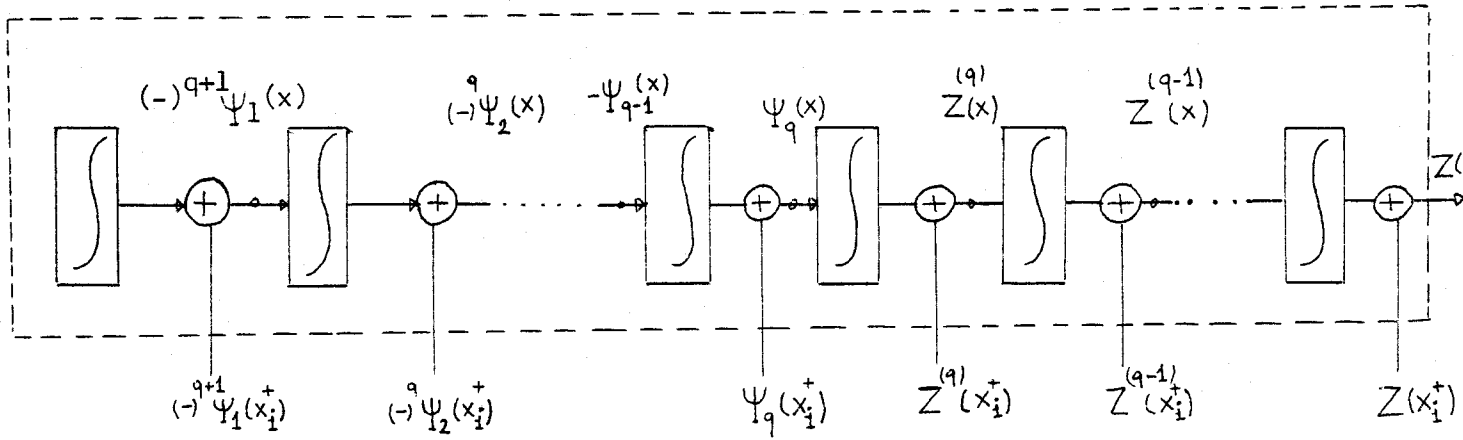


Fig - 34

L'équation de progression définissant complètement  $Z(x)$  dans chaque intervalle  $[x_i, x_{i+1}]$  est obtenue par intégration de l'équation d'état précédente : (II - 5 - 19) à partir de conditions initiales  $v(x_i^+)$ .

Il vient :

$$\forall h \in [0, x_{i+1} - x_i[ ; \quad \forall i=1, \dots, n-1$$

(II - 5 - 20)

$$v(x_i + h) = M(h) \cdot v(x_i^+);$$

$$M(h) = \begin{bmatrix} 1 & h & h^2/2 & \dots & h^{(2q-1)}/(2q-1)! \\ 0 & 1 & h & & \vdots \\ \vdots & & & & \vdots \\ 0 & \cdot & \cdot & \cdot & \cdot & \cdot & h \\ & & & & & & 1 \end{bmatrix}$$



La dernière composante  $v_{2q}(x_{i+1}^+)$  est issue de la condition de transversalité (II - 5 - 13) que l'on peut écrire sous la forme :

$$(-1)^{q+1} \cdot \Psi_{11}(x_{i+1}^+) = (-1)^{q+1} \cdot \Psi_{11}(x_{i+1}^+) + (-1)^{q+1} \rho_{i+1}.$$

$$(Z(x_{i+1}^-) - z_{i+1}) \quad (\text{II} - 5 - 23)$$

En injectant cette relation dans (II - 5 - 22) et en remplaçant  $v(x_{i+1}^-)$  par (II - 5 - 21) on obtient :

$$v(x_{i+1}^+) = \begin{bmatrix} 1 & \dots & \dots & \dots & \dots & \dots \\ 0 & & & & & \\ \vdots & & & & & \\ 0 & & & 1 & & \\ \vdots & & & & & \\ 0 & & & & & 1 \\ (-1)^{q+1} \rho_{i+1} & 0 & \dots & 0 & & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & h_{i+1} & & h_{i+1}^{2q-1}/(2q-1)! \\ 0 & 1 & h_{i+1} & \vdots \\ \vdots & & 1 & \vdots \\ \vdots & & & 1 & h_{i+1} \\ 0 & \dots & \dots & \dots & 1 \end{bmatrix} + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ (-1)^q \rho_{i+1} z_{i+1} \end{bmatrix} \quad (\text{II} - 5 - 24)$$

$$v^T(x) = [Z(x), Z'(x), \dots, Z^{q-1}(x), \Psi_q(x), -\Psi_{q-1}(x), \dots, (-1)^q \Psi_2(x), (-1)^{q+1} \Psi_1(x)]$$

Soit sous forme abrégée :

$$v(x_{i+1}^+) = (N(i+1).M(x_{i+1}-x_i)). v(x_i^+) + Y1(i+1) \quad (\text{II} - 5 - 25)$$

En notant :

$$A(i+1) = N(i+1) \cdot M(x_{i+1}-x_i) \quad (\text{II} - 5 - 26)$$

$$B(i+1) = Y1(i+1)$$

(II - 5 - 25) s'écrit :

$$v(x_{i+1}^+) = A(i+1) \cdot v(x_i^+) + B(i+1) \quad (\text{II} - 5 - 27)$$

En appliquant une stratégie analogue à celle du paragraphe (II - 4 - 3 - 2), on calcule les conditions limites  $v(x_i^+)$  à partir de l'initialisation de  $v(x_1^+)$ .

II - 5 - 5 - 2 : Calcul du vecteur  $v(x_1^+)$  :

L'équation (II - 5 - 25) itérée le long du réseau jusqu'au point  $x_n^-$  aboutit à l'équation matricielle désormais classique,

$$v(x_n^-) = M(h_n) \left\{ \prod_{k=3}^n A(k) \right\} \cdot v(x_1^+) + M(h_n) \cdot \left\{ \sum_{k=3}^n \left( \prod_{e=k+1}^n A(e) \right) \cdot B(k) \right\}$$

(II - 5 - 28)

qui constitue un système de  $2q$  équations linéaires.

Compte tenu des conditions aux limites (II - 5 - 11) ,  
 (resp), (II - 5 - 12) , pour lesquelles les inconnues sont  $v_1(x_1^+)$ , .....  
 $v_q(x_1^+)$  et  $v_{2q}(x_1^+)$  (resp  $v_1(x_n^-)$ ,  $v_2(x_n^-)$ , ...,  $v_q(x_n^-)$  et  $v_{2q}(x_n^-)$  ou  
 bien de la fixation des  $(q-1)$  premières dérivées en  $x_1^+$  (resp  $(q-1)$  premières  
 dérivées en  $x_n^-$ ) pour lesquelles les inconnues sont  $v_1(x_1^+)$  et  $v_{q+1}(x_1^+)$ ,  
 .....,  $v_{2q}(x_1^+)$  (resp  $v_1(x_n^-)$  et  $v_{q+1}(x_n^-)$  .....  $v_{2q}(x_n^-)$ , on peut extraire  
 du système (II - 5 - 28) un système de  $q$  équations à  $q+1$  inconnues.

En exprimant  $\Psi_1(x_1^+)$  et  $\Psi_1(x_n^-)$  respectivement en fonction  
 de  $Z(x_1)$  et  $Z(x_n^-)$  : (II - 5 - 11 et II - 5 - 12), on obtient un  
 système de  $q$  équations à  $q$  inconnues.

La  $(q+1)$ ième inconnue du vecteur  $v(x_1^+)$  est alors  
 issue à partir de la condition limite :

$$\Psi_1(x_1^+) = \int_1(Z(x_1) - z_1).$$

EXEMPLE :  $q = 2$  :

$$v(x_n^-) = P \cdot v(x_1^+) + q ; \quad P \in \mathbb{R}^{4 \times 4} ; \quad q \in \mathbb{R}^{4 \times 1} \quad (\text{II - 5 - 29})$$

1° - Si  $Z'(x_1^+)$  fixé et  $\Psi_2(x_n^-) = 0$  :

(II - 5 - 30)

$$\begin{bmatrix} (P_{31} - \int_1 \cdot P_{34}) & & & P_{33} \\ \int_n (P_{11} - P_{14} \cdot 1) - P_{41} + \int_1 \cdot P_{44} & & (\int_n \cdot P_{13} - P_{43}) & \end{bmatrix} \cdot \begin{bmatrix} Z(x_1^+) \\ \Psi_2(x_1^+) \end{bmatrix} =$$

$$\begin{bmatrix} -q_3 - \rho_1 \cdot P_3 - z_1 - P_{32} \cdot Z'(x_1^+) \\ \rho_n (z_n - q_1 - \rho_1 \cdot z_1 \cdot P_{14} - P_{12} \cdot Z'(x_1^+) + P_{42} \cdot Z'(x_1^+) - P_{44} \cdot z_1 + q_4) \end{bmatrix}$$

2° - Si  $Z'(x_1^+)$  et  $Z'(x_n^-)$  fixés :

$$\begin{bmatrix} (P_{21} - P_{24} \cdot \rho_1) & P_{23} \\ \rho_n (P_{11} - \rho_1 P_{14}) - P_{41} + P_{44} & \rho_n \cdot P_{13} - P_{43} \end{bmatrix} \cdot \begin{bmatrix} Z(x_1^+) \\ \Psi_2(x_1^+) \end{bmatrix} = \begin{matrix} \text{(II - 5 - 31)} \\ \\ \end{matrix}$$

$$\begin{bmatrix} Z'(x_n^-) - P_{22} \cdot Z'(x_1^+) - q_2 - P_{24} \cdot \rho_1 z_1 \\ \rho_n (z_n - q_1 - \rho_1 \cdot z_1 \cdot P_{12} - P_{12} \cdot z_1 \cdot Z'(x_1^+) + P_{42} \cdot Z'(x_1^+) - P_{44} \cdot z_1 + q_4) \end{bmatrix}$$

3° - Si  $\Psi_2(x_1^+) = 0$  et  $Z'(x_n^-)$  fixé :

(II - 5 - 32)

$$\begin{bmatrix} (P_{21} - P_{24} \cdot \rho_1) & P_{22} \\ (\rho_n (P_{11} - \rho_1 \cdot P_{14}) - P_{41} + \rho_1 \cdot P_{44}) & \rho_n \cdot P_{12} - P_{42} \end{bmatrix} \cdot \begin{bmatrix} Z(x_1^+) \\ Z'(x_1^+) \end{bmatrix} =$$



$$= \begin{bmatrix} Z'(x_n^-) - P_{24} \cdot \rho_1 \cdot z_1^{-q_2} \\ \rho_n (z_n^{-q_1} - \rho_1 \cdot P_{14} \cdot z_1) - P_{44} \cdot z_1^{+q_4} \end{bmatrix}$$

4° - Si  $\Psi_2(x_1^+) = \Psi_2(x_n^-) = 0$  :

(II - 5 - 33)

$$\begin{bmatrix} P_{31} - P_{34} \cdot \rho_1 & P_{32} \\ \rho_n (P_{11} - \rho_1 \cdot P_{14}) - P_{41} + \rho_1 \cdot P_{44} & \rho_n \cdot P_{12} - P_{42} \end{bmatrix} \cdot \begin{bmatrix} Z(x_1^+) \\ Z'(x_1^+) \end{bmatrix} =$$

$$\begin{bmatrix} -q_3 - P_{34} \cdot \rho_1 \cdot z_1 \\ \rho_n (z_n^{-q_1} - \rho_1 \cdot P_{14} \cdot z_1) - P_{44} \cdot z_1 + q_4 \end{bmatrix}$$

La détermination des vecteurs  $v(x_1^+)$  ;  $\forall i=2, \dots, n-1$  utilise la propriété de continuité de  $Z(x)$ , pour les  $(2q-1)$  premières composantes :

Il vient :

$$v_j(x_{i+1}^-) = v_j(x_{i+1}^+) \quad \forall i=1, \dots, n-1$$

$$\forall j=1, \dots, 2q-1$$

La dernière composante est obtenue à partir de la dernière ligne de l'équation de progression et de la condition limite (II - 5 - 15) :

$$v_{2q}(x_{i+1}^-) = v_{2q}(x_i^+)$$

soit

$$v_{2q}(x_{i+1}^+) = v_{2q}(x_i^+) + \rho_{i+1} \cdot (z_{i+1} - v_1(x_{i+1}))$$

#### II - 5 - 6 : REALISATION DU PROCESSUS D'AJUSTEMENT :

Le schéma hybride développé au paragraphe précédent (§ II - 4) est applicable au processus d'ajustement sous réserve de nullité des coefficients  $a_i$  (cf Fig - 31).

Nous présentons ci-joint l'organigramme correspondant à la séquence d'opérations explicitées précédemment.

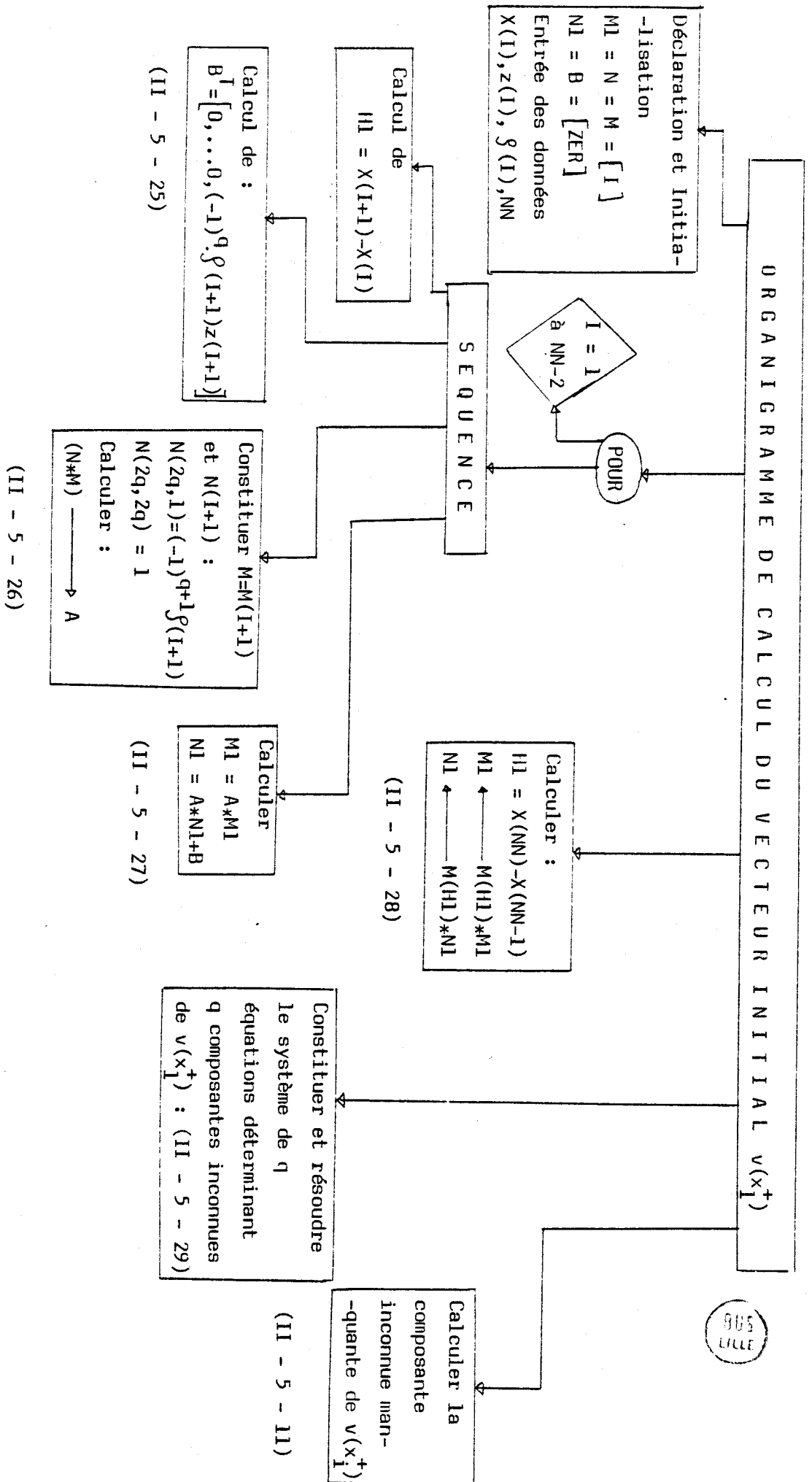
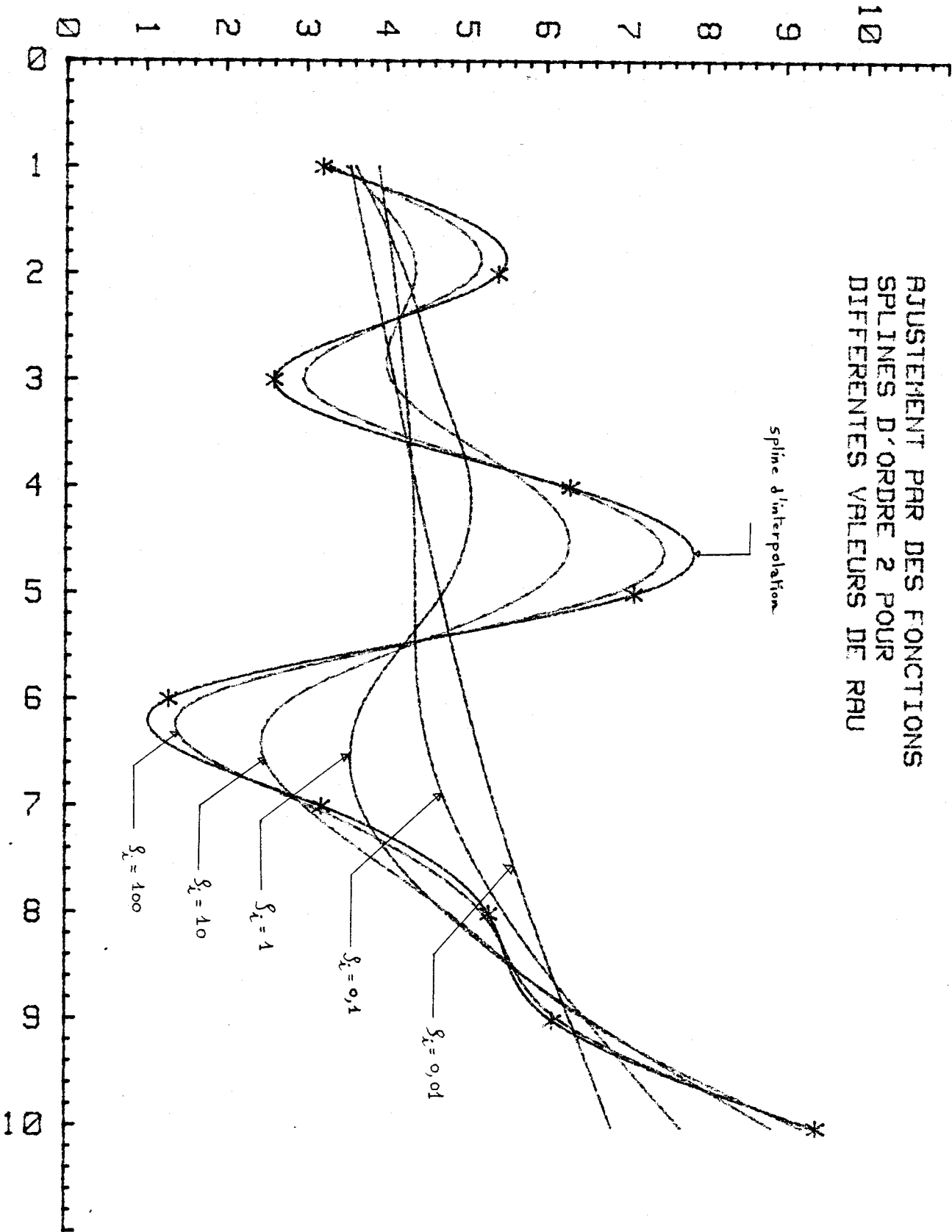


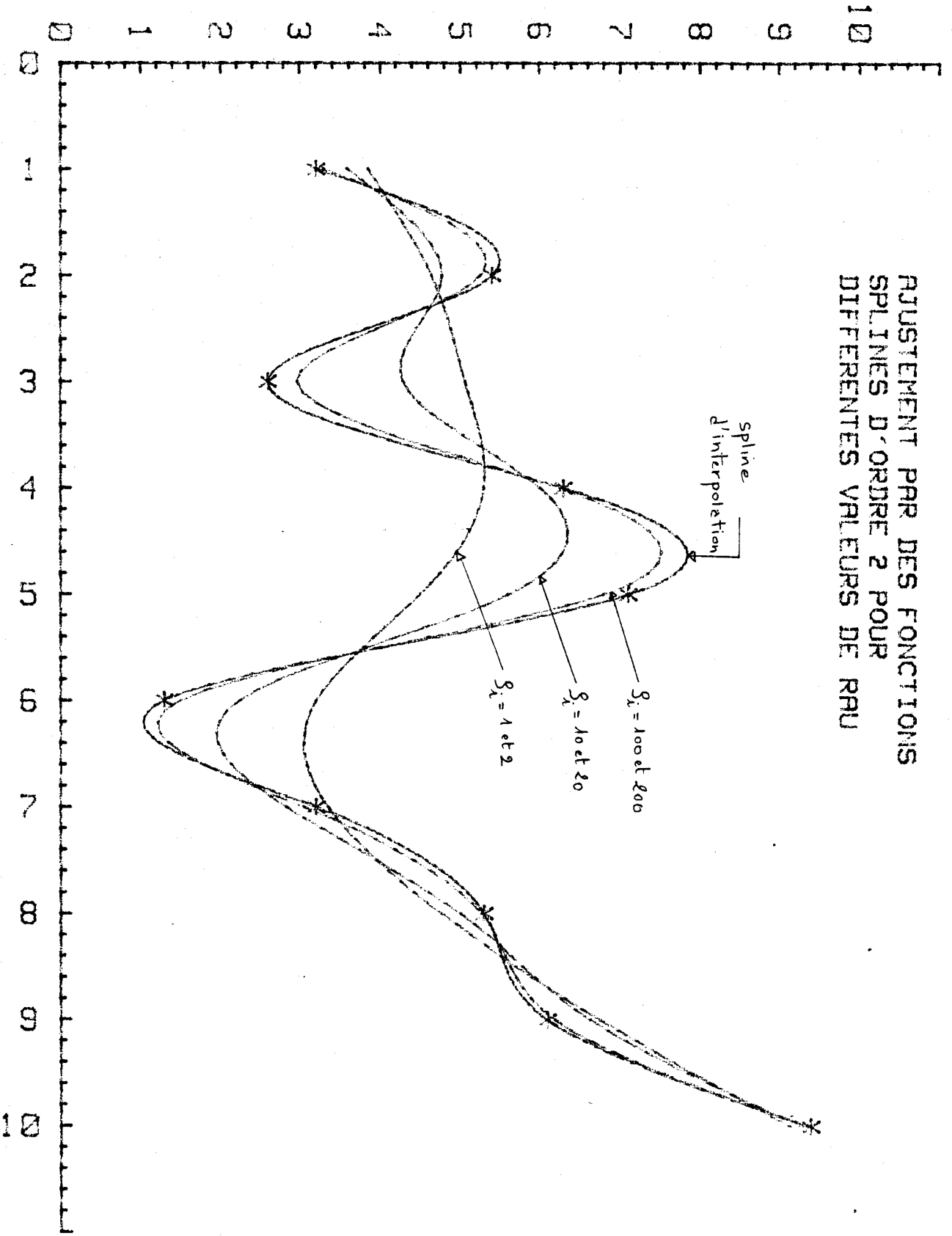
Fig - 35

ADJUSTEMENT PAR DES FONCTIONS  
SPLINES D'ORDRE 2 POUR  
DIFFERENTES VALEURS DE RAU

spline d'interpolation



AJUSTEMENT PAR DES FONCTIONS  
SPLINES D'ORDRE 2 POUR  
DIFFERENTES VALEURS DE  $R_{\lambda}$



II - 6 : INSTABILITE NUMERIQUE :

Lorsque le nombre  $n$  de points à interpoler, d'un réseau  $\{x_i, y_i ; i=1, \dots, n\}$  est élevé, le résultat obtenu à partir de l'application de l'algorithme précédent peut être le suivant (cf Fig - II-6-I)

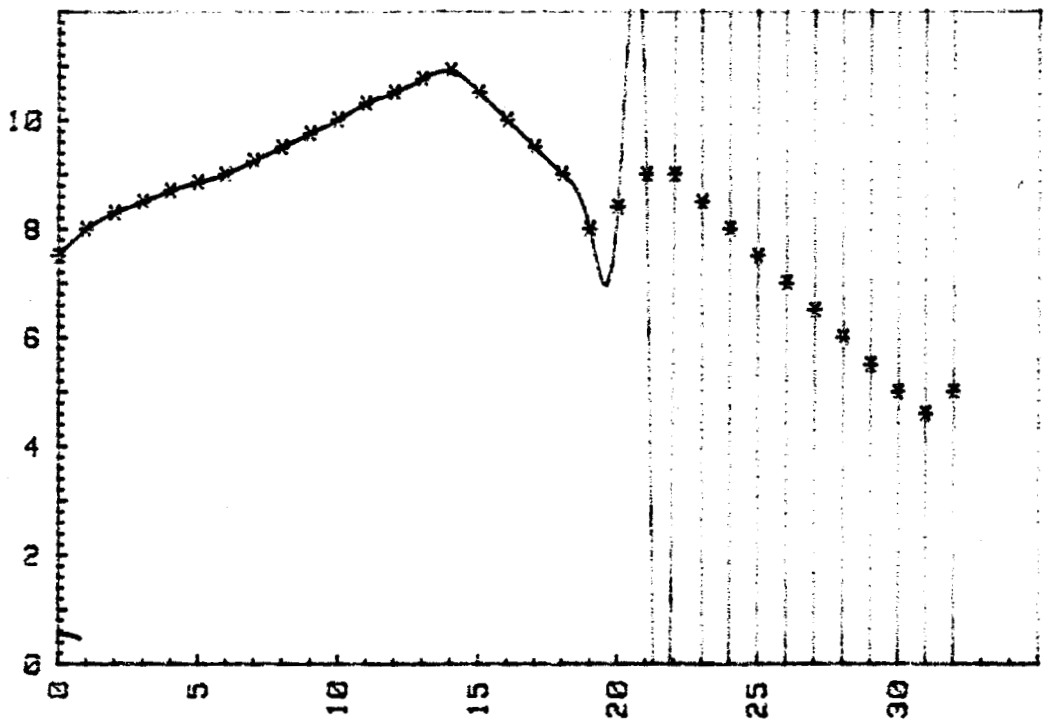


Fig - II-6-I

Le comportement de la courbe décrite sur ce

graphique ne peut en aucun cas s'expliquer à partir des seules données d'interpolation.

Pour s'en convaincre, il suffit de résoudre le même problème avec un programme dans lequel les variables sont représentées en simple précision puis avec le même programme où les variables sont définies en double précision.

Il existe des données pour lesquelles les résultats du premier programme divergent (cf Fig -II-6-2) alors que l'interpolation issue du deuxième programme est encore admissible (cf Fig - II-6-3)

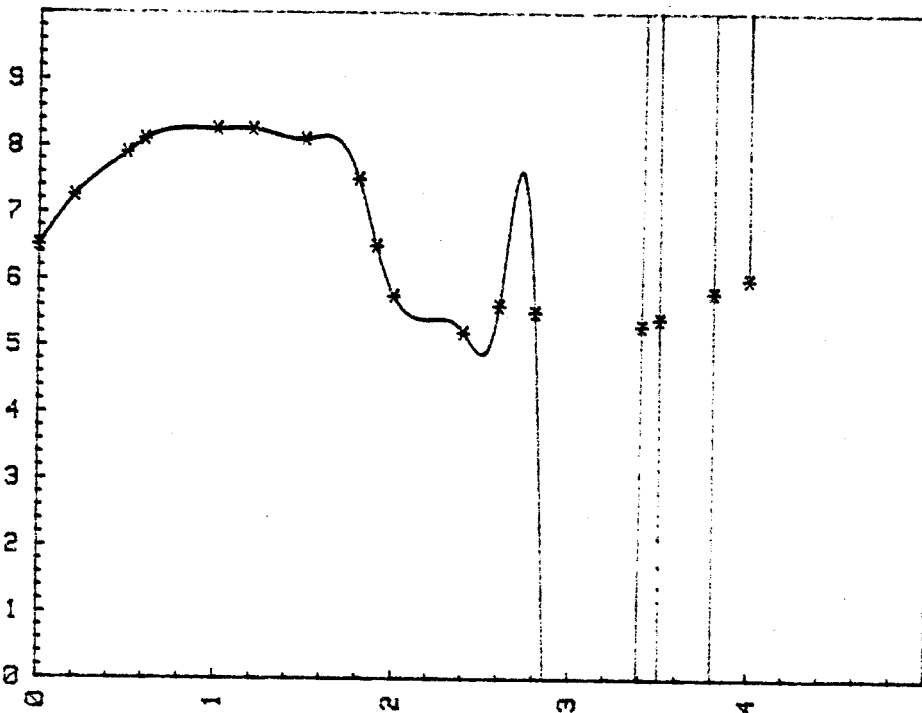


Fig - II-6-2

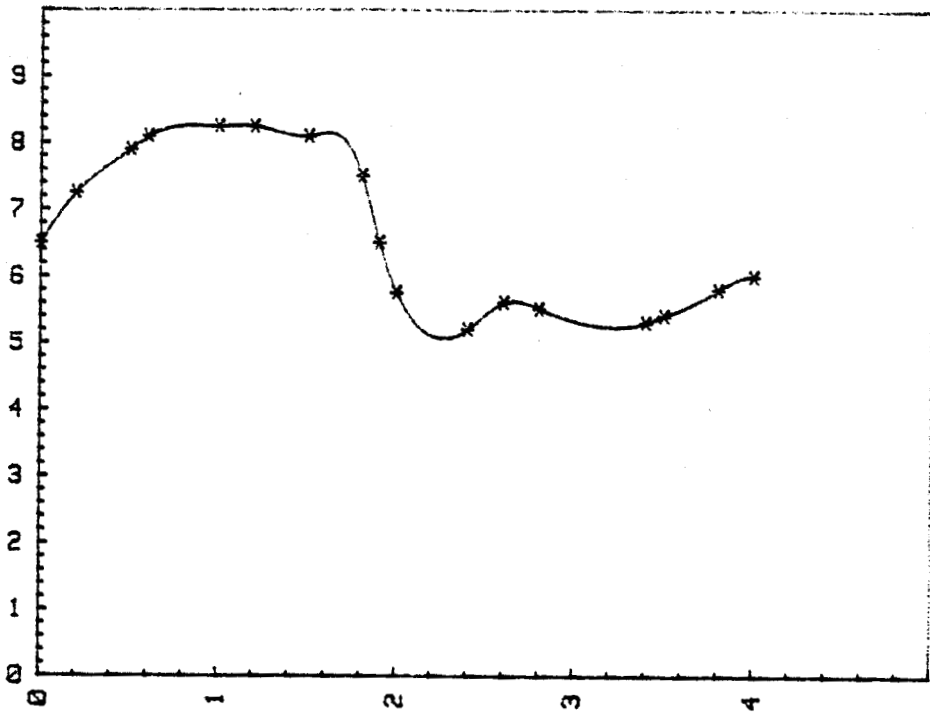


Fig - II-6-3

Le comportement instable de la courbe d'interpolation est donc explicable à cause des erreurs introduites par la troncature des nombres représentés en mémoire du calculateur.

On a ainsi affaire à un problème d'instabilité numérique.

Considérons maintenant le mécanisme de cette instabilité.



L'algorithme du "processus interpolateur" détermine, à partir d'un réseau de points, un vecteur  $v(x_1^+)$  obtenu comme la solution d'un système linéaire (CHAP II; II-2-37)

Ce vecteur  $v(x_1^+)$  induit l'interpolation sur l'intervalle  $[x_1, x_2[$ .

De même, l'interpolation sur un intervalle  $[x_i, x_{i+1}[$  est caractérisée par un vecteur de paramètre  $v(x_i^+)$ .

Ce dernier est comme on l'a vu, relié au précédent par une relation affine :

$$v(x_{i+1}^+) = A(i+2) \cdot v(x_i^+) + b(i+2). \quad (\text{II} - 6 - 1)$$

Cette dernière relation nous permet de calculer la suite  $\{ v(x_i^+) ; i=2, \dots, n-1 \}$  à partir du calcul initial de  $v(x_1^+)$ .

Considérons maintenant une erreur  $\delta v(x_1^+)$  introduite lors du calcul de  $v(x_1^+)$ .

On note  $\tilde{v}(x_1^+)$  l'estimation (contenue dans la mémoire du calculateur) de  $v(x_1^+)$ , il vient alors :

$$v(x_1^+) = \tilde{v}(x_1^+) + \delta v(x_1^+). \quad (\text{II-6-2})$$

L'erreur induite  $\delta v(x_2^+)$  sur l'estimation  $\tilde{v}(x_1^+)$  est alors, dans le cas le plus favorable où  $A(3)$  et  $b(3)$  sont représentés sans erreur, déterminée par la différence des deux équations d'état suivantes :

$$v(x_2^+) = A(3).v(x_1^+) + b(3) \quad \text{en théorie} \quad (\text{II-6-3})$$

$$\tilde{v}(x_2^+) = A(3).v(x_1^+) + b(3) \quad \text{par calcul.}$$

$$\text{Il vient donc :} \quad \delta v(x_2^+) = A(3) . \delta v(x_1^+)$$

$$\text{et en itérant} \quad \delta v(x_i^+) = \left[ \prod_{j=3}^{i+1} A(j) \right] . \delta v(x_1^+) \quad (\text{II-6-4})$$

L'erreur initiale  $\delta v(x_1^+)$  est ainsi répercutée sur les paramètres d'interpolation des segments  $[x_i, x_{i+1}]$  du réseau.

Dans le cas où le rayon spectral de  $A(i)$  est égal à 1, l'erreur commise dans le calcul de  $v(x_1^+)$  se répercuté sur  $v(x_i^+)$  en se cumulant avec les erreurs de troncature commises en calculant :

$$v(x_{i+1}^+) = A(i+2) \cdot v(x_i^+) + b(i+2)$$

Dans le cas où le rayon spectral de  $A(i)$  est supérieur à 1, alors  $\delta v(x_i^+)$  tend vers l'infini lorsque  $i$  tend vers l'infini.

On peut montrer par ailleurs que le rayon spectral de  $A(i)$  est toujours supérieur ou égal à 1.

Une manière de parer à ces erreurs consiste alors à rendre indépendante l'erreur  $\delta v(x_{i+1})$  et les erreurs précédentes.

## II - 6 - 1 : STRATEGIE EN "BOUCLE FERMEE" :

La méthode proposée est issue de considérations expérimentales sur les courbes obtenues en cas d'instabilité numérique.

Le début du tracé de la courbe est admissible puis diverge et ceci, quelque soient les données qu'on a pu rentrer.

Le remède à l'instabilité est alors simple.

Chaque tracé doit être celui d'un début de  
courbe.

Pour cela, on considère le réseau :

$$\{x_i, y_i ; i=1, \dots, n ; \lceil y'(x_1) \rceil \lceil y'(x_n) \rceil\} \quad (\text{II-6-5})$$

On calcule  $v(x_1)$  en faisant une erreur, on  
trace la fonction obtenue sur  $[x_1, x_2]$  et ce faisant, on  
obtient  $v(x_2), v'(x_2), v''(x_2)$ ,

On considère alors le nouveau réseau

$$\{(x_i, y_i) ; i=2, \dots, n ; y'(x_2), \lceil y'(x_n) \rceil\} \quad (\text{II-6-6})$$

raccordé au précédent à l'ordre 1 et on calcule alors  $v(x_2^+)$ , après  
avoir fait des erreurs, comme si on ne connaissait pas  $v(x_1^+)$ , etc..

Du fait de l'unicité de la solution, cette valeur  
de  $v(x_2^+)$  est égale à celle déterminée par le pas précédent dans  
le cas où il n'y a pas d'erreur de calcul.

Dans le cas contraire, il y a discontinuité de la  
dérivée seconde aux noeuds.

En d'autres termes, on recale la dérivée seconde de manière à limiter l'amplitude de la courbe obtenue.

L'arbre programmatique correspondant à cette solution est le suivant :

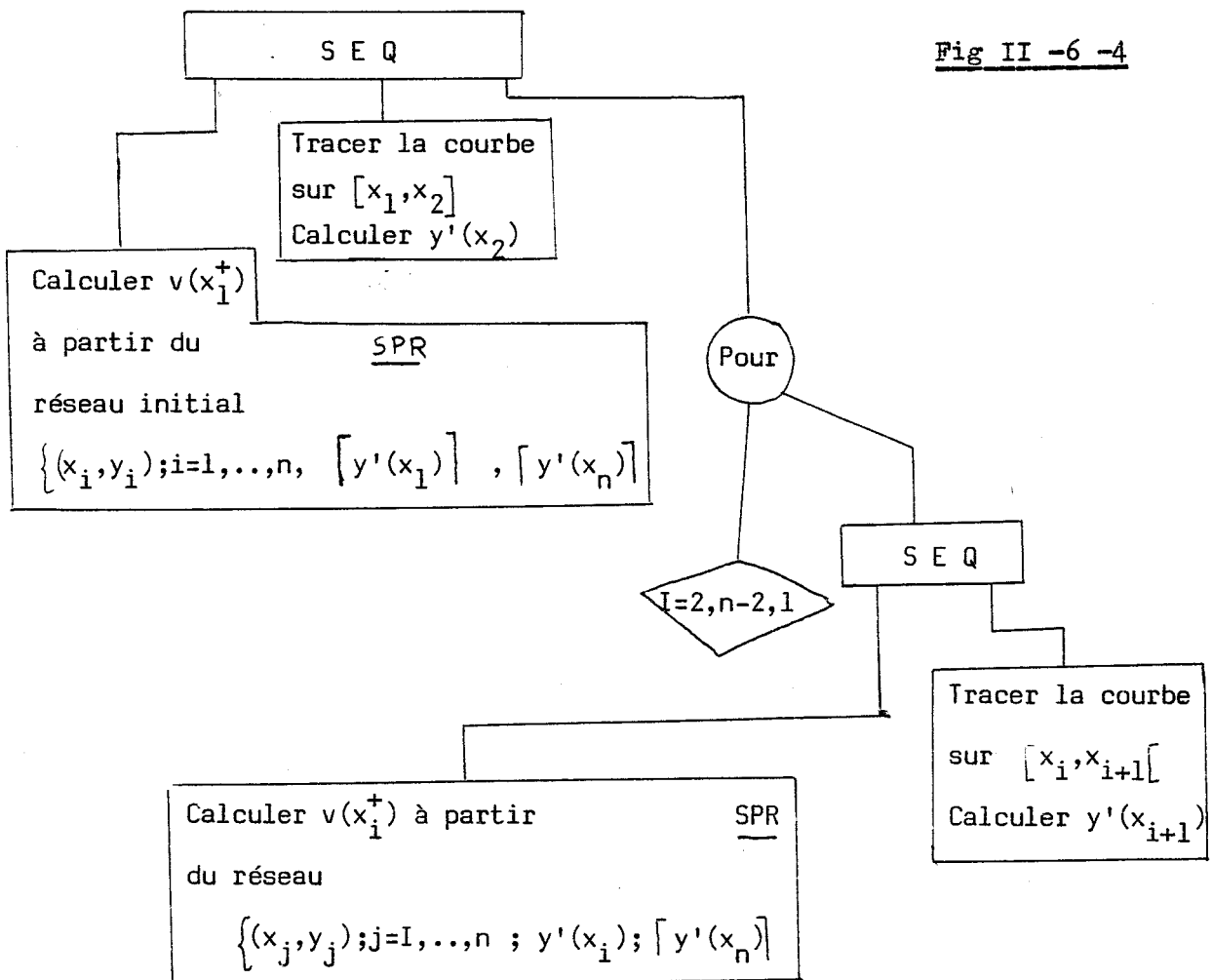


Fig II -6 -4

La procédure SPR est celle exposée dans le précédent paragraphe.

Cette utilisation de l'algorithme du processus interpolateur est naturellement très lourde en temps de calcul mais la fiabilité des résultats est alors bien supérieure.

A titre d'exemple, les courbes suivantes représentent, sur le même diagramme, les solutions issues d'une application directe de l'algorithme du processus interpolateur et de l'application "en boucle fermée" du même algorithme .

II - 6 - 2 : STRUCTURE DU PROCESSUS INTERPOLATEUR  
EN BOUCLE FERMEE :

De manière à mettre en évidence le bouclage introduit dans l'algorithme, au niveau du bloc de calcul, nous définissons la structure du processus interpolateur en boucle fermée suivante ( Fig II -6 -5)

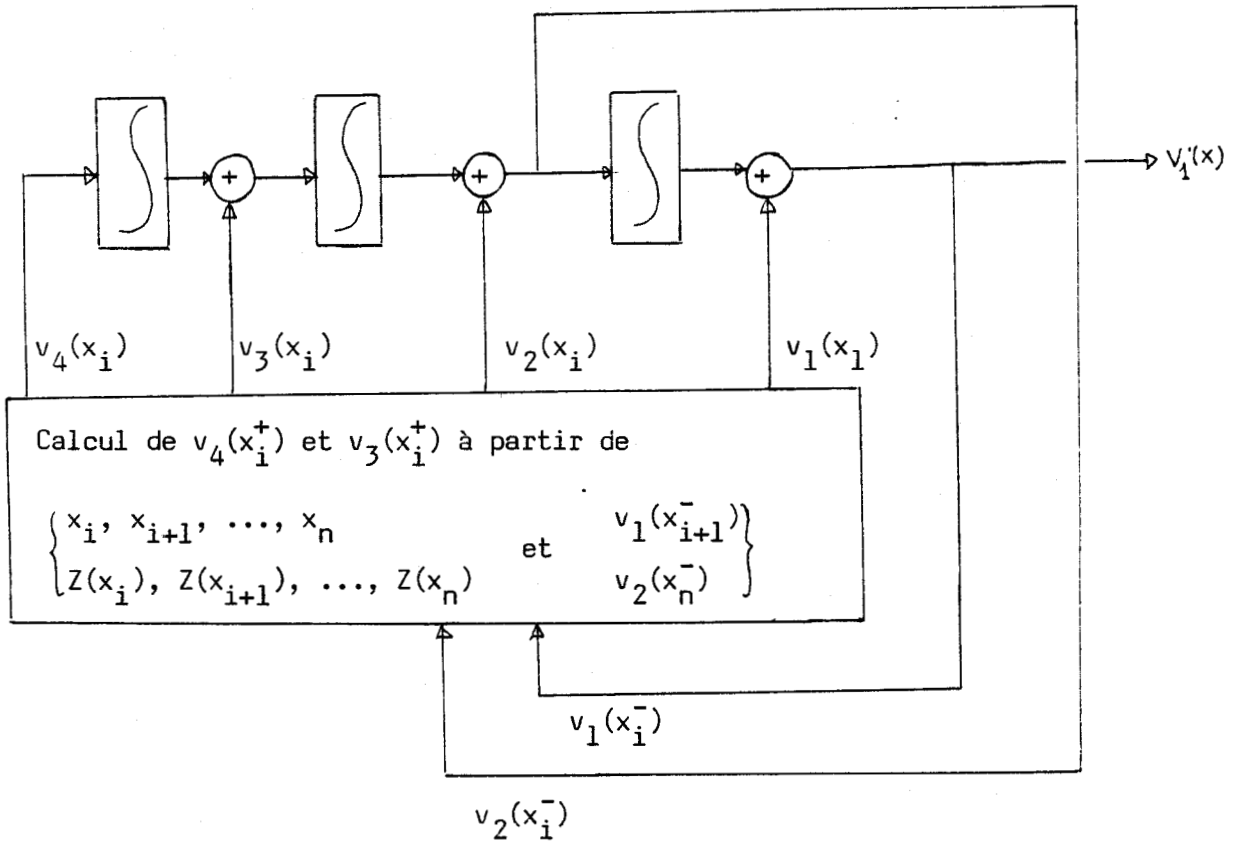


Fig II-6-5 : STRUCTURE DU PROCESSUS INTERPOLATEUR EN BOUCLE FERMEE

Cette structure met en évidence l'existence d'une boucle de retour entre les calculateurs analogique et numérique et correspond bien à une structure en boucle fermée.

En particulier, la condition initiale  $v_3(x_i)$  (qui n'est autre que la commande  $u$ ) est déterminée à partir du vecteur d'état  $Z^T(x_i) = [v_1(x_i^-), v_2(x_i^-)]$  à cette abscisse, et des composantes  $v_1(x_{i+1}^-)$  et  $v_2(x_n^-)$  (resp  $v_3(x_n^-)$ ) futures, conformément arbre programmatique (Fig II-6-4)

II - 6 - 2 - 1 : Réalisation du processus

interpolateur en boucle fermée :

Le schéma correspond<sup>ant</sup> à une réalisation hybride est illustré (Fig II -6 -6)

Le calculateur analogique effectue par intégration le calcul de la trajectoire sur tout intervalle  $[x_i, x_{i+1}]$ .

Ce calcul est initialisé au point  $x=x_1$  à partir du transfert par le calculateur numérique des composantes initiales :

- $Z'''(x_1)$  sur l'entrée B0 de l'intégrateur N°1,
- $Z''(x_1)$  sur l'entrée B1 de l'intégrateur N°2 et  $x_2$  sur l'entrée B2 du comparateur.

Le calculateur analogique est alors en mode opération.

Lorsque la sortie du comparateur commute, ce qui signifie que  $x$  est sorti de l'intervalle  $[x_1, x_2]$ , le calculateur analogique transfère les valeurs  $Z(x_2^-)$  et  $Z'(x_2^-)$  au calculateur numérique qui fournit à son tour les nouvelles entrées  $-Z'''(x_2)$  sur la borne B0,  $-Z''(x_2)$  sur la borne B1 et l'entrée  $x_3$  sur la borne B2.

Ce processus est ainsi répété suivant le schéma (Fig II-6-7) jusqu'à ce que  $x$  devienne égal à  $x_n$ , ce qui déclenche l'arrêt du calculateur analogique.



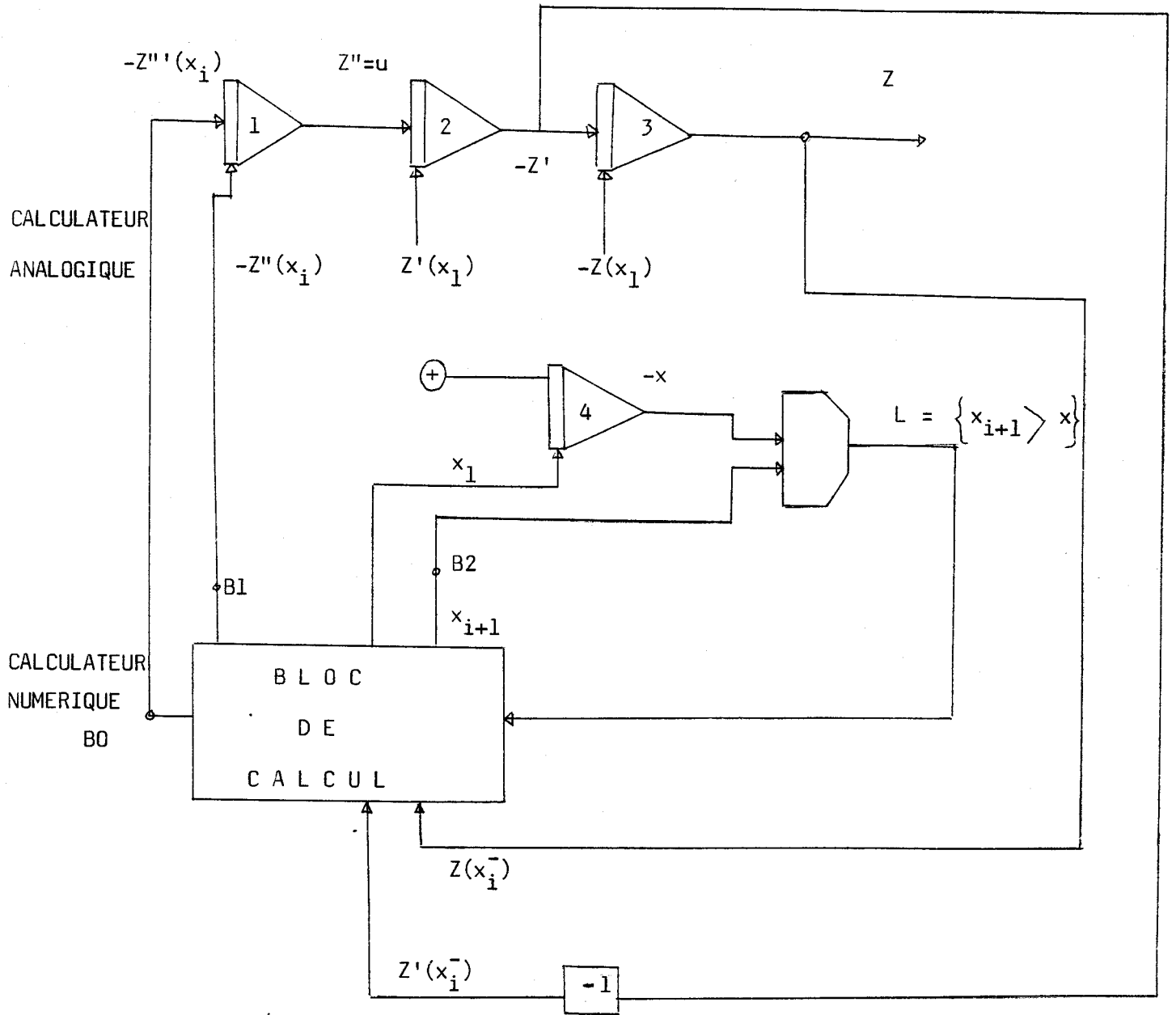


Fig II -6 -6 REALISATION HYBRIDE DU PROCESSUS INTERPOLATEUR EN BF :  
FONCTIONS SPLINES D'INTERPOLATION D'ORDRE 2

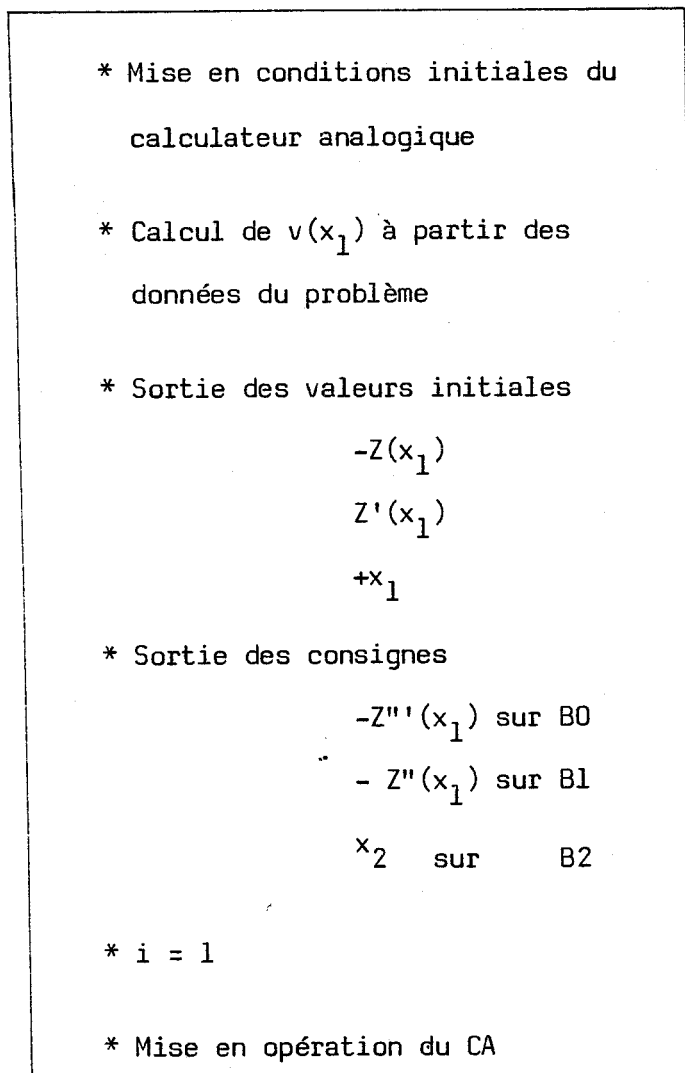
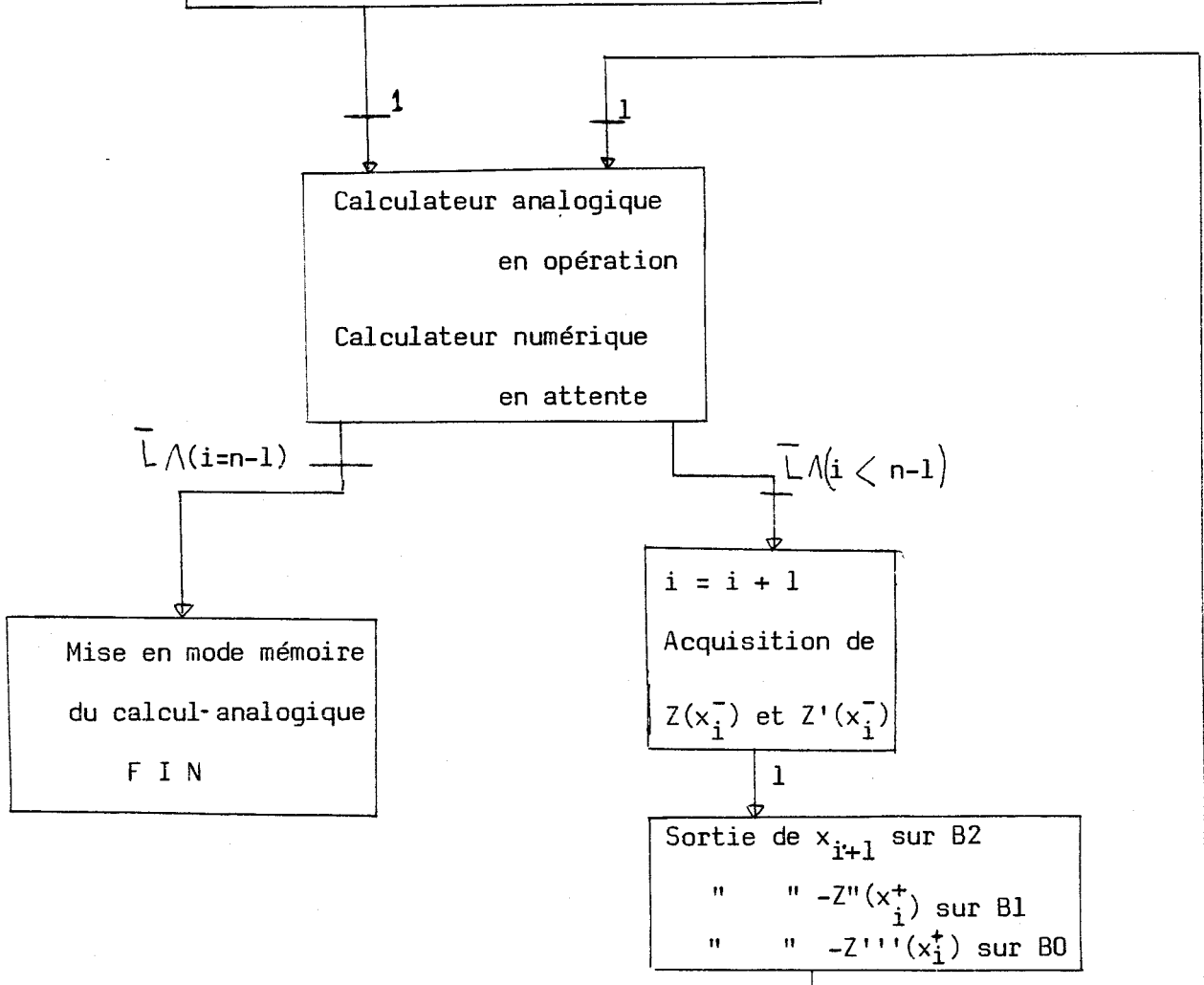


Fig II-6-7



II - 6 - 3 : METHODE DE DECOMPOSITION EN SOUS RESEAUX :

Une deuxième méthode de résolution du problème d'instabilité numérique consiste à décomposer la courbe à générer en une succession de morceaux de courbes numériquement stables, définies sur des sous-réseaux, puis à les raccorder entre eux.

Chaque morceau de courbe est représenté par une fonction spline d'ordre  $q$ , satisfaisant les propriétés (P1  $\longrightarrow$  P4 : cf § II - 2 - 1), et considéré comme la trajectoire issue d'un processus interpolateur.

La structure et le calcul de chacun des processus interpolateur sont établis par l'algorithme présenté aux paragraphes précédents.

Le raccordement de ces trajectoires est réalisé de manière à assurer la continuité de la fonction ainsi que de ses dérivées jusqu'à l'ordre  $(q-1)$ .

Si nous faisons choix d'une subdivision  $R_j$  du réseau  $R = [x_1, x_n]$ , tel que (II - 6 - 7)

$$\cup R_j = R$$

(II - 6 - 7)

$$\text{Si } R_j = [x_1, \dots, x_k] \text{ et } R_{j+1} = [x_k, \dots, x_1] ; k < 1 \leq n$$

sont deux sous-réseaux consécutifs vérifiant les conditions

suyvantes (II - 6 - 8) :

$$x_i < x_{i+1} \quad \forall i=1, 1-1 \quad (\text{II} - 6 - 8)$$

et si on désigne par  $v_{j+1}(x)$  le vecteur associé à la trajectoire définie sur  $R_{j+1}$ , (II - 6 - 9)

$$v_{j+1}^I(x) = \left[ Z_{j+1}(x), Z'_{j+1}(x), \dots, Z_{j+1}^{(q)}(x), \Psi_{q_{j+1}}(x), \dots, \pm \Psi_{1_{j+1}}(x) \right] \quad (\text{II} - 6 - 9)$$

alors la continuité entre les deux trajectoires  $Z_j(x)$  et  $Z_{j+1}(x)$  impose les conditions aux limites suivantes (II - 6 - 10) :

$$* Z_{j+1}^{i-1}(x_k^+) = Z_j^{i-1}(x_k^-) \text{ fixées en } x_k^+ ; \forall i=1, \dots, q.$$

et (II - 6 - 10)

$$* Z_{j+1}^{i-1}(x_1^-) \text{ fixés en } x_1^- ; \forall i=1, \dots, q$$

$$\text{sinon } \Psi_{i_{j+1}}(x_1^-) = 0 \quad \forall i=2, \dots, q$$

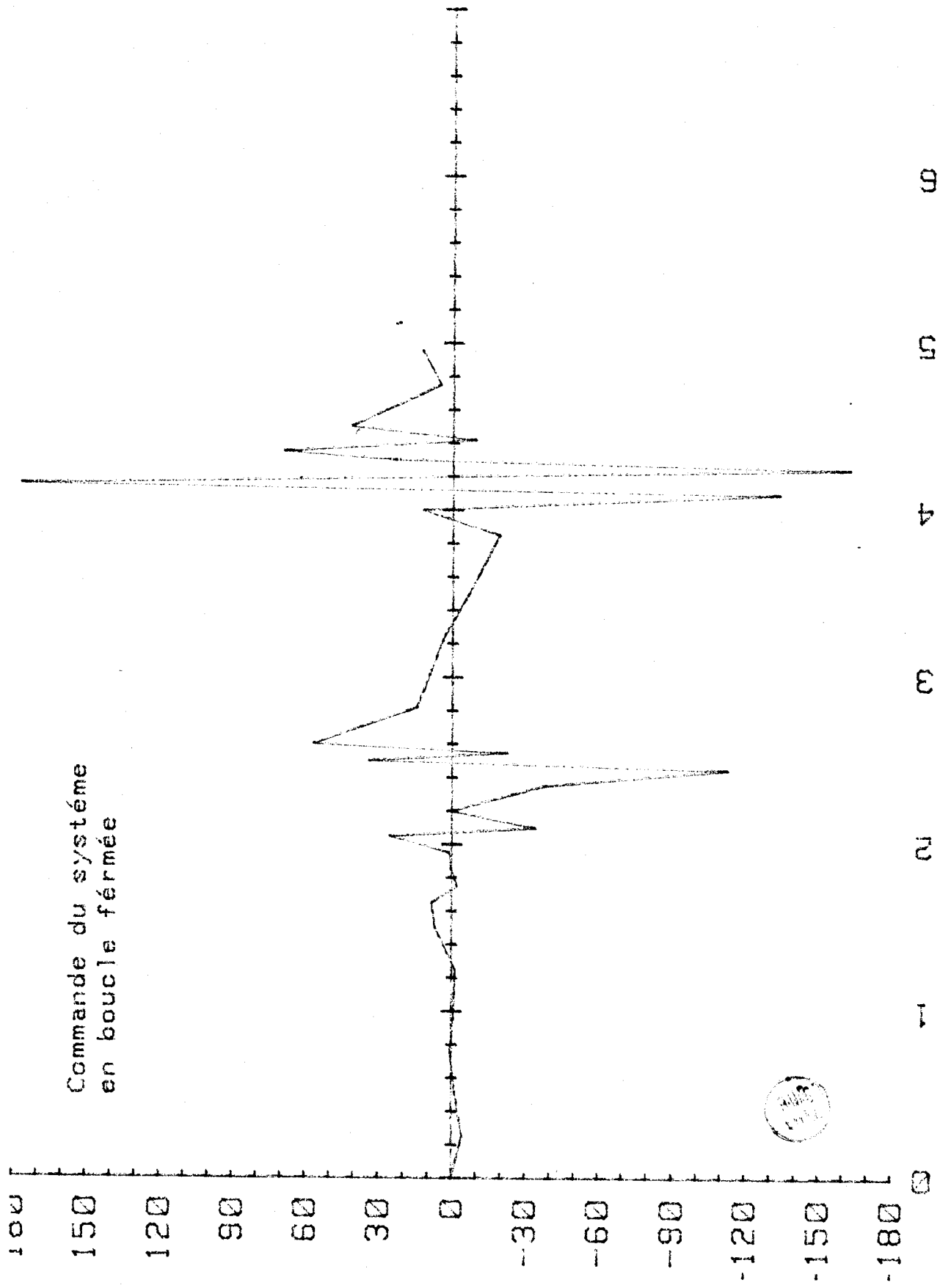
Par rapport à la méthode du processus interpolateur en boucle fermée, cette technique présente l'avantage d'être plus rapide puisqu'en général, une subdivision de R en trois ou quatre sous-réseaux suffit.

Néanmoins, la courbe obtenue n'est pas en général optimale au sens du critère dans la mesure où on a la relation (II - 6 - 11) :

$$\min J \neq \min \{ J_1 + J_2 + \dots + J_n \} \quad (\text{II - 6 - 11})$$

Par ailleurs, il n'existe à priori aucun renseignement permettant de réaliser de manière optimum (temps minimum de traitement) la subdivision du réseau initial, de sorte qu'à chaque choix correspond une solution.

Commande du système  
en boucle fermée



II-7 : COMPARAISON DES ALGORITHMES

Dans ce paragraphe, nous effectuons une comparaison en nombre d'opérations nécessités par différents algorithmes présentés au chapitre I avec celui de la méthode du processus interpolateur. Etant donné la nature du problème d'interpolation de données ponctuelles, nous restreignons la comparaison respectivement aux méthodes d'intégration et de transport, en esquissant la méthode des fonctions de BEZIER (cf. chap I), et des fonctions B-splines.

Dans ce contexte, nous proposons de comptabiliser le nombre d'opérations exigé par l'algorithme en boucle ouverte, au cas des fonctions splines d'interpolation d'ordre 2 (chap II, § II-2) en tenant compte de la spécificité des matrices et vecteurs intervenant dans l'algorithme.

Dans ce cas la matrice A (cf II-2-30) est de dimension (4,4) et le vecteur B (cf II-2-29) de dimension (4,1).

La détermination du vecteur  $v(x_1^+)$  requiert, dans un premier temps, le calcul à chaque pas de la matrice A pour laquelle,

5 divisions
11 multiplications
5 additions

sont exigées, soit au total :

5 (N-2) divisions
11 (N-2) multiplications
5 (N-2) additions

D'autre part,  $v(x_1^+)$  nécessite le calcul de produits matriciels, dont le nombre d'opérations est, compte-tenu de la forme de A et B.

64 (N-2)- 41 additions
81 (N-2)- 51 multiplications
2 (N-2) + soustractions

A ce niveau, il y a lieu d'effectuer la résolution d'un système d'ordre 2 pour lequel 13 opérations sont nécessaires.

Le calcul des vecteurs  $v(x_i^+)$  en tout point  $x_i$  du réseau;  
 $\forall i=2, \dots, n-1$  nécessite :

7 (N-3) multiplications
3 (N-3) divisions
3 (N-3) additions
3 (N-3) soustractions

Au total, l'algorithme requiert :

99 (N-2)-41 multiplications
72 (N-2)-37 additions
8 (N-2)+6 divisions
5 (N-2) soustractions

Avant de comparer la complexité des opérations, il est judicieux de spécifier les remarques suivantes.

REMARQUES :

1) Le nombre d'opérations établi précédemment peut être diminué dans des proportions non négligeables, si les abscisses  $x_i$  sont équidistantes. En particulier, l'algorithme ne requiert, dans ce cas, qu'une seule fois le calcul de la matrice A.



2) Par ailleurs, le dénombrement des opérations ne tient pas compte de la propriété suivante de la matrice A.

La première ligne de  $A^2$  (et par conséquent de  $A^i$  pour tout  $i=3, \dots, N-2$ ) est nulle.

3) Cette méthode établit, de façon directe, les composantes de  $v(x_i^+)$ , sans opérer de transformations sur les relations.

4) De manière à minimiser le calcul de la fonction spline d'interpolation en un point  $x \in [x_i, x_{i+1}]$ , l'algorithme ne requiert pas de schéma supplémentaire du type AITKEN-NEVILLE à arguments répétés.  
(cf. § II-1)

5) Si l'on ne tient pas compte de la forme, des matrices N, M, A et B, le nombre total d'opérations requis par une fonction spline d'ordre q est :

$\times 4 q^2 (4q+1).(N-2) + q^2+q-1$	multiplications
$\times 2 q (8q^2-4q-1).(q^2-1)$	additions
$\times$ Inversion d'une matrice d'ordre q pour l'établissement du vecteur $v(x_1^+)$ , et :	
$\times (2q-1).(N-1) .$	$\left\{ \begin{array}{l} (2q-j) \text{ multiplications} \\ (2q-1) \text{ additions} \\ (2q-j-1) \text{ divisions} \end{array} \right. \quad \forall j=2, \dots, 2q-1$
$\times (N-2) .$	$\left\{ \begin{array}{l} 4q-3 \text{ multiplications} \\ 2q-1 \text{ additions} \\ 2q-2 \text{ divisions} \end{array} \right.$

Au chapitre I, nous avons étudié la méthode d'intégration sur les fonctions splines d'ordre 2. Dans ce cas, la méthode consiste à déterminer les dérivées secondes de la fonction spline en tout point  $x_i$ , en résolvant le système d'équations d'ordre  $n-2$ . (MARCHOUK, 1980)

$$A \cdot M = H \cdot f \quad (\text{II-7-1})$$

avec :

- { A matrice de dimension  $(n-2, n-2)$
- { M vecteur colonne de dimension  $(n-2)$
- { H matrice de dimension  $(n-2, n)$
- { f vecteur colonne de dimension  $(n)$

La détermination du vecteur M (lorsque les dérivées secondes aux extrémités sont nulles) requiert les opérations suivantes :

- Inversion d'une matrice carrée d'ordre  $(n-2)$
- $(2n-2) \times (n-2)$  multiplications
- $2 (n-2)^2$  additions

Ce nombre est supérieur à celui requis par l'algorithme du processus interpolateur pour un nombre de points suffisamment grand. Par ailleurs, lorsque  $n$  est élevé, l'inversion de la matrice A augmente d'une part la complexité en nombre d'opérations, ainsi que les risques d'instabilité.

On recourt à des algorithmes de résolution qui sont plus stables.

Par ailleurs, pour des valeurs de  $q > 2$ , la méthode d'intégration nécessite qu'on résolve  $(q-1)$  systèmes d'équations du type.

$$A \cdot M = H \cdot f$$

En effet considérons à titre d'exemple le cas des splines d'interpolation d'ordre 3.

Dans chaque intervalle  $[x_i, x_{i+1}]$ ,  $f(x)$  s'écrit

$$\begin{aligned}
 f(x) = & f^{(4)}(x_i) \frac{(x_{i+1} - x)^5}{120 h_{i+1}} + f^{(4)}(x_{i+1}) \frac{(x - x_i)^5}{120 h_{i+1}} + A_{i+1} \frac{(x_{i+1} - x)^3}{6 h_{i+1}} \\
 & + B_{i+1} \frac{(x - x_i)^3}{6 h_{i+1}} + C_{i+1} \frac{x_{i+1} - x}{h_{i+1}} + D_{i+1} \frac{x - x_i}{h_{i+1}}
 \end{aligned}$$

(II-7-2)

(II-7-2)

avec

$A_{i+1}$ ,  $B_{i+1}$ ,  $C_{i+1}$  et  $D_{i+1}$  étant les constantes d'intégration

Pour les calculer, il y a lieu de disposer de quatre relations, dont deux sont établies à partir de la propriété d'interprétation.

Il vient :

$$\begin{aligned}
 f(x_i) &= f^4(x_i) \frac{h_{i+1}^4}{120} + A_{i+1} \frac{h_{i+1}^2}{6} + C_{i+1} \\
 f(x_{i+1}) &= f^{(4)}(x_{i+1}) \frac{h_{i+1}^4}{120} + B_{i+1} \frac{h_{i+1}^2}{6} + D_{i+1}
 \end{aligned}$$

(II-7-3)

Les deux autres relations sont fournies par exemple par les dérivées secondes en  $x_i$  et  $x_{i+1}$

$$\begin{aligned}
 f''(x_i) &= f^{(4)}(x_i) \frac{h_{i+1}^2}{6} + A_{i+1} \\
 f''(x_{i+1}) &= f^{(4)}(x_{i+1}) \frac{h_{i+1}^2}{6} + B_{i+1}
 \end{aligned}$$

(II-7-4)

Pour déterminer  $f''(x_i)$ ,  $f''(x_{i+1})$  et  $f^4(x_i)$ ,  $f^4(x_{i+1})$  il y a lieu de résoudre deux systèmes du type  $A.M = H.f$ . Ces systèmes sont obtenus en écrivant respectivement la relation de continuité pour  $f'(x)$  et  $f^{(3)}(x)$  en  $x_i$ . ( cf chap. I, § I-4-2)

II-7-2 : COMPARAISON AVEC LA METHODE DE TRANSPORT

La détermination de la fonction spline d'ordre 2 (res ordre q) par la méthode de transport des relations consiste à résoudre en tout point  $x_i$  un système d'équations linéaires d'ordre 2 (resp  $2q-2$ ) entre les composantes  $v_2(x_i^+)$  et  $v_3(x_i^+)$  ( resp  $v_j(x_i^+)$  ;  $\forall j = 2, \dots, 2q-1$ ) du vecteur  $v(x_i^+)$ .

Dans chaque intervalle  $[x_i, x_{i+1}]$ , la fonction  $f(x)$  est définie à partir des données  $v_1(x_i)$ ,  $v_1(x_{i+1})$  et des valeurs des  $q-1$  premières dérivées en ces points suivant le schéma de AITKEN-NEVILLE; (M. PETIT, 1971)

Pour  $q=2$ , il vient :

$$\left\{ \begin{array}{ll} x_i & v_1(x_i) \\ x_i & v_1(x_i) \\ x_{i+1} & v_1(x_{i+1}) \\ x_{i+1} & v_1(x_{i+1}) \end{array} \right. \begin{array}{l} \\ v_2(x_i) \\ A \\ v_2(x_{i+1}) \end{array} \begin{array}{l} \\ \underline{\underline{B}} \\ \\ C \end{array} \begin{array}{l} \\ \\ D \\ \end{array} \quad (II-7-5)$$

avec  $h_{i+1} = x_{i+1} - x_i$ , les coefficients A, B, C et D sont donnés par :

$$\begin{array}{ll} A = \frac{v_1(x_{i+1}) - v_1(x_i)}{h_{i+1}} & B = \frac{A - v_2(x_i)}{h_{i+1}} \\ C = \frac{v_2(x_{i+1}) - A}{h_{i+1}} & D = \frac{C - B}{h_{i+1}} \end{array} \quad (II-7-6)$$

f (x) s'écrit alors :

$$f(x) = V_1(x_i) + (x - x_i) \cdot (V_2(x_i) + (x - x_i) (B + D(x - x_{i+1})))$$

Dans ce cas, le nombre d'opérations requis par l'algorithme (M. PETIT, 1971) est :

13 (N-1)	soustractions
38 (N-1)	multiplications
11 (N-1)	additions
4 (N-1)	divisions

Ce nombre est manifestement inférieur à celui nécessité par l'algorithme proposé pour calculer les vecteurs  $v(x_i^+)$ .

Néanmoins, le calcul de la fonctionspline en un point X quelconque de l'intervalle  $]x_i, x_{i+1}[$ , requiert, par rapport à l'algorithme du processus interpolateur, deux fois plus d'opérations(17 au lieu de 9).

Pour une valeur N donnée, le calcul de f(x) en 20 points (par exemple) de chaque intervalle  $]x_i, x_{i+1}[$  nécessite :

- \* 180 (N-1) opérations pour l'algorithme du processus interpolateur
- \* 340 (N-1) opérations pour la méthode de transport

Le tableau ci-dessous illustre la comparaison entre les deux algorithmes, pour 20 points intermédiaires et différentes valeurs de N.

N	ALGORITHME DE TRANSPORT			ALGORITHME PROCESSUS INTERPOLATEUR		
	$v(x_i)$	$f(x)$	Total	$v(x_i)$	$f(x)$	Total
10	594	3060	3654	1400	1620	3020
100	6534	33660	40194	17960	17820	35780
1000	65934	339660	405594	183560	179820	363380

Cette méthode semble bien être moins coûteuse et envisageable dans une utilisation graphique.

### II-7-3 COMPARAISON AVEC LA METHODE DES B-SPLINES

Compte-tenu du fait qu'il est pratiquement impossible de trouver la séquence  $\mathcal{C} = \{t_i\}_{i=1}^{n+k}$  définissant les B-splines, pour un réseau de points  $x_i$  donné, il est néanmoins possible de comparer les deux méthodes en supposant la séquence donnée par [cf. chap I] (II-7-7)

$$t_{i+k} = \frac{(x_{i+1} + \dots + x_{i+k-1})}{(k-1)}; \quad \forall i=1, \dots, n-k \quad (\text{II-7-7})$$

Le calcul de la fonction spline qui interpole le réseau nécessite [DE BOOR, 1978].

- Le calcul des B- splines qui utilise une équation de récurrence
- Le calcul de la valeur en tout point  $x_i$  des B-splines différentes de zéro
- La résolution d'un système d'équations linéaires d'ordre  $n$  par application de la méthode d'élimination de GAUSS (par exemple)

Sans entrer dans les détails de calcul, il semble que le nombre d'opérations requis soit aussi important que dans la méthode du processus interpolateur.

REM :

Lorsqu'on veut dessiner une forme géométrique qui requiert une grande représentation de données  $x_i$ , il est plus intéressant de connaître les coefficients du polynôme d'approximation. On utilise alors une représentation polynomiale pour morceaux qui, lorsque les points  $x_i$  sont équidistants, diminue considérablement le nombre d'opérations.

Compte-tenu de cette remarque, et de la difficulté de calculer la séquence  $\mathcal{C}$ , la méthode des B-splines demeure, problématique pour ce type de problème.

#### III-7-4 COMPARAISON AVEC LA METHODE DE BEZIER :

Mise à part sa justification faite au chapitre I comme étant une excellente méthode de conception, l'algorithme de BEZIER est néanmoins utilisée pour la reconstitution de formes géométriques.

Dans ce contexte, la courbe de BEZIER est assujettie à passer par les points  $x_i$  du réseau et son calcul est effectué à partir de la détermination des vecteurs  $\vec{a}_i$  qui constituent le polygone. ( cf Chap I)

Ces vecteurs sont solution d'un système d'équation dont l'ordre dépend du nombre de contraintes de points de passage. [BEZIER,1968] Lorsque ce nombre est élevé, le calcul des vecteurs  $\vec{a}_i$  se ramène à la résolution d'un système d'ordre élevé.

Une fois ces vecteurs calculés, il y a lieu d'effectuer le calcul des vecteurs  $\vec{b}_j$ .

Dans ce sens, le nombre d'opérations requis est le suivant :

\* Calcul des vecteurs  $\vec{B}_j$  :

$$\sum_{i=0}^{n-2} \{ (n-1)-i \} \text{ soustractions} \quad (\text{II-7-8})$$

où n est le nombre de sommets du polygone

\* Calcul des  $C_n^j$  .

$$\sum_{i=0}^n \{ (n-1)-i \} \text{ additions} \quad (\text{II-7-9})$$

\* Point courant  $\vec{P}_i(u)$

L'utilisation du schéma de HORNER requiert :

$$\left\{ \begin{array}{l} m \quad \text{multiplications} \\ \quad \quad \text{et} \\ n \quad \text{additions} \end{array} \right. \quad (\text{II-7-10})$$

où m est le nombre de côtés ( $m=n-1$ ) du polygone. Il y a lieu de compléter ce bilan en rajoutant m multiplications et n additions pour le calcul de la dérivée du point courant. (cas de contraintes de tangence aux points éventuels de raccordement.)

Si l'on s'en tient au décompte du nombre d'opérations nécessaires pour calculer les vecteurs  $\vec{a}_i$ , l'algorithme du processus interpolateur semble être plus rapide dans la mesure où il requiert uniquement la résolution d'un seul système d'ordre q.

D'autres critères vont dans ce sens et en particulier l'augmentation du degré de la courbe en fonction du nombre de points d'interpolation, et l'assignation arbitraire des valeurs du paramètre u, aux différentes contraintes de passage.



## II- 8 CONCLUSION

L'algorithme présenté dans ce chapitre permet de résoudre de manière aisée, le problème de calcul des fonctions splines d'interpolation initialement posé.

La formulation du problème dans l'espace d'état et l'application de la théorie de la commande optimale permettent une modélisation standard d'un système dynamique capable

- \* De générer les différentes classes de fonctions splines à partir de quelques transformations simples du logiciel, liées à la fonctionnelle à minimiser.
- \* De tenir compte des différentes conditions aux limites exigées, suivant la nature du problème.
- \* De parer, à partir d'une transformation simple, aux problèmes d'instabilité numérique qu'on pourrait détecter lorsque le nombre de données à interpoler devient élevé.
- \* D'être réalisé simplement sur un système hybride de visualisation.

Ce dernier point permettrait, pour les consoles graphiques issues du principe des oscilloscopes à mémoire (TEKTRONIX), de doter ces appareils d'opérateurs analogiques capables de réaliser les interpolations splines circulaires classiques. Un gain de temps appréciable pourrait ainsi être obtenu lors de la visualisation des lignes courbes. Par ailleurs, un tel système peut-être envisagé comme un outil pour la génération des surfaces dans le plan et dans l'espace. Ce point met en évidence la nécessité d'étudier les modifications introduites par l'algorithme pour la synthèse des courbes gauches.

Dans ce contexte, nous proposons d'introduire, de manière esquissée, l'étude des fonctions splines paramétrées. Celles-ci permettent aussi l'interpolation de données par des fonctions autres que celles (du type  $y=f(x)$ ) traitées jusqu'à présent. (cas des données engendrant une courbe fermée par exemple).

### CHAPITRE III

### CHAPITRE III : MODELISATION DES COURBES NON UNIMODALES.

#### APPLICATION A LA SYNTHÈSE DES COURBES GAUCHES.

---

Ce chapitre va nous permettre d'introduire la notion de "fonctions splines paramétrées", nécessaires pour la résolution de problèmes initialement définis sur des réseaux  $\{x_i\}_{i=1}^n$  non monotones.

Dans un premier volet, nous étudions la cas des courbes planes définies pour l'équation de la forme  $y=f(x)$ , à partir de l'application de l'algorithme du processus interpolateur au cas des fonctions splines d'ordre 2.

Dans un deuxième volet, nous envisageons l'application à la synthèse des courbes gauches susceptibles d'être générées par des fraiseuses travaillant en trois dimensions.

Le problème de l'instabilité numérique possible de l'algorithme est résolu suivant la technique de systèmes bouclés présentée au chapitre précédent.

III - 1. COURBES PLANES  $y=f(x)$

III - 1 - 1 : FONCTIONS SPLINES PARAMETREES

D'ORDRE 2 :

Nous nous intéressons dans ce paragraphe à la détermination des courbes définies sur des réseaux  $\{ (x_i, y_i) ; i=1, \dots, n \}$  quelconques pour lesquels les suites  $\{ x_i \}$  et  $\{ y_i \}$  ne sont plus nécessairement croissantes.

Ce type de courbes est fréquemment rencontré en construction mécanique, et constitue en général une ou plusieurs parties d'un objet mécanique.

Ces parties se raccordent en général à des portions linéaires et/ou circulaires, et sont telles qu'on ne peut plus les représenter globalement par des relations  $y = f(x)$  ou  $x = f(y)$ .

En conséquence, l'algorithme du processus interpolateur ne peut être envisagé directement pour ces types de courbes.

La figure ci-dessous illustre un exemple de ce

cas :

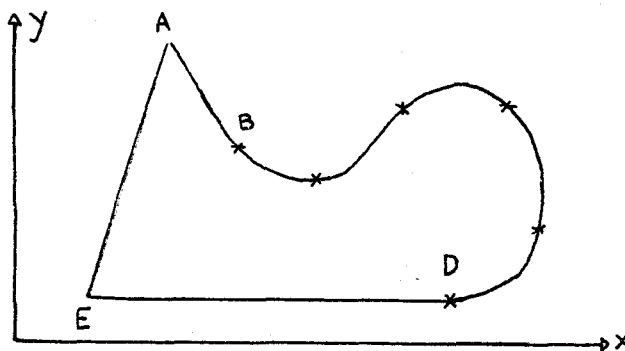


Fig III-1-1

Les parties AB, DE et EA étant linéaires, leur interpolation (calcul d'un grand nombre de points intermédiaires situés entre les extrémités) requiert uniquement la donnée des extrémités.

Par contre, la partie BD est une courbe qui ne correspond plus à un graphe de fonction, de sorte que l'on ne peut appliquer l'algorithme du processus interpolateur, et ce indépendamment du choix de l'échantillonnage.

Les courbes entrant dans cette catégorie sont notées courbes "non unimodales".

Pour résoudre ce type de problèmes, on peut soit utiliser une base de fonctions (B - splines), soit définir des fonctions splines paramétrées dont nous proposons l'étude.

### III - 1 - 2 : PROPRIETES :

Considérons le problème de l'interpolation par une courbe  $\mathcal{C}$  d'un réseau de points

$$\left\{ P_i ; i=1, \dots, n ; P_i = (x_i ; y_i) \right\} \quad (\text{III-1-1})$$

Un paramétrage monotone de la courbe  $\mathcal{C}$  permet l'ordonnement de ces points ; on le définit de la façon suivante :

$$[t_1, t_n] \subset \mathbb{R}$$

$$\mathcal{C} = \left\{ P(t) \in \mathbb{R}^2, t \in [t_1, t_n] ; P(t) = (x(t), y(t)) \right\}$$

$$\forall i=1, \dots, n-1 ; t_{i+1} > t_i \quad (\text{III-1-2})$$

$$\forall i=1, \dots, n ; P(t_i) = P_i$$

REMARQUE :

Dans le cas où l'une des suites  $\{x_i\}$  ou  $\{y_i\}$  est monotone, cette suite est un paramétrage. C'est en particulier le cas des fonctions traitées dans le chapitre précédent.

L'interpolation de l'ensemble donné de points est définie par les propriétés suivantes :

T1) L'arc de courbe est de classe  $C^2$  sur  $[t_1, t_n]$

T2)  $\forall i=1, \dots, n ; P(t_i) = P_i$

T3) L'arc de courbe minimise une fonctionnelle de forme :

$$J = \int_{t_1}^{t_n} [(x''(t))^2 + (y''(t))^2] \cdot dt \quad (\text{III-1-3})$$

Cette fonctionnelle étant additive, sa minimisation s'opère en minimisant indépendamment les quantités suivantes :

$$J_1 = \int_{t_1}^{t_n} (x''(t))^2 dt \quad \text{et} \quad J_2 = \int_{t_1}^{t_n} (y''(t))^2 dt \quad (\text{III-1-4})$$

On est ainsi amené à définir l'interpolation satisfaisant les propriétés énoncées par ses deux composantes  $x(t)$  et  $y(t)$ .

Chacune de ses composantes est caractérisée par les propriétés suivantes :

T11)  $x(t)$  (resp  $y(t)$ ) est de classe  $C^2 [t_1, t_n]$

T21)  $\forall i = 1, \dots, n$

$$\dot{x}(t_i) = x_i \quad (\text{resp } y(t_i) = y_i)$$

T31)  $x(t)$  (resp  $y(t)$ ) minimise la fonctionnelle de forme  $J_1$  (resp  $J_2$ ) ;  $\forall t \in [t_1, t_n]$

et peut être considérée comme la trajectoire issue du système dynamique de sortie  $x(t)$  (resp  $y(t)$ ) suivant :

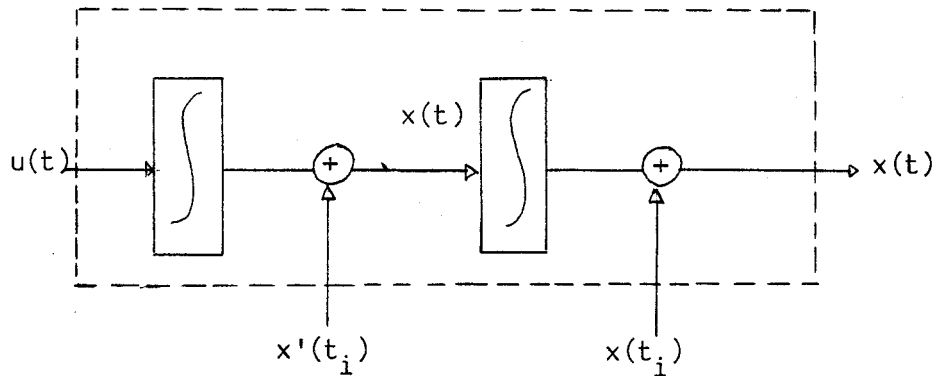


Fig III-I-2

La détermination de la courbe  $\mathcal{C}$  d'interpolation se ramène à celle des commandes  $u(t)$  de deux processus interpolateurs générant un point mobile  $(x(t), y(t))$  qui parcourt  $\mathcal{C}$ .

Chaque commande minimise la fonctionnelle quadriatique :

$$J(u(t)) = \int_{t_1}^{t_n} u^2(t) dt \quad (\text{III-1-5})$$

et peut être déterminée par la problématique du chapitre II.

Il nous reste à prononcer sur le choix du paramétrage. On sait que la fonction longueur d'une courbe constitue un paramétrage intrinsèque.

Si on considère le paramétrage (fréquemment utilisé dans la pratique) comme étant celui qui approxime la



longueur de l'arc de courbe définie par l'intégrale :

$$J = \int_{t_i}^{t_{i+1}} \sqrt{(x(t))^2 + (y(t))^2} . dt \quad (\text{III-1-6})$$

alors le réseau  $\left\{ t_i \right\}_{i=1}^n$  est défini par la récurrence suivante :

$$t_1 = 0 ; t_{i+1} = t_i + \sqrt{(x_{i+1}-x_i)^2 + (y_{i+1}-y_i)^2} \quad (\text{III-1-7})$$

$$\forall i = 1, \dots, n-1$$

où  $x_i$  et  $y_i$  ,  $\forall i=1, \dots, n$  sont les valeurs en  $t_i$  des fonctions splines  $x(t)$  et  $y(t)$  d'ordre 2.

La paramétrisation est importante et résoud en particulier le problème défini en introduction. Si  $t$  est assimilé à un temps, les conditions de transversalité fixent les composantes de la vitesse aux extrémités du réseau, et pas seulement les tangentes (cf THILLIEZ , 1967).

Nous proposons maintenant de généraliser l'algorithme à la synthèse des courbes dans l'espace.

III - 2 . APPLICATION A LA SYNTHÈSE DES COURBES GAUCHES :

La problématique est analogue à celle définie précédemment. Elle permet de représenter dans l'espace une courbe qui interpole des données de points à partir d'une représentation paramétrique.

Dans ce contexte, une courbe gauche est envisagée comme la trajectoire suivie par un point mobile, et issue d'un système dynamique.

Etant donné un réseau ordonné de points,

$$\left\{ P_i ; i = 1, \dots, n ; P_i = \{ x_i, y_i, z_i \} \right\} \quad (\text{III-2-1})$$

le choix d'un paramétrage monotone défini de la façon suivante :

$$\begin{aligned} & \exists t_1, t_n \in \mathbb{R} \\ \mathcal{C} = & \left\{ P(t) \in \mathbb{R}^3, t \in [t_1, t_n] ; P(t) = \{ x(t), y(t), z(t) \} \right\} \\ & \forall i=1, \dots, n-1 ; t_{i+1} > t_i \\ & \forall i=1, \dots, n ; P(t_i) = P_i \end{aligned} \quad (\text{III-2-2})$$

permet de définir l'interpolation de l'ensemble donné des points, par l'ensemble des propriétés suivantes :

T<sub>1</sub>' ) L'arc de courbe est de classe  $C^{2q-2}$  sur  $[t_1, t_n]$

T<sub>2</sub>' )  $\forall i=1, \dots, n ; P(t_i) = P_i$

T<sub>3</sub>' ) L'arc de courbe minimise une fonctionnelle de forme :

$$J = \int_{t_1}^{t_n} \left\{ [L(x(t))]^2 + [L(y(t))]^2 + [L(z(t))]^2 \right\} dt$$

où  $L(\cdot)$  est un opérateur différentiel linéaire d'ordre  $q$  :

$$L = \left( \frac{d}{dt} \right)^q + \sum_{i=1}^q a_i \left( \frac{d}{dt} \right)^{i-1}$$

Cette définition comprend , en particulier ,  
celle des fonctions splines polynomiales d'ordre 2 dans l'espace  
pour lesquelles l'opérateur  $L$  est l'opérateur dérivé seconde.

$J$  étant additive, on minimise indépendamment  
la fonctionnelle suivante :

$$J_1 = \int_{t_1}^{t_n} (L(f(t)))^2 dt \quad \text{pour } f(t) = x(t), y(t) \text{ ou } z(t)$$

(III-2-3)

On est ainsi amené à définir l'interpolation satisfaisant aux propriétés énoncées  $T'_1$ ,  $T'_2$ ,  $T'_3$  par ses trois composantes  $x(t)$ ,  $y(t)$  et  $z(t)$ .

La deuxième composante par exemple est caractérisée par les trois propriétés suivantes :

$T'_{11})$   $y(t)$  est de classe  $C^{2q-2}$  sur  $[t_1, t_n]$

$T'_{21})$   $\forall i=1, \dots, n$  ;  $y(t_i) = y_i$

$T'_{31})$   $y(t)$  ;  $t \in [t_1, t_n]$  minimise la fonctionnelle

$$J(y(t)) = \int_{t_1}^{t_n} (L(y(t)))^2 dt$$

Le schéma bloc correspondant est le suivant :

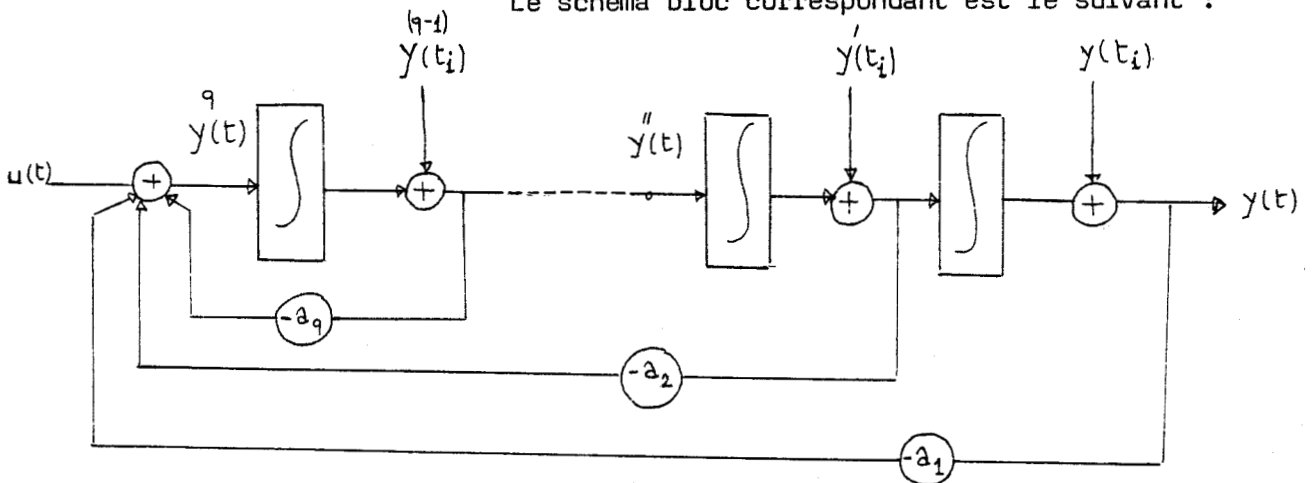


Fig III-2-I

où  $u(t) = L(y(t))$ .

La détermination de la courbe  $\mathcal{C}$  se ramène à celle des commandes  $u(t)$  des trois processus interpolateurs générant le point mobile  $(x(t), y(t), z(t))$  qui parcourt  $\mathcal{C}$ .

Chaque commande minimise la fonctionnelle quadratique :

$$J(u(t)) = \int_{t_1}^{t_n} u^2(t) \cdot dt \quad (\text{III-2-4})$$

et sa détermination utilise l'algorithme du processus interpolateur (cf Chapitre II).

Le réseau  $\{t_i\}_n$  est défini à partir de la donnée de :  $\{x_i, y_i, z_i\}_{i=1}$ , par la relation :

$$t_1 = 0 ; t_{i+1} = t_i + \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2} \quad (\text{III-2-5})$$

$$\forall i = 1, \dots, n-1$$

III - 3 . CONCLUSION :

La définition de courbes interpolant un réseau de points, à partir d'une loi paramétrique, trouve toute son importance dans plusieurs cas.

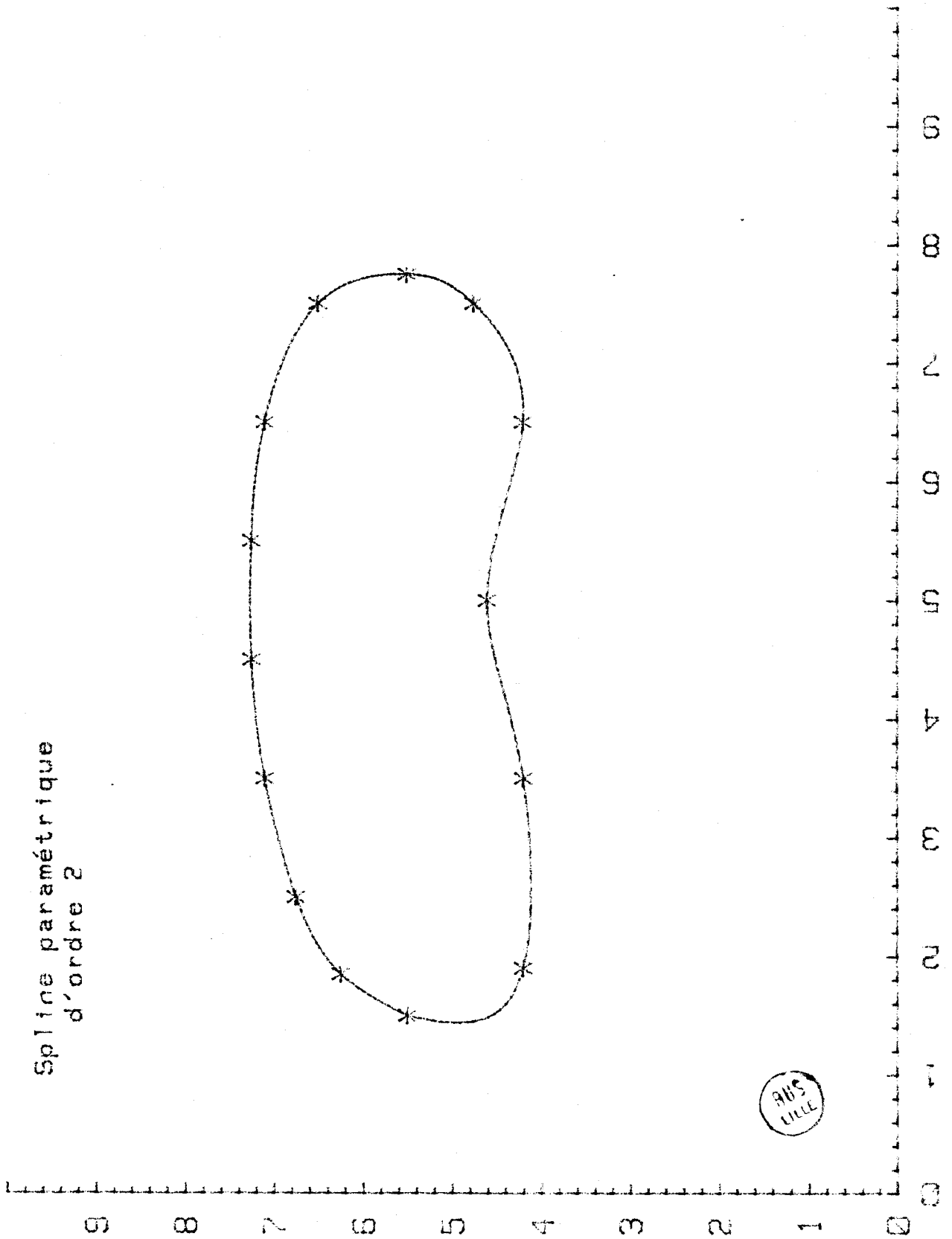
Elle permet tout d'abord d'apporter une solution aisée au problème de génération des courbes qu'on ne peut décrire par des applications de type  $y = f(x)$ .

Par ailleurs, une telle représentation permet à la machine de fonctionner avec plusieurs axes (en particulier pour la manipulation de courbes et surfaces gauches) suivant ses possibilités.

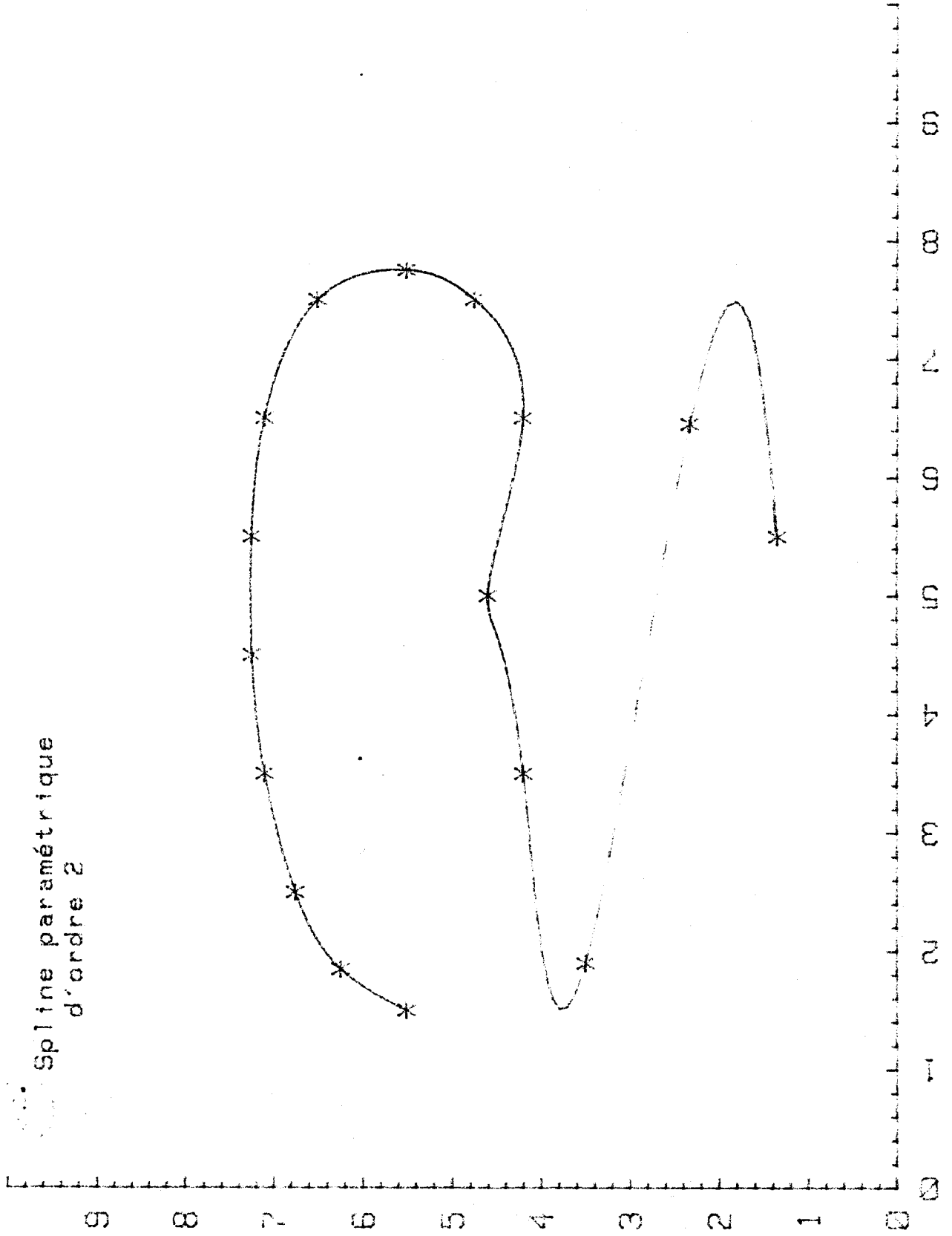
La méthode exposée dans ce chapitre requiert deux fois l'algorithme du processus interpolateur en boucle ouverte pour les courbes planes paramétrées et trois fois cet algorithme pour les courbes gauches.

Elle est mise à profit pour la génération des surfaces gauches définies sur des réseaux rectangulaires, au chapitre IV.

Spline paramétrique  
d'ordre 2

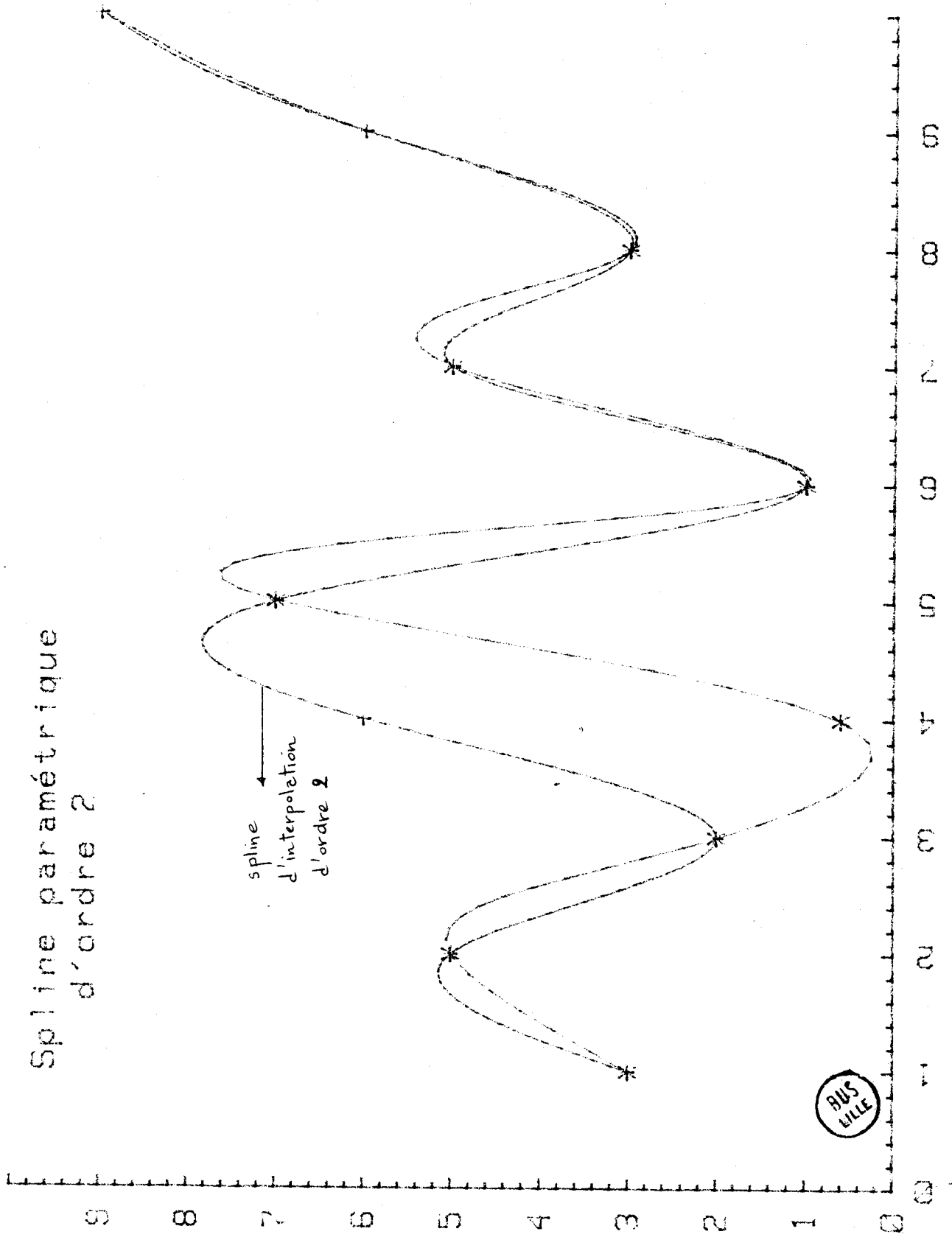


Spline paramétrique  
d'ordre 2





# Spline paramétrique d'ordre 2



## CHAPITRE IV

Chapitre IV : MODELISATION DES SURFACES

---

IV-1 INTRODUCTION :

Les fonctions splines bidimensionnelles ont plus particulièrement été étudiées par ATTEIA [1968], THOMANN [1970], et DEBOOR [1978], pour la représentation de surfaces définies sur des réseaux rectangulaires et/ou circulaires.

Les premiers ont introduit la notion de splines de type plaque mince [PAIHUA, 1978], et DE BOOR [1978] définit une surface comme le produit tensoriel de fonctions splines unidimensionnelles.

Dans ce contexte, le problème de détermination d'une fonction spline bidimensionnelle définie sur un réseau rectangulaire peut s'énoncer de la manière suivante :

Parmi les fonctions  $f(x,y)$  qui prennent les valeurs  $Z_{ij}$  aux noeuds d'un réseau rectangulaire bidimensionnelle  $\{x_i, y_j; i=1, \dots, n; j=1, \dots, m\}$  déterminer celle qui satisfait les propriétés annoncées au chapitre I, et qui minimise la fonctionnelle suivante

$$J(f) = \int_{x_1}^{x_n} \int_{y_1}^{y_m} \left\{ \left( \frac{\partial^2 f}{\partial x^2} \right)^2 + 2 \left( \frac{\partial^2 f}{\partial x \partial y} \right)^2 + \left( \frac{\partial^2 f}{\partial y^2} \right)^2 \right\} dx dy$$

D'autres types de fonctionnelles peuvent être utilisées. C'est ainsi que pour des splines de type plaque mince en dimension deux, on cherche à minimiser :

$$J(f) = \int_{x_1}^{x_n} \int_{y_1}^{y_m} \left| \frac{\partial^2 f}{\partial x^2} + 2 \frac{\partial^2 f}{\partial x \partial y} + \frac{\partial^2 f}{\partial y^2} \right|^2 dx \cdot dy$$

L'utilisation de l'une ou l'autre des méthodes conduit à la résolution algébrique d'un système d'équations linéaire.

A cette fin, différentes techniques ont pu être proposées pour minimiser le coût des opérations ainsi que la capacité mémoire utilisée. [PAIHUA, 1978].

En utilisant la notion de produit tensionnel de fonctions splines sous-tension d'ordre 2, conjointement avec les résultats établis au chapitre précédent, on montre, dans ce chapitre, que le problème initialement posé, se ramène à l'étude respective de chacune des courbes constituant la surface.

Dans ce sens, la surface qui interpole le réseau rectangulaire est définie à partir de courbes génératrices et directrices, interprétées comme les trajectoires issues de systèmes dynamiques (cf. CHAP. II)

Cette technique permet entre autre, de restreindre le volume de calcul initialement prévu, et ne nécessite qu'une faible capacité mémoire.

#### IV- 2 ETUDE DES SURFACES DEFINIES SUR DES RESEAUX RECTANGULAIRES:

Soit D le réseau défini par :

$$D = \left\{ (x_i, y_j); \forall_i = 1, \dots, n; \forall_j = 1, \dots, m; \text{ tel que :} \right. \\ \left. x_i < x_{i+1}; \text{ et } y_j < y_{j+1} \right\} \quad (\text{IV-2-1})$$

et soient  $Z(x_i, y_j) = Z_{ij}$  la valeur prise par la fonction spline sous-tension bidimensionnelle aux différents noeuds  $(x_i, y_j)$  de D.

Le problème consiste à déterminer la surface de classe  $C^{2,2}$   $[x_1, x_n] \times [y_1, y_m]$  qui interpole le réseau D :

Nous proposons de définir sa forme à partir de celles des courbes génératrices

$$- Z(x, y_0 = \text{cste} \in [y_1, y_m]) \quad (\text{IV-2-2})$$

et des courbes directrices définies par

$$- Z(x_0 = \text{cste} \in [x_1, x_n], y) \quad (\text{IV-2-3})$$

Cette approche se justifie en particulier par la propriété de douceur de la surface ainsi obtenue.

Chaque courbe est ainsi spécifiée comme une interpolation définie sur un réseau unidimensionnel. Elle peut ainsi être élaborée par un processus interpolateur suivant la méthode exposée au chapitre II.

A ce niveau, on considère qu'on a construit, suivant la méthodologie proposée, un sous programme permettant d'élaborer l'interpolation à partir des données d'un réseau.

PAR EXEMPLE

SUBROUTINE SPLST 2 (N,X,Y, TI, YPI, TF,YPE,  $\sigma$ , V)

ARGUMENTS D'ENTREE :

N : Nombre de points à interpoler

$\begin{cases} X(N) \\ Y(M) \end{cases}$  ; définit le réseau  $x_i, y_j, x_i = X(I), y_j = Y(J)$

TI : argument logique vrai si la tangente initiale  $y'(x_1)$  est donnée.  
(dans ce cas elle est donnée par YPI)

TF : Le couple TF, YPF fonctionne comme ( TI, YPI) dans le cas de la tangente au point final.

$\sigma$  : paramètre de tension

ARGUMENTS DE SORTIE:

V(4, N-1): les paramètres  $V(J,I)$  ;  $J = 1,2,3,4$ , définissent l'interpolation sur  $[x(I), x(I+1)]$

La génération d'une surface se déroule alors en deux étapes qui sont :

- 1) Interpolation des n courbes directrices définies par les réseaux unidimensionnels

$$R_i = \{ y_j, Z_{ij}; \forall j = 1, \dots, m; \forall i = 1, \dots, n \}$$

- 2) A partir de la connaissance des n directrices  $Z(x_i, y_0 = \text{cste})$  on peut déterminer toute génératrice  $Z(x, y_0 = \text{cste})$

Le problème est ainsi résolu conformément au schéma suivant (Fig

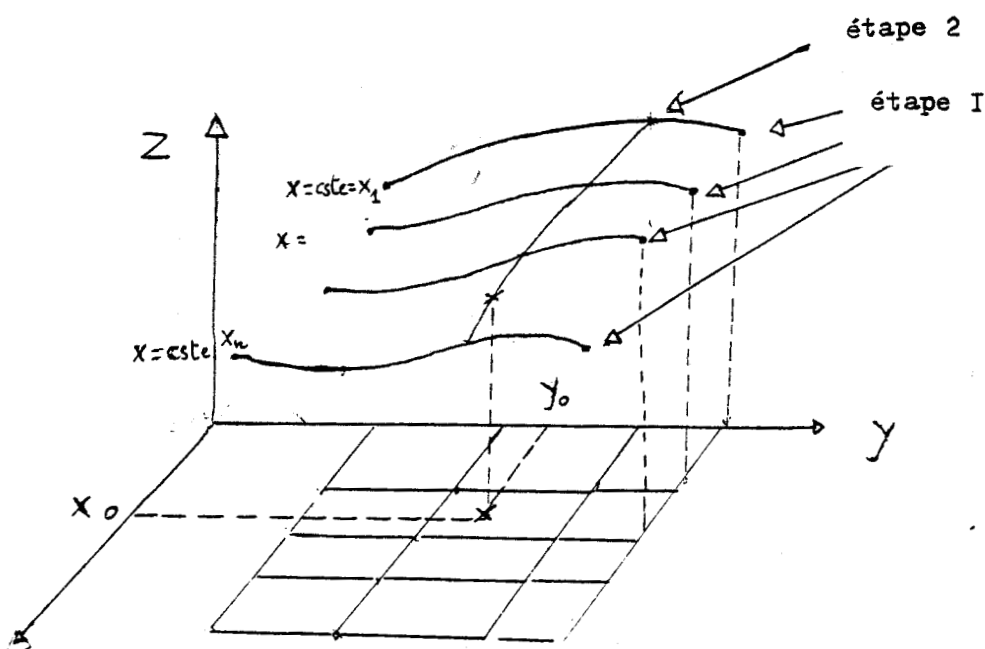


Fig-IV-2-1

Nous proposons de montrer qu'il est possible de restreindre le nombre d'opérations nécessitées par l'étape 2, en le rendant indépendant du nombre de génératrices que l'on souhaite calculer/tracer.

IV-2-1 : ANALYSE DE L'ETAPE 1 :

Si  $y$  est une valeur appartenant à l'intervalle  $[y_1, y_n]$ , à la suite de l'étape 1 on connaît la valeur  $Z(x_i, y)$ , pour tout  $i = 1, \dots, n$ . On est ainsi amené à résoudre  $n$  fois une interpolation de  $m$  points par l'algorithme exposé, au chapitre II.

IV-2-2 ANALYSE DE L'ETAPE 2 :

L'interpolation du réseau  $\{x_i, y ; z(x_i, y) ; \forall i = 1, \dots, n$   
 $y \in [y_1, y_m]\}$  est effectuée par un processus interpolateur dont le bloc de  
 calcul définit les composantes du vecteur  $v(x_1^+, y = \text{cste})$ , pour toute valeur  
 $y = \text{cste}$  fixée ;  $y \in [y_1, y_m]$ .

En définissant les composantes du vecteur  $v(x_1, y)$  comme suit :

$$V^T(x_1, y) = \left[ v(x_1, y), \frac{\partial v(x_1, y)}{\partial x}, \frac{\partial^2 v(x_1, y)}{\partial x^2}, \frac{\partial^3 v(x_1, y)}{\partial x^3} \right]$$

avec  $v(x_1, y) = z(x_1 = \text{cste}, y)$  (IV-2.4)

alors deux des composantes sont données initialement.

Ce sont  $v(x_1, y)$  et  $\frac{\partial v(x_1, y)}{\partial x} = 0$  si on suppose la dérivée  
 première non fixé au point initial, les deux autres composantes sont issues  
 de l'équation de récurrence [II-3-3, chap II]

$$V(x_n^-, y = \text{cste}) = P \cdot v(x_1^+, y = \text{cste}) + q$$

avec  $P \in \mathbb{R}^{4 \times 4}$  et  $q \in \mathbb{R}^4$  (IV-2.5)

Le réseau étant rectangulaire, la matrice P est indépendante de y,  
 de sorte que seul q en dépend, par l'intermédiaire du vecteur  $y_1 (i+2)$   
 [II-3-20, chap II]

Pour déterminer le vecteur initial  $v(x_1^+, y)$  il est nécessaire  
 de définir une deuxième relation. Celle-ci est fournie par la première  
 ligne de l'équation de progression au premier pas (i=1) :

$$V(x_2^-, y = \text{cste}) = M(x_2 - x_1) V(x_1^+, y = \text{cste})$$
 (IV-2.7)

soit  $y = (y = \text{cste})$ , il vient :

$$M(1,2) \frac{\partial v}{\partial x}(x_1^+, y) + M(1,4) \frac{\partial^3 v(x_1^+, y)}{\partial x^3} = v(x_2^+, y) - M(1,1) V(x_1^+, y)$$

où  $M(1,1) = 1$  et  $M(1,2), M(1,4)$  ne dépendent que des  $x_i$ .

Cette relation conjointement à celle issue de la 3 ème ligne de l'équation,

$$V(x_{\bar{n}}, y=cste) = P v(x_1^+, y) + q$$

aboutit au système matriciel suivant, quelque soit  $y \in [y_1, y_m]$ .

$$M(1,2) = C_{11} ; M(1,4) = C_{12} ; M_1(3,2) = C_{21}, M_1(3,4) = C_{22}$$

$$d_1 = M(1,1) \text{ et } d_2 = M_1(3,1)$$

$$\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial x} v(x_1^+, y) \\ \frac{\partial^3}{\partial x^3} v(x_1^+, y) \end{bmatrix} = \begin{bmatrix} v(x_2^+, y) - d_1 v(x_1^+, y) \\ -d_2 v(x_1^+, y) - N_1(3,1) \end{bmatrix} \quad (\text{IV-2-8})$$

valable quel que soit  $y = cste \in [y_1, y_m]$

La matrice C d'éléments  $[c_{ij}]$  est indépendante de y, de sorte que son calcul n'est effectué qu'une seule fois. Il en est de même pour les coefficients  $d_1$  et  $d_2$  intervenant dans le deuxième membre. Cette remarquable simplicité est inhérente au caractère rectangulaire du réseau de points à interpoler.

La détermination des trajectoires définies sur les réseaux  $\{y_j, z_{ij}\}$  est en tout point analogue à celle exposée au chapitre deux pour les fonctions splines sous-tension d'ordre, de sorte que nous convenons d'exposer uniquement l'organigramme relatif au calcul des trajectoires génératrices.

De même, la réalisation hybride d'un processeur capable de générer les surfaces décrites précédemment à partir d'un processeur numérique chargé des opérations algébriques et d'un processeur spécialisé dans les opérations d'intégration (en technologie analogique par exemple) est une extension naturelle de la méthode proposée au chapitre II.



REMARQUE :

Cette méthode se généralise aisément au cas des surfaces fermées, et fait appel aux résultats exposés au précédent chapitre. Dans le cas des surfaces non unimodales, il convient d'utiliser une représentation paramétrique [c f. chap III], et d'appliquer la présente méthode.

ORGANIGRAMME DE CALCUL D'UNE SURFACE DEFINIE SUR LE RESEAU RECTANGULAIRE

$X(I), Y(J), Z(I, J) : I = \dots, NN ; J = \dots, MM$

305  
11/12

Entrer les données  
 $X(NN), Y(MM),$   
 $Z(NN, MM), V_{aly}(RR)$

POUR

$K = 1$   
à  
 $NN$

- Calcul des trajectoires  
DIRECTRICES sur les ré-  
seaux :

$$R_K = \{ Y(I), Z(K, I), \\ I = 1, \dots, MM-2 \}$$

- Sortie des composantes  
 $V(K, J, I) ; J = 1, \dots, 4$   
 $I = 1, \dots, MM-$

SS-PROGRAMME ' SPLST 2 '

ETAPE I

- Calcul des valeurs des  
sommets des génératrices  
pour toute valeur de :

$Y = cste$  ou bien  $V_{aly}(I) ;$

$I = 1, \dots, RR$

- Sortie des valeurs dans  
le tableau :

$F(RR, NN)$

POUR

$L = 1$   
à  
 $RR$

- Calcul des trajectoires  
GENRATRICES définies sur  
réseaux :

$$R_j = \{ X(I), F(L, I) ; I = 1, \dots, NN-2 \\ L = 1, \dots, RR \}$$

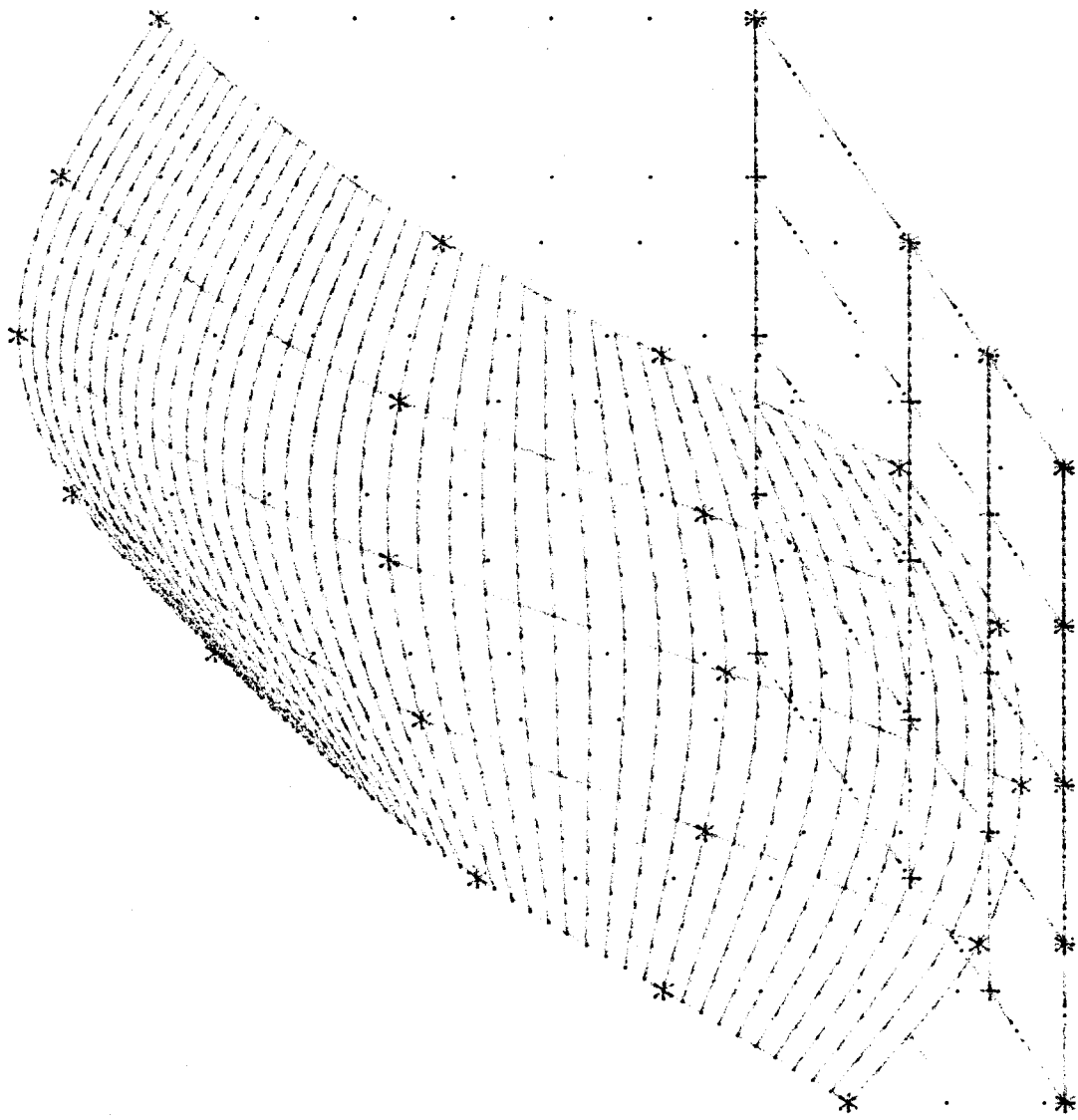
- Sortie des composantes :

$$G(L, J, K) ; L = 1, \dots, RR \\ J = 1, \dots, 4 \\ K = 1, \dots, NN-$$

ETAPE 2

Fig-IV-2-2

sig=1



CONCLUSION

Au cours de cette étude, nous avons montré le lien étroit existant entre les théories de l'interpolation et de la commande optimale des systèmes dynamiques linéaires.

Dans ce sens, les résultats théoriques obtenus sont analogues à ceux définis à l'origine, par l'interpolation.

D'un point de vue pratique, l'algorithme résultant de cette étude permet le maniement aisé et rapide des différentes catégories des fonctions Splines existant, par une simple transformation du logiciel correspondant.

La possibilité de raccordement de courbes induite par l'algorithme, permet de l'incorporer à un logiciel de C.A.O., de manière à décrire, plus particulièrement, les contours de pièces mécaniques.

L'utilisation de la technique de processus interpolateur en boucle fermée, apporte une solution efficace au problème d'instabilité numérique pouvant résulter de la manipulation d'un grand nombre de données.

ANNEXE I

POSITION DU PROBLEME DE LA COMMANDE D'UN SYSTEME LINEAIRE  
DYNAMIQUE

Soit  $[x_1, x_n]$  un intervalle fermé  $\subset \mathbb{R}$ , et notons pour toute valeur  $x \in [x_1, x_n]$ :

$u$  le vecteur de commande à  $m$  composante,  $u \in \mathbb{R}^m$

$z$  le vecteur de sortie à 1 composante,  $z \in \mathbb{R}^1$

$\bar{z}$  le vecteur d'état à  $q$  composantes,  $\bar{z} \in \mathbb{R}^q$

Le modèle du processus sera définie par une équation vectorielle d'évolution.

$$\frac{d}{dx} (\bar{z}(x)) = f [\bar{z}(x), u(x), x]; \quad x \in [x_1, x_n] \quad (I-1)$$

et par une équation vectorielle algébrique de sortie

$$z(x) = C \cdot [\bar{z}(x), x] \quad (I-2)$$

De manière à assurer l'unicité de la solution, de l'équation vectorielle (I-1), pour des conditions initiales déterminés  $\bar{z}(x_1)$  et une loi de commande  $u(x)$  donnée, nous supposons que les composantes de  $f$  sont continues en  $\bar{z}$ ,  $u$  et continument dérivables en  $\bar{z}$ . En d'autres termes

$$f^i (\bar{z}(x), u(x), x), \quad \frac{\partial f^i}{\partial \bar{z}} (\bar{z}(x), u(x)) \quad \text{et} \quad \frac{\partial f^i}{\partial x} (\bar{z}(x), u(x))$$

$$\forall \quad i = 1, \dots, q$$

(I-3)

sont définies et continues sur  $\mathbb{R}^q \times U$  ( $U$  désigne le domaine de commandes  $u(x) \in U$ ).

Le processus évolue entre les abscisses  $x_1$  et  $x_n$  auxquelles sont affectées des valeurs  $\bar{z}(x_1)$  et  $\bar{z}(x_n)$  du vecteur d'état. Si on définit une loi de commande ( $i.e$  est fait le choix d'une commande admissible  $u=u(x)$ , alors (quelles que soient les conditions initiales  $\bar{z}(x_1)$ ), la loi

$$\dot{\bar{z}} = \bar{z}(x) \quad (I-4)$$

régissant l'évolution du système décrit par l'équation (I-1) est définie de manière univoque.

On associe au problème précédent constitué d'un système (I-1) et de conditions initiales et terminales spécifiant  $\bar{z}(x_1)$  et  $\bar{z}(x_n)$ , la fonctionnelle  $J$  suivante (I-5)

$$J = \int_{x_1}^{x_n} r(\bar{z}, u, x) dx \quad (I-5)$$

où  $r(\cdot, \cdot, \cdot)$  est une fonction scalaire continue par rapport à ses arguments et continument dérivable par rapport à  $\bar{z}$  et  $x$ . Alors le problème de la commande optimale du processus décrit par (I-1) consiste à trouver parmi les commandes admissibles  $u = u(x)$  transférant le point représentatif du système de  $\bar{z}(x_1)$  à  $\bar{z}(x_n)$  celle qui minimise la fonctionnelle (I-5) (Fig-1)

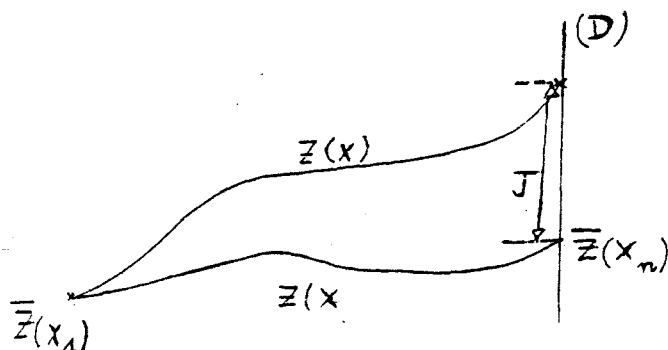


Fig-1 : Transfert de l'état  $\bar{z}(x_1)$  à l'état  $\bar{z}(x_n)$  en minimisant  $J$

FORMULATION DE LA CLASSE DE PROBLEMES ETUDIES

Le système dynamique étudié est caractérisé par les équations (I-1) et (I-2) avec  $m=1=1$ .

Le vecteur d'état  $\bar{z}(x)$  est continue et continument dérivable jusqu'à l'ordre  $(q-1)$  par rapport à  $x$ .

$$\frac{d\bar{z}(x)}{dx}, \quad \frac{d^{(j)}\bar{z}(x)}{dx^{(j)}} \quad \text{continues pour } x \in [x_1, x_n], \quad j=1, \dots, q-1$$

$x_1, x_n, \bar{z}(x_1), \bar{z}(x_n)$  sont des valeurs fixées à priori.  $\bar{z}(x_1)$  et  $\bar{z}(x_n)$  obéissent à des conditions aux limites telles qu'à l'abscisse  $x_1$  (resp  $x_n$ ), l'état  $\bar{z}(x_1)$  (resp  $\bar{z}(x_n)$ ) est astreint à se déplacer sur une variété  $V_1$  (resp  $V_n$ ) de l'espace d'état  $\mathbb{R}^q$ , de sorte que le problème est à états extrêmes partiellement contraints (Fig-2)

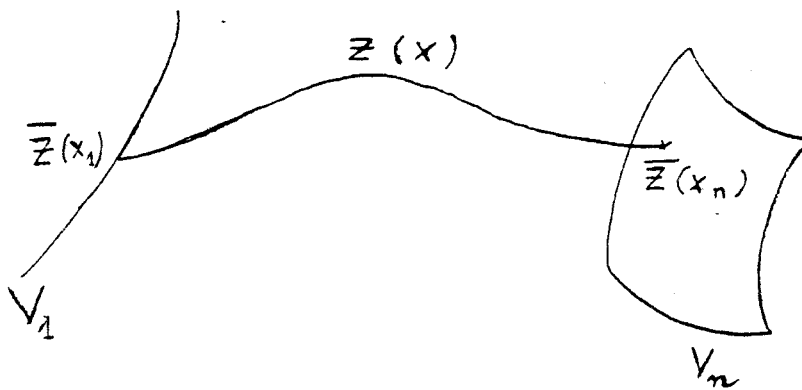


Fig-2 Problème à extrémités libres

Par ailleurs si  $\bar{z}(x_1), \dots, \bar{z}(x_k)$  sont des points de l'espace de phase et s'il existe une commande  $u_i(x)$  transférant le point représentatif de  $\bar{z}(x_i)$  en  $\bar{z}(x_{i+1})$  et fait prendre à la fonctionnelle (I-5) la valeur  $J_i, \forall i=1, \dots, k-1$ , il existe alors une commande  $u(x)$  qui transfère le point représentatif de  $\bar{z}(x_1)$  en  $\bar{z}(x_k)$  et fait prendre à la fonctionnelle (I-5) la valeur  $J_1 + J_2 + \dots + J_k$ .



La résolution de ce problème utilise le principe du maximum (ou minimum) de PONTRYAGUINE, qui est une extension des résultats du calcul des variations, et s'appuie sur la notion d'HAMILTONIEN.

### PRINCIPE DU MAXIMUM DE PONTRYAGUINE

Soit le processus dynamique d'équation d'état :

$$\frac{d \bar{z}(x)}{dx} = f(\bar{z}(x), u(x), x)$$

et  $u(x)$  une loi de commande admissible transférant le système d'un état initial  $(\bar{z}(x_1), x_1)$  à un état final  $(\bar{z}(x_n), x_n)$

Pour que  $u(x)$  soit la loi de commande relative au critère.

$$J = \int_{x_1}^{x_n} r(\bar{z}(x), v(x), x) dx$$

et  $\bar{z}(x)$  la trajectoire optimale correspondante, il est nécessaire qu'existe une fonction vectorielle  $\psi(x)$

$$\psi^T(x) = [\psi_1(x), \dots, \psi_n(x)]$$

continue et non nulle correspondant aux fonctions  $u(x)$  et  $\bar{z}(x)$  tel que :

$\psi(x)$  et  $\bar{z}(x)$  soient solution du système d'équation différentielles suivant :

$$\frac{d \bar{z}}{dx} = \frac{\partial H(\bar{z}(x), \psi(x), u(x), x)}{\partial \psi}$$

$$\frac{d \psi}{dx} = - \frac{\partial H(\bar{z}(x), \psi(x), u(x), x)}{\partial \bar{z}}$$

$$\text{où } H(\bar{z}, \psi, u, x) = -r(\bar{z}, u, x) + \psi^T \cdot f(\bar{z}, u, x)$$

est une fonction scalaire notée HAMILTONIEN

b/ La fonction H atteint son maximum (ou minimum) pour la commande  $u=u(x)$ .

$$\max_u H(\bar{z}(x), \psi(x), u, x) = H(\bar{z}(x), \psi(x), u, x)$$

$$u \in \mathcal{U}$$

ou de manière équivalente :

$$H(\bar{z}(x), \psi(x), u(x), x) \leq H(\bar{z}(x), \psi(x), u, x)$$

pour tout  $u \in \mathcal{U}$

c/  $\psi(x)$  satisfait les conditions de transversalité suivantes :

$$\begin{cases} \psi(x_1) \text{ orthogonal à } V_1 \\ \psi(x_n) \text{ orthogonal à } V_n \end{cases}$$

Ces relations (a), (b) et (c) peuvent être obtenues à partir de la variation de la fonctionnelle

$$J(u(x)) = \int_{x_1}^{x_n} r(\bar{z}(x), u(x)) dx \quad (I-13)$$

en supposant connues les fonctions  $u(x)$  et  $\bar{z}(x)$  minimisantes et en effectuant des variations infiniment petites  $\delta u(x)$  et  $\delta \bar{z}(x)$ .

En associant aux contraintes (I-7) des multiplicateurs de LAGRANGE qui sont des fonctions de  $x$ , il vient

$$J_1(u) = \int_{x_1}^{x_n} \left\{ r(\bar{z}(x), u(x)) + \psi^T(x) \cdot [f(\bar{z}(x), u(x)) - \dot{\bar{z}}] \right\} dx \quad (I-14)$$

En posant :

$$H(\bar{z}(x), u(x), \psi(x)) = r(\bar{z}(x), u(x)) + \psi^T(x) \cdot f(\bar{z}(x), u(x)) \quad (I-15)$$

Il vient :

$$J_1(u) = \int_{x_1}^{x_n} (H - \psi^T \cdot \bar{z}) dx \quad (I-16)$$

La variation de cette fonctionnelle due à des variations infiniment petites  $\delta u$  et  $\delta \bar{z}$ , s'écrit :

$$J_1(u) = \int_{x_1}^{x_n} \left\{ H_{\bar{z}}^T \cdot \delta \bar{z}(x) + H_u(x) \cdot \delta u(x) - \psi^T(x) \cdot \delta \bar{z}(x) \right\} dx \quad (I-17)$$

$$\text{or } \int_{x_1}^{x_n} -\psi^T(x) \cdot \delta \bar{z}(x) dx = \left[ -\psi^T(x) \cdot \delta \bar{z}(x) \right]_{x_1}^{x_n} + \int_{x_1}^{x_n} \dot{\psi}^T(x) \cdot \delta \bar{z}(x) dx$$

$$\text{soit } \delta J_1(u) = \psi^T(x_1) \cdot \delta \bar{z}(x_1) - \psi^T(x_n) \cdot \delta \bar{z}(x_n) + \int_{x_1}^{x_n} (H_{\bar{z}}^T + \dot{\psi}^T(x)) \cdot \delta \bar{z}(x) + H_u(x) \cdot \delta u(x) \cdot dx$$

La condition de minimum de  $J_1$ , quelles que soient  $\delta u$  et  $\delta \bar{z}$  est donnée par :

$$J_1(u^*(u)) = 0$$

Soit :

$$\boxed{\begin{aligned} \frac{d}{dx} [\psi(x)] &= -H_{\bar{z}}(x) \\ H_u(x) &= 0 \\ [\psi^T(x) \cdot \delta \bar{z}(x)] &= 0 \\ & \quad x=x_1 \\ & \quad x=x_n \end{aligned}}$$

L'application de ces résultats au problème spécifique des fonctions splines d'interpolation, dans tout intervalle  $]x_i, x_{i+1}[$  ne contenant pas de contraintes se justifie dans la mesure où la fonctionnelle à minimiser est additive.

## B I B L I O G R A H I E

-----

- M. ATTEIA : " Etude de certains noyaux et théorie des fonctions splines en Analyse numérique ". Publications de l'institut de mathématiques appliquées de GRENOBLE, 1966
- T. BENNANI : " Générateur numérique de trajectoires par interpolation polynomiale "  
Thèse de 3ème cycle , 1975 , ORSAY
- P. BEZIER : "Procédé de définition numérique des courbes et surfaces non mathématiques - Système UNISURF " ,1968
- R. BOUDAREL , J. DELMAS , P. GUICHET : " Commande optimale des processus ", TOME 2 , DUNOD - PARIS - 1968
- C. DE BOOR : " A practical guide to Splines ". Applied Mathematical Sciences; 27  
Springer-Verlag NEW YORK HEIDELBERG BERLIN , 1978
- ENCYCLOPEDIE UNIVERSALIS : "Approximation des fonctions"  
Volume II , p.989,990 .Analyse numérique
- A. FELDBAUM : " Principes théoriques des systèmes asservis optimaux"  
Editions MIR . MOSCOU , 1973
- F.R. GANTMACHER : "Théorie des matrices " , TOME I, DUNOD , PARIS, 1966
- J.C. GILLES, M. CLIQUE : " La représentation d'état pour l'étude des systèmes dynamiques ". Tome I, variables et équations d'état. Editions EYROLLES , 1975
- J.C. GILLES, M. CLIQUE : " La représentation d'état pour l'étude des systèmes dynamiques ". Tome II, commande et observation. Editions EYROLLES , 1975

R. HAJ NASSAR, D. MEIZEL, P. BIELEC : "Détermination des fonctions Spline d'ajustement par la théorie de la commande optimale des processus linéaires "  
Proceedings of the First I.A.S.T.E.D. Symposium on Applied Informatics . LILLE, 1983

R. HAJ NASSAR, D. MEIZEL, P. BIELEC : " Une procédure rapide de calcul et tracé des fonctions splines d'ordre 2 ", Actes du congrès IASTED-ICD 82. TUNIS. Septembre 1982.

R. HAJ NASSAR, D. MEIZEL, P. BIELEC : " Application d'un outil de simulation à la conception des surfaces gauches: Notion de PROCESSUS INTERPOLATEUR ". Congrès IMACS -NANTES-1983

P. de LARMINAT & Y. THOMAS : "Automatique des systèmes linéaires " Commande (Tome 3) , 1977  
Collection Flammarion-Sciences

P. J. LAURENT : " Approximation et Optimisation "; Collection Enseignement des Sciences.  
I3, HERMANN , PARIS, 1972;

G. MARCHOUK : " Méthode de calcul numérique ".  
Editions MIR - MOSCOU , 1980

D. MEIZEL, R. HAJ NASSAR, P. BIELEC : " Une procédure rapide d'interpolation de données ponctuelles.  
Actes du congrès IASTED-ICD 82, TUNIS-Septembre 1982

P. NASLIN : " Théorie de la commande et conduite optimale " .  
Collection DUNOD. 1968

L. PAIHUA MONTES : " Quelques méthodes numériques pour le calcul de fonctions splines à une et plusieurs variables ".  
Thèse de DOCTORAT 3<sup>ème</sup> cycle. Analyse numérique,  
Université Scientifique et Médicale de GRENOBLE  
Institut National Polytechnique de GRENOBLE, 1978

M. PETIT : "Quelques programmes graphiques d'analyse numérique "  
THESE de DOCTORAT 3<sup>ème</sup> cycle, NANCY , , 1971

- D. PILCHER : " Smooth Approximation of Parametric curves and surfaces ".Ph.D THESIS - Mathematics Department, University of Utah , 1973
- L. PONTRYAGUINE, V. BOLTJANSKI, R. GAMKRELIDZE, E. MITCHENKO :  
"Théorie mathématique des processus optimaux"  
Editions MIR - MOSCOU, 1974.
- M. RIAUX : " Quelques logiciels et leurs applications en CFAO ".  
Suites-Journées Production et Automatismes. 1980  
Direction des Méthodes de la Régie Nationale des usines  
RENAULT.
- J. RICE : " The Approximation of functions .Advanced topics ";  
1969 Vol II. Nonlinear and Multivariate Theory ; ADDISON  
-WESLEY PUBLISHING COMPANY. LONDON.CALIFORNIA MENLO PARK
- L. SCHUMAKER : " Spline Functions, Basic Theory ". Pure and Applied  
Mathematics. A Wiley-Interscience Publication.  
John Wiley & Sons , NEW YORK, 1981.
- J. THILLIEZ : " La commande numérique des machines"  
DUNOD, PARIS, 1967
- J. THOMANN : " Détermination et construction de fonctions splines à  
deux variables définies sur un domaine rectangulaire ou  
circulaire "  
Thèse 3<sup>ème</sup> cycle, LILLE , 1970
- J. TODD : " Survey of numerical analysis " .Mc GRAW-HILL BOOK  
Company, INC, 1962; NEW YORK , SAN FRANCISCO



## RESUME

Le travail présenté propose une approche originale dans la détermination des fonctions SPLINE d'interpolation.

A l'inverse de la plupart des modèles d'approximation existants, nous envisageons une courbe d'interpolation comme une trajectoire d'un point passant successivement par toutes les données à interpoler. Une surface d'interpolation peut alors être définie par sa famille de génératrices.

Chaque courbe est définie à partir de la problématique des fonctions Splines (simples ou généralisées).

Le problème étant formalisé comme un problème de commande optimale des processus, nous en proposons une solution sous forme d'un algorithme séquentiel ou parallèle noté "processus interpolateur".

Cette formulation permet en particulier d'apporter des solutions naturelles aux problèmes d'instabilité numérique.

- MOTS - CLES :
- STABILITE NUMERIQUE (DYNAMIQUE)
  - CONCEPTION ASSISTEE PAR ORDINATEUR
  - COMMANDE OPTIMALE
  - (PROCESSUS)
  - FONCTION MATHEMATIQUE
  - SPLINE CUBIQUE
  - SPLINE GENERALISEE
  - FONCTION SPLINE
  - INTERPOLATION (EUR)
  - ANALYSE NUMERIQUE