



**HAL**  
open science

# Quelques programmes graphiques d'analyse numérique

Marc Petit

► **To cite this version:**

Marc Petit. Quelques programmes graphiques d'analyse numérique. Modélisation et simulation. Université Joseph-Fourier - Grenoble I, 1971. Français. NNT: . tel-00282860

**HAL Id: tel-00282860**

**<https://theses.hal.science/tel-00282860v1>**

Submitted on 28 May 2008

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

N° d'ordre

# THESE

présentée à

L'UNIVERSITE SCIENTIFIQUE ET MEDICALE DE GRENOBLE

pour obtenir

LE GRADE DE DOCTEUR DE TROISIEME CYCLE

" Mathématiques Appliquées - Informatique "

par

*Marc PETIT*

QUELQUES PROGRAMMES GRAPHIQUES

D'ANALYSE NUMERIQUE

Thèse soutenue le 25 Septembre 1971 devant la commission d'examen :

Monsieur GASTINEL

Président

Monsieur BOLLIET

Examineur

Monsieur LAURENT

Examineur



Président : Monsieur Michel SOUTIF  
Vice-Président : Monsieur Gabriel CAU

PROFESSEURS TITULAIRES

MM. ANGLÉS D'AURIAC Paul	Mécanique des fluides
ARNAUD Georges	Clinique des maladies infectieuses
ARNAUD Paul	Chimie
AYANT Yves	Physique approfondie
BARBIER Jean-Claude	Physique expérimentale
BARBIER Reynold	Géologie appliquée
BARJON Robert	Physique nucléaire
BARNOUD Fernand	Biosynthèse de la cellulose
BARRIE Joseph	Clinique chirurgicale
BENOIT Jean	Radioélectricité
BESSON Jean	Electrochimie
BEZES Henri	Chirurgie générale
BLAMBERT Maurice	Mathématiques pures
BOLLIET Louis	Informatique (IUT B)
BONNET Georges	Electrotechnique
BONNET Jean-Louis	Clinique ophtalmologique
BONNET-EYMARD Joseph	Pathologie médicale
BONNIER Etienne	Electrochimie Electrometallurgie
BOUCHERLE André	Chimie et Toxicologie
BOUCHEZ Robert	Physique nucléaire
BRAVARD Yves	Géographie
BRISSONNEAU Pierre	Physique du Solide
BUYLE-BODIN Maurice	Electronique
CABANAC Jean	Pathologie chirurgicale
CABANEL Guy	Clinique rhumatologique et hydrologique
CALAS François	Anatomie
CARRAZ Gilbert	Biologie animale et pharmacodynamie
CAU Gabriel	Médecine légale et Toxicologie
CAUQUIS Georges	Chimie organique
CHABAUTY Claude	Mathématiques pures
CHATEAU Robert	Thérapeutique
CHENE Marcel	Chimie papetière
COEUR André	Pharmacie chimique
CONTAMIN Robert	Clinique gynécologique
COUDERC Pierre	Anatomie Pathologique
CRAYA Antoine	Mécanique
Mme DEBELMAS Anne-Marie	Matière médicale
MM. DEBELMAS Jacques	Géologie générale
DEGRANGE Charles	Zoologie
DESSAUX Georges	Physiologie animale
DODU Jacques	Mécanique appliquée
DREYFUS Bernard	Thermodynamique
DUCROS Pierre	Cristallographie
DUGOIS Pierre	Clinique de Dermatologie et Syphiligraphie

FAU René	Clinique neuro-psychiatrique
FELICI Noël	Electrostatique
GAGNAIRE Didier	Chimie physique
GALLISSOT François	Mathématiques pures
GALVANI Octave	Mathématiques pures
GASTINEL Noël	Analyse numérique
GERBER Robert	Mathématiques pures
GIRAUD Pierre	Géologie
KLEIN Joseph	Mathématiques pures
Mme KOFLER Lucie	Botanique et physiologie végétale
MM. KOSZUL Jean-Louis	Mathématiques pures
KRAVTCHENKO Julien	Mécanique
KUNTZMANN Jean	Mathématiques appliquées
LACAZE Albert	Thermodynamique
LACHARME Jean	Biologie végétale
LATURAZE Jean	Biochimie pharmaceutique
LEDRU Jean	Clinique médicale B
LLIBOUTRY Louis	Géophysique
LOUP Jean	Géographie
Mlle LUTZ Elisabeth	Mathématiques pures
MM. MALGRANGE Bernard	Mathématiques pures
MALINAS Yves	Clinique obstétricale
MARTIN-NOEL Pierre	Séméiologie médicale
MASSEPORT Jean	Géographie
MAZARE Yves	Clinique médicale A
MICHEL Robert	Minéralogie et Pétrographie
MOURIQUAND Claude	Histologie
MOUSSA André	Chimie nucléaire
NEEL Louis	Physique du Solide
OZENDA Paul	Botanique
PAUTHENET René	Electrotechnique
PAYAN Jean-Jacques	Mathématiques pures
PEBAY-PEYROULA Jean-Claude	Physique
PERRET René	Servomécanismes
PILLET Emile	Physique industrielle
RASSAT André	Chimie systématique
RENARD Michel	Thermodynamique
REULOS René	Physique industrielle
RINALDI Renaud	Physique
ROGET Jean	Clinique de pédiatrie et de puériculture
SANTON Lucien	Mécanique
SEIGNEURIN Raymond	Microbiologie et Hygiène
SENGEL Philippe	Zoologie
SILBERT Robert	Mécanique des fluides
SOUTIF Michel	Physique générale
TANCHE Maurice	Physiologie
TRAYNARD Philippe	Chimie générale
VAILLAND François	Zoologie
VAUQUOIS Bernard	Calcul électronique
Mme VERAIN Alice	Pharmacie galénique
VERAIN André	Physique
Mme VEYRET Germaine	Géographie
MM. VEYRET Paul	Géographie
VIGNAIS Pierre	Biochimie médicale
YOCCOZ Jean	Physique nucléaire théorique

PROFESSEURS ASSOCIES

MM.	BULLEMER Bernhard	Physique
	RADHAKRISHNA Pidatala	Thermodynamique

PROFESSEURS SANS CHAIRE

MM.	AUBERT Guy	Physique
Mme	BARBIER Marie-Jeanne	Electrochimie
MM.	BARRA Jean	Mathématiques appliquées
	BEAUDOING André	Pédiatrie
	BERTRANDIAS Jean-Paul	Mathématiques appliquées
	BIAREZ Jean-Pierre	Mécanique
	BONNETAIN Lucien	Chimie minérale
Mme	BONNIER Jane	Chimie générale
MM.	CARLIER Georges	Biologie végétale
	COHEN Joseph	Electrotechnique
	COUMES André	Radioélectricité
	DEPASSEL Roger	Mécanique des Fluides
	DEPORTES Charles	Chimie minérale
	DESRE Pierre	Métallurgie
	DOLIQUE Jean-Michel	Physique des plasmas
	GAUTHIER Yves	Sciences biologiques
	GEINDRE Michel	Electroradiologie
	GIDON Paul	Géologie et Minéralogie
	GLENAT René	Chimie organique
	HACQUES Gérard	Calcul numérique
	JANIN Bernard	Géographie
Mme	KAHANE Josette	Physique
MM.	LATREILLE René	Chirurgie générale
	LAURENT Pierre	Mathématiques appliquées
	MULLER Jean-Michel	Thérapeutique
	PERRIAUX Jean-Jacques	Géologie et minéralogie
	POULOUJADOFF Michel	Electrotechnique
	REBECQ Jacques	Biologie (CUS)
	REVOL Michel	Urologie
	REYMOND Jean-Charles	Chirurgie générale
	ROBERT André	Chimie papetière
	SARRAZIN Roger	Anatomie et chirurgie
	SARROT-REYNAULD Jean	Géologie
	SIBILLE Robert	Construction Mécanique
	SIROT Louis	Chirurgie générale
Mme	SOUTIF Jeanne	Physique générale
M.	VALENTIN Jacques	Physique nucléaire

MAITRES DE CONFERENCES ET MAITRES DE CONFERENCES AGREGES

Mle	AGNIUS-DELORD Claudine	Physique pharmaceutique
	ALARY Josette	Chimie analytique
MM.	AMBLARD Pierre	Dermatologie
	AMBROISE-THOMAS Pierre	Parasitologie
	ARMAND Yves	Chimie

BEGUIN Claude	Chimie organique
BELORIZKY Elie	Physique
BENZAKEN Claude	Mathématiques appliquées
Mme BERTRANDIAS Françoise	Mathématiques pures
MM. BLIMAN Samuel	Electronique (EIE)
BLOCH Daniel	Electrotechnique
Mme BOUCHE Liane	Mathématiques (CUS)
MM. BOUCHET Yves	Anatomie
BOUSSARD Jean-Claude	Mathématiques appliquées
BOUVARD Maurice	Mécanique des Fluides
BRIERE Georges	Physique expérimentale
BRODEAU François	Mathématiques (IUT B)
BRUGEL Lucien	Energétique
BUISSON Roger	Physique
BUTEL Jean	Orthopédie
CHAMBAZ Edmond	Biochimie médicale
CHAMPETIER Jean	Anatomie et organogénèse
CHARACHON Robert	Oto-Rhino-Laryngologie
CHIAVERINA Jean	Biologie appliquée (EFP)
CHIBON Pierre	Biologie animale
COHEN-ADDAD Jean-Pierre	Spectrométrie physique
COLOMB Maurice	Biochimie médicale
CONTE René	Physique
CROUZET Guy	Radiologie
DURAND Francis	Métallurgie
DUSSAUD René	Mathématiques (CUS)
Mme ETERRADOSSI Jacqueline	Physiologie
MM. FAURE Jacques	Médecine légale
GAVEND Michel	Pharmacologie
GENSAC Pierre	Botanique
GERMAIN Jean-Pierre	Mécanique
GIDON Maurice	Géologie
GRIFFITHS Michael	Mathématiques appliquées
GROULADE Joseph	Biochimie médicale
HOLLARD Daniel	Hématologie
HUGONOT Robert	Hygiène et médecine préventive
IDELMAN Simon	Physiologie animale
IVANES Marcel	Electricité
JALBERT Pierre	Histologie
JOLY Jean-René	Mathématiques pures
JOUBERT Jean-Claude	Physique du Solide
JULLIEN Pierre	Mathématiques pures
KAHANE André	Physique générale
KUHN Gérard	Physique
Mme LAJZEROWICZ Jeannine	Physique
MM. LAJZEROWICZ Joseph	Physique
LANCIA Roland	Physique atomique
LE JUNTER Noël	Electronique
LEROY Philippe	Mathématiques
LOISEAUX Jean-Marie	Physique nucléaire
LONGEQUEUE Jean-Pierre	Physique nucléaire
LUU DUC Cuong	Chimie organique
MACHE Régis	Physiologie végétale
MAGNIN Robert	Hygiène et Médecine préventive
MARECHAL Jean	Mécanique
MARTIN-BOUYER Michel	Chimie (CUS)
MAYNARD Roger	Physique du Solide
MICOUD Max	Maladies infectieuses
MOREAU René	Hydraulique (INP)

	NÈGRE Robert	Mécanique
	PARAMELLE Bernard	Pneumologie
	PECCOUD François	Analyse (IUT B)
	PEFFEN René	Métallurgie
	PELMONT Jean	Physiologie animale
	PERRET Jean	Neurologie
	PERRIN Louis	Pathologie expérimentale
	PFISTER Jean-Claude	Physique du Solide
	PHELIP Xavier	Rhumatologie
Mle	PIERY Yvette	Biologie animale
	RACHAIL Michel	Médecine interne
	RACINET Claude	Gynécologie et obstétrique
	RICHARD Lucien	Botanique
Mme	RINAUDO Marguerite	Chimie macromoléculaire
MM.	ROMIER Guy	Mathématiques (IUT B)
	ROUGEMONT (DE) Jacques	Neuro-chirurgie
	STIEGLITZ Paul	Anesthésiologie
	STOEBNER Pierre	Anatomie pathologique
	VAN CUTSEM Bernard	Mathématiques appliquées
	VEILLON Gérard	Mathématiques appliquées (INP)
	VIALON Pierre	Géologie
	VOOG Robert	Médecine interne
	VIROUSSOS Constantin	Radiologie
	ZADWORNÝ François	Electronique

#### MAITRES DE CONFERENCES ASSOCIES

MM.	BOUDOURIS Georges	Radioélectricité
	CHEEKE John	Thermodynamique
	GOLDSCHMIDT Hubert	Mathématiques
	YACOUD Mahmoud	Médecine légale

#### CHARGES DE FONCTIONS DE MAITRES DE CONFERENCES

Mme	BERIEL Hélène	Physiologie
Mme	RENAUDET Jacqueline	Microbiologie





*Je tiens à remercier :*

*Monsieur le Professeur Noël GASTINEL,  
Directeur du Laboratoire de Calcul, qui a bien voulu me faire l'honneur de  
présider le jury de cette Thèse, et d'en diriger les travaux.*

*Monsieur Louis BOLLIET,  
Professeur à l'Institut Universitaire de Technologie de GRENOBLE, pour ses  
encouragements répétés et l'intérêt amical qu'il a porté à ce travail.*

*Monsieur Pierre Jean LAURENT,  
Professeur à l'Institut de Mathématiques Appliquées de GRENOBLE, qui a suivi  
avec intérêt mon travail et m'a aidé de ses critiques et de ses conseils.*

*Mes remerciements vont également à tous mes collègues  
du Laboratoire, en particulier Madame LONCHAMPT et Monsieur BELLOT.*

*Je tiens également à remercier les services de  
dactylographie et de reproduction qui ont permis la réalisation matérielle  
de ce travail.*

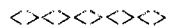


*A ma femme*

*A mon fils*



# TABLE DES MATIERES



	Page
<u>INTRODUCTION</u>	1
<u>CHAPITRE 1 - MATERIEL UTILISE</u>	
1.1 La configuration .....	3
1.1.1 L'unité centrale .....	3
1.1.2 Les périphériques .....	3
1.1.2.1 Le terminal graphique IBM 2250- Modèle 1.....	4
1.2 Le Software .....	8
1.2.1 Les systèmes .....	8
1.2.1.1 CP-67 .....	8
1.2.1.2 CMS .....	8
1.2.2 Les Langages .....	9
1.2.2.1 Le langage EXEC .....	10
1.2.2.1.1 Description des commandes de contrôle .....	10
1.2.2.2 GSP .....	11
1.2.2.3 Fortran et Assembleur 360 .....	11
<u>CHAPITRE 2 - LE SOUS-SYSTEME PGAN</u>	
2.1 Introduction .....	14
2.2 Description du sous-système .....	15
2.2.1 Le programme de contrôle .....	15
2.2.2 Les commandes du sous-système.....	18
2.2.2.1 Les commandes d'appel de programmes.....	19
2.2.2.2 Les commandes systèmes .....	25
2.2.3 Les programmes généraux .....	29

	Page
2.2.3.1 Gestion de l'écran .....	30
2.2.3.2 Le crayon optique .....	32
2.2.3.3 Le clavier alphanumérique .....	32
2.2.3.4 Le clavier de fonctions .....	33
2.2.4 Liste et description des fonctions .....	33
2.2.4.1 Gestion et enchaînement des fonctions .....	35
2.2.5 Entrée des données pour les programmes généraux .....	38
2.2.5.1 Description du contenu des fichiers .....	38
2.2.6 Les sous-programmes de service .....	43
2.3 Quelques détails de programmation .....	43

### CHAPITRE 3 - METHODES MATHEMATIQUES

3.1 Introduction .....	45
3.2 Interpolation .....	45
3.2.1 Spline d'interpolation d'ordre q .....	47
3.2.1.1 Propriétés .....	48
3.2.1.2 Mise en oeuvre de l'algorithme .....	51
3.2.1.2.1 Evaluation des q-1 premières dérivées .....	52
3.2.1.2.2 Calcul de la valeur du polynôme .....	56
3.2.1.3 Algorithmes en ALGOL .....	58
3.3 Ajustement .....	63
3.3.1 Spline d'ajustement d'ordre q .....	64
3.3.1.1 Propriétés .....	65
3.3.1.2 Mise en oeuvre de l'algorithme .....	67
3.3.1.2.1 Calcul de la valeur de la fonction et de la valeur de ses q-1 premières dérivées. ....	67
3.3.1.2.2 Calcul de la valeur du polynôme .....	72
3.3.1.3 Algorithmes en ALGOL .....	72
3.3.2 Ajustement polynomial au sens des moindres carrés .....	78
3.3.2.1 Propriétés de la matrice B .....	80
3.3.2.2 Mise en oeuvre de l'algorithme .....	82
3.4 Recherche de courbes de niveaux sur un maillage rectangulaire ....	84

	Page
3.4.1 Fonction spline d'interpolation à 2 variables sur un domaine rectangulaire .....	86
3.4.2 Stratégie utilisée .....	88
3.5 Résolution des systèmes différentiels du premier ordre .....	90
3.5.1 Rappel de la méthode de Runge-Kutta .....	91
<u>CHAPITRE 4 - RESULTATS OBTENUS</u>	
4.1 Introduction .....	92
4.2 Résultats théoriques .....	93
4.3 Autres résultats .....	103
4.3.1 Calcul du bilan de masse d'un glacier .....	103
4.3.2 Résolution d'un système différentiel du 1er ordre .....	105
4.3.3 Autres exemples .....	110
<u>CONCLUSION</u> .....	114
<u>REFERENCES BIBLIOGRAPHIQUES</u> .....	116





## INTRODUCTION

Le but de ce travail est de permettre à un utilisateur, qu'il soit programmeur ou non, d'utiliser un terminal graphique en mode conversationnel, pour la résolution de problèmes classiques d'Analyse Numérique.

Un terminal graphique utilisé dans cet environnement, met à la disposition de l'utilisateur des moyens extrêmement puissants, à savoir, visualisation et conversation. Tout utilisateur est de ce fait en liaison étroite avec son programme et peut voir l'influence de la valeur de certains paramètres sur le tracé des courbes ou les résultats numériques. Il peut aussi prendre une ou plusieurs décisions, les transmettre instantanément à son programme afin de donner une nouvelle orientation à la résolution de son problème.

De nombreuses applications dans plusieurs domaines utilisent intensivement ces nouvelles ressources, notamment: la création, la mise à jour et l'édition de fichiers, le dessin de maquettes de voitures ou d'avions, la description et manipulation formelle de formules mathématiques etc...

L'analyse numérique restait encore un parent pauvre; c'est pourquoi nous avons voulu fournir un ensemble de programmes interactifs pour les applications suivantes:

- L'interpolation.
- L'ajustement.
- La recherche de courbes de niveau sur un maillage.
- La résolution de systèmes différentiels.

En effet, ce type de problèmes se prête particulièrement au traitement graphique, car les solutions cherchées peuvent être visualisées sur l'écran cathodique du terminal sous forme de courbes. Il est toujours plus agréable d'interpréter une courbe que des colonnes de chiffres.

Nous avons écrit un sous-système: P.G.A.N. (Programmes Graphiques d'Analyse Numérique) permettant d'aborder les sujets énoncés plus haut. Nous nous sommes attachés à rendre l'emploi de ce sous-système aussi souple et facile que possible; il n'est nullement besoin de connaître la programmation et le fonctionnement du terminal graphique pour utiliser P.G.A.N. Un ensemble de commandes et de fonctions élémentaires sont disponibles et tout utilisateur peut les apprendre rapidement.

Nous nous proposons de décrire ce système qui a été réalisé à l'Institut de Mathématiques Appliquées de GRENOBLE (I.M.A.G.).

## C H A P I T R E I

### MATERIEL UTILISE

Nous décrirons brièvement les moyens employés à l'I.M.A.G. pour la réalisation de ce travail. Nous considèrerons deux aspects: La configuration du matériel et le software utilisé.

#### I.1 LA CONFIGURATION - Figure 1 -

L'ensemble de traitement est constitué par un ordinateur I.B.M. 360/67 modèle 1 [1]. Il bénéficie de dispositifs spéciaux en particulier la D.A.T. (Dynamic Address Translator). Nous distinguons l'unité centrale et les périphériques qui lui sont connectés.

##### I.1.1 L'unité centrale.

Elle possède:

- Une mémoire de 1024 K octets.
- 16 registres généraux ou d'index de 4 octets chacun.
- 4 registres virgules flottantes de 8 octets.
- 5 types d'interruption.
- Un cycle mémoire de 750 ns. Une instruction, en moyenne, s'exécute en 2  $\mu$ s.

##### I.1.2 Les périphériques.

On appelle périphérique tout ce qui peut être connecté à l'unité

centrale par l'intermédiaire d'un canal. Un canal, est une "ligne" permettant aux informations (impulsions électriques) de circuler entre l'unité centrale et les périphériques. On peut distinguer :

- 2 tambours magnétiques d'une capacité de 4.000.000 d'octets chacun.
- 2 batteries de 8 disques magnétiques et une de 4, d'une capacité totale de 560 millions d'octets.
- 4 dérouleurs de bandes magnétiques (3\*9 pistes + 1 \* 7 pistes).
- Deux lecteurs/perforateurs de cartes.
- Deux imprimantes papier (1200 lignes/minute).
- Un terminal graphique I.B.M. 2250 modèle 1.
- 70 terminaux légers du type machine à écrire.

Nous accorderons un intérêt particulier au terminal graphique.

#### I.1.2.1 Le Terminal Graphique IBM 2250 modèle 1 [2] -Figure 2-

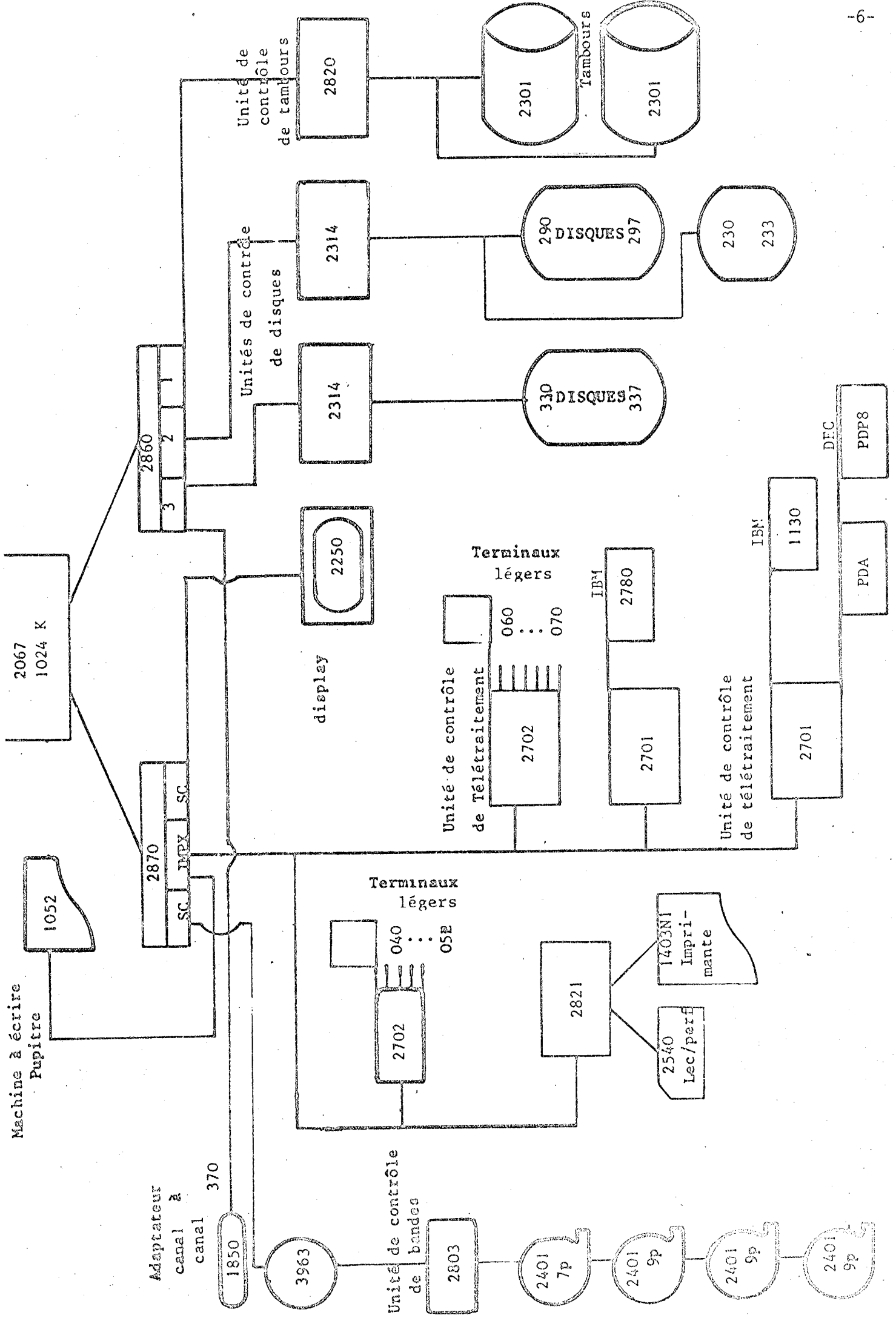
Il est composé d'un certain nombre d'éléments.

- a- Un écran cathodique permettant l'affichage d'une image générée au moyen de trois éléments de base: Le point, le vecteur, le caractère.
- b- Un clavier de fonction: C'est, un ensemble de 32 touches banales numérotées de 0 à 31. L'enfoncement de l'une des touches provoque une interruption au niveau de l'ordinateur. Elle communique à celui-ci le numéro de la Touche. Il exécute alors la fonction associée à l'emploi de cette touche. Cette fonction peut être un appel à un sous-programme, une réponse à une question du programme, etc...
- c- Un clavier alphanumérique: semblable à celui d'une machine à écrire, il permet de composer sur l'écran n'importe quelle chaîne de caractères. On dispose de 63 caractères et de deux touches "END" et "CANCEL" qui permettent, respectivement, d'envoyer à l'ordinateur la chaîne affichée ou de la détruire.

- d- Un crayon optique: constitué d'une cellule photo-électrique, il permet de récupérer les coordonnées d'un point de l'écran au moyen d'une interruption.
- e- Une mémoire d'entretien de 8 K octets, où se trouve le programme graphique correspondant à l'image observée. L'image affichée est fugitive, pour la rendre stable, elle est régénérée n fois par seconde ( $n \leq 40$ ). Ceci veut dire que le 2250 se comporte comme un petit ordinateur, qui exécute n fois le programme graphique par seconde.

Au moyen du clavier de fonction, du clavier alphanumérique ou du crayon optique l'utilisateur peut stopper le déroulement de son programme et engager le dialogue avec ce dernier.

REMARQUE. Seules les interruptions prévues par le programme sont intelligibles par celui-ci.



SECRET - CONFIDENTIAL

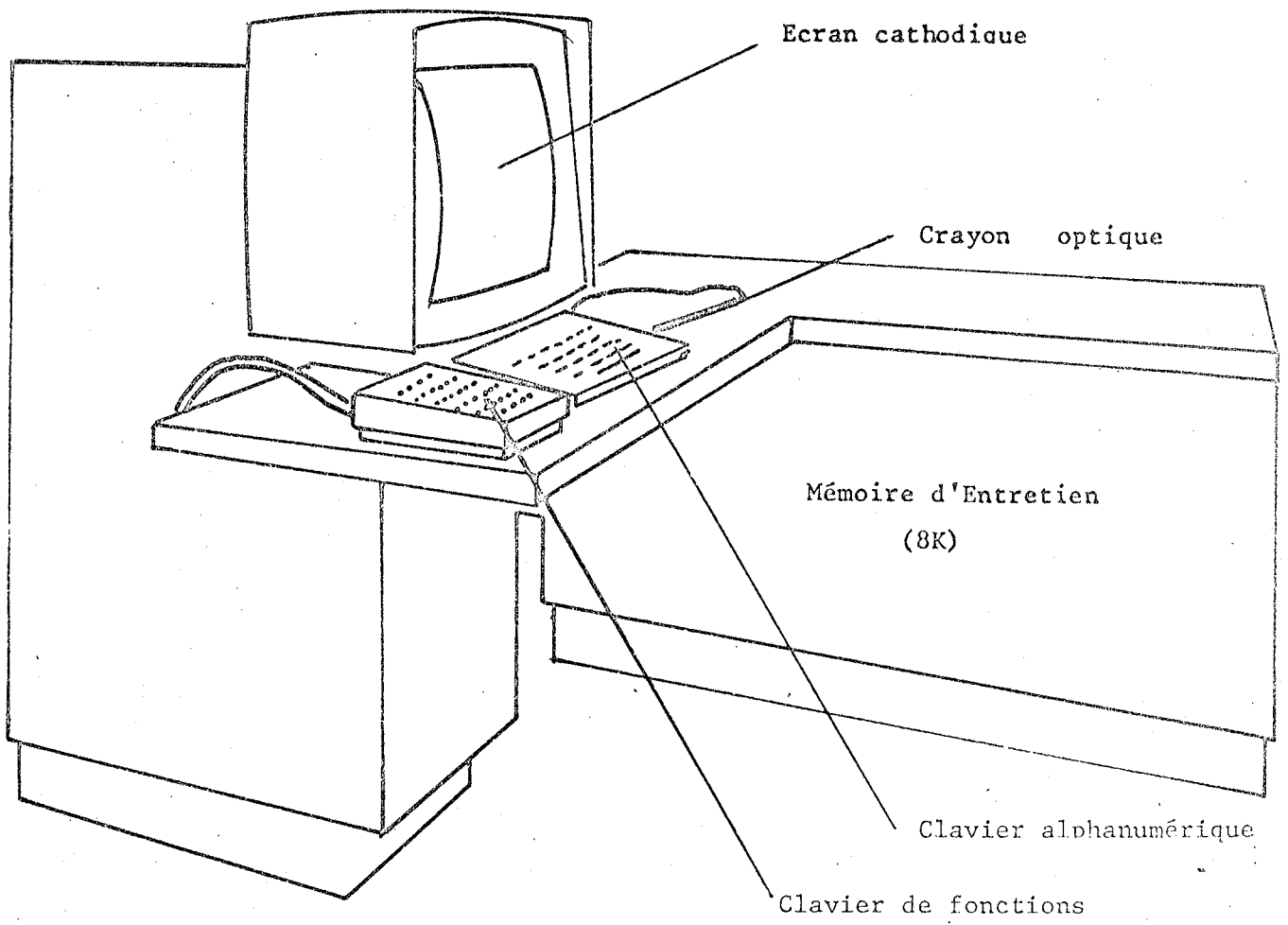


Figure 2 I.B.M. 2250 unit, modèle 1



## I.2 LE SOFTWARE

Nous donnerons également une brève description des systèmes et langages utilisés pour la réalisation de P.G.A.N.

### I.2.1 Les systèmes

Ils sont au nombre de deux et dans l'ordre.

#### I.2.1.1 CP-C7 (Control Program - 67) [3]

Ce système de partage de temps et de ressources fonctionne sur un ordinateur IBM-360/67 et génère, pour chaque utilisateur, une machine virtuelle analogue à un ordinateur IBM-360 standard. Les machines virtuelles sont indépendantes les unes des autres, et sont activées successivement pendant un court intervalle de temps par le système CP-67. Ces dernières diffèrent entre elles par leurs caractéristiques (taille mémoire, organes d'entrées-sorties...). La figure 3 donne la configuration de la machine virtuelle sur laquelle a été réalisé le système P.G.A.N.

Tout système, fonctionnant sur un ordinateur IBM-360 standard, peut fonctionner sur une telle machine virtuelle en particulier, DOS, OS, PCP, MFT, MVT, DOS-APL, CMS etc... Nous étudierons plus particulièrement ce dernier.

#### I.2.1.2 CMS (Cambridge Monitor System) [3]

Ce système conversationnel est chargé dans la machine virtuelle générée par CP-67. C'est-à-dire qu'il facilite au mieux, le dialogue entre l'utilisateur et sa machine virtuelle.

Il met à la disposition de l'utilisateur un grand nombre de facilités, en particulier :

- Un langage de commandes.
- Un sous-système de création et de modification de fichier: EDIT.
- Un ensemble important de composants (assembleurs et compilateurs).
- Un sous-système de mise au point de programmes: DEBUG.
- Un macro-langage de commandes: EXEC.

Les commandes déclenchent, soit l'exécution d'un service, soit le passage à un sous-système.

D'une part, certaines commandes permettent l'appel de compilateurs tels que: Fortran, Algol, PL/1, Snobol, Lisp, etc...

D'autre part, le macro-langage EXEC [3] permet de définir des macro-commandes. C'est une de ces macro-commandes qui donnera le contrôle au système P.G.A.N.

### I.2.2 Les langages

Nous avons, pour l'écriture de P.G.A.N., utilisé principalement le langage Fortran IV [4]. Une partie, cependant, qu'on pourrait appeler le programme de contrôle est écrite à l'aide du langage EXEC de CMS. Il constitue une macro-commande qui gère l'ensemble des ressources offertes par P.G.A.N. Il pourrait d'ailleurs s'écrire en Fortran. Nous avons préféré utiliser ce dernier pour des raisons de souplesse, de simplicité et de gain de temps. Les programmes Fortran emploient intensivement une bibliothèque de sous-programmes, pour l'utilisation du terminal graphique 2250, appelée G.S.P. (Graphic Subroutine Package) [5]. Enfin quelques sous-programmes sont écrits en assembleur 360 [6].

### I.2.2.1 Le langage EXEC.

Les éléments du langage sont constitués par les commandes de CMS proprement dites, dont on trouvera la description dans [3] et par des commandes de contrôle. Ces dernières permettent à l'utilisateur de construire en fonction des précédentes, des macro-commandes plus spécialement adaptées à ses besoins. C'est une commande semblable qui gère tous les appels aux programmes de P.G.A.N.

#### I.2.2.1.1 Description des commandes de contrôle.

Elles sont formées à l'aide d'étiquettes, de variables et de mots de contrôle.

a- Les étiquettes. Ex: -10, -ABC

b- Les variables sont constituées par:

- Des variables simples. Ex: &TOTO, &NUM1
- Des variables numériques. Ex: &0, &1, &\*
- Des variables à mot-clé.

&INDEXO ... &INDEX9

&GLOBALO ... &GLOBAL9

&LINENUM

c- Parmi les mots de contrôle, on distingue ceux de

- Branchement. &QUIT

&EXIT

&GØTØ

&SKIP

&CØNTINUE

- Branchement conditionnel. &ERROR

&IF

&LØØP

- Contrôle de lecture. &ARGS  
&READ  
&STACK  
&BEGSTACK  
&ENDSTACK

- Contrôle de service. &COMMENT  
&PRINT  
&TYPEOUT  
&TIME  
&SPACE

SERVICE POLYCOPIE  
MATHÉMATIQUES APPLIQUÉES  
Université de GRENOBLE

#### I.2.2.2 G.S.P.

C'est un ensemble de 57 sous-programmes écrits en assembleur 360, utilisable en Cobol, PL/1, Fortran et assembleur 360 sous "l'Operating System 360". Ces sous-programmes permettent de définir des figures (Graphic Data Set) et dans ces figures, créer et manipuler des éléments. Ces éléments pouvant être des points, des segments, des lignes, des textes.

Pour fonctionner sous CMS, GSP a été modifié par C.HANS [7] et O.LECARME [7] [8]. Cette version utilisable en Fortran et assembleur 360 est celle que nous avons employée dans les divers programmes du P.G.A.N.

#### I.2.2.3 Fortran IV et Assembleur 360.

L'existence de GSP sous CMS permettait l'utilisation de Fortran ou de l'assembleur pour l'écriture de P.G.A.N. Nous avons préféré Fortran, pour des raisons de commodité, il est plus simple en effet, de décrire un algorithme en Fortran, qu'en assembleur. Etant donnée l'utilisation généralisée de Fortran IV,

les programmes gardent de ce fait une certaine indépendance vis à vis de la machine.

Pour ce qui concerne l'Assembleur, il n'a été utilisé que lorsque Fortran était défaillant (Ex: Test de l'existence d'un fichier) ou lorsqu'un sous-programme Fortran conduisait à un temps d'exécution trop important (Ex: Affichage de données numériques à l'écran du 2250).

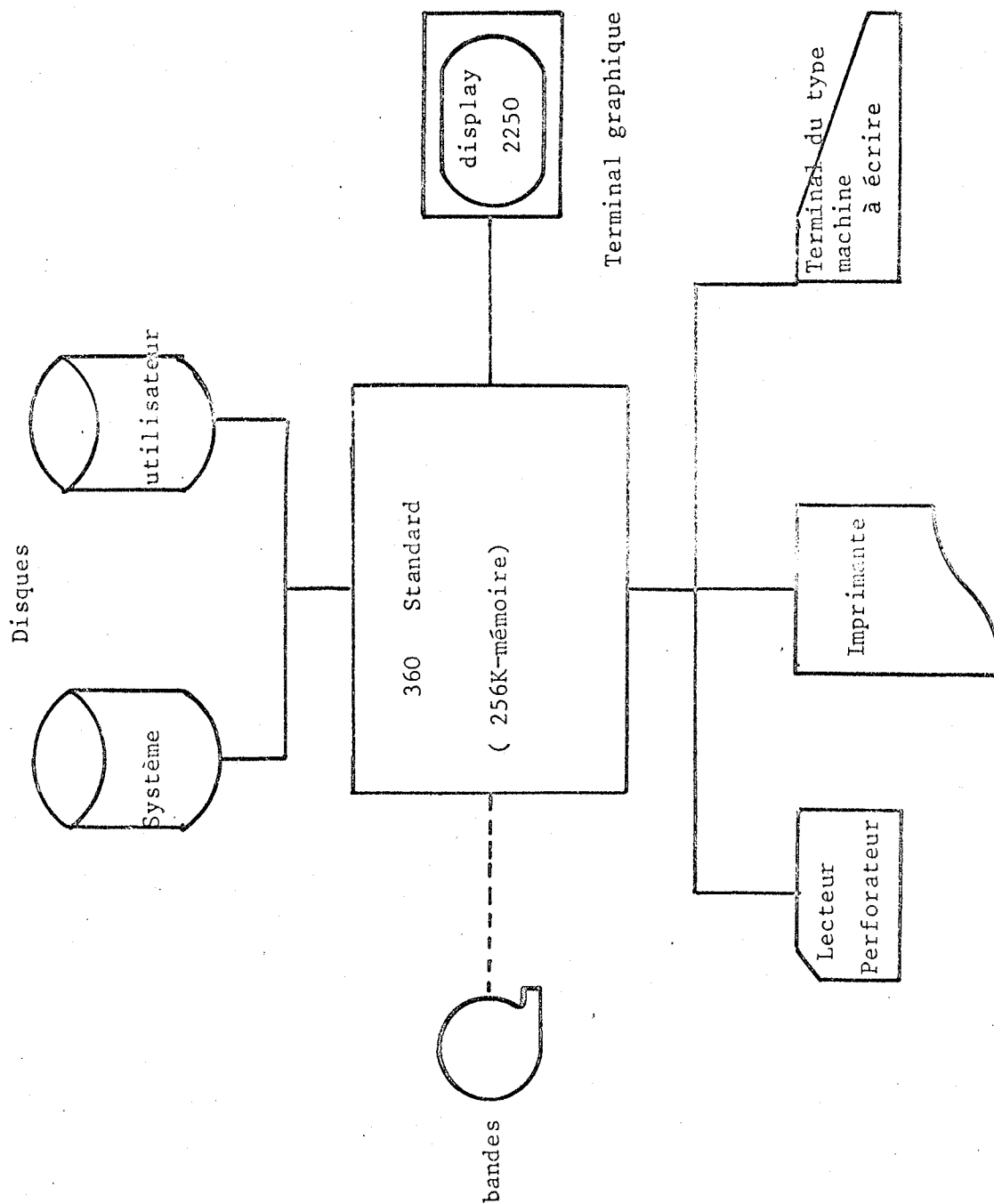


FIGURE 3. CONFIGURATION DE LA MACHINE VIRTUELLE



## C H A P I T R E 2

### LE SOUS SYSTEME PGAN

#### 2.1 INTRODUCTION

Le système PGAN est destiné aux programmeurs (physiciens ou non) qui veulent traiter d'une manière conversationnelle et interactive sur un matériel précis (Cf. Chapitre 1) un ensemble de problèmes mathématiques particuliers développés au chapitre 3.

Je ne détaillerai pas tous les avantages que peut présenter un système conversationnel, mais j'indiquerai ceux qui dans le cadre de nos travaux présentent le plus d'intérêt.

Le premier et peut-être un des plus importants est le gain de temps; En effet, avec un tel système, l'utilisateur peut avoir presque instantanément la solution de son problème, alors que dans une exploitation de type 'Batch', le temps d'attente de la liste des résultats peut devenir prohibitif. Dans un cas, il attendra quelques secondes, dans l'autre, quelques heures.

Un second avantage est lié à l'action immédiate de l'utilisateur sur son programme; il peut, à tout instant modifier la valeur de certains paramètres, voir sur l'écran les résultats de sa modification et déterminer, par approximations successives, la meilleure valeur des paramètres de son problème. L'utilisateur peut réaliser partiellement ce même travail en 'batch' mais au prix de nombreuses exécutions successives.

L'avantage le plus important est sans doute la possibilité donnée à l'utilisateur de voir globalement la solution de son problème, exprimée sous forme de courbe. Rien n'est plus facile au physicien que d'interpréter



une courbe, plutôt que de consulter des colonnes de chiffres qui seront finalement reportés sur du papier millimétré, afin d'obtenir une idée du phénomène.

Enfin, nous avons voulu faire de ce système un outil puissant et d'emploi facile, d'où des commandes et fonctions simples et claires mais efficaces dans le domaine où elles s'appliquent. PGAN demande peu de connaissances en programmation: l'instruction d'affectation fortran, les notions les plus élémentaires de CMS, en particulier l'éditeur graphique. De ce fait, tout utilisateur peut l'apprendre et l'utiliser rapidement.

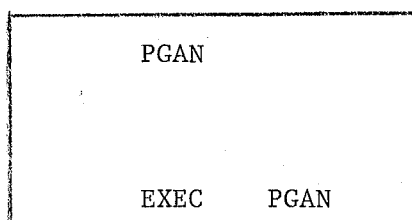
## 2.2 DESCRIPTION DU SOUS-SYSTEME.

Nous voulons décrire PGAN de telle sorte que ce chapitre soit également un manuel pour l'utilisateur. Le sous-système est composé d'un programme de contrôle, de 9 programmes généraux et de 37 sous-programmes de service. Pour CMS, le sous-système se comporte comme un programme ordinaire; ceci veut dire que lorsque le programme est activé, l'utilisateur ne converse qu'avec lui. Le programme de contrôle se comporte comme un superviseur qui met en oeuvre dans un ordre logique, les différents programmes du sous-système (cf.figure 4) en fonction des demandes de l'utilisateur.

### 2.2.1 Le programme de contrôle.

C'est un programme conversationnel, écrit à l'aide du langage EXEC. De ce fait, il constitue une macro-commande du système CMS. L'utilisateur lui donne le contrôle en émettant à partir du terminal les commandes:

ou



*EXEMPLES*

1)

pgan

UNIVERSITE DE GRENOBLE

LE SYSTEME P.G.A.N. EST EN FONCTION...

2)

exec pgan

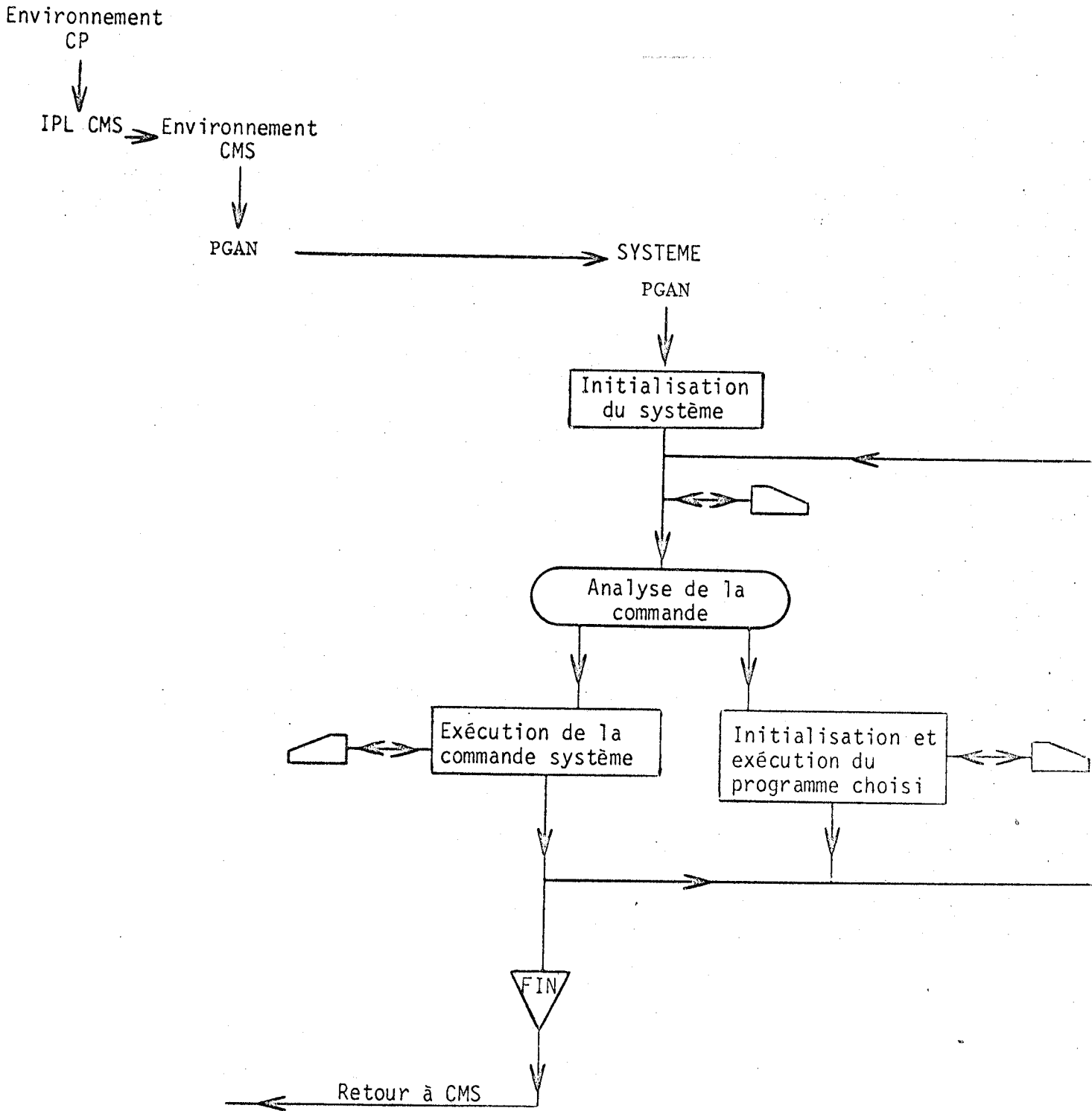
UNIVERSITE DE GRENOBLE

LE SYSTEME P.G.A.N. EST EN FONCTION...

Ce programme contrôle au moyen de commandes, l'exécution des programmes généraux ou des fonctions du système mis à la disposition de l'utilisateur. Ces programmes généraux peuvent à leur tour appeler d'autres sous-programmes. Vis à vis de PGAN, ces programmes se situent au même niveau. Ils permettent de résoudre les problèmes d'analyse numérique évoqués dans l'introduction et développés dans le chapitre 4.

En d'autres termes, on peut considérer ce sous système comme une véritable bibliothèque de programmes traitant de problèmes particuliers. En effet, l'utilisateur dispose d'une commande spéciale qui lui permet d'ajouter au sous-système n'importe quel programme et sous-programme de son choix, pourvu qu'il respecte les conventions d'écriture des programmes généraux. Par ce moyen, il peut construire un sous-système personnalisé et augmenter ainsi le domaine d'application. Nous reviendrons sur cette commande lors de l'étude des fonctions du sous-système.

Lorsque PGAN a pris le contrôle, il guide, vérifie et contrôle les différentes demandes d'exécution par un échange de messages avec l'utilisateur. Pour cela PGAN reçoit des commandes ou des réponses (oui ou non) à des questions. La figure 4 montre à quels niveaux se situent ces échanges.



Légende.


 Echange de messages entre le système et l'utilisateur

Figure 4. Logique du sous-système

### 2.2.2 Les commandes du sous-système .

Parmi ces commandes, nous en distinguerons deux sortes: Celles qui permettent l'appel d'un programme général et celles qui réalisent une fonction du sous-système.

#### Liste des commandes

INTERSP2	: Appelle le programme d'interpolation par des fonctions splines d'ordre 2.
INTERSPQ	: Appelle le programme d'interpolation par des fonctions splines d'ordre $q$ . $3 \leq q \leq 6$ .
AJUSP2	: Appelle le programme d'ajustement par des fonctions splines d'ordre 2.
AJUSPQ	: Appelle le programme d'ajustement par des fonctions splines d'ordre $q$ . $3 \leq q \leq 6$ .
AJUSMDR	: Appelle le programme d'ajustement au sens des moindres carrés.
TOMAN	: Appelle un programme qui prépare les données pour la commande suivante.
GRILLE	: Appelle un programme qui calcule les courbes de niveau sur une grille.
SYSDIFRK	: Appelle un programme qui résout un système différentiel du premier ordre par une méthode de Runge-Kutta.
BENSON	: Appelle un programme qui dessine à l'aide d'un traceur Benson, une image de l'écran du 2250.
LISTE	: Liste au Terminal toutes les commandes du système.
LISTLIB	: Liste au Terminal le nom de tous les programmes et sous-programmes du système.

CMS : Permet à l'utilisateur de taper au terminal un certain nombre de commandes CMS.

ADD : Permet à l'utilisateur d'ajouter au système un programme et ses sous-programmes et de créer ainsi une nouvelle commande.

DELETE : Permet à l'utilisateur d'enlever du système un programme et ses sous-programmes et de supprimer la commande correspondante.

FIN : Permet de sortir du système et de revenir dans l'environnement CMS.

#### 2.2.2.1 Les commandes d'appel de programmes.

Ce sont les 9 premières de la liste ci-dessus. Elles se composent d'un mot-clé sans paramètre, dont le nom mnémorique rappelle le programme appelé.

##### INTERSP2

Format:

INTERSP2

Fonction: Charge en mémoire et calcule le programme d'interpolation par des fonctions splines d'ordre 2.  
(cf.3.2.1).

Exemple: TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME

intersp2

INTERPOLATION PAR DES FONCTIONS SPLINES D'ORDRE 2

ENTREZ VOUS VOS DONNEES PAR FICHER? REpondre OUI OU NON

—

non

EXECUTION BEGINS...

...EXECUTION ENDED

T=3.42/7.53 06.46.45

INTERSPQ

Format:

INTERSPQ

Fonction: Charge en mémoire et exécute le programme d'interpolation par des fonctions splines d'ordre  $q$  ( $3 \leq q \leq 6$ ) . (cf.3.2.1)

Exemple: TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME

—  
interspq  
INTERPOLATION PAR DES FONCTIONS SPLINES D'ORDRE Q  
ENTREZ VOUS VOS DONNEES PAR FICHER? REPONDRE OUI OU NON  
—  
oui  
VOTRE FICHER INTERSPQ N'EXISTE PAS.  
VEUILLEZ LE CREER COMME: FILE INTERSPQ P1

AJUSP2

Format:

AJUSP2

Fonction: charge en mémoire et exécute le programme d'ajustement par des fonctions splines d'ordre 2. (cf.3.3.1).

Exemple: TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME

—  
ajusp2  
AJUSTEMENT PAR DES FONCTIONS SPLINES D'ORDRE 2  
ENTREZ VOUS VOS DONNEES PAR FICHER? REPONDRE OUI OU NON  
—  
non  
REPONDEZ OUI OU NON !  
—  
non  
EXECUTION BEGINS...  
...EXECUTION ENDED  
T=4.40/8.20 06.48.26

AJUSPQ

Format:

AJUSPQ

Fonction: charge en mémoire et exécute le programme d'ajustement par des fonctions splines d'ordre  $q$  ( $3 \leq q \leq 6$ ).  
(Cf. 3.3.1).

Exemple: TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME.

```
-  
ajuspq  
AJUSTEMENT PAR DES FONCTIONS SPLINES D'ORDRE Q  
ENTREZ VOUS VOS DONNEES PAR FICHER? REpondRE OUI OU NON  
-  
non  
EXECUTION BEGINS...  
...EXECUTION ENDED  
T=3.05/06.19 06.49.42
```

AJUSMDR

Format:

AJUSMDR

Fonction: charge en mémoire et exécute le programme d'ajustement au sens des moindres carrés. (cf. 3.3.2).

Exemple: TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME

```
-  
ajusmdr  
AJUSTEMENT AU SENS DES MOINDRES CARRÉS  
ENTREZ VOUS VOS DONNEES PAR FICHER? REpondRE OUI OU NON  
-  
non  
EXECUTION BEGINS...  
... EXECUTION ENDED  
T-3.08/6.27 06.51.57
```

TOMAN

Format:

TOMAN

Fonction: Charge en mémoire et exécute le programme qui crée les données indispensables à l'exécution du programme appelé par la commande GRILLE. Pour cela, on utilise une fonction spline d'interpolation à deux variables sur un domaine rectangulaire (cf.3.4.1).

Exemple: TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME

—  
toman

CREATION PAR LE PROGRAMME TOMAN DU FICHER:  
FILE GRILLE P1  
EXECUTION BEGINS...  
...EXECUTION ENDED  
T=24.87/27.23 07.42.13

GRILLE

Format:

GRILLE

Fonction: Charge en mémoire et exécute le programme qui calcule les courbes de niveaux sur un réseau de points (cf.3.4.2).

Exemple: TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME

—  
grille

RECHERCHE DE COURBES DE NIVEAU SUR UNE GRILLE

ENTREZ VOUS VOS DONNEES PAR FICHER? REpondre OUI OU NON

—  
oui  
EXECUTION BEGINS...  
...EXECUTION ENDED  
T=3.29/7.71 08.24.33



SYSDIFRK

Format:

SYSDIFRK

Fonction: Charge en mémoire et exécute le programme qui résoud un système différentiel du premier ordre par une méthode de Runge Kutta.(cf.3.5).

Exemple: TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME

sysdifrk

RESOLUTION D'UN SYSTEME DIFFERENTIEL DU 1ER ORDRE  
PAR UNE METHODE DE RUNGE-KUTTA.

DEFINITION DE VOTRE PROBLEME

CREATION DU FICHIER DEBRK  
EXECUTION BEGINS...  
...EXECUTION ENDED

EDITION GRAPHIC DU FICHIER DEBRK  
EXECUTION BEGINS...  
...EXECUTION ENDED

EXECUTION BEGINS...  
...EXECUTION ENDED  
T=5.78/14.54 08.12.09

BENSON

Format:

BENSON

Fonction: Charge en mémoire et exécute le programme qui restitue sur bande magnétique une image de l'écran préalablement copiée dans un fichier (FILE BENSON P). Cette bande, montée sur le traceur digital Benson modèle 121, restitue sur papier l'image initiale de l'écran.

Exemple: TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME

—  
benson

CREATION D'IMAGE(S) POUR LE BENSON  
ON DEMANDE A L'OPERATEUR D'ATTACHER UNE BANDE A VOTRE  
MACHINE COMME 180. LORSQUE VOUS AUREZ SA REPOSE  
REPONDEZ PAR OUI OU NON SELON LE CAS

—  
FROM OPERATOR: C'EST FAIT

oui  
DEV 180 ATTACHED  
EXECUTION BEGINS...  
DEV 180 DETACHED  
...EXECUTION ENDED  
T=4.64/6.34 08.28.16  
VOULEZ-VOUS CONSERVER VOTRE FICHER 'FILE BENSON'?  
REPONDE OUI OU NON

—  
oui

REMARQUES:

- Exemple d'une commande utilisateur

TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME

—  
progl

EXECUTION DE LA COMMANDE UTILISATEUR:  
PROG1

ENTREZ VOUS VOS DONNEES PAR FICHER ? REPONDE OUI OU NON

—  
non  
EXECUTION BEGINS...  
...EXECUTION ENDED  
T=1.64/4.47 17.47.36

- Exemple de commande qui n'existe pas.

TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME

—  
bidule

THE FOLLOWING NAMES ARE UNDEFINED: BIDULE  
ERREUR : CETTE COMMANDE N'EXISTE PAS.  
RETAPEZ VOTRE COMMANDE

—  
fin

R; T=1.42/3.43 17.53.36

#### 2.2.2.2 Les commandes systèmes

Ce sont les 6 dernières du paragraphe 2.2.2. Elles se composent d'un mot clé suivi ou non de paramètres. Leur nom rappelle la fonction réalisée.

#### LISTE

Format:

LISTE

Fonction: Liste au Terminal toutes les commandes du sous-système, y compris celles rajoutées par l'utilisateur.

Exemple: TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME

—  
liste

LISTE DES COMMANDES DU SYSTEME

INTERSP2  
INTERSPQ  
AJUSP2  
AJUSPQ  
AJUSMDR  
TOMAN  
GRILLE  
SYSDIFRK  
BENSON  
LISTE  
LISTLIB  
CMS  
ADD  
DELETE  
FIN

LISTLIB

Format:

LISTLIB

Fonction: Liste au Terminal les noms des programmes et sous-programmes du sous-système, y compris ceux rajoutés par l'utilisateur.

Exemple: TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME

listlib

LISTE DES PROGRAMMES ET SOUS-PROGRAMMES DU SYSTEME

ENTRY	INDEX	SIZE
POINT	2	47
INIT1	49	18
INITI	67	20
AXES	87	26
EFACPT	113	26
ADDPT	139	30
SPL2IN	169	42

.  
.  
.  
TITRE 3644 49  
IMAGE 3693 119  
FIN  
QIR  
RETOUR  
COORDX  
COORDY

49 ENTRYS IN LIBE

LISTE DES PROGRAMMES ET SOUS-PROGRAMMES DE L'UTILISATEUR

ENTRY	INDEX	SIZE
PROG1	2	14
SP1	16	13
SP2	29	13

3 ENTRYS IN LIBE

CMS

Format:

CMS

Fonction: Permet à l'utilisateur de revenir dans l'environnement de CMS et de taper au terminal un ensemble de commandes CMS. (100 au maximum). Ce nombre peut être changé à l'initialisation du système.

Exemple:

TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME

```
—  
cms  
TAPEZ VOS COMMANDES CMS ( MAX=100 )  
POUR TERMINER TAPEZ: &GOTO - 71  
fortran prog1 sp1 sp2  
  COMPILING: SP1  
  COMPILING: SP2  
&goto -71
```

ADD

Format:

ADD &1 &2 ... &10

Cette commande peut avoir de 1 à 10 paramètres, Chacun d'eux représente le nom (6 caractères maximum) d'un fichier de type TEXT (voir [3]).

Fonction: Elle permet d'incorporer au système un programme fortran compilé et ses sous-programmes, et de créer une nouvelle commande.

-&1 est le nom du programme fortran que l'on veut ajouter au système et c'est aussi le nom de la nouvelle commande qui permettra d'appeler ce programme.

-&i, i=2 à 10, représentent les sous-programmes utilisés par le programme &1.

Exemple:

TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME

—  
add prog1 sp1 sp2  
DEFAULT TABS SET.  
EDIT:  
FIN  
INPUT:  
EDIT:

DELETE

Format:

DELETE &1 &2 ... &10

Cette commande peut avoir de 1 à 10 paramètres, chacun d'eux représente le nom (6 caractères maximum) d'un programme ou sous-programme du système.

Fonction: Réalise l'opération inverse de la commande ADD, c'est-à-dire supprime du système un programme et ses sous-programmes et supprime la commande correspondante.

- &1 est le nom du programme et de la commande
- &i, i=2 à 10 représente les sous-programmes.

Exemples:

TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME

—  
delete prog1 sp1 sp2  
WIPEOUT!!!  
DEFAULT TABS SET.  
EDIT:  
PROG1  
EQF:

FIN

Format:

FIN

Fonction: Restitue la machine de l'utilisateur dans l'état où elle se trouvait à l'initialisation du système, imprime éventuellement le fichier résultat de la session, et revient dans l'environnement de CMS.

Exemple:

TAPEZ LE NOM D'UNE COMMANDE DU SYSTEME

—

fin

R; T=2.38/4.49 06.57.51

### 2.2.3 Les programmes généraux

Ce sont des programmes conversationnels et interactifs, exclusivement écrits en Fortran IV. A chacun d'eux est associée une commande qui permet d'en demander l'exécution (cf.2.2.2.1). Le nom de la commande désigne ainsi le programme correspondant. Chacun de ces programmes généraux utilise largement les sous-programmes de GSP (cf.1.2.2.2) et ceux du sous-système.

Le principe de fonctionnement de chaque programme est le suivant:

- L'échange de messages avec l'utilisateur
- La réception des réponses à des questions
- La réception des interruptions provoquées par l'utilisateur

se font exclusivement par le terminal graphique (cf.1.1.2.1). Le programme visualise sur l'écran du 2250, des messages, des diagnostics ou des figures (axes, courbes, valeurs numériques). L'utilisateur répond ou agit sur le programme par l'intermédiaire du clavier de fonctions, du clavier alphanumérique ou du crayon optique. Ces éléments sont utilisés de la même manière par tous les programmes généraux.

### 2.2.3.1 Gestion de l'écran

L'écran permet d'afficher des points, des vecteurs, des lignes, des caractères (petits ou grands). Il est partagé en 4 zones.

- La zone titre: Elle reçoit la chaîne de caractères qui indique la nature du programme utilisé.

- La zone message/diagnostic.

Chaque fois que le programme a besoin d'informations, il envoie à l'utilisateur dans cette zone un message ou un diagnostic pour l'inviter à prendre une décision.

- La zone résultat.

A mesure que le programme se déroule, l'utilisateur peut voir dans cette zone s'afficher les résultats (axes, axes + courbes, valeurs numériques).

- La zone de lecture.

C'est dans cette zone que le programme vient lire la valeur numérique des paramètres dont il a besoin. La présence du curseur (cf.2.2.3.3), signale que l'utilisateur peut composer à l'aide du clavier alphanumérique, la valeur numérique du paramètre à lire.

La figure 5 donne l'emplacement de ces zones sur l'écran.



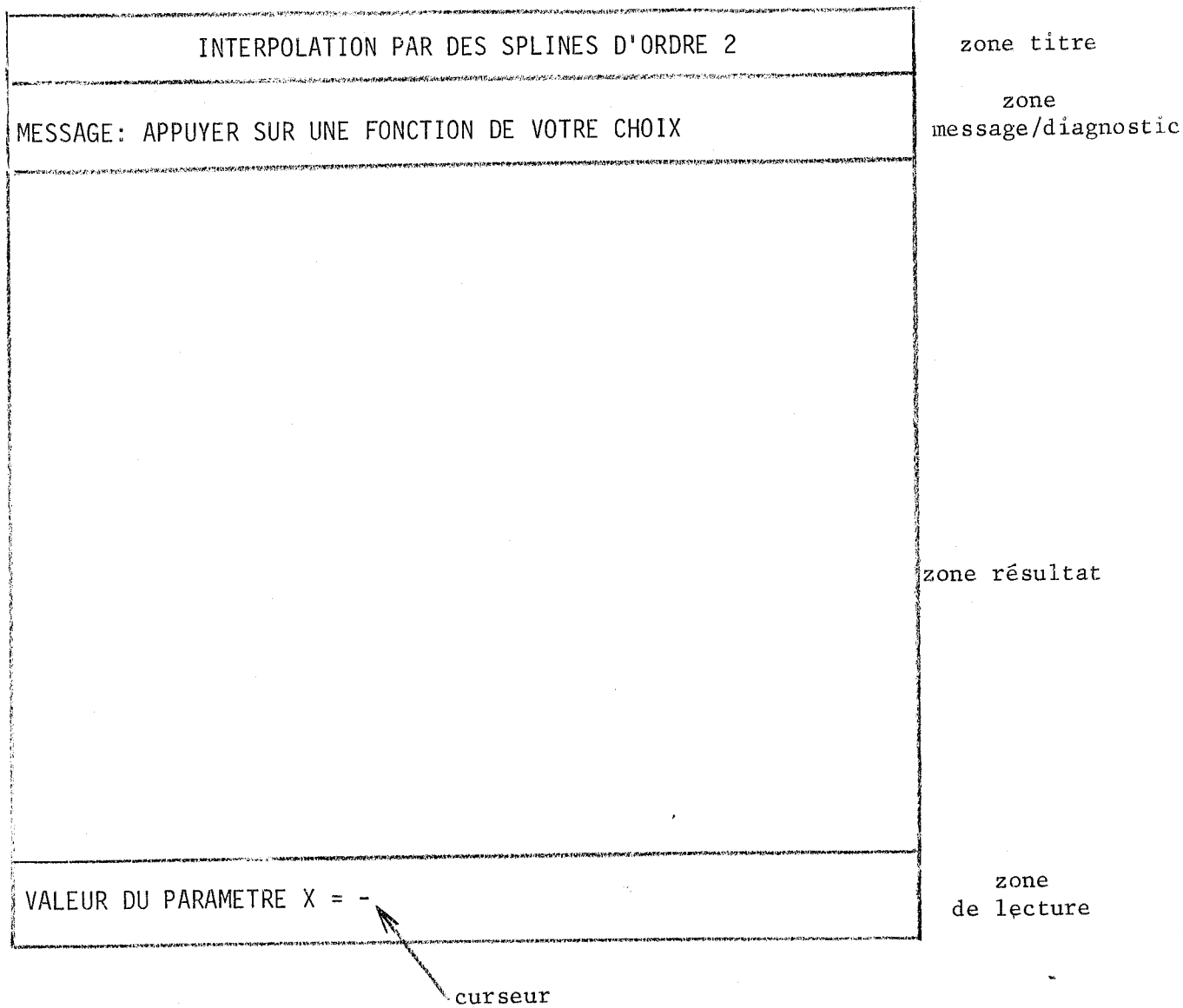


FIGURE 5 - Gestion de l'écran-

#### 2.2.3.2 Le crayon optique.

Il est utilisé dans deux phases différentes des programmes généraux.

- Dans la phase 'Entrée de données', il donne à l'utilisateur la possibilité de montrer un par un, les points qu'il a sélectionnés en pressant le crayon optique sur l'emplacement de l'écran correspondant.
- Dans la phase 'Modifier', il permet, en montrant un point donné, de l'effacer, le modifier, ou d'en ajouter un nouveau.

#### 2.2.3.3 Le clavier alphanumérique.

Il sert à composer dans la zone de lecture, lorsque le curseur est présent, la valeur numérique demandée par le programme. Deux touches spéciales 'Advance' et 'Back Space' font respectivement avancer ou reculer le curseur pour corriger les erreurs de frappe. En pressant simultanément sur les touches 'ALTN CODING' et 'END' le programme est averti qu'il peut récupérer la valeur numérique composée dans la zone de lecture. De cette manière, l'utilisateur peut agir sur son programme en modifiant la valeur numérique des paramètres et constater dans la zone résultat l'influence de ses modifications.

#### 2.2.3.4 Le clavier de fonctions.

Il se compose de 32 touches numérotées de 0 à 31 qui peuvent être allumées par programme. A chacune d'elles peut être associée une fonction (cf.1.1.2.1).

Nous nous sommes définis un ensemble de fonctions

aussi générales que possible, pouvant s'appliquer aussi bien aux programmes du système qu'à ceux de l'utilisateur. L'activation d'une fonction se fait en pressant la touche correspondante.

#### 2.2.4 Liste et description des fonctions

Elles sont au nombre de 19 (figure 6) et ont un rôle bien précis pour tous les programmes du système. Elles ne peuvent pas être modifiées par l'utilisateur. Par contre, celui-ci peut créer de nouvelles fonctions pour ses propres programmes et affecter à chacune d'elles une des touches laissées libres. Actuellement, nous avons les fonctions suivantes:

DEBUT

Lance l'exécution du programme en cours.

FIN

Termine l'exécution du programme en cours.

FICHER

Provoque la lecture des données du programme en cours, sur le fichier de données correspondant.

CRAYON

Permet l'utilisation du crayon optique pour afficher les données du programme en cours.

ACCEPTER

Est en relation directe avec la fonction précédente. Elle permet d'entrer les coordonnées du point montré sur l'écran par le crayon optique.

TERMINER

Supprime l'action de la fonction CRAYON.

CALCULER

Résoud, en fonction des données, le problème particulier lié au programme en cours.

RESULTATS

Affiche à l'écran du 2250 une partie des résultats numériques du programme en cours.

AVANCER ou RECULER

Sont en relation directe avec la fonction RESULTATS. Leur utilisation permet d'avoir accès à l'ensemble des résultats du calcul effectué.

FIG-COUR

Visualise la figure courante. On appelle figure courante, la représentation, sous forme de courbes ou lignes, des résultats du calcul effectué.

CONSERVER

Est à rapprocher de la fonction FIG-COUR précédente. Elle copie la figure courante dans une figure qu'on appelle PRECEDENTE. A l'initialisation du programme la figure précédente est vide.

FIG-PREC

Visualise la figure précédente. C'est-à-dire la superposition de l'ensemble des figures courantes que l'on a conservé.

HARD-COPY

Copie dans un fichier (FILE BENSON P) l'image actuelle de l'écran du 2250. On fera le rapprochement avec la commande BENSON (cf.2.2.2).

IMPRIMER

Ecrit dans un fichier destiné à l'imprimante les résultats numériques du calcul effectué. L'ensemble des résultats est automatiquement listé en fin de session.

MODIFIER

Permet à l'utilisateur de modifier grâce aux 3 fonctions qui suivent, les coordonnées d'un point de l'écran.

EFFACER-PT

Efface le point de l'écran montré par le crayon optique et supprime des données les coordonnées correspondantes.

AJOUTER-PT

Visualise le point de l'écran montré par le crayon optique et ajoute aux données les coordonnées correspondantes.

MODIFIER-PT

Permet de modifier, grâce au clavier alphanumérique, les coordonnées du point de l'écran montré par le crayon optique.

2.2.4.1 Gestion et enchaînement des fonctions.

Chaque fonction a une signification unique, quel que soit le programme qui l'utilise et chaque programme peut faire appel à toutes ou à une partie de ces fonctions.

Leur enchaînement est hiérarchisé. Certaines fonctions se situent dans un même niveau de la hiérarchie et conduisent d'une phase des traitements à une autre. D'autres, par contre, conduisent d'un niveau de la hiérarchie à un autre niveau, soit par ascendance soit par descendance (cf.figure 7).

Pour l'utilisateur, l'emploi de ces fonctions est simplifié à l'extrême puisque à chaque étape du déroulement de son programme, seules les touches liées aux fonctions autorisées sont allumées.

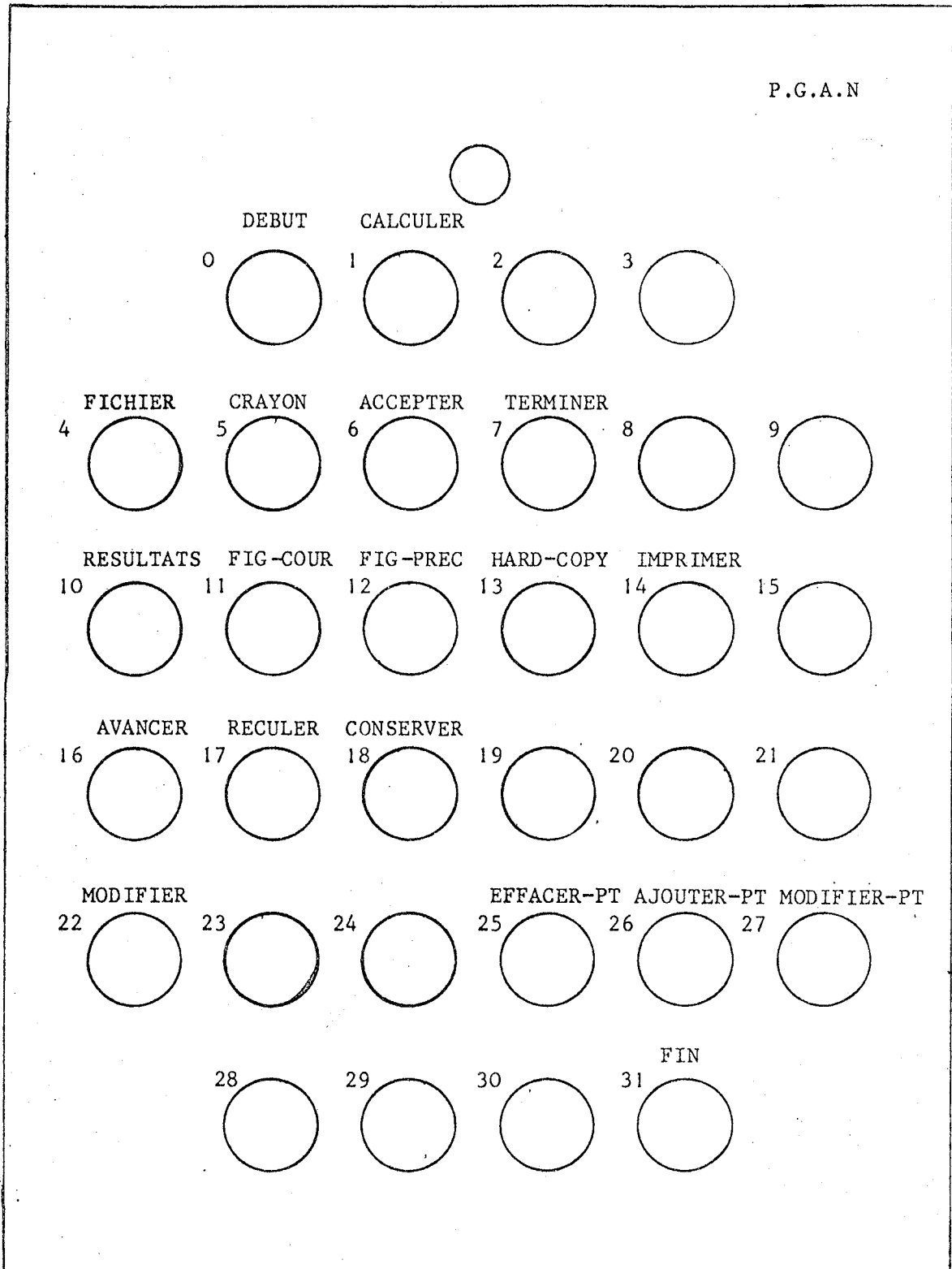


FIGURE 6 -Clavier de fonctions-

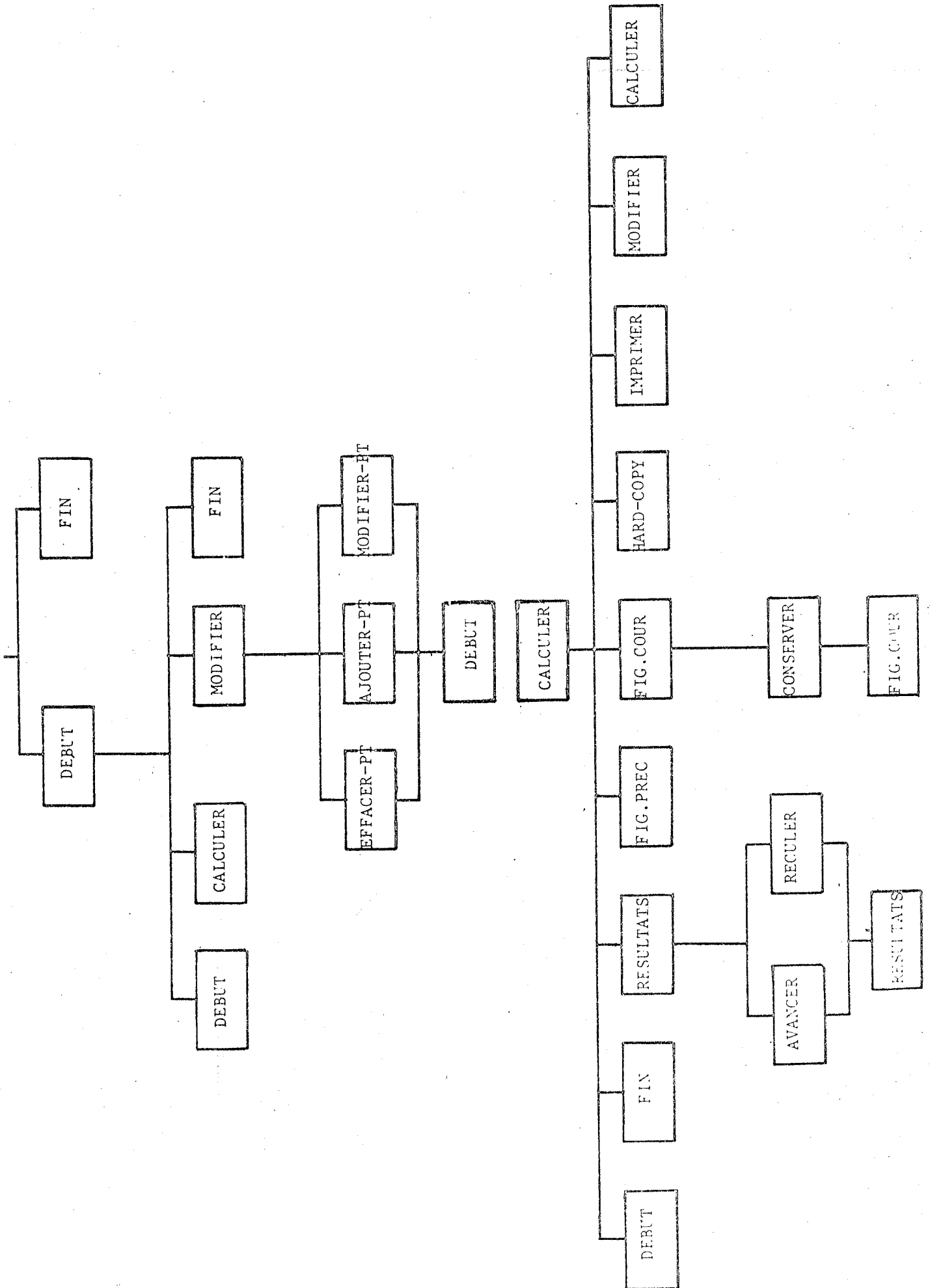


FIGURE 7 - Enchaînement et hiérarchie des fonctions -

### 2.2.5. Entrée des données pour les programmes généraux.

Elle est contrôlée par les deux fonctions FICHER et CRAYON.

La première, FICHER, effectue la lecture des données dans un fichier créé sur disque par l'utilisateur et qui doit avoir les caractéristiques CMS suivantes:

FILENAME	FILETYPE	MODE
FILE	nom de la commande	P1

Exemple: Le fichier donné pour le programme INTERSP2 doit être créé comme suit:

FILE	INTERSP2	P1
------	----------	----

Ces fichiers ne contiennent que des valeurs entières de format fortran 'I' et des valeurs réelles de format '5E15.7'.

La deuxième, CRAYON, visualise et permet de récupérer les coordonnées des points de l'écran montrés par le crayon optique.

REMARQUE: Certains programmes n'utilisent pas ce mécanisme, ce sont les programmes TOMAN, GRILLE, BENSON, SYSDIFRK. (Les données sont trop ou pas assez nombreuses).

Enfin, pour tous les programmes, les paramètres nécessaires sont transmis aux programmes par l'intermédiaire du clavier alphanumérique.

#### 2.2.5.1 Description du contenu des fichiers.

Chaque fichier varie d'un programme à l'autre. Nous donnerons pour chacun des programmes, le contenu détaillé du fichier qui lui est associé.

- Les programmes INTERSP2, INTERSPQ, AJUSP2, AJUSPQ, AJUSMDR



opèrent sur des données analogues. Les fichiers correspondants doivent contenir dans cet ordre:

un nombre entier N, un tableau X, un tableau Y.

N est le nombre de points traités, il doit être écrit avec un format fortran: FORMAT (I2).

X et Y sont des tableaux de dimension N, ils contiennent les coordonnées des points traités et doivent être écrits avec un format fortran: FORMAT (5E15.7).

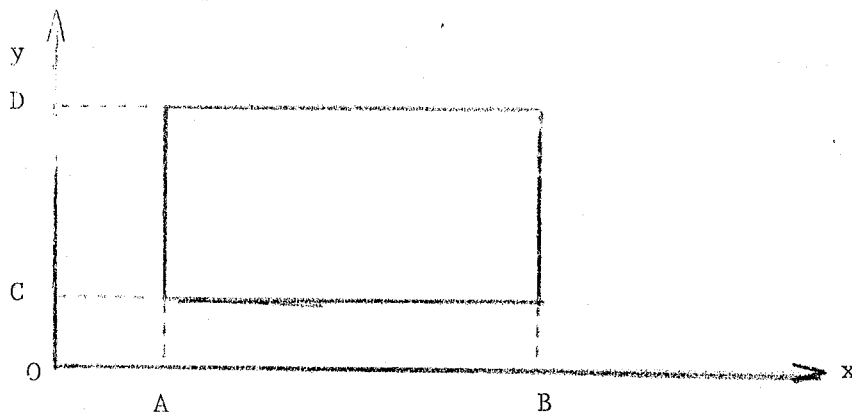
Exemple:

6  
0.1250000E 00 0.3750000E 00 0.6250000E 00 0.8750000E 00 0.1125000E 01  
0.1375000E  
0.7801341E 02 0.7622451E 02 0.7313470E 02 0.6799371E 02 0.5935732E 02  
0.4566502E 02

- Le programme TOMAN opère sur un fichier qui doit contenir dans cet ordre:

quatre nombres (A,B,C,D), quatre nombres entiers (L,N,M, MAX), un tableau U, un tableau V, un tableau ALFA, un tableau X, et un tableau Y.

A, B, C, D doivent être écrits avec un format fortran: FORMAT (4E15.7). Ce sont les abscisses et ordonnées d'un domaine rectangulaire:



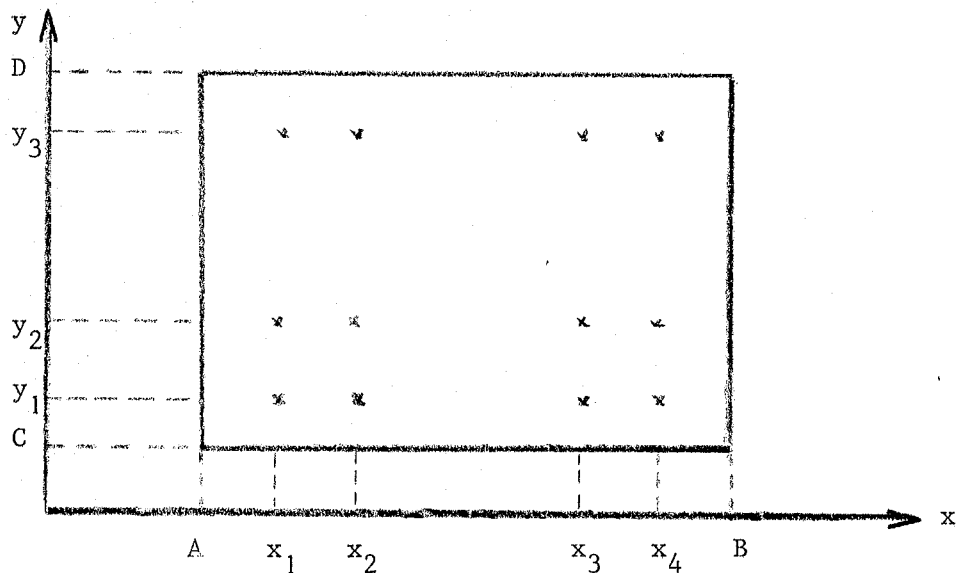
L,N,M,MAX doivent être écrits avec un format fortran:

FORMAT (4I3). MAX indique le nombre de termes à sommer pour une série simple, on prendra:  $30 \leq \text{MAX} \leq 300$ .

U, V, ALFA sont des tableaux de dimension L, ils doivent être écrits avec un format fortran: FORMAT(5E15.7). Ils sont remplis de la manière suivante:

U et V contiennent les valeurs des coordonnées de L points situés à l'intérieur du domaine (A,B,C,D) où sont connues les valeurs de la fonctions spline d'interpolation à deux variables. Ces valeurs sont dans le tableau ALFA.

X et Y sont des tableaux de dimension N et M respectivement. Ils doivent être écrits avec un format fortran: FORMAT(5E15.7). Ils contiennent respectivement les abscisses x et les ordonnées y des points du quadrillage, pas forcément régulier, où l'on veut calculer la valeur de la fonction spline d'interpolation à deux variables.



## Exemple:

```

-0.6000000E 03  0.6000000E 03  -0.6000000E 03  0.6000000E 03
16 11  9 30
-0.2142699E 03  0.1393000E 03  0.3929999E 01  0.2400699E 03 -0.2522899E 03
 0.3987397E 03  0.2100200E 03  0.1688998E 03  0.1080300E 03 -0.2533998E 02
-0.9077998E 02 -0.2793198E 03  -0.4935498E 03  -0.3212097E 03 -0.3576099E 03
 0.2911199E 03
-0.1305000E 02  0.3370999E 03  -0.1604900E 03  -0.2737898E 03 -0.1666400E 03
 0.5309999E 01 -0.1219500E 03  0.3752998E 02  0.1773600E 03 -0.3107999E 02
 0.2524699E 03  0.1260100E 03  0.1669998E 02  -0.6049999E 01  0.3357397E 03
 0.4122898E 03
-0.1000000E 01 -0.2239999E 01  -0.1089999E 01  -0.1149999E 01 -0.8099999E 00
-0.6799999E 00 -0.1429999E 01  -0.1649999E 01  -0.1770000E 01 -0.1030000E 01
-0.1299999E 01 -0.1109999E 01  -0.7699999E 00  -0.1229999E 01 -0.1020000E 01
00.1549999E 01
-0.5000000E 03 -0.4000000E 03  -0.3000000E 03  -0.2000000E 03 -0.1000000E 03
 0.0 0.1000000E 03  0.2000000E 03  0.3000000E 03  0.4000000E 03
 0.5000000E 03
 0.4000000E 03 -0.3000000E 03  -0.2000000E 03  -0.1000000E 03  0.0
 0.1000000E 03  0.2000000E 03  0.3000000E 03  0.4000000E 03

```

- Le programme GRILLE opère sur le fichier créé par le programme TOMAN ou sur un fichier qui doit contenir dans cet ordre:

Deux nombres entiers (N,M), un tableau X, un tableau Y et un tableau Z.

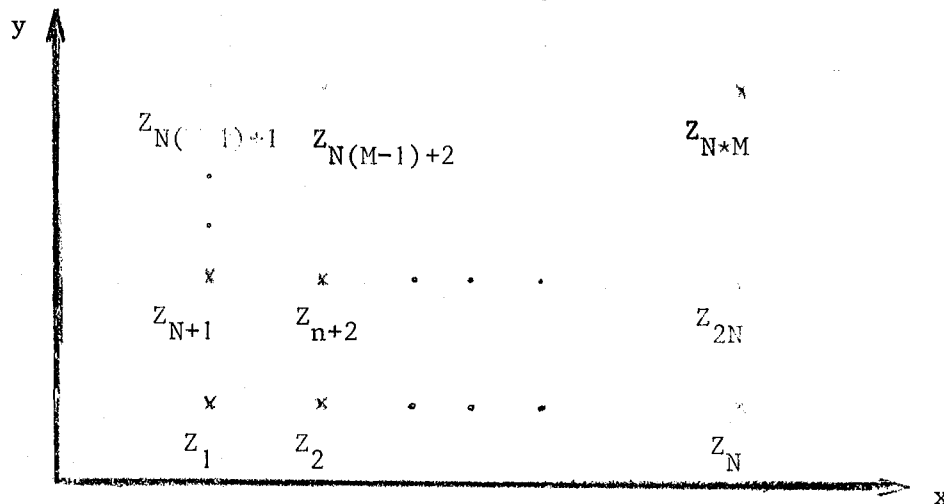
N et M doivent être écrits avec un format fortran: FORMAT(2I2).

X et Y sont des tableaux de dimension N et M respectivement.

Ils doivent être écrits avec un format fortran: FORMAT(5E15.7).

Ils contiennent respectivement les abscisses x et les ordonnées y des points du quadrillage, pas forcément régulier.

Le tableau Z, de dimension N\*M, doit être écrit avec un format fortran: FORMAT(5E15.7), Z(I) est la valeur au point du quadrillage de coordonnées (X(I), Y(I)). Z est rempli dans l'ordre indiqué par le graphique:



Exemple:

8

0.1250000E-01	0.3750000E-01	0.6250000E-01	0.8750000E-01	0.1125000E+00
0.1375000E+00				
0.3125000E-02	0.9375000E-02	0.1562500E-01	0.2187500E-01	0.2812500E-01
0.3437500E-01	0.4062500E-01	0.4687500E-01		
0.7801341E+C1	0.7622451E+01	0.7313470E+01	0.6799371E+01	0.5935732E+01
0.4566502E+01	0.2342937E+02	0.2288508E+02	0.2196306E+02	0.2043172E+02
0.1785621E+02	0.1374011E+02	0.3913626E+02	0.3820126E+02	0.3668111E+02
0.3416532E+02	0.2992564E+02	0.2304271E+02	0.5498700E+02	0.5361122E+02
0.5151828E+02	0.4807218E+02	0.4225234E+02	0.3258970E+02	0.7108375E+02
0.6918328E+02	0.6655378E+02	0.6225181E+02	0.5497725E+02	0.4256921E+02
0.8767805E+02	0.8513637E+02	0.8201724E+02	0.7695195E+02	0.6839339E+02
0.5339405E+02	0.1058403E+03	0.1025549E+03	0.9900287E+02	0.9328165E+02
0.8365422E+02	0.6643918E+02	0.1176836E+03	0.1142638E+03	0.1102939E+03
0.1039686E+03	0.9346522E+02	0.7465489E+02		

- Le programme SYSDIF1 ne lit ses données que par l'intermédiaire du clavier alphanumérique.
- Le programme BENSON lit ses données dans un fichier qui est créé ou complété par la fonction HARD-COPY.
- Les programmes de l'utilisateur doivent lire leurs données sur un fichier: 'FILE nom du programme P1', c'est la seule restriction.

### 2.2.6 Les sous-programmes de service.

Le nombre important de ces sous-programmes nous interdit d'en donner la liste. La plupart sont écrits en Fortran, les autres en Assembleur 360. Nous indiquerons brièvement la nature des services rendus:

- Initialisation du 2250.
- Gestion des Entrées-Sorties au 2250.
- Manipulation des points affichés au 2250.
- Evaluation des algorithmes des méthodes mathématiques utilisées.

### 2.3 QUELQUES DETAILS DE PROGRAMMATION.

Ce paragraphe a pour but d'indiquer le nombre et le contenu des divers fichiers du système, comment sont traitées les erreurs et que faire lorsque le sous-système se 'casse'.

- a- Le sous-système comporte 5 fichiers dont les caractéristiques CMS sont:

Filename	Filetype	Mode
PGAN	EXEC	P1
PGANLIB	TXTLIB	P1
PGANLIB1	TXTLIB	P1
PGAN	LISTE	P1
SAVEPGAN	EXEC	P1

- Le premier de ces fichiers contient le programme de contrôle.
  - Le deuxième contient tous les programmes généraux et les sous-programmes du sous-système.
  - Le troisième contient les programmes et sous-programmes que l'utilisateur veut ajouter au sous-système.
  - Le quatrième contient la liste des commandes du système.
  - Le cinquième sert à sauvegarder les fichiers de l'utilisateur et ceux du sous-système lorsque celui-ci se 'casse'.
- b- Le traitement des erreurs est préventif et dynamique. En effet, dès que l'utilisateur émet une commande le programme de contrôle teste la validité de celle-ci. Si elle n'est pas conforme, un diagnostic apparaît au terminal. Si la valeur d'un paramètre est incorrecte, le programme général visualise dans la zone de l'écran prévue à cet effet, un message d'erreur. Dans les deux cas, l'utilisateur doit, soit redonner sa commande, soit corriger la valeur du paramètre.
- c- Un certain nombre de fichiers est nécessaire au bon fonctionnement du sous-système, si pour une raison quelconque, le système vient à se 'casser', il faut restaurer ces fichiers avant de redonner le contrôle au sous-système. Une commande: SAVEPGAN, indépendante de celui-ci, réalise cette opération lorsqu'elle est émise depuis le Terminal. Elle remet en état les fichiers du sous-système et éventuellement ceux de l'utilisateur qui ont pu être perturbés.



## C H A P I T R E 3

### METHODES MATHÉMATIQUES

#### 3.1 INTRODUCTION.

Ce chapitre regroupe l'ensemble des méthodes mathématiques programmées dans les divers programmes de ce système.

L'utilisation croissante des fonctions splines dans différents domaines de l'analyse numérique et en particulier dans celui de l'approximation, nous a conduit à leur donner ici un relief particulièrement important. En effet, l'approximation polynomiale ordinaire ne conduit pas toujours à des résultats satisfaisants notamment lorsque le degré des polynômes est élevé. Par contre, l'utilisation des fonctions splines évite cet inconvénient. C'est pourquoi, nous trouverons dans ce système et ci-après deux aspects importants de l'approximation: l'interpolation et l'ajustement par des fonctions splines et ceci, pour différents ordres. Pour les méthodes classiques, seul l'ajustement au sens des moindres carrés est abordé.

Enfin, nous traitons également deux autres problèmes de l'analyse numérique: La recherche de courbes de niveau sur un domaine rectangulaire, toujours à l'aide de fonctions splines et la résolution de systèmes différentiels du premier ordre.

#### 3.2 INTERPOLATION.

Soient  $n$  abscisses distinctes appartenant à l'intervalle  $[a,b]$



telles que :

$$a < x_1 < x_2 < \dots < x_n < b$$

et  $n$  nombres réels  $y_i$ ,  $i = 1 \dots n$ . En général les valeurs  $y_i$  représentent les valeurs en  $x_i$  d'une certaine fonction  $f$ .

L'interpolation consiste à substituer à  $f$ , une fonction  $\tilde{f}$ , qui, pour la valeur  $x_i$ , prenne la valeur  $y_i$ ,  $i = 1 \dots n$ . On peut chercher comme fonction  $\tilde{f}$  un polynôme dont le degré ne dépasse pas  $n-1$ , c'est alors l'interpolation polynomiale classique. Le remplacement de  $f$  par  $\tilde{f}$  permet ainsi :

- si  $f$  est connue mais d'expression analytique compliquée, de lui substituer une fonction aisément calculable.  
(Schéma de Horner si  $\tilde{f}$  est un polynôme).
- si  $f_n$  est mesurée pour les abscisses  $x_i, i=1 \dots n$ , de permettre une évaluation approchée de  $f$  pour  $x \neq x_i$ .

Cette méthode peut présenter de graves inconvénients: il est toujours difficile de manipuler numériquement des polynômes de degré élevé et le polynôme aura tendance à osciller de façon anormale entre les abscisses  $x_i$ . Pour remédier à cette difficulté, on préfère subdiviser l'intervalle  $[a,b]$  en  $p$  intervalles et dans chacun de ces  $p$  intervalles, pratiquer une interpolation polynomiale de faible degré (2,3,4,...): c'est l'interpolation polynomiale par morceaux. Une difficulté subsiste, on ne voit pas comment vont se raccorder les différents morceaux. Pour lever cette dernière difficulté, on peut utiliser l'interpolation par des fonctions splines [ 9 ]. C'est en fait une interpolation polynomiale par morceaux, mais avec raccordement des dérivées entre les divers morceaux. On veut dire par là que non seulement la fonction interpolante est continue sur  $[a,b]$  mais un certain nombre de ses dérivées le sont aussi.

On étudiera le cas où les morceaux de polynômes seront de degré  $q$  ( $q = 2,3,4,5,6$ ). On donnera un algorithme particulier dans le cas où  $q = 2$ .

### 3.2.1 Spline d'interpolation d'ordre q.

Soient  $n$  abscisses fixes et distinctes appartenant à l'intervalle  $[a, b]$  telles que

$$a < x_1 < x_2 < \dots < x_n < b$$

avec  $n \geq q$  et  $n$  nombres réels  $y_i$ ,  $i = 1 \dots n$ .

On montre [10] que parmi toutes les fonctions de  $H^q[a, b]$ , il en existe une et une seule, notée  $\sigma$  qui vérifie:

$$\sigma(x_i) = y_i \quad i = 1 \dots n$$

et

$$\int_a^b [\sigma^{(q)}(x)]^2 dx \leq \int_a^b [f^{(q)}(x)]^2 dx \quad \text{pour tout } f \in H^q[a, b] \\ \text{vérifiant } f(x_i) = y_i \quad i=1 \dots n.$$

$\sigma$  est appelée fonction spline d'interpolation d'ordre q et est caractérisée par les propriétés suivantes:

P1:  $\sigma$  est un polynôme de degré  $\leq 2q-1$  sur chaque intervalle  $]x_i, x_{i+1}[$ ,  $i=1 \dots (n-1)$ .

P2:  $\sigma$  est un polynôme de degré  $\leq q-1$  sur chacun des intervalles  $[a, x_1[$  et  $]x_n, b]$ .

P3: En chacun des points  $x_i$ ,  $i=1 \dots n$ , on a:

$$\sigma(x_i) = y_i$$

P4: La fonction spline  $\sigma$  et ses dérivées jusqu'à l'ordre  $2q-2$  inclus, sont continues sur  $[a, b]$ , c'est-à-dire:

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

Le problème est donc de déterminer dans chaque intervalle la fonction spline  $\sigma$ : son expression est un polynôme de degré  $2q-1$  dont il faut déterminer les  $2q$  coefficients.

REMARQUE 1: Un polynôme de degré  $2q-1$  est entièrement déterminé par sa valeur en un point et les valeurs de ses  $2q-1$  premières dérivées. On peut prendre les  $2q$  inconnues suivantes:

$$\sigma(x), \sigma'(x), \sigma''(x), \dots, \sigma^{(2q-1)}(x)$$

introduisons  $Z(x)$  le vecteur suivant:

$$Z(x) = \begin{pmatrix} \sigma(x) \\ \sigma'(x) \\ \vdots \\ \sigma^{(2q-1)}(x) \end{pmatrix}$$

### 3.2.1.1 Propriétés.

P5: Si l'on connaît  $q$  relations linéaires (indépendantes de la relation  $\sigma(x_i) = y_i$ ) vérifiées par  $Z(x_i^-)$

$$M(x_i^-) \cdot Z(x_i^-) = b(x_i^-) \quad (3.1)$$

où  $M(x_i^-)$  est une matrice  $(q, 2q)$  et  $b(x_i^-)$  un vecteur  $(q, 1)$ , alors on peut trouver  $q$  relations linéaires du même type vérifiées par  $Z(x_i^+)$ .

#### Démonstration

$\sigma^{(2q-1)}(x)$  n'étant pas continue en  $x_i$ , l'expression (3.1) n'est pas vérifiée en  $x_i^+$ . En éliminant l'inconnue  $\sigma^{(2q-1)}(x_i^-)$  entre les  $q$  relations données par (3.1), on obtient

q-1 nouvelles relations linéaires entre les q-1 premières inconnues du vecteur  $Z(x_i^-)$ . Du fait de la continuité (P4) ces q-1 relations sont aussi vérifiées en  $x_i^+$ . On prend comme q<sup>ième</sup> relation en  $x_i^+$  :

$$\sigma(x_i) = y_i \quad (P3)$$

Ces q relations se notent sous forme matricielle par :

$$M(x_i^+) \cdot Z(x_i^+) = b(x_i^+) \quad (3.2)$$

avec  $M(x_i^+)$  et  $b(x_i^+)$  de la forme :

$$M(x_i^+) = \begin{array}{cccccc} & & & & 2q & \\ & & & & & \\ * & * & \dots & * & 0 & \\ * & * & \dots & * & 0 & \\ \cdot & \cdot & & \cdot & \cdot & \\ \cdot & \cdot & & \cdot & \cdot & \\ \cdot & \cdot & & \cdot & \cdot & \\ * & * & \dots & * & 0 & \\ 1 & 0 & \dots & 0 & 0 & \end{array} \left. \vphantom{\begin{array}{cccccc} * & * & \dots & * & 0 & \\ * & * & \dots & * & 0 & \\ \cdot & \cdot & & \cdot & \cdot & \\ \cdot & \cdot & & \cdot & \cdot & \\ \cdot & \cdot & & \cdot & \cdot & \\ * & * & \dots & * & 0 & \\ 1 & 0 & \dots & 0 & 0 & \end{array}} \right\} q$$

$$b(x_i^+) = \begin{array}{c} * \\ * \\ \\ \\ * \\ y_i \end{array} \left. \vphantom{\begin{array}{c} * \\ * \\ \\ \\ * \\ y_i \end{array}} \right\} 2q$$

(3.2) a bien la forme de (3.1).

P6: Si dans un intervalle  $]x_i, x_{i+1}[$ , on connaît q relations linéaires qui se notent sous forme matricielle

$$M(x_i^+) \cdot Z(x_i^+) = b(x_i^+) \quad (3.3)$$

il existe en  $x_{i+1}^-$ , q relations linéaires analogues.

Démonstration

Posons  $h_i = x_{i+1}^- - x_i^+$  et utilisant l'identité de Taylor pour  $F$ , polynôme de degré  $\leq 2q-1$ .

$$F(b) = F(a) + (b-a) F'(a) + \dots + \frac{(b-a)^{2q-1}}{(2q-1)!} F^{(2q-1)}(a)$$

on peut écrire en  $x_{i+1}^-$  les relations suivantes:

$$\sigma(x_{i+1}^-) = \sigma(x_i^+) + h_i \sigma'(x_i^+) + \dots + \frac{h_i^{2q-1}}{(2q-1)!} \sigma^{(2q-1)}(x_i^+)$$

$$\sigma'(x_{i+1}^-) = \sigma'(x_i^+) + h_i \sigma''(x_i^+) + \dots + \frac{h_i^{2q-2}}{(2q-2)!} \sigma^{(2q-2)}(x_i^+)$$

.

.

.

$$\sigma^{(2q-1)}(x_{i+1}^-) = \sigma^{(2q-1)}(x_i^+)$$

relations qui s'écrivent sous forme matricielle:

$$Z(x_{i+1}^-) = T(h_i) \cdot Z(x_i^+) \tag{3.4}$$

avec

$$T(h_i) = \begin{pmatrix} 1 & h_i & \frac{h_i^2}{2!} & \dots & \frac{h_i^{2q-1}}{(2q-1)!} \\ 0 & 1 & h_i & \dots & \frac{h_i^{2q-2}}{(2q-2)!} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdot & \cdot & 1 \end{pmatrix}$$

en écrivant l'identité de Taylor à partir de  $x_{i+1}^-$  on obtient

$$Z(x_i^+) = T(-h_i) \cdot Z(x_{i+1}^-) \quad (3.5)$$

en portant (3.5) dans (3.3) on obtient

$$M(x_i^+) \cdot T(-h_i) \cdot Z(x_{i+1}^-) = b(x_i^+)$$

en posant

$$M(x_{i+1}^-) = M(x_i^+) \cdot T(-h_i)$$

$$b(x_{i+1}^-) = b(x_i^+)$$

on obtient

$$M(x_{i+1}^-) \cdot Z(x_{i+1}^-) = b(x_{i+1}^-) \quad (3.6)$$

forme qui est analogue à (3.3).

### 3.2.1.2 Mise en oeuvre de l'algorithme.

Le but de cette méthode est d'évaluer, pour un  $x$  donné, appartenant à l'intervalle  $[a, b]$ , la valeur correspondante de la fonction spline d'interpolation d'ordre  $q$ . Pour calculer cette valeur, il faut connaître dans chaque intervalle  $] x_i, x_{i+1} [$   $i = 1 \dots n$ , le polynôme de degré  $\leq 2q-1$ .

REMARQUE 2: Un polynôme de degré  $\leq 2q-1$  est entièrement déterminé si on connaît deux valeurs de ce polynôme en deux points distincts, et les  $(q-1)$  valeurs de ses  $(q-1)$  premières dérivées en ces mêmes points.

Cette remarque va éclairer la méthode utilisée. On connaît par la propriété 3, la valeur du polynôme en tous les points  $x_i$ ,  $i = 1 \dots n$ . Reste à déterminer la valeur des  $(q-1)$  premières

dérivées en tous les points  $x_i$ ,  $i = 1 \dots n$ .

Le calcul de  $\sigma$  se fait en deux étapes:

- 1) Evaluation des  $q-1$  premières dérivées aux points  $x_i$ ,  $i = 1 \dots n$ .
- 2) Calcul de la valeur du polynôme.

### 3.2.1.2.1 Evaluation des $q-1$ premières dérivées.

Nous prenons comme inconnu le vecteur

$$Z(x) = \begin{pmatrix} \sigma(x) \\ \sigma'(x) \\ \cdot \\ \cdot \\ \cdot \\ (2q-1) \\ \sigma(x) \end{pmatrix}$$

Pour calculer  $Z(x)$  et en particulier les  $(q-1)$  premières dérivées  $\sigma^{(i)}(x)$ ,  $i = 1 \dots (q-1)$ , il faut connaître en tous points  $x_i$ ,  $i = 1 \dots n$ ,  $2q$  relations linéaires entre ces  $2q$  inconnues. La détermination de ces  $2q$  relations va se faire en deux étapes.

①

Dans une première phase, en balayant l'intervalle  $[a, b]$  de gauche à droite, on détermine en chaque point  $x_i^+$ ,  $i = 1 \dots n$   $q$  relations linéaires entre les inconnues:

$$\sigma(x_i^+), \sigma'(x_i^+), \dots, \sigma^{(2q-1)}(x_i^+)$$

Initialisation

En  $x_1^+$  on écrit sous forme matricielle les propriétés 3 et 4 qui expriment, en tenant compte de la propriété 2, la continuité des dérivées. On obtient les relations linéaires:

$$M(x_1^+) \cdot Z(x_1^+) = b(x_1^+)$$

où  $M(x_1^+)$  est une matrice  $(q, 2q)$  et  $b(x_1^+)$  un vecteur  $(q, 1)$ . Ce système est de la forme:

$$\begin{pmatrix} 0 & 0 & \dots & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 0 & 1 & \dots & 0 \\ & & & & & & & \\ 0 & 0 & \dots & 0 & 0 & \dots & 1 & 0 \\ 1 & 0 & \dots & 0 & 0 & \dots & 0 & \end{pmatrix} \times \begin{pmatrix} \sigma(x_1^+) \\ \sigma'(x_1^+) \\ \\ \sigma^{(2q-2)}(x_1^+) \\ \sigma^{(2q-1)}(x_1^+) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \\ 0 \\ y_1 \end{pmatrix}$$

- on passe au point  $x_2^-$  en utilisant la propriété 6, qui nous donne

$$M(x_2^-) \cdot Z(x_2^-) = b(x_2^-)$$

avec

$$M(x_2^-) = M(x_1^+) \cdot T(-h_1)$$

$$b(x_2^-) = b(x_1^+)$$

Remarque: La forme de  $T(-h_1)$  est donnée dans la démonstration de la propriété 6.



- On passe au point  $x_2^+$  en utilisant la propriété 5, qui nous donne le système linéaire

$$M(x_2^+) \cdot Z(x_2^+) = b(x_2^+)$$

où  $M(x_2^+)$  et  $b(x_2^+)$  sont de la forme:

$$M(x_2^+) = \begin{array}{cccc|c} * & * & \dots & * & 0 \\ & & & & 0 \\ \cdot & \cdot & & \cdot & \cdot \\ \cdot & \cdot & & \cdot & \cdot \\ * & * & \dots & * & 0 \\ 1 & 0 & \dots & 0 & 0 \end{array} \quad b(x_2^+) = \begin{array}{c|c} * \\ * \\ \cdot \\ \cdot \\ * \\ y_2 \end{array} \quad (3.7)$$

- On passe de  $x_2^+$  en  $x_3^+$  comme l'on est passé de  $x_1^+$  en  $x_2^+$ , et on applique cette stratégie jusqu'au point  $x_n^+$ .

②

Dans une deuxième phase, en balayant l'intervalle  $[ a, b ]$  de droite à gauche, on détermine en chaque point  $x_i^-$ ,  $i = n, (n-1) \dots 1$  q nouvelles relations linéaires entre les  $2q$  inconnues:

$$\sigma(x_i^-), \sigma'(x_i^-), \dots, \sigma^{(2q-1)}(x_i^-)$$

Initialisation

En  $x_n^-$  on écrit sous forme matricielle les propriétés 3 et 4 qui expriment, en tenant compte

de la propriété 2, la continuité des dérivées. On obtient les relations linéaires:

$$M(x_n^-).Z(x_n^-) = b(x_n^-)$$

avec  $M(x_n^-) = M(x_1^+)$  et  $b(x_n^-) = b(x_1^+)$  (en remplaçant  $y_1$  par  $y_n$ ).

- On passe au point  $x_{n-1}^+$  en utilisant la propriété 6 qui nous donne:

$$M(x_{n-1}^+).Z(x_{n-1}^+) = b(x_{n-1}^+)$$

avec

$$M(x_{n-1}^+) = M(x_n^-).T(h_n)$$

$$b(x_{n-1}^+) = b(x_n^-)$$

- On passe au point  $x_{n-1}^-$  en utilisant la propriété 5 qui nous donne

$$M(x_{n-1}^-).Z(x_{n-1}^-) = b(x_{n-1}^-)$$

où  $M(x_{n-1}^-)$  et  $b(x_{n-1}^-)$  ont une forme analogue à (3.7).

- On passe du point  $x_{n-1}^-$  au point  $x_{n-2}^-$  comme l'on est passé de  $x_n^-$  en  $x_{n-1}^-$  et on applique cette stratégie jusqu'au point  $x_1^-$ .

En résumé on obtient en chaque point  $x_i^-$  et  $x_i^+$ ,  $i = 1 \dots n$ ,  $q$  relations linéaires qui proviennent

de ①, et q relations linéaires qui proviennent de ②, respectivement. Au total, 2q relations linéaires entre 2q inconnues. De ces 2q relations linéaires, on peut tirer aisément 2q-2 relations linéaires vérifiées par les 2q-2 inconnues suivantes:

$$\sigma'(x_i), \sigma''(x_i), \dots, \sigma^{(2q-2)}(x_i)$$

Les inconnues sont calculées en résolvant ce système linéaire de 2q-2 équations à 2q-2 inconnues à l'aide de la méthode de Gauss.

Cette méthode a donné naissance d'une part aux commandes INTERSP2 et INTERSPQ du système PGAN (Cf.2.2.2) et d'autre part aux deux procédures Algol SPL2IN et SPLQIN décrites au paragraphe (3.2.1.3).

Remarque. L'extrême simplicité du cas q = 2, qui conduit à un système linéaire de 2 équations à 2 inconnues, a été traité séparément.

### 3.2.1.2.2 Calcul de la valeur du polynôme

Pour calculer la valeur du polynôme, on utilise une méthode de Lagrange-Hermite à argument répété sur deux points (Schéma de Aitken-Neville).

#### Rappel de la méthode

Si en deux points  $x_i$  et  $x_{i+1}$ , on connaît les valeurs  $P_i$  et  $P_{i+1}$  du polynôme ainsi que les valeurs de ses q-1 premières dérivées en ces mêmes points et si on construit le tableau suivant (ici q = 3).

$$\begin{array}{rcccc}
 x_i & P_i & & & \\
 & & P'_i & & \\
 x_i & P_i & & \frac{P''_i}{2!} & \\
 & & P'_i & D & \\
 x_i & P_i & & B & G \\
 & & A & E & I \\
 x_{i+1} & P_{i+1} & & C & H \\
 & & P'_{i+1} & F & \\
 x_{i+1} & P_{i+1} & & \frac{P''_{i+1}}{2!} & \\
 & & P'_{i+1} & & \\
 x_{i+1} & P_{i+1} & & & 
 \end{array}$$

avec

$$\begin{aligned}
 A &= \frac{P_{i+1} - P_i}{x_{i+1} - x_i}, & B &= \frac{A - P'_i}{x_{i+1} - x_i}, & C &= \frac{P'_{i+1} - A}{x_{i+1} - x_i}, \\
 D &= \frac{B - \frac{P''_i}{2}}{x_{i+1} - x_i}, & E &= \frac{C - B}{x_{i+1} - x_i}, & F &= \frac{\frac{P''_{i+1}}{2} - C}{x_{i+1} - x_i} \\
 G &= \frac{E - D}{x_{i+1} - x_i}, & H &= \frac{F - E}{x_{i+1} - x_i}, & I &= \frac{H - G}{x_{i+1} - x_i}
 \end{aligned}$$

On montre [11] que le polynôme cherché s'écrit sous la forme:

$$\begin{aligned}
 P(x) = & P_i + P'_i(x-x_i) + \frac{P''_i}{2!}(x-x_i)^2 + D(x-x_i)^3 + G(x-x_i)^3(x-x_{i+1}) \\
 & + I(x-x_i)^3(x-x_{i+1})^2
 \end{aligned}$$

Cette méthode a donné lieu à 2 procédures Algol: VALQSP et VAL2SP pour les cas  $2 < q \leq 6$  et  $q=2$  respectivement (cf.3.2.1.3).

1. Cas général ( $2 < q \leq 6$ )

PROCEDURE SPLQIN(X, Y, N, Q, DER, BOOL);  
TABLEAUX Y, DER; ENTIER Q, N; BOOLEEN BOOL;  
 COMMENTAIRE X ET Y SONT DES TABLEAUX REELS DE DIMENSION (/1:N/)  
 FOURNIS PAR L'UTILISATEUR QUI REPRESENTENT LES COORDONNEES DES  
 POINTS DONNES.

CETTE PROCEDURE CALCULE DANS LE TABLEAU DER(/1:N, 1:2\*Q-2/) LA VALEUR EN X(/I/) DES DERIVEES JUSQU'A L'ORDRE 2\*Q-2 DE LA FONCTION SPLINE D'INTERPOLATION D'ORDRE Q PRENANT LES VALEURS Y(/I/) AUX POINTS X(/I/) (I=1 A N). N EST UN ENTIER QUI INDIQUE LE NOMBRE DE POINTS. Q EST ENTIER QUI INDIQUE L'ORDRE DE LA FONCTION SPLINE D'INTERPOLATION. BOOL EST UN BOOLEEN QUI PREND LA VALEUR FAUX SI LE CALCUL N'EST PAS POSSIBLE. CETTE PROCEDURE UTILISE LA PROCEDURE GRESOLSYSLINE OU GRESOLPIV POUR LA RESOLUTION DES SYSTEMES LINEAIRES.

CETTE PROCEDURE PEUT S'EMPLOYER AVEC LA REELLE PROCEDURE VALQSP QUI CONNAISSANT LE TABLEAU DER CALCULE LA VALEUR DE LA FONCTION SPLINE EN UN POINT DONNE;

DEBUT TABLEAU M(/1:N, 1:Q-1, 1:2\*Q-2/), S(/1:N, 1:Q-1/),  
M1(/1:Q-1, 1:2\*Q/), S1(/1:Q-1/), A(/1:2\*Q-2, 1:2\*Q-2/),  
B, Z(/1:2\*Q-2/), HH, AA(/1:2\*Q/);  
PROCEDURE GRESOLSYSLINE(A, B, X, N, IMPOSSIBLE); CORPS DE PROCEDURE  
 CETTE PROCEDURE EST DECRITE DANS LE LIVRE:

PROCEDURES ALGOL EN ANALYSE NUMERIQUE  
 TOME 1, EDITION DU CNRS;

REEL H, S2, S3; ENTIER I, J, K, K1;  
PHASE1: I:=1; HH(/2\*Q/):=1.0;  
POUR J:=1 PAS 1 JUSQUA Q-1 FAIRE  
DEBUT  
POUR K:=1 PAS 1 JUSQUA 2\*Q FAIRE  
M1(/J, K/):=SI K=J+Q ALORS 1.0 SINON 0.0; S1(/J/):=0.0;  
FIN;  
ITER1 :  
POUR J:=1 PAS 1 JUSQUA Q-1 FAIRE  
DEBUT  
POUR K:=1 PAS 1 JUSQUA 2\*Q-2 FAIRE  
M(/I, J, K/):=M1(/J, K+1/);  
S(/I, J/):=S1(/J/)-M1(/J, 1/)\*Y(/I/);  
FIN;  
SI I=N ALORS ALLERA PHASE2;  
COMMENTAIRE CALCUL DES RELATIONS DE GAUCHE A DROITE;  
H:=X(/I+1/)-X(/I/);  
POUR J:=2 PAS 1 JUSQUA 2\*Q FAIRE  
HH(/2\*Q-J+1/):=HH(/2\*Q-J+2/)\*(-H)/(J-1);  
POUR J:=1 PAS 1 JUSQUA Q-1 FAIRE  
DEBUT  
POUR K:=1 PAS 1 JUSQUA 2\*Q FAIRE  
AA(/K/):=M1(/J, K/);  
POUR K:=1 PAS 1 JUSQUA 2\*Q FAIRE  
DEBUT S2:=0.0;  
POUR K1:=1 PAS 1 JUSQUA K FAIRE  
S2:=S2+AA(/K1/)\*HH(/2\*Q-K+K1/);  
M1(/J, K/):=S2;  
FIN FIN;  
COMMENTAIRE ELIMINATION LIGNE PAR LIGNE;  
POUR J:=Q-1 PAS -1 JUSQUA 2 FAIRE  
DEBUT S2:=M1(/J-1, 2\*Q/); S3:=-M1(/J, 2\*Q/); M1(/J, 2\*Q/):=0.0;  
POUR K:=1 PAS 1 JUSQUA 2\*Q-1 FAIRE  
M1(/J, K/):=M1(/J, K/)\*S2+M1(/J-1, K/)\*S3;  
S1(/J/):=S1(/J/)\*S2+S1(/J-1/)\*S3;  
FIN;

```

S2:=HH(/1/);S3:=-M1(/1,2*Q/);M1(/1,2*Q/):=0.0;
POUR J:=1 PAS 1 JUSQUA 2*Q-1 FAIRE
M1(/1,J/):=M1(/1,J/)*S2+HH(/2*Q-J+1/)*S3;
S1(/1/):=S1(/1/)*S2+Y(/1/)*S3;
NORMA1:
POUR J:=1 PAS 1 JUSQUA Q-1 FAIRE
DEBUT S2:=S1(/J/)*S1(/J/);
POUR K:=1 PAS 1 JUSQUA 2*Q-1 FAIRE
S2:=S2+M1(/J,K/)*M1(/J,K/);
S2:=SQRT(S2);
S1(/J/):=S1(/J/)/S2;
POUR K:=1 PAS 1 JUSQUA 2*Q-1 FAIRE
M1(/J,K/):=M1(/J,K/)/S2; FIN;
I:=I+1; ALLERA ITER1;
PHASE2:I:=N; BOOL:=VRAI;
POUR J:=1 PAS 1 JUSQUA Q-1 FAIRE
DEBUT
POUR K:=1 PAS 1 JUSQUA 2*Q FAIRE
M1(/J,K/):=SI K=J+Q ALORS 1.0 SINON 0.0; S1(/J/):=0.0;
S1(/J/):=0.0; FIN;
ITER2:
POUR J:=1 PAS 1 JUSQUA Q-1 FAIRE
DEBUT
POUR K:=1 PAS 1 JUSQUA 2*Q-2 FAIRE
DEBUT A(/J,K/):=M(/I,J,K/);
A(/Q-1+J,K/):=M1(/J,K+1/); FIN;
B(/J/):=S(/I,J/);
B(/Q-1+J/):=S1(/J/)-Y(/I/)*M1(/J,1/);
FIN;
GRESOLSYSLINE(A,B,Z,2*Q-2,TERM);
POUR J:=1 PAS 1 JUSQUA 2*Q-2 FAIRE

DER(/I,J/):=Z(/J/);
SI I=1 ALORS ALLERA FIN;
H:=X(/I/)-X(/I-1/);
POUR J:=2 PAS 1 JUSQUA 2*Q FAIRE
HH(/2*Q-J+1/):=HH(/2*Q-J+2/)*H/(J-1);
POUR J:=1 PAS 1 JUSQUA Q-1 FAIRE
DEBUT
POUR K:=1 PAS 1 JUSQUA 2*Q FAIRE
AA(/K/):=M1(/J,K/);
POUR K:=1 PAS 1 JUSQUA 2*Q FAIRE
DEBUT S2:=0.0;

POUR K1:=1 PAS 1 JUSQUA K FAIRE
S2:=S2+AA(/K1/)*HH(/2*Q-K+K1/);
M1(/J,K/):=S2;
FIN FIN;
COMMENTAIRE ELIMINATION LIGNE PAR LIGNE;
POUR J:=Q-1 PAS -1 JUSQUA 2 FAIRE
DEBUT S2:=M1(/J-1,2*Q/);S3:=-M1(/J,2*Q/);M1(/J,2*Q/):=0.0;
POUR K:=1 PAS 1 JUSQUA 2*Q-1 FAIRE
M1(/J,K/):=M1(/J,K/)*S2+M1(/J-1,K/)*S3;
S1(/J/):=S1(/J/)*S2+S1(/J-1/)*S3;
FIN;
S2:=HH(/1/);S3:=-M1(/1,2*Q/);M1(/1,2*Q/):=0.0;
POUR J:=1 PAS 1 JUSQUA 2*Q-1 FAIRE
M1(/1,J/):=M1(/1,J/)*S2+HH(/2*Q-J+1/)*S3;
S1(/1/):=S1(/1/)*S2+Y(/1/)*S3;
NORMA2:
POUR J:=1 PAS 1 JUSQUA Q-1 FAIRE
DEBUT S2:=S1(/J/)*S1(/J/);

```

```

POUR K:=1 PAS 1 JUSQUA 2*Q-1 FAIRE
S2:=S2+MI(/J,K/)*MI(/J,K/);
S2:=SQRT(S2);
S1(/J/):=S1(/J/)/S2;
POUR K:=1 PAS 1 JUSQUA 2*Q-1 FAIRE
MI(/J,K/):=MI(/J,K/)/S2;

```

```

FIN;
I:=I-1; ALLERA ITER2;
TERM: BOOL:=FAUX;
FIN: FIN SPLQIN;

```

REEL PROCEDURE VALOSP(X,Y,DER,N,Q,Z,BOOL);  
 TABLEAU X,Y,DER; ENTIER N,Q; REEL Z; BOOLEEN BOOL;  
 COMMENTAIRE X ET Y SONT DES TABLEAUX REELS DE DIMENSION (/1:N/)  
 FOURNIS PAR L'UTILISATEUR QUI REPRESENTENT LES COORDONNEES DES  
 POINTS DONNES.

CETTE REELLE PROCEDURE CALCULE EN Z LA VALEUR DE LA  
 FONCTION SPLINE D'INTERPOLATION D'ORDRE Q PRENANT LES VALEURS  
 Y(/I/) AUX POINTS X(/I/) (I=1 A N). LE TABLEAU DER CONTIENT LES  
 VALEURS DES 2\*Q-2 PREMIERES DERIVEES DE LA FONCTION SPLINE EN  
 CES POINTS. N EST LE NOMBRE DE POINTS. BOOL EST UN BOOLEEN QUI  
 PREND LA VALEUR FAUX SI Z N'APPARTIENT PAS A L'INTERVALLE OU EST  
 DEFINIE LA FONCTION SPLINE. LE TABLEAU DER EST FOURNI PAR LA PROCEDURE  
 SPLQIN;

```

DEBUT REEL S,T,U,H; ENTIER I,J,K; TABLEAU LH1,LH2(/1:Q/);
  BOOL:=VRAI;
SI(Z<X(/1/))|(Z>X(/N/))ALORSDEBUTBOOL:=FAUX;ALLERAFIN;FIN;

```

```

  POUR I:=2 PAS 1 JUSQUA N FAIRE
    SI(Z<X(/I/)) ALORS ALLERA SUITE;
SUITE: H:=X(/I/)-X(/I-1/); T:=Z-X(/I-1/); U:=T-H;
  LH1(/I/):=LH2(/I/):=(Y(/I/)-Y(/I-1/))/H;
  K:=1;

```

```

  POUR J:=2 PAS 1 JUSQUA Q FAIRE
DEBUT K:=K*(J-1);
  LH1(/J/):=(LH1(/J-1/)-DER(/I-1,J-1/)/K)/H;
  LH2(/J/):=(DER(/I,J-1/)/K-LH2(/J-1/))/H;
  FIN;

```

```

  POUR K:=2 PAS 1 JUSQUA Q FAIRE
DEBUT LH2(/2/):=(LH2(/2/)-LH1(/K/))/H;
  POUR J:=3 PAS 1 JUSQUA Q FAIRE
    LH2(/J/):=(LH2(/J/)-LH2(/J-1/))/H;

```

```

  FIN;
  K:=1;
  POUR J:=1 PAS 1 JUSQUA Q-1 FAIRE K:=K*J;
  S:=LH2(/Q/);
  POUR J:=Q-1 PAS-1 JUSQUA 2 FAIRE S:=S*U+LH2(/J/);
  S:=S*U*T+LH1(/Q/)*T+DER(/I-1,Q-1/)/K;

```

```

  POUR J:=Q-1 PAS -1 JUSQUA 2 FAIRE
DEBUT K:=K*/J; S:=S*T+DER(/I-1,J-1/)/K; FIN;
  VALOSP:=S*T+Y(/I-1/);

```

```

FIN;
FIN VALOSP ;

```

2. Cas  $q = 2$ PROCEDURE SPL2IN(X, Y, N, DF, BOOL);TABLEAU X, Y, DF; ENTIER N; BOOLEEN BOOL;COMMENTAIRE X ET Y SONT DES TABLEAUX REELS DE DIMENSION (/1:N/)  
FOURNIS PAR L'UTILISATEUR QUI REPRESENTENT LES COORDONNEES DES  
POINTS DONNES.CETTE PROCEDURE CALCULE DANS LE TABLEAU DF(/1:N/)  
LA VALEUR EN X(/1/) DE LA DERIVEE PREMIERE DE LA FONCTION SPLINE  
D'INTERPOLATION D'ORDRE 2 PRENANT LES VALEURS Y(/1/) AUX  
POINTS X(/1/) (I=1 A N). N EST UN ENTIER QUI INDIQUE LE NOMBRE DE  
POINTS.BOOL EST UN BOOLEEN QUI PREND LA VALEUR FAUX SI LE CALCUL  
N'EST PAS POSSIBLE. CETTE PROCEDURE PEUT S'EMPLOYER AVEC LA  
REELLE PROCEDURE VAL2SP QUI CONNAISSANT LE TABLEAU DF  
CALCULE LA VALEUR DE LA FONCTION SPLINE EN UN POINT DONNE;DEBUTREEL M1, M2, M3, S, S1, F1, F2, F3, F4, NOR, H, H2, H3;ENTIER I; REELTABLEAU B, C, D(/1:N/);BOOL:=VRAI;LEFTRIGHT: M1:=M2:=S:=0.0; M3:=1.0;POUR I:=1 PAS 1 JUSQUA N-1 FAIREDEBUT B(/1/):=M2; C(/1/):=M3; D(/1/):=S-M1\*Y(/1/);H:=X(/1+1/)-X(/1/); H2:=H\*H/2.0; H3:=H2\*H/3.0;F1:=M1; F2:=-H\*M1+M2; F3:=H2\*M1-H\*M2+M3; F4:=-H3\*M1+H2\*M2-H\*M3;M1:=F1\*H3+F4; M2:=-F2\*H3-H\*F4; M3:=F3\*H3+H2\*F4; S:=H3\*S+F4\*Y(/1/);NOR:=SQRT(M1\*M1+M2\*M2+M3\*M3+S\*S);M1:=M1/NOR; M2:=M2/NOR; M3:=M3/NOR; S:=S/NOR;FIN LEFTRIGHT;B(/N/):=M2; C(/N/):=M3; D(/N/):=S-M1\*Y(/N/);RIGHTLEFT: M1:=M2:=S:=0.0; M3:=1.0;POUR I:=N PAS-1 JUSQUA 2 FAIREDEBUT NOR:=C(/1/)\*M2-M3\*B(/1/); S1:=S-M1\*Y(/1/);SI ABS(NOR) = 0.0 ALORSDEBUTBOOL:=FAUX; ALLERA FIN; FINDF(/1/):=(C(/1/)\*S1-M3\*D(/1/))/NOR;H:=X(/1/)-X(/1-1/); H2:=H\*H/2; H3:=H2\*H/3;F1:=M1; F2:=H\*M1+M2; F3:=H2\*M1+H\*M2+M3; F4:=H3\*M1+H2\*M2+H\*M3;M1:=F4-H3\*F1; M2:=F4\*H-H3\*F2; M3:=F4\*H2-H3\*F3; S:=F4\*Y(/1/)-H3\*S;NOR:=SQRT(M1\*M1+M2\*M2+M3\*M3+S\*S);M1:=M1/NOR; M2:=M2/NOR; M3:=M3/NOR; S:=S/NOR;FIN RIGHTLEFT;NOR:=C(/1/)\*M2-M3\*B(/1/); S:=S-M1\*Y(/1/);SI ABS(NOR) = 0.0 ALORSDEBUTBOOL:=FAUX; ALLERA FIN; FINDF(/1/):=(C(/1/)\*S-M3\*D(/1/))/NOR;FIN:FIN SPL2IN;REEL PROCEDURE VAL2SP(X, Y, DF, N, Z, BOOL);TABLEAUX X, Y, DF; ENTIER N; REEL Z; BOOLEEN BOOL;COMMENTAIRE X ET Y SONT DES TABLEAUX REELS DE DIMENSION (/1:N/)  
FOURNIS PAR L'UTILISATEUR QUI REPRESENTENT LES COORDONNEES DES  
POINTS DONNES.CETTE REELLE PROCEDURE CALCULE EN Z LA VALEUR DE LA  
FONCTION SPLINE D'INTERPOLATION D'ORDRE 2 PRENANT LES  
VALEURS Y(/1/) AUX POINTS X(/1/) (I=1 A N). LE TABLEAU  
DF(/1:N/) CONTIENT LES VALEURS DE LA DERIVEE PREMIERE  
DE LA FONCTION SPLINE EN CES POINTS. N EST LE NOMBRE DE  
POINTS. BOOL EST UN BOOLEEN QUI PREND LA VALEUR FAUX  
SI Z N'APPARTIENT PAS A L'INTERVALLE OU EST DEFINIE LA  
FONCTION SPLINE. LE TABLEAU DF EST FOURNI PAR LA PROCEDURE  
SPL2IN;DEBUT REEL A, B, H, T, ALPHA, BETA, GAMMA, DELTA;BOOL:=VRAI;SI (Z<X(/1/)) | (Z>X(/N/)) ALORS



```
DEBUT BOOL:=FAUX; ALLERA FIN; FIN;  
POUR I:=2 PAS 1 JUSQUA N FAIRE  
SI (Z<X(/I/)) ALORS ALLERA SUITE;  
SUITE: H:=X(/I/)-X(/I-1/); T:=Z-X(/I-1/); A:=Y(/I-1/);  
B:=DF(/I-1/);  
ALPHA:=(Y(/I/)-A)/H; BETA:=(ALPHA-B)/H;  
GAMMA:=(DF(/I/)-ALPHA)/H; DELTA:=(GAMMA-BETA)/H;  
VAL2SP:=A+T*(B+T*(BETA+DELTA*(T-H)));  
FIN:FIN VAL2SP;
```

### 3.3 AJUSTEMENT

Soient  $n$  abscisses distinctes appartenant à l'intervalle  $[a, b]$  telles que:

$$a < x_1 < x_2 < \dots < x_n < b$$

et  $n$  nombres réels  $y_i$ ,  $i = 1 \dots n$ . En général les valeurs  $y_i$  représentent en  $x_i$ , les valeurs expérimentales d'une certaine fonction  $f$ . Ces valeurs de par leur nature sont souvent entachées d'erreurs. Il est intéressant, dans ce cas, de chercher une fonction  $\tilde{f}$  ne prenant pas exactement ces valeurs, mais qui réalise un compromis entre l'approximation de  $f$  et le lissage des valeurs expérimentales.

L'ajustement consiste à trouver un critère qui permette de réaliser ce compromis. Nous étudierons les cas où  $\tilde{f}$  appartient aux espaces fonctionnels  $H^q [a, b]$  et  $\mathcal{P}_m$ . Le premier cas nous conduira à l'ajustement par des fonctions splines d'ordre  $q$ , le second à l'ajustement polynomial au sens des moindres carrés.

- Pour définir le caractère lisse d'une fonction de  $H^q [a, b]$  on peut choisir par exemple

$$J_q(f) = \int_a^b [f^{(q)}(x)]^2 dx$$

de même, la quantité

$$\epsilon(f) = \sum_{i=1}^n [f(x_i) - y_i]^2$$

peut être choisie comme caractère d'approximation.

Le compromis entre le lissage et l'approximation sera réalisé lorsque l'on voudra minimiser la quantité

$$J_q(f) + \rho \epsilon(f) \quad \rho > 0.$$

Le paramètre  $\rho$  mesure l'importance relative du lissage et de l'approximation.

- Pour  $\mathcal{S}_m$ , on peut choisir le critère d'approximation et de lissage de la méthode des moindres carrés qui consiste à minimiser la quantité

$$Q(P) = \sum_{i=1}^n (y_i - P(x_i))^2 \quad \text{avec } P \in \mathcal{S}_m$$

### 3.3.1 Spline d'ajustement d'ordre q.

Soient  $n$  abscisses fixes et distinctes de l'intervalle  $[a, b]$  telles que:

$$a < x_1 < x_2 < \dots < x_n < b$$

avec  $n \geq q$  et  $n$  nombres réels  $y_i$ ,  $i = 1 \dots n$ .

On montre [12] que parmi toutes les fonctions de  $H^q [a, b]$ , il en existe une et une seule notée  $\sigma$  qui vérifie :

$$\int_a^b [\sigma^{(q)}(x)]^2 dx + \rho \sum_{i=1}^n [\sigma(x_i) - y_i]^2 \leq \int_a^b [f^{(q)}(x)]^2 dx + \sum_{i=1}^n [f(x_i) - y_i]^2$$

pour tout  $f \in H^q [a, b]$  ( $\rho$  est un coefficient positif).

$\sigma$  est appelée fonction spline d'ajustement d'ordre  $q$  sur  $[a, b]$  et est caractérisée par les propriétés suivantes:

P7:  $\sigma$  est un polynôme de degré  $\leq 2q-1$  sur chaque intervalle  $]x_i, x_{i+1}[$   $i=1 \dots (n-1)$ .

P8:  $\sigma$  est un polynôme de degré  $\leq q-1$  sur chacun des intervalles  $[a, x_1[$  et  $]x_n, b]$ .

P9: En chaque point  $x_i$ ,  $i=1\dots n$ , on a la relation

$$\sigma(x_i) + \frac{(-1)^q}{\rho} [\sigma^{(2q-1)}(x_i^+) - \sigma^{(2q-1)}(x_i^-)] = y_i$$

P10: La fonction spline  $\sigma$  et ses dérivées jusqu'à l'ordre  $2q-2$  inclus, sont continues, c'est-à-dire:

$$\begin{aligned} \sigma^{(j)}(x_i^-) &= \sigma^{(j)}(x_i^+) & i=1\dots n \\ & & j=0\dots(2q-2) . \end{aligned}$$

Le problème est de déterminer dans chaque intervalle la fonction spline  $\sigma$  : son expression est un polynôme de degré  $\leq 2q-1$  dont il faut déterminer les  $2q$  coefficients. En tenant compte de la remarque 1, on choisit les inconnues:

$$\sigma(x), \sigma'(x), \dots, \sigma^{(2q-1)}(x)$$

et on pose

$$Z(x) = \begin{pmatrix} \sigma(x) \\ \sigma'(x) \\ \cdot \\ \cdot \\ \sigma^{(2q-1)}(x) \end{pmatrix}$$

### 3.3.1.1 Propriétés .

P11: Si l'on connaît  $q$  relations linéaires vérifiées par  $Z(x_i^-)$  :

$$M(x_i^-) \cdot Z(x_i^-) = b(x_i^-) \quad (3.8)$$

où  $M(x_i^-)$  est une matrice  $(q, 2q)$  et  $b(x_i^-)$  un vecteur  $(q, 1)$   
 alors on peut trouver  $q$  relations linéaires du même type  
 vérifiées par  $Z(x_i^+)$  .

Démonstration

$\sigma^{(2q-1)}(x)$  n'étant pas continue en  $x_i$ , l'expression  
 (3.8) n'est pas valable en  $x_i^+$ . En remplaçant, dans les  $q$   
 relations linéaires données par (3.8), l'inconnue  $\sigma^{(2q-1)}(x_i^-)$   
 par

$$(-1)^q \cdot \rho \cdot \sigma(x_i^-) + \sigma^{(2q-1)}(x_i^+) - (-1)^q \cdot \rho \cdot y_i ,$$

valeur tirée de la propriété 9, on obtient  $q$  nouvelles relations  
 linéaires vérifiées en  $x_i^+$ , du fait de la continuité (P10) de  
 la fonction et de ses  $2q-2$  premières dérivées. Ces  $q$  relations  
 linéaires se notent sous forme matricielle par:

$$M(x_i^+) \cdot Z(x_i^+) = b(x_i^+)$$

forme analogue à (3.8) mais en  $x_i^+$  .

P12: Si dans un intervalle  $]x_i, x_{i+1}[$  il existe en  $x_i^+$   $q$  relations  
 linéaires qui se notent sous forme matricielle:

$$M(x_i^+) \cdot Z(x_i^+) = b(x_i^+)$$

il existe en  $x_{i+1}^-$ ,  $q$  relations linéaires analogues.

La démonstration a été faite en P6.

### 3.3.1.2 Mise en oeuvre de l'algorithme .

Le but de cette méthode est d'évaluer, pour un  $x$  donné de l'intervalle  $[a,b]$ , la valeur correspondante de la fonction spline d'ajustement d'ordre  $q$ . Pour calculer cette valeur, il faut connaître, dans chaque intervalle  $]x_i, x_{i+1}[$   $i=1\dots(n-1)$ , le polynôme de degré  $\leq 2q-1$ . La remarque 2 éclairera la méthode utilisée. Cette fois  $\sigma(x_i)$  n'est pas connu, il faut donc déterminer sa valeur en chaque point  $x_i$ ,  $i=1\dots n$ , ainsi que la valeur des  $(q-1)$  premières dérivées en ces mêmes points. Cette méthode se partage en deux parties:

- 1) Calcul de la valeur de la fonction et de ses  $q-1$  premières dérivées.
- 2) Calcul de la valeur du polynôme.

#### 3.3.1.2.1 Calcul de la valeur de la fonction et de la valeur de ses $q-1$ premières dérivées.

Nous prendrons comme inconnu le vecteur

$$Z(x) = \begin{pmatrix} \sigma(x) \\ \sigma'(x) \\ \cdot \\ \cdot \\ \cdot \\ \sigma^{(2q-1)}(x) \end{pmatrix}$$

Pour calculer ces inconnues et en particulier,  $\sigma(x)$  et les  $q-1$  premières dérivées, il faut connaître en tous points  $x_i$ ,  $i=1\dots n$ ,  $2q$  relations linéaires entre ces  $2q$  inconnues. La détermination de ces

2q relations va se faire en deux étapes.

①

Dans une première phase, en balayant l'intervalle [ a, b ] de gauche à droite, on détermine en chaque point  $x_i^+$ ,  $i=1 \dots n$ , q relations linéaires entre les inconnues:

$$\sigma(x_i^+), \sigma'(x_i^+), \dots, \sigma^{(2q-1)}(x_i^+).$$

Initialisation

- En  $x_1^+$  on écrit sous forme matricielle les propriétés 9 et 10 qui expriment, en tenant compte de la propriété 8, la continuité des dérivées. On obtient les relations linéaires:

$$M(x_1^+).Z(x_1^+) = b(x_1^+)$$

où  $M(x_1^+)$  est une matrice (q,2q) et  $b(x_1^+)$  un vecteur (q,1). Ce système est de la forme :

$$\begin{array}{c} \begin{array}{cccccc} \xleftarrow{q} & & & \xrightarrow{q} & & \\ \left| \begin{array}{cccccc} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ (-1)^q \cdot \rho & 0 & & & 0 & 1 \end{array} \right| & \cdot & \begin{array}{c} \left| \begin{array}{c} \sigma(x_1^+) \\ \sigma'(x_1^+) \\ \vdots \\ \sigma^{(2q-2)}(x_1^+) \\ \sigma^{(2q-1)}(x_1^+) \end{array} \right| & = & \begin{array}{c} \left| \begin{array}{c} 0 \\ 0 \\ \vdots \\ 0 \\ (-1)^q \cdot \rho \cdot y_1 \end{array} \right| \end{array} \end{array}$$

- On passe en  $x_2^-$  en utilisant la propriété 12, qui nous donne:

$$M(x_2^-) \cdot Z(x_2^-) = b(x_2^-)$$

avec

$$M(x_2^-) = M(x_2^+) \cdot T(-h_1)$$

$$b(x_2^-) = b(x_1^+)$$

Remarque. La forme de  $T(-h_1)$  est donnée dans la démonstration de la propriété 6.

- On passe au point  $x_2^+$  en utilisant la propriété 11 qui nous donne les relations linéaires:

$$M(x_2^+) \cdot Z(x_2^+) = b(x_2^+)$$

où  $M(x_2^+)$  et  $b(x_2^+)$  ont la forme :

$$M(x_2^+) = \begin{array}{cccc|cc} * & * & \dots & * & * & \\ * & * & & * & * & \\ \cdot & & & \cdot & \cdot & \\ \cdot & & & \cdot & \cdot & \\ \cdot & & & \cdot & \cdot & \\ * & * & & * & * & \\ * & * & & * & * & \end{array} \quad b(x_2^+) = \begin{array}{c} * \\ * \\ \cdot \\ \cdot \\ \cdot \\ * \\ * \end{array} \quad (3.9)$$

- On passe du point  $x_2^+$  au point  $x_3^+$  comme l'on est passé de  $x_1^+$  en  $x_2^+$  et on applique cette stratégie jusqu'au point  $x_n^+$ .



②

Dans une deuxième phase, en balayant l'intervalle [ a, b ] de droite à gauche, on détermine en chaque point  $x_i^-$ ,  $i=1..n$ , q nouvelles relations linéaires entre les 2q inconnues:

$$\sigma(x_i^-), \sigma'(x_i^-), \dots, \sigma^{(2q-1)}(x_i^-) .$$

Initialisation

En  $x_n^-$ , on écrit sous forme matricielle les propriétés 9 et 10, qui expriment, en tenant compte de la propriété 8, la continuité des dérivées. On obtient les relations linéaires:

$$M(x_n^-) \cdot Z(x_n^-) = b(x_n^-)$$

où  $M(x_n^-)$  est une matrice (q,2q) et  $b(x_n^-)$  un vecteur (q.1). Ce système est de la forme:

$$\begin{array}{cccccccc|cccc} 0 & 0 & \dots & 0 & 1 & 0 & \dots & 0 & \sigma(x_n^-) & & 0 \\ 0 & 0 & \dots & 0 & 0 & 1 & \dots & 0 & \sigma'(x_n^-) & & 0 \\ \vdots & & & & & & & & \vdots & & \\ \vdots & & & & & & & & \vdots & & \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & 1 & 0 & & \\ (-1)^q \cdot \rho & 0 & \dots & \dots & \dots & \dots & \dots & 0 & -1 & & (-1) \cdot \rho \cdot \dot{y}_n \end{array}$$

$$\begin{array}{c} x \\ \sigma^{(2q-2)}(x_n^-) \\ \sigma^{(2q-1)}(x_n^-) \end{array} =$$

- On passe au point  $x_{n-1}^+$  en utilisant la propriété 12 qui nous donne les relations linéaires:

$$M(x_{n-1}^+) \cdot Z(x_{n-1}^+) = b(x_{n-1}^+)$$

avec

$$M(x_{n-1}^+) = M(x_n^-) \cdot T(h_n)$$

$$b(x_{n-1}^+) = b(x_n^-)$$

Remarque. La forme de  $T(h_n)$  est donnée dans la démonstration de la propriété 6.

- On passe au point  $x_{n-1}^-$  en utilisant la propriété 11 qui nous donne les relations linéaires:

$$M(x_{n-1}^-) \cdot Z(x_{n-1}^-) = b(x_{n-1}^-)$$

où  $M(x_{n-1}^-)$  et  $b(x_{n-1}^-)$  ont une forme analogue à (3.9).

- On passe du point  $x_{n-1}^-$  au point  $x_{n-2}^-$  comme l'on est passé de  $x_n^-$  à  $x_{n-1}^-$  et on applique cette stratégie jusqu'au point  $x_1^-$ .

En résumé, on obtient en chaque point  $x_i^-$  et  $x_i^+$ ,  $i=1 \dots n$ ,  $q$  relations linéaires qui proviennent de (1) et (2), entre les inconnues :

$$\sigma(x_i^+), \sigma'(x_i^+), \dots, \sigma^{(2q-1)}(x_i^+)$$

et

$$\sigma(x_i^-), \sigma'(x_i^-), \dots, \sigma^{(2q-1)}(x_i^-) .$$

Remarque. Dans la phase (2) au cours du calcul des q relations linéaires en  $x_i^-$ , on obtient à un moment q relations linéaires, en  $x_i^+$  qui sont ajoutées à celles obtenues dans la phase (1), pour donner 2q relations linéaires entre les inconnues:

$$\sigma(x_i^+), \sigma'(x_i^+), \dots, \sigma^{(2q-1)}(x_i^+) .$$

Les inconnues sont calculées en résolvant ce système linéaire par la méthode de Gauss.

Cette méthode a donné naissance d'une part aux commandes AJUSP2 et AJUSPQ du système PGAN (cf.2.2.2) et d'autre part aux deux procédures Algol AJU2SP et AJUQSP décrites au paragraphe (3.3.1.3).

Remarque. Le cas, q=2, qui conduit à un système linéaire de quatre équations à quatre inconnues, est traité séparément.

### 3.3.1.2.2 Calcul de la valeur du polynôme

La méthode utilisée est celle exposée au paragraphe (3.2.1.2.2) qui conduit dans ce cas aux deux procédures Algol VALQAJ et VAL2AJ pour les cas  $2 < q \leq 6$  et q=2 respectivement (cf.3.3.1.3).

## 3.3.1.3. Algorithmes en ALGOL

1. Cas général ( $2 < q \leq 6$ )

```

PROCEDURE AJUOSP(X, Y, N, Q, RAU, F, DER, BOOL);
TABLEAUX Y, F, DER; ENTIER Q, N; BOOLEEN BOOL; REEL RAU;
COMMENTAIRE X ET Y SONT DES TABLEAUX REELS DE DIMENSION (/1:N/)
FOURNIS PAR L'UTILISATEUR QUI REPRESENTENT LES COORDONNEES DES
POINTS DONNES.
CETTE PROCEDURE CALCULE DANS LES TABLEAUX F ET DER DE DIMENSIONS (/1:N/)
ET (/1:N, 1:2*Q-1/) RESPECTIVEMENT, LA VALEUR DE LA FONCTION SPLINE
D'AJUSTEMENT D'ORDRE Q ET DE SES 2*Q-1 PREMIERES DERIVEES AUX POINTS
X(/I/) (I=1 A N). N EST UN ENTIER QUI INDIQUE LE NOMBRE DE POINTS DONNES.
Q EST UN ENTIER QUI DONNE L'ORDRE DE LA FONCTION SPLINE D'AJUSTEMENT.
BOOL EST UN BOOLEEN QUI PREND LA VALEUR FAUX SI LE CALCUL N'EST PAS
POSSIBLE. RAU EST PARAMETRE REEL FOURNI PAR L'UTILISATEUR QUI MESURE LE
COMPROMIS ENTRE L'AJUSTEMENT ET L'APPROXIMATION (RAU>0).
CETTE PROCEDURE UTILISE LA PROCEDURE GRESOLSYSLINE OU GRESOLPIV POUR LA
RESOLUTION DES SYSTEMES LINEAIRES.
CETTE PROCEDURE PEUT S'EMPLOYER AVEC LA REELLE PROCEDURE VALQAJ QUI
CONNAISSANT LES TABLEAUX F ET DER CALCULE LA VALEUR DE LA FONCTION EN
UN POINT DONNE;
DEBUT TABLEAU M(/1:N, 1:Q, 1:2*Q/), S(/1:N, 1:Q/),
M1(/1:Q, 1:2*Q/), S1(/1:Q/), A(/1:2*Q, 1:2*Q/),
B, Z(/1:2*Q/), HH, AA(/1:2*Q/);
PROCEDURE GRESOLSYSLINE (A, B, X, N, IMPOSSIBLE); CORPS DE PROCEDURE
CETTE PROCEDURE EST DECRITE DANS LE LIVRE:
PROCEDURES ALGOL EN ANALYSE NUMERIQUE
TOME 1, EDITION DU CNRS;
REEL H, S2, S3; ENTIER I, J, K, K1, CTE;
PHASE1: I:=1; HH(/2*Q/):=1.0; CTE:=(-1)**Q; BOOL:=VRAI;
POUR J:=1 PAS 1 JUSQUA Q FAIRE
DEBUT
POUR K:=1 PAS 1 JUSQUA 2*Q FAIRE
M1(/J, K/):=SI K=J+Q ALORS 1.0 SINON 0.0; S1(/J/):=0.0;
FIN;
M1(/Q, 1/):=CTE*RAU; S1(/Q/):=CTE*RAU*Y(/1/);
ITER1 :
POUR J:=1 PAS 1 JUSQUA Q FAIRE
DEBUT
POUR K:=1 PAS 1 JUSQUA 2*Q FAIRE
M(/1, J, K/):=M1(/J, K/);
S(/1, J/):=S1(/J/);
FIN;
SI I=N ALORS ALLERA PHASE2;
COMMENTAIRE CALCUL DES RELATIONS DE GAUCHE A DROITE;
H:=X(/I+1/)-X(/I/);
POUR J:=2 PAS 1 JUSQUA 2*Q FAIRE
HH(/2*Q-J+1/):=HH(/2*Q-J+2/)*(-H)/(J-1);
POUR J:=1 PAS 1 JUSQUA Q FAIRE
DEBUT
POUR K:=1 PAS 1 JUSQUA 2*Q FAIRE
AA(/K/):=M1(/J, K/);
POUR K:=1 PAS 1 JUSQUA 2*Q FAIRE
DEBUT S2:=0.0;
POUR K1:=1 PAS 1 JUSQUA K FAIRE
S2:=S2+AA(/K1/)*HH(/2*Q-K+K1/);
M1(/J, K/):=S2;
FIN FIN;
COMMENTAIRE ELIMINATION LIGNE PAR LIGNE;
POUR J:=1 PAS 1 JUSQUA Q FAIRE
DEBUT
M1(/J, 1/):=M1(/J, 1/)+CTE*RAU*M1(/J, 2*Q/);

```

$S1(/J/) := S1(/J/) + CTE * RAU * M1(/J, 2*Q/) * Y(/I+1/);$

FIN;

NORMA1:

POUR J:=1 PAS 1 JUSQUA Q FAIRE

DEBUT S2:=S1(/J/)\*S1(/J/);

POUR K:=1 PAS 1 JUSQUA 2\*Q FAIRE

$S2 := S2 + M1(/J, K/) * M1(/J, K/);$

$S2 := \text{SQRT}(S2);$

$S1(/J/) := S1(/J/) / S2;$

POUR K:=1 PAS 1 JUSQUA 2\*Q FAIRE

$M1(/J, K/) := M1(/J, K/) / S2;$  FIN;

I:=I+1; ALLERA ITER1;

PHASE2: I:=N;

POUR J:=1 PAS 1 JUSQUA Q FAIRE

DEBUT

POUR K:=1 PAS 1 JUSQUA 2\*Q FAIRE

$M1(/J, K/) := \text{SI } K=J+Q \text{ ALORS } 1.0 \text{ SINON } 0.0;$

$S1(/J/) := 0.0;$  FIN;

$M1(/Q, 1/) := CTE * RAU;$   $M1(/Q, 2*Q/) := -1.0;$   $S1(/Q/) := CTE * RAU * Y(/N/);$

ITER2:

POUR J:=1 PAS 1 JUSQUA Q FAIRE

DEBUT

POUR K:=1 PAS 1 JUSQUA 2\*Q FAIRE

$A(/J, K/) := M(/I, J, K/);$   $A(/J+Q, K/) := M1(/J, K/);$  FIN;

$B(/J/) := S(/I, J/);$   $B(/J+Q/) := S1(/J/);$  FIN;

POUR J:=1 PAS 1 JUSQUA Q FAIRE

DEBUT

$A(/J, 1/) := A(/J, 1/) - CTE * RAU * A(/J, 2*Q/);$

$B(/J/) := B(/J/) - CTE * RAU * A(/J, 2*Q/) * Y(/I/);$

FIN;

GRESOLSYSLINE(A, B, Z, 2\*Q, TERM);

$F(/I/) := Z(/I/);$

POUR J:=1 PAS 1 JUSQUA 2\*Q-1 FAIRE

$DER(/I, J/) := Z(/J+1/);$

SI I=1 ALORS ALLERA FIN;

$H := X(/I/) - X(/I-1/);$

POUR J:=2 PAS 1 JUSQUA 2\*Q FAIRE

$HH(/2*Q-J+1/) := HH(/2*Q-J+2/) * H/(J-1);$

POUR J:=1 PAS 1 JUSQUA Q FAIRE

DEBUT

POUR K:=1 PAS 1 JUSQUA 2\*Q FAIRE

$AA(/K/) := M1(/J, K/);$

POUR K:=1 PAS 1 JUSQUA 2\*Q FAIRE

DEBUT S2:=0.0;

POUR K1:=1 PAS 1 JUSQUA K FAIRE

$S2 := S2 + AA(/K1/) * HH(/2*Q-K+K1/);$

$M1(/J, K/) := S2;$

FIN FIN;

COMMENTAIRE ELIMINATION LIGNE PAR LIGNE;

POUR J:=1 PAS 1 JUSQUA Q FAIRE

DEBUT

$M1(/J, 1/) := M1(/J, 1/) - CTE * RAU * M1(/J, 2*Q/);$

$S1(/J/) := S1(/J/) - CTE * RAU * M1(/J, 2*Q/) * Y(/I-1/);$

FIN;

NORMA2:

POUR J:=1 PAS 1 JUSQUA Q FAIRE

DEBUT S2:=S1(/J/)\*S1(/J/);

POUR K:=1 PAS 1 JUSQUA 2\*Q FAIRE

$S2 := S2 + M1(/J, K/) * M1(/J, K/);$

$S2 := \text{SQRT}(S2);$

```

S1(/J/):=S1(/J)/S2;
POUR K:=1 PAS 1 JUSQUA 2*Q FAIRE
M1(/J,K/):=M1(/J,K)/S2;
FIN;
I:=I-1; ALLERA ITER2;
TERM: BOOL:=FAUX;
FIN: FIN AJUQSP;
REEL PROCEDURE VALQAJ(X,Y,N,Q,F,DER,Z,BOOL);
    TABLEAU X,Y,F,DER; ENTIER N,Q; REEL Z; BOOLEEN BOOL;
    COMMENTAIRE X ET Y SONT DES TABLEAUX REELS DE DIMENSION (/1:N/)
    FOURNIS PAR L'UTILISATEUR QUI REPRESENTENT LES COORDONNEES DES
    POINTS DONNES, F ET DER SONT DES TABLEAUX DE DIMENSIONS (/1:N/)
    ET (/1:N,1:2*Q-1/) RESPECTIVEMENT, FOURNIS PAR LA PROCEDURE AJUQSP.
    CETTE REELLE PROCEDURE CALCULE EN Z LA VALEUR DE LA FONCTION SPLINE
    D'AJUSTEMENT D'ORDRE Q, PRENANT LES VALEURS F(/I/) AUX POINTS X(/I/)
    (I=1 A N). LE TABLEAU DER CONTIENT LES VALEURS DES 2*Q-1 PREMIERES
    DERIVEES DE LA FONCTION SPLINE AUX POINTS X(/I/) DONNES (I=1 A N).
    LE TABLEAU F CONTIENT LES VALEURS DE LA FONCTION SPLINE AUX POINTS
    X(/I/) (I=1 A N). BOOL EST UN BOOLEEN QUI PREND LA VALEUR FAUX SI Z
    N'APPARTIENT PAS A L'INTERVALLE OU EST DEFINIE LA FONCTION SPLINE;
DEBUT REEL S,T,U,H; ENTIER I,J,K; TABLEAU LH1,LH2(/1:Q/);
    BOOL:=VRAI;
SI (Z<X(/1/))|(Z>X(/N/))ALORSDEBUTBOOL:=FAUX;ALLERAFIN;FIN;
    POUR I:=2 PAS 1 JUSQUA N FAIRE
        SI (Z<X(/I/)) ALORS ALLERA SUITE;
SUITE: H:=X(/I/)-X(/I-1/); T:=Z-X(/I-1/); U:=T-H;
        LH1(/I/):=LH2(/I/):=(F(/I/)-F(/I-1/))/H;
        K:=1;
        POUR J:=2 PAS 1 JUSQUA Q FAIRE
DEBUT K:=K*(J-1);
        LH1(/J/):=(LH1(/J-1/)-DER(/I-1,J-1/)/K)/H;
        LH2(/J/):=(DER(/I,J-1/)/K-LH2(/J-1/))/H;
        FIN;
        POUR K:=2 PAS 1 JUSQUA Q FAIRE
DEBUT LH2(/2/):=(LH2(/2/)-LH1(/K/))/H;
        POUR J:=3 PAS 1 JUSQUA Q FAIRE
            LH2(/J/):=(LH2(/J/)-LH2(/J-1/))/H;
        FIN;
        K:=1;
    POUR J:=1 PAS 1 JUSQUA Q-1 FAIRE K:=K*J;
        S:=LH2(/Q/);
        POUR J:=Q-1 PAS-1 JUSQUA 2 FAIRE S:=S*U+LH2(/J/);
        S:=S*U*T+LH1(/Q/)*T+DER(/I-1,Q-1/)/K;
        POUR J:=Q-1 PAS -1 JUSQUA 2 FAIRE
DEBUT K:=K*/J; S:=S*T+DER(/I-1,J-1/)/K; FIN;
        VALQAJ:=S*T+F(/I-1/);
FIN;
FIN VALQAJ;

```

PROCEDURE AJU2SP(X,Y,N,RAU,F,DF,BOOL);  
TABLEAU X,Y,F,DF; ENTIER N; BOOLEEN BOOL; REEL RAU;  
 COMMENTAIRE X ET Y SONT DES TABLEAUX REELS DE DIMENSION (/1:N/)  
 FOURNIS PAR L'UTILISATEUR QUI REPRESENTENT LES COORDONNEES DES  
 POINTS DONNES.

CETTE PROCEDURE CALCULE DANS LES TABLEAUX F ET DF DE DIMENSION (/1:N/)  
 LA VALEUR DE LA FONCTION SPLINE D'AJUSTEMENT D'ORDRE 2 ET DE SA  
 DERIVEE PREMIERE AU POINT X(/1/) (I=1 A N). N EST UN ENTIER QUI INDIQUE  
 LE NOMBRE DE POINTS DONNES.

BOOL EST UN BOOLEEN QUI PREND LA VALEUR FAUX SI LE CALCUL N'EST PAS  
 POSSIBLE. RAU EST PARAMETRE REEL FOURNI PAR L'UTILISATEUR QUI MESURE LE  
 COMPROMIS ENTRE L'AJUSTEMENT ET L'APPROXIMATION (RAU>0).

CETTE PROCEDURE PEUT S'EMPLOYER AVEC LA REELLE PROCEDURE VAL2AJ QUI  
 CONNAISSANT LES TABLEAUX F ET DF CALCULE LA VALEUR DE LA FONCTION EN  
 UN POINT DONNE;

DEBUT ENTIER I,J,K;

REEL M1,M2,M3,M4,M5,M6,M7,F1,F2,F3,F4,F5,F6,F7,F8,S1,S2,NOR,H,H2,H3,S;  
REEL SP1,SP2;

TABLEAU M(/1:N,1:2,1:5/); BOOL:=VRAI;

LEFTRIGHT: M1:=M2:=M5:=M6:=S1:=0.0;M3:=M7:=1.0;M4:=RAU;S2:=RAU\*Y(/1/);

POUR I:=1 PAS 1 JUSQUA N-1 FAIRE

DEBUT M(/1,1,1/):=M1;M(/1,1,2/):=M2;M(/1,1,3/):=M3;M(/1,1,5/):=S1;

M(/1,2,1/):=M4;M(/1,2,2/):=M5;M(/1,2,3/):=M6;M(/1,2,4/):=M7;

M(/1,2,5/):=S2;

H:=X(/1+1/)-X(/1/); H2:=H\*H/2.0; H3:=H2\*H/3.0;

F1:=M1; F2:=-H\*M1+M2; F3:=H2\*M1-H\*M2+M3;F4:=-H3\*M1+H2\*M2-H\*M3;

F5:=M4; F6:=-H\*M4+M5; F7:=H2\*M4-H\*M5+M6;F8:=-H3\*M4+H2\*M5-H\*M6+M7;

M1:=F4\*F5-F8\*F1; M2:=F4\*F6-F8\*F2;M3:=F4\*F7-F8\*F3;S1:=F4\*S2-F8\*S1;

NOR:=(M1\*M1+M2\*M2+M3\*M3+S1\*S1); NOR:=SQRT(NOR);

M1:=M1/NOR; M2:=M2/NOR; M3:=M3/NOR; S1:=S1/NOR;

M4:=F5+RAU\*F8; M5:=F6; M6:=F7; M7:=F8; S2:=S2+RAU\*F8\*Y(/1+1/);

NOR:=(M4\*M4+M5\*M5+M6\*M6+M7\*M7+S2\*S2); NOR:=SQRT(NOR);

M4:=M4/NOR; M5:=M5/NOR; M6:=M6/NOR; M7:=M7/NOR;S2:=S2/NOR;

FIN LEFTRIGHT;

M(/N,1,1/):=M1;M(/N,1,2/):=M2;M(/N,1,3/):=M3;M(/N,1,5/):=S1;

M(/N,2,1/):=M4;M(/N,2,2/):=M5;M(/N,2,3/):=M6;M(/N,2,4/):=M7;

M(/N,2,5/):=S2;

RIGHTLEFT: M1:=M2:=M5:=M6:=S1:=0.0;M3:=1.0;M7:=-1.0;M4:=RAU;

S2:=RAU\*Y(/N/);

POUR I:=N PAS -1 JUSQUA 2 FAIRE

DEBUTS:M(/1,2,4/):=SP1;SP1:=M(/1,2,1/)-RAU\*S;SP2:=M(/1,2,5/)-RAU\*S\*Y(/1/);

F1:=M4\*S-M7\*SP1;F2:=M5\*S-M7\*M(/1,2,2/);

F3:=M6\*S-M7\*M(/1,2,3/);F4:=S2\*S-M7\*SP2;F5:=M3\*F1-F3\*M1;

F6:=M3\*F2-F3\*M2;F7:=M3\*F4-F3\*S1;S:=M(/1,1,3/);

F1:=S\*M1-M3\*M(/1,1,1/);F2:=S\*M2-M3\*M(/1,1,2/);

F3:=S\*S1-M3\*M(/1,1,5/);

F(/1/):=(F2\*F7-F6\*F3)/(F2\*F5-F6\*F1);DF(/1/):=(F7-F5\*F(/1/))/F6;

H:=X(/1/)-X(/1-1/); H2:=H\*H/2.0; H3:=H2\*H/3.0;

F1:=M1; F2:=+H\*M1+M2; F3:=H2\*M1+H\*M2+M3;F4:=+H3\*M1+H2\*M2+H\*M3;

F5:=M4; F6:=+H\*M4+M5; F7:=H2\*M4+H\*M5+M6;F8:=+H3\*M4+H2\*M5+H\*M6+M7;

M1:=F4\*F5-F8\*F1; M2:=F4\*F6-F8\*F2;M3:=F4\*F7-F8\*F3;S1:=F4\*S2-F8\*S1;

NOR:=(M1\*M1+M2\*M2+M3\*M3+S1\*S1); NOR:=SQRT(NOR);

M1:=M1/NOR; M2:=M2/NOR; M3:=M3/NOR; S1:=S1/NOR;

M4:=F5-RAU\*F8; M5:=F6; M6:=F7; M7:=F8; S2:=S2-RAU\*F8\*Y(/1-1/);

NOR:=(M4\*M4+M5\*M5+M6\*M6+M7\*M7+S2\*S2); NOR:=SQRT(NOR);

M4:=M4/NOR; M5:=M5/NOR; M6:=M6/NOR; M7:=M7/NOR;S2:=S2/NOR;

FIN RIGHTLEFT;

F1:=M4; F2:=M5; F3:=M6; F4:=S2;

F5:=M3\*F1-F3\*M1;

```

F6:=M3*F2-F3*M2;F7:=M3*F4-F3*S1;S:=M(/1,1,3/);
F1:=S*M1-M3*M(/1,1,1/);F2:=S*M2-M3*M(/1,1,2/);
F3:=S*S1-M3*M(/1,1,5/);
F(/1/):=(F2*F7-F6*F3)/(F2*F5-F6*F1);DF(/1/):=(F7-F5*F(/1/))/F6;
FIN AJU2SP;
REEL PROCEDURE VAL2AJ(X,Y,F,DF,N,Z,BOOL);
TABLEAU X,Y,F,DF; ENTIER N; REEL Z; BOOLEEN BOOL;
COMMENTAIRE X ET Y SONT DES TABLEAUX REELS DE DIMENSION (/1:N/)
FOURNIS PAR L'UTILISATEUR QUI REPRESENTENT LES COORDONNEES DES
POINTS DONNES,F ET DF SONT DES TABLEAUX DE DIMENSION (/1:N/)
FOURNIS PAR LA PROCEDURE AJU2SP.
CETTE REELLE PROCEDURE CALCULE EN Z LA VALEUR DE LA FONCTION SPLINE
D'AJUSTEMENT D'ORDRE 2, PRENANT LES VALEURS F(/1/) AUX POINTS X(/1/)
(I=1 A N). LE TABLEAU DF CONTIENT LES VALEURS DE LA DERIVEE PREMIERE
DE LA FONCTION SPLINE AUX POINTS X(/1/) DONNES (I=1 A N).
BOOL EST UN BOOLEEN QUI PREND LA VALEUR FAUX SI Z N'APPARTIENT PAS
A L'INTERVALLE OU EST DEFINIE LA FONCTION SPLINE;
DEBUT REEL A,B,H,T,ALPHA,BETA,GAMMA,DELTA;
    BOOL:=VRAI;
    SI (Z<X(/1/)) | (Z>X(/N/)) ALORS
    DEBUT BOOL:=FAUX; ALLERA FIN; FIN;
    POUR I:=2 PAS 1 JUSQUA N FAIRE
    SI (Z<X(/I/)) ALORS ALLERA SUITE;
SUITE: H:=X(/I/)-X(/I-1/); T:=Z-X(/I-1/); A:=F(/I-1/);
    B:=DF(/I-1/);
    ALPHA:=(F(/I/)-A)/H; BETA:=(ALPHA-B)/H;
    GAMMA:=(DF(/I/)-ALPHA)/H; DELTA:=(GAMMA-BETA)/H;
    VAL2AJ:=A+T*(B+T*(BETA+DELTA*(T-H)));
FIN: FIN VAL2AJ;

```



### 3.3.2 Ajustement polynomial au sens des moindres carrés.

Soient  $m+1$  abscisses distinctes et réelles telles que:

$$x_0 < x_1 < \dots < x_m$$

et  $m+1$  nombres réels  $y_i$ ,  $i=0,1,\dots,m$ . En général les valeurs  $y_i$  sont les valeurs en  $x_i$  d'une fonction  $f$ , connue partout ou seulement pour les abscisses  $x_i$ .

Soient

$$\mathcal{P}_n = \{ \text{polynômes de degré } \leq n \} .$$

Peut-on trouver un polynôme  $P \in \mathcal{P}_n$  tel que:

$$P(x_i) = y_i, \quad i=0,1,\dots,m,$$

si un tel polynôme existe, il s'écrit sous forme canonique

$$P(x) = \sum_{j=0}^n a_j x^j$$

et doit vérifier

$$\sum_{j=0}^n a_j \cdot x_i^j = y_i \quad i=0,1,\dots,m$$

Ceci est un système linéaire de  $m+1$  équations à  $n+1$  inconnues. On montre [13] que:

- si  $n > m$  il y a une solution avec une indétermination de degré  $(n-m)$ .
- si  $n < m$  il n'y a en général pas de solution.
- si  $n = m$  il y a une et une seule solution.

Dans le cas présent, nous sommes intéressés par un polynôme de degré  $n < m$ , car dans la pratique le degré  $n$  choisi est petit par rapport au nombre de points  $m+1$ . Nous serons même un peu plus stricts en imposant  $n \leq 8$ .

Nous avons vu que pour  $n < m$  il n'existe en général pas de polynôme  $P \in \mathcal{P}_n$  qui vérifie les  $m+1$  conditions

$$P_n(x_i) = y_i \quad i=0,1,\dots,m .$$

Dans ce cas, considérons la quantité

$$Q(P) = \sum_{j=0}^m (y_j - P_n(x_j))^2 \quad (3.10)$$

Le problème de l'ajustement au sens des moindres carrés consiste à déterminer, s'il existe un élément  $P \in \mathcal{P}_n$  tel que (3.10) soit minimum.

Introduisons les notations suivantes:

$$P_n(x) = a_0 + a_1 x + \dots + a_n x^n$$

et considérons les vecteurs définis par:

$$U_j = \begin{pmatrix} x_0^j \\ x_1^j \\ \vdots \\ x_m^j \end{pmatrix} \in \mathbb{R}^{m+1}, \quad y = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_m \end{pmatrix} \in \mathbb{R}^{m+1}, \quad a = \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} \in \mathbb{R}^{n+1}$$

alors  $Q(P)$  prend la forme

$$Q(P) = \left\| y - \sum_{j=0}^n a_j \cdot u_j \right\|^2 \quad (\text{norme euclidienne})$$

$Q(P)$  sera minimum si  $\sum_{j=0}^n a_j \cdot u_j$  est la projection orthogonale de  $y$

sur la variété linéaire de  $R^{m+1}$  engendrée par les  $u_j, j=0,1,\dots,n$ .

Ecrivons que la quantité  $(y - \sum_{j=0}^n a_j \cdot u_j)$  est orthogonale à tous les vecteurs  $u_j, j=0,1,\dots,n$ .

$$\left\langle y - \sum_{j=0}^n a_j \cdot u_j, u_p \right\rangle = 0 \quad p = 0,1,\dots,n$$

c'est-à-dire

$$\sum_{j=0}^n a_j \langle u_j, u_p \rangle = \langle y, u_p \rangle \quad p=0,1,\dots,n \quad (3.11)$$

Soit la matrice  $M \in \mathcal{M}_{(m+1, n+1)}(p)$ , telle que chaque colonne soit respectivement égale aux vecteurs  $u_j, j=0,1,\dots,n$ .

La formule (3.11) devient:

$$M^T \cdot M \cdot a = M^T \cdot y \quad (3.12)$$

Posons  $B = M^T \cdot M$  et  $C = M^T \cdot y$ , alors (3.12) devient:

$$B \cdot a = C \quad (3.13)$$

### 3.3.2.1 Propriétés de la matrice B

1. Elle est de dimensions  $(n+1, n+1)$  donc d'un ordre relativement petit par rapport à  $m+1$ .

2. Elle est symétrique et réelle

$$- B^T = (M^T \cdot M)^T = M^T \cdot M$$

$$B^T = B$$

- réelle car tous ses éléments sont réels.

3. Elle est définie positive ( $x^T \cdot B \cdot x > 0 \quad \forall x \in \mathbb{R}^{n+1} \quad x \neq 0$ )

En effet

$$x^T \cdot B \cdot x = x^T \cdot M^T \cdot M \cdot x = ||M \cdot x||^2 \geq 0 \quad \forall x \in \mathbb{R}^{n+1}$$

Montrons que si  $x \neq 0$  alors  $x^T \cdot B \cdot x > 0$ .

Raisonnons par l'absurde: s'il existait  $x \neq 0$  avec

$$x^T \cdot B \cdot x = 0$$

celà s'écrirait  $||M \cdot x||^2 = 0$  soit  $M \cdot x = 0$

or

$$M = \begin{pmatrix} 1 & x_0 & \dots & x_0^n \\ 1 & x_1 & \dots & x_1^n \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ 1 & x_m & \dots & x_m^n \end{pmatrix}$$

extrayons de M, la matrice carrée principale  $\tilde{M}$  formée des n+1 premières lignes de M :

$$\tilde{M} = \begin{pmatrix} 1 & x_0 & \dots & x_0^n \\ 1 & x_1 & \dots & x_1^n \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ 1 & x_n & \dots & x_n^n \end{pmatrix}$$

Dire que  $M.x = 0$  avec  $x \neq 0$  revient à dire

$$\tilde{M}.x = 0 \quad \text{avec} \quad x \neq 0$$

La matrice  $\tilde{M}$  est une matrice de Vandermonde, donc

$$D(\tilde{M}) = \prod_{i > j} (x_i - x_j)$$

or tous les  $x_i$  sont distincts (par hypothèse) donc

$$D(\tilde{M}) \neq 0$$

La matrice  $\tilde{M}$  est donc non singulière, par conséquent

$$\tilde{M}.x = 0 \quad \text{avec} \quad x \neq 0 \text{ est impossible}$$

de même  $M.x = 0$  est impossible avec  $x \neq 0$ , ce qui achève la démonstration.

### 3.3.2.2 Mise en oeuvre de l'algorithme .

Pour calculer les coefficients du polynôme  $P(x)$  qui minimise (3.10) nous sommes amenés à résoudre le système linéaire (3.13) avec  $B$  matrice carrée  $(n+1, n+1)$ , symétrique définie positive. Dans ces conditions l'algorithme de Choleski [ 14 ] semble tout indiqué. Cette méthode a donné naissance à la commande AJUSMDR du système PGAN (Cf.2.2.2) .

#### Rappel de l'algorithme de Choleski

Soit  $B$  symétrique, définie, positive, cherchons  $R$  triangulaire inférieure telle que  $B = R.R^T$ .

Nous allons procéder par identification et posons

$$R = (r_{ij}) \quad \text{avec} \quad r_{ij} = 0 \quad \text{si} \quad i < j$$

On doit donc avoir,

$$b_{ij} = \sum_{k=1}^i r_{ik} \cdot r_{kj}^T = \sum_{k=1}^i r_{ik} \cdot r_{jk} \quad i \leq j$$

$$b_{ij} = \sum_{k=1}^i r_{ik} \cdot r_{jk} \quad \text{pour } i \leq j \quad (3.14)$$

- Identifions la première colonne.

Cela donne pour  $r_{11}$

$$b_{11} = r_{11} \cdot r_{11} = r_{11}^2$$

$$r_{11} = \pm \sqrt{b_{11}} \neq 0$$

on choisira la détermination positive donc

$$r_{11} = + \sqrt{b_{11}}$$

pour les termes  $r_{j1}$ , on aura

$$b_{ij} = r_{11} \cdot r_{j1}$$

$$r_{j1} = \frac{b_{1j}}{r_{11}} \quad j = 2, 3, \dots, n$$

- Identifions la  $i^{\text{ème}}$  colonne

Cela donne pour  $r_{ii}$

$$b_{ii} = \sum_{k=1}^i r_{ik} \cdot r_{ik} \quad (i=j)$$

$$r_{ii} = \sqrt{b_{ii} - \sum_{k=1}^{i-1} r_{ik}^2} \quad (i=j) \quad (3.15)$$

on a choisi la détermination positive.

Pour les termes  $r_{ji}$  ( $j > i$ ), on aura:

$$b_{ij} = \sum_{k=1}^{i-1} r_{ik} \cdot r_{jk} + r_{ii} \cdot r_{ji}$$

$$r_{ji} = \frac{1}{r_{ii}} \left[ b_{ij} - \sum_{k=1}^{i-1} r_{ik} \cdot r_{jk} \right] \quad (j > i) \quad (3.16)$$

A l'aide des formules (3.15) et (3.16) on détermine la matrice R.

Une fois R trouvée, le système (3.13) s'écrit

$$RR^T a = C$$

On pose  $R^T a = Y$  et l'on résoud les deux systèmes

$$\begin{cases} RY = C \\ R^T a = Y \end{cases} \quad (3.17)$$

qui sont tous les deux triangulaires donc très faciles à résoudre.

### 3.4 RECHERCHE DE COURBES DE NIVEAUX SUR UN MAILLAGE RECTANGULAIRE.

Soient n abscisses distinctes appartenant à l'intervalle [ a, b ] telles que:

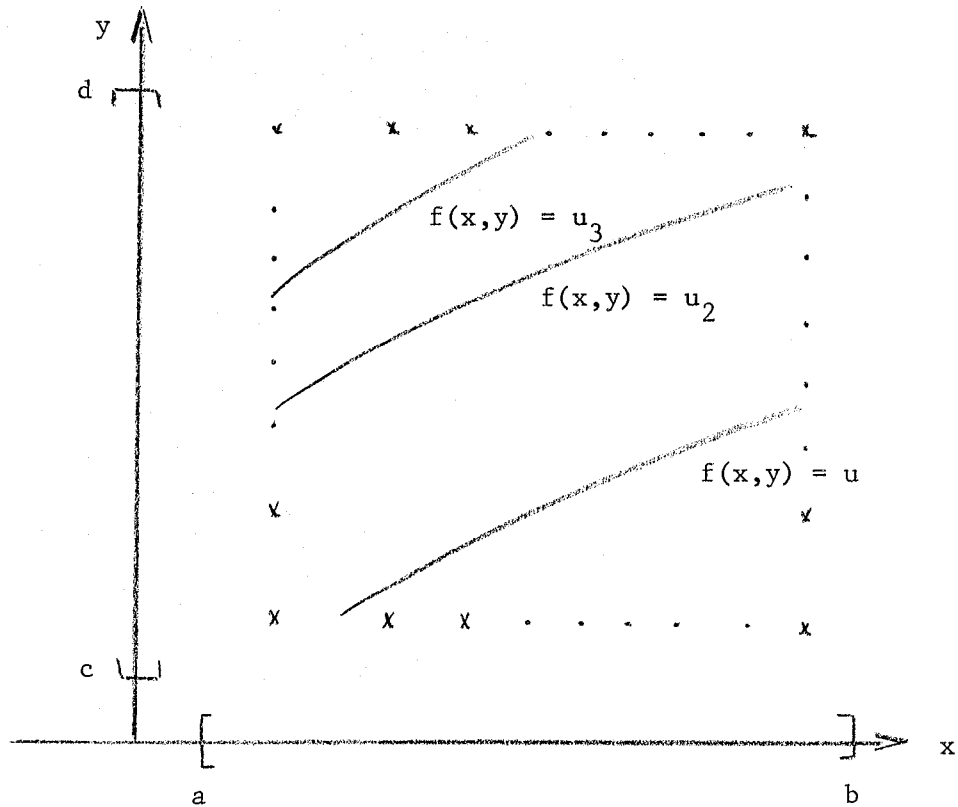
$$a \leq x_1 < x_2 < \dots < x_n \leq b$$

et m ordonnées distinctes appartenant à l'intervalle [c,d] telles que:

$$c \leq y_1 < y_2 < \dots < y_m \leq d$$

et  $n * m$  nombres réels  $z_i$ ,  $i=1, \dots, n * m$ . Les  $z_i$  représentent les valeurs au point  $(x_i, y_i)$ , d'une certaine fonction  $f(x, y)$ .

Le problème est de déterminer une approximation des graphes  $f(x, y) = \text{cte}$ .



Pour résoudre ce problème, il faut connaître les  $x_i$ ,  $y_j$  et  $z_k$ ,  $i=1 \dots n$ ;  $j=1 \dots m$ ,  $k=1 \dots n * m$ . S'ils ne le sont pas, on va indiquer, moyennant quelques informations, un moyen de déterminer un maillage rectangulaire à l'aide de fonctions splines d'interpolation à deux variables sur un domaine rectangulaire.



3.4.1 Fonction spline d'interpolation à 2 variables sur un domaine rectangulaire.

Soient:

- E l'espace vectoriel topologique des fonctions réelles  $u(t)$  définies sur un ouvert  $\Omega$  de  $R^2$ :  $t=(t_1, t_2) \in \Omega$  (dans le cas présent, l'intérieur d'un rectangle).
- N le sous-espace vectoriel fermé de E engendré par:

$$t \rightarrow 1; \quad t \rightarrow t_1; \quad t \rightarrow t_2$$

- G l'ensemble convexe fermé de E:

$$G = \{ u \in E : u(t^i) = \alpha_i, \quad 1 \leq i \leq n \} \quad \text{où}$$

$t^1, t^2, \dots, t^n$  sont n points fixés distincts de  $\Omega$   
 $\alpha_1, \alpha_2, \dots, \alpha_n$  sont n nombres réels.

- V le sous-espace vectoriel de E, des fonctions réelles définies continues sur  $\Omega$ , dont les dérivées du 1er et du 2ème ordre, au sens des distributions, appartiennent à  $\mathcal{L}^2(\Omega)$ .

Si les  $\alpha_i$  sont tels que:  $G \cap N = \emptyset$  et si  $n > 3$ , ou montre [ 15 ] qu'il existe un élément  $\sigma \in V$  et un seul, appartenant à G et tel que:

$$J(\sigma, \sigma) = \int_{\Omega} \left[ \left( \frac{\partial^2 \sigma}{\partial t_1^2} \right)^2 + 2 \left( \frac{\partial^2 \sigma}{\partial t_1 \partial t_2} \right)^2 + \left( \frac{\partial^2 \sigma}{\partial t_2^2} \right)^2 \right] d\omega(t)$$

soit minimum.

$\sigma$  est la fonction spline d'interpolation et est de la forme:

$$\sigma(t) = p(t) + \sum_{j=1}^{n-3} a_j [c_0^j K(t, t^j) + c_1^j K(t, t^{j+1}) + c_2^j K(t, t^{j+2}) + c_3^j K(t, t^{j+3})]$$

où  $p \in N$  et  $a_j \in R$ .

les coefficients  $C_0^j, C_1^j, C_2^j, \text{ et } C_3^j \in \mathbb{R}$  ( $j = 1, 2, \dots, n-3$ ) sont tels que:

$$a) \rho_j = C_0^j \delta_{t^j} + C_1^j \delta_{t^{j+1}} + C_2^j \delta_{t^{j+2}} + C_3^j \delta_{t^{j+3}}, \text{ soient } n-3$$

éléments indépendants du dual  $E'$  de  $E$ .  $\delta_{t^j}$ : mesure de Dirac au point  $t^j$ .

$$b) \langle \rho_j, 1 \rangle = \langle \rho_j, t_1 \rangle = \langle \rho_j, t_2 \rangle = 0$$

$K(t, \theta)$  ( $t$  et  $\theta \in \Omega$ ) est le noyau reproduisant d'Arousuzjn-Bergman du sous-espace Hilbertien  $\mathcal{K}$  de  $E$ :

$$\mathcal{K} = \{ u \in V : \int_{\partial\Omega} u dl = \int_{\partial\Omega} u t_1 dl = \int_{\partial\Omega} u t_2 dl = 0 \},$$

muni du produit scalaire

$$(u|V) = J(u, V) = \int_{\Omega} \left[ \frac{\partial^2 u}{\partial t_1^2} \cdot \frac{\partial^2 V}{\partial t_1^2} + 2 \frac{\partial^2 u}{\partial t_1 \cdot \partial t_2} \cdot \frac{\partial^2 V}{\partial t_1 \partial t_2} + \frac{\partial^2 u}{\partial t_2^2} \cdot \frac{\partial^2 V}{\partial t_2^2} \right] d\omega(t)$$

le calcul du noyau  $K(t, \theta)$  est basé sur l'expression

$$K = \sum_{i \in I} e_i \otimes e_i \quad \text{où } (e_i)_{i \in I}$$

C'est une base Hilbertienne de  $\mathcal{K}$ .

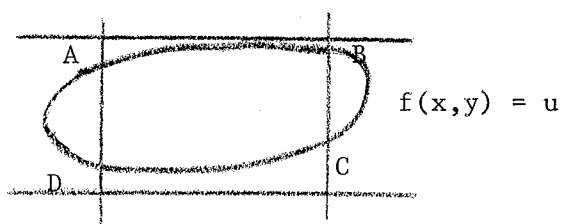
En résumé, cet algorithme nous permet, connaissant quelques valeurs de la fonction dans un domaine rectangulaire, de calculer sur un maillage rectangulaire, la valeur que prendrait une fonction aussi lisse que possible (dans le sens précisé ci-dessus) prenant les mêmes valeurs en ces points.

Cet algorithme décrit dans [ 16 ] a donné lieu à la commande TOMAN du système PGAN. (cf. 2.2.2).

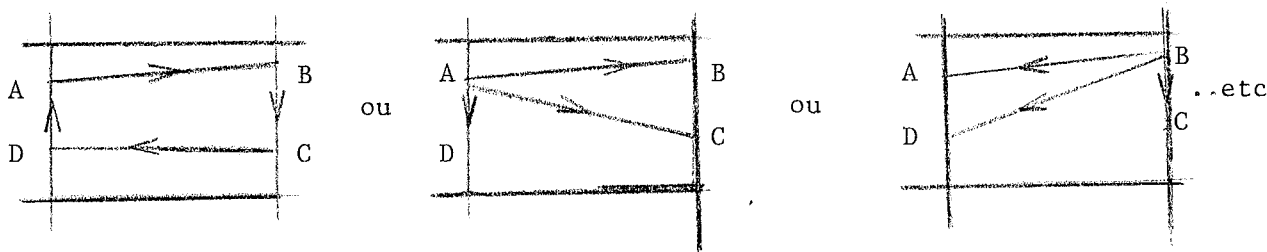
### 3.4.2 Stratégie utilisée

Le maillage rectangulaire étant donné ou calculé par la méthode indiquée au paragraphe 3.4.1, la recherche des courbes de niveaux n'est pas résolue pour autant. Ce problème est en réalité difficile, les exemples suivants vont illustrer cette difficulté.

a- Considérons le maillage suivant où une courbe de niveau coupe la même maille en 4 points A, B, C, D:

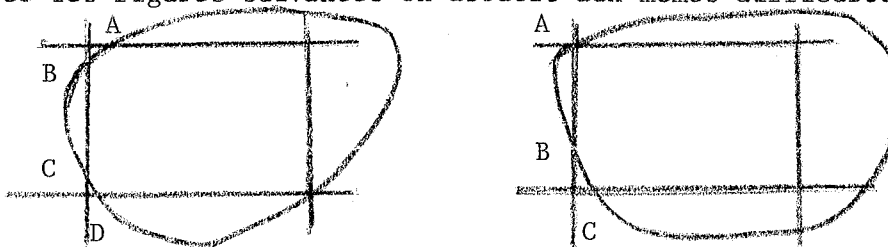


Par interpolation, on peut arriver à connaître la valeur des coordonnées des points A,B,C,D. Pour tracer la courbe de niveau, une hypothèse de travail serait de joindre entre eux les points tels que:  $f(x,y) = u$ , qui nous conduit dans ce cas aux figures suivantes:



Comme on peut le remarquer, la solution est loin d'être unique, en effet, il est impossible de connaître l'ordre dans lequel les points doivent être joints.

b- Avec les figures suivantes on aboutit aux mêmes difficultés:



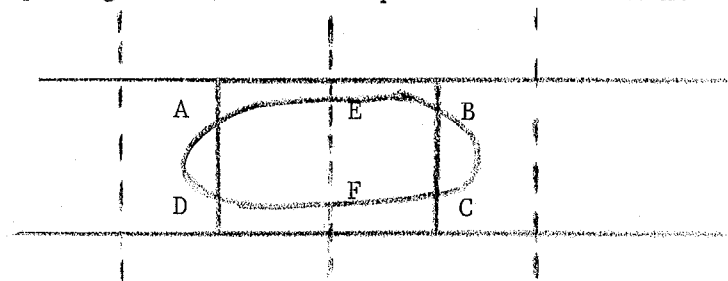
Ces quelques exemples nous ont conduit à faire l'hypothèse supplémentaire suivante: Une courbe de niveau ne peut pas couper une maille en plus de 2 points.

La courbe de niveau sera obtenue en joignant l'ensemble des points où la fonction prend la même valeur.

Lorsque les cas (a) ou (b) se produiront, on adoptera la technique suivante: On forme un nouveau maillage en partageant alternativement en deux chaque intervalle en x ou en y, jusqu'à ce que notre hypothèse soit vérifiée. Si le nombre de points du nouveau maillage dépasse une certaine limite pour laquelle notre hypothèse ne s'est pas vérifiée, la méthode sera dite impossible. Le nouveau maillage est calculé à l'aide de fonctions splines d'interpolation d'ordre 2(cf.3.2.1).

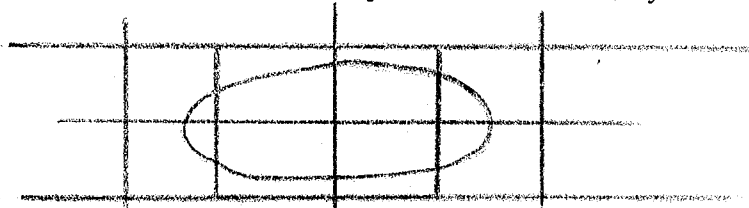
Pour l'exemple a, on obtient:

- en partageant en deux chaque intervalle en x.



impossible.

- en partageant en deux chaque intervalle en y.



hypothèse vérifiée

Cette méthode ne prétend pas résoudre tous les problèmes mais, peut, dans certains cas, donner des résultats intéressants (cf.4 ). Elle a donné lieu à la commande GRILLE du système PGAN cf(2.2.2).

### 3.5 RESOLUTION DES SYSTEMES DIFFERENTIELS DU PREMIER ORDRE.

On s'intéresse aux systèmes différentiels de N équations de la forme:

$$\left\{ \begin{array}{l} Z_1'(S) = F_1(S, Z_1(S), \dots, Z_N(S)) \\ \vdots \\ Z_N'(S) = F_N(S, Z_1(S), \dots, Z_N(S)) \end{array} \right.$$

avec les conditions initiales suivantes:

$$\left\{ \begin{array}{l} Z_1(x_0) = y_1^0 \\ \vdots \\ Z_N(x_0) = y_N^0 \end{array} \right.$$

dont les solutions sont:  $y_1(S), \dots, y_N(S)$ .

En introduisant les notations suivantes:

$$T = \begin{bmatrix} z_1 \\ \vdots \\ z_N \end{bmatrix}, \quad F = \begin{bmatrix} F_1 \\ \vdots \\ F_N \end{bmatrix}, \quad Y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}, \quad Y_0 = \begin{bmatrix} y_1^0 \\ \vdots \\ y_N^0 \end{bmatrix}, \quad \tilde{T} = [z_1, \dots, z_N]$$

Le système s'écrit:

$$\boxed{\begin{array}{l} T'(S) = F(S, \tilde{T}) \\ T(x_0) = Y_0 \end{array}}$$

La solution de ce système sera obtenue par la méthode de Runge-Kutta dont on rappelle brièvement la méthode.

### 3.5.1 Rappel de la méthode de Runge-Kutta

Partant de:  $\bar{Y}(x_0) = Y(x_0) = Y_0$

on évalue l'approximation  $\bar{Y}(x_{n+1})$  de  $Y(x_{n+1})$  à partir de  $\bar{Y}(x_n)$  de la façon suivante:

On pose:

$$Y_{no} = \bar{Y}(x_n)$$

et

$$Y_{n\alpha} = Y_{no} + h \sum_{\beta=0}^{\alpha-1} A_{\alpha\beta} F[ x_{n\beta}, \tilde{Y}_{n\beta} ], \quad \alpha=1, \dots, q$$

avec  $x_{n\alpha} = x_n + \theta_\alpha h$ ,  $\theta_0 = 0$ ,  $\theta_1 \neq 0$ ,  $\theta_q = 1$ .

Nous prenons:

$$\bar{Y}(x_{n+1}) = Y_{nq}$$

Nous avons

$$\bar{Y}(x_{n+1}) = Y(x_{n+1}) + O(h^q)$$

Cette méthode a donné naissance à la commande SYSDIFRK du système PGAN (cf.2.2.2) et on trouvera dans [17] la description détaillée de cet algorithme ainsi que la procédure Algol correspondante.



## C H A P I T R E 4

### RESULTATS OBTENUS

#### 4.1 INTRODUCTION.

Ce chapitre a pour but de mettre en relief, sur des exemples concrets, les résultats que l'on peut obtenir du système PGAN. Principalement orienté vers des problèmes dont la solution peut s'exprimer sous forme de courbes, il permet, grâce aux commandes et fonctions du système de:

- Sélectionner rapidement sa méthode
- Comparer diverses méthodes
- Visualiser sous forme de courbes la solution
- Visualiser les résultats numériques correspondant à la solution calculée.
- Comparer diverses solutions correspondant à des valeurs différentes du même paramètre.
- Prendre une copie solide (hard copy) de l'image affichée au 2250.
- Sortir un listing des résultats.

Toutes ces possibilités offertes par le système en font un outil extrêmement pratique pour traiter de façon conversationnelle et interactive les problèmes évoqués au chapitre 3. Dans une certaine mesure, il peut venir en aide au chercheur pour étudier certaines propriétés des courbes; rechercher le nombre de points minimum pour qu'une courbe ait une allure donnée, faire des rapprochements entre diverses courbes, chose difficile à réaliser si on ne dispose que de colonnes de chiffres. Chacun de ces aspects, pratique et



théorique sera illustré par un ou plusieurs exemples.

#### 4.2 RESULTATS THEORIQUES.

Lors de l'étude des fonctions splines dépendant d'un paramètre et ajustant un certain nombre de points, étude rendue possible grâce à la commande AJUSPQ du système PGAN, on a pu constater sur l'écran du 2250 (figure 8) que pour certains jeux d'abscisses, les diverses courbes obtenues avaient tendance à passer par un ou plusieurs points fixes et ceci pour de très grandes variations du paramètre (de  $10^{+6}$  à  $10^{-6}$  par exemple). On a cherché à formaliser cette propriété et on est arrivé aux résultats suivants: Etant données  $n$  abscisses fixes et distinctes de l'intervalle  $[a,b]$

$$a < x_1 < x_2 < \dots < x_n < b$$

et  $n$  nombres réels  $y_i$ ,  $i=1\dots n$ , si l'on considère la fonction "spline d'ajustement" d'ordre  $q$  ( $n \geq q$ ) définie par:

$$\sigma_\rho(x) = \sum_{j=0}^{q-1} \alpha_j x^j + \sum_{i=1}^n d_i \frac{(x-x_i)_+^{2q-1}}{(2q-1)!}, \quad x \in [a,b] \quad (4.1)$$

$$\sigma_\rho(x) + \frac{(-1)^q}{\rho} d_i = y_i, \quad i=1\dots n \quad (4.2)$$

$$\sum_{i=1}^n d_i (x_i)^k = 0, \quad k=0,1\dots(q-1) \quad (4.3)$$

on montre [12] que  $\sigma_\rho(x)$  minimise la quantité

$$\int_a^b [f^{(q)}(x)]^2 dx + \rho \sum_{i=1}^n (f(x_i) - y_i)^2, \quad f \in H^q[a,b]$$

(avec  $\rho > 0$ ).

On se propose de démontrer le théorème suivant:

THEOREME : Il existe des jeux d'abscisses  $x_i, i=1 \dots n$ , pour lesquels les fonctions  $\sigma_\rho(x)$  ont, en certaines abscisses, des valeurs constantes, indépendantes de  $\rho$ .

Démonstration

L'expression (4.1) dépend de  $n+q$  paramètres:  $\alpha_j, j=0, 1 \dots (q-1)$  et  $d_i, i=1 \dots n$ ; d'autre part, les expressions (4.2) et (4.3) conduisent à un système linéaire de  $n+q$  relations linéaires entre ces  $n+q$  paramètres. Il s'écrit sous forme matricielle.

$\frac{(-1)^q}{\rho}$	0	.	.	.	0	1	$x_1$	$x_1^2$	...	$x_1^{q-1}$	$d_1$	$y_1$
$\frac{(x_2 - x_1)^{2q-1}}{(2q-1)!}$	$\frac{(-1)^q}{\rho}$	0	...	0	1	$x_2$	$x_2^2$	...	$x_2^{q-1}$		$d_2$	$y_2$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\frac{(x_n - x_1)^{2q-1}}{(2q-1)!}$	$\frac{(x_n - x_2)^{2q-1}}{(2q-1)!}$	.	.	.	$\frac{(-1)^q}{\rho}$	1	$x_n$	$x_n^2$	...	$x_n^{q-1}$	$d_n$	$y_n$
1	1	.	.	.	1	0	0	0	...	0	$\alpha_0$	0
$x_1$	$x_2$	.	.	.	$x_n$	0	0	0	...	0	$\alpha_1$	0
$x_1^2$	$x_2^2$	.	.	.	$x_n^2$	0	0	0	...	0	$\alpha_2$	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	0
$x_1^{q-1}$	$x_2^{q-1}$	.	.	.	$x_n^{q-1}$	0	0	0	...	0	$\alpha_{q-1}$	0

Introduisons les notations suivantes:

$$B = \begin{pmatrix} 1 & x_1 & x_1^2 & \dots & x_1^{q-1} \\ 1 & x_2 & x_2^2 & \dots & x_2^{q-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^{q-1} \end{pmatrix}, D = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{pmatrix}, Y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, \alpha = \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{q-1} \end{pmatrix}$$

$$K = \begin{pmatrix} 0 & 0 & \dots & 0 \\ \frac{(x_2 - x_1)^{2q-1}}{(2q-1)!} & 0 & & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \frac{(x_n - x_1)^{2q-1}}{(2q-1)!} & \frac{(x_n - x_2)^{2q-1}}{(2q-1)!} & \dots & 0 \end{pmatrix}, \Psi(x) = \begin{pmatrix} \frac{(x-x_1)^{2q-1}}{(2q-1)!} \\ \frac{(x-x_2)^{2q-1}}{(2q-1)!} \\ \vdots \\ \frac{(x-x_n)^{2q-1}}{(2q-1)!} \end{pmatrix}, \mu(x) = \begin{pmatrix} 1 \\ x \\ \vdots \\ x^{q-1} \end{pmatrix}$$

Les vecteurs D et  $\alpha$  sont déterminés par le système:

$$\left. \begin{aligned} \left[ \frac{(-1)^q}{\rho} I + K \right] D + B\alpha &= Y \\ B^T D &= 0 \end{aligned} \right\} \quad (4.5)$$

multiplions à gauche la première relation de (4.5) par  $B^T$

$$\frac{(-1)^q}{\rho} B^T D + B^T K D + B^T B\alpha = B^T Y$$

en tenant compte de la deuxième relation (4.5), on obtient le système:

$$\left. \begin{aligned} B^T K D + B^T B\alpha &= B^T Y \\ B^T D &= 0 \end{aligned} \right\} \quad (4.6)$$

Soient deux vecteurs quelconques U et V d'ordre q :

$$U = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_q \end{pmatrix}, \quad V = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_q \end{pmatrix}$$

On forme alors toutes les combinaisons linéaires entre les vecteurs D et  $\alpha$ , indépendantes de  $\rho$ , en écrivant :

$$U^T B^T K D + U^T B^T B\alpha + V^T B^T D = U^T B^T Y$$

ou

$$U^T B^T B\alpha + [U^T B^T K + V^T B^T] D = \text{cte} \quad (\text{indépendante de } \rho) \quad (4.7)$$

Or, avec les notations introduites précédemment, (4.1) peut s'écrire :

$$\sigma_{\rho}(x) = \mu^T(x) \cdot \alpha + \Psi^T(x) \cdot D \quad (4.8)$$

si l'on a :

$$\sigma_{\rho}(x) = \text{cte} \quad \forall \rho > 0$$

on aura

$$\mu^T(x) \cdot \alpha + \Psi^T(x) \cdot D = \text{cte} \quad \forall \rho > 0 .$$

Ceci est une expression linéaire entre les vecteurs D et  $\alpha$ ; or toutes les expressions linéaires entre D et  $\alpha$  sont données par (4.5), en particulier, il existera, deux vecteurs U et V tels que

$$\mu^T(x) \cdot \alpha + \Psi^T(x) \cdot D = U^T B^T B \alpha + [ U^T B^T K + V^T B^T ] D$$

en identifiant terme à terme on obtient :

$$\mu^T(x) = U^T B^T B$$

$$\Psi^T(x) = U^T B^T K + V^T B^T$$

ou encore

$$\mu(x) = B' B U$$

$$\Psi(x) = B V + K^T B U$$

(4.9)

ce qui peut être considéré comme un système de  $n+q$  équations entre ( $U, V, x, x_i$ ), c'est-à-dire  $n + 2q + 1$  inconnues.

Remarque. Si les graphes de  $\sigma_{\rho}$  ont un ou plusieurs points fixes correspondant à un jeu d'abscisses  $x_i, i=1 \dots n$ , alors pour tout choix des ( $y_i$ ), l'abscisse de ce ou ces points fixes reste constante.

Pour s'en assurer, il suffit de remarquer que les relations données par (4.9) ne font pas intervenir Y pour la détermination de x, U et V.

Pour déterminer ces points, on dispose de n+q relations données par (4.9) entre les n + 2q + 1 inconnues:

$$x, x_i (i = 1..n), U \text{ et } V.$$

On peut s'en fixer q à priori et déterminer les n+q restantes, dépendant d'un paramètre arbitraire a. En général, on se fixera à priori q abscisses. Explicitons les calculs sur un cas simple (n=3, q=2), en prenant les données suivantes:

$$x_1 = 0, \quad x_2 = 1, \quad x_3 = a \quad (a > 1),$$

si on peut trouver x, U, V, (a) tel que (4.9) soit satisfait, la propriété du ou des points fixes sera réalisée. En développant (4.9) on obtient les équations:

$$\left. \begin{aligned} 1 &= 3 u_1 + (1 + a) u_2 \\ x &= (1+a)u_1 + (1+a^2) u_2 \\ \frac{(x)^3}{3!} &= V_1 + \frac{1}{3!} (u_1 + u_2) + \frac{a^3}{3!} (u_1 + a u_2) \\ \frac{(x-1)^3}{3!} &= V_1 + V_2 + \frac{(a-1)^3}{3!} (u_1 + a u_2) \\ \frac{(x-a)^3}{3!} &= V_1 + a V_2 \end{aligned} \right\} (4.10)$$

Cherchons s'il existe un point fixe pour les graphes de  $\sigma_\rho$  tel que:  $0 < x < 1$ . Dans ce cas les équations (4.10) deviennent:

$$1 = 3 u_1 + (1+a) u_2$$

$$x = (1+a)u_1 + (1+a^2) u_2$$

$$\frac{x^3}{6} = V_1 + \frac{1}{6}(u_1 + u_2) + \frac{a^3}{6} (u_1 + a u_2)$$

$$0 = V_1 + V_2 + \frac{(a-1)^3}{6} (u_1 + a u_2)$$

$$0 = V_1 + a V_2$$

qui, après élimination de  $u_1$ ,  $u_2$ ,  $V_1$ ,  $V_2$  conduisent à l'équation en  $a$  et  $x$  suivante

$$(a^2 - a + 1)x^3 - (2a^3 - 2a^2 + 1)x + a(a-1)^2 = 0$$

qui pour  $a = 2$  donne l'équation:

$$3x^3 - 9x + 2 = 0$$

dont les racines sont:

$$- 1.833986$$

$$0.2260739$$

$$1.607913$$

seule la deuxième racine correspond à l'hypothèse faite:  $0 < x < 1$

$$x = 0.2260739$$

De même, cherchons s'il existe un point fixe pour les graphes de  $\sigma_\rho$  tel que  $1 < x < a$  ( $a=2$ ); Dans ce cas les équations (4.10) conduisent après élimination de  $u_1$ ,  $u_2$ ,  $v_1$ ,  $v_2$ , à l'équation en  $a$  et  $x$  suivante:

$$(a^2 - a + 1)x^3 - 3a(a^2 - a + 1)x^2 + (2a^4 - a^3 - a^2 + 4a - 1)x - a^2(a^2 - 2a + 2) = 0$$

qui pour  $a = 2$  donne l'équation:

$$3x^3 - 18x^2 + 27x - 8 = 0$$

dont les racines sont:

3.833986

0.3920878

1.773926

Seule la troisième racine correspond à l'hypothèse faite:  $1 < x < \bar{a}$

$$x = 1.773926$$

On obtient deux points fixes pour les graphes de  $\sigma_p$  dont les abscisses sont:

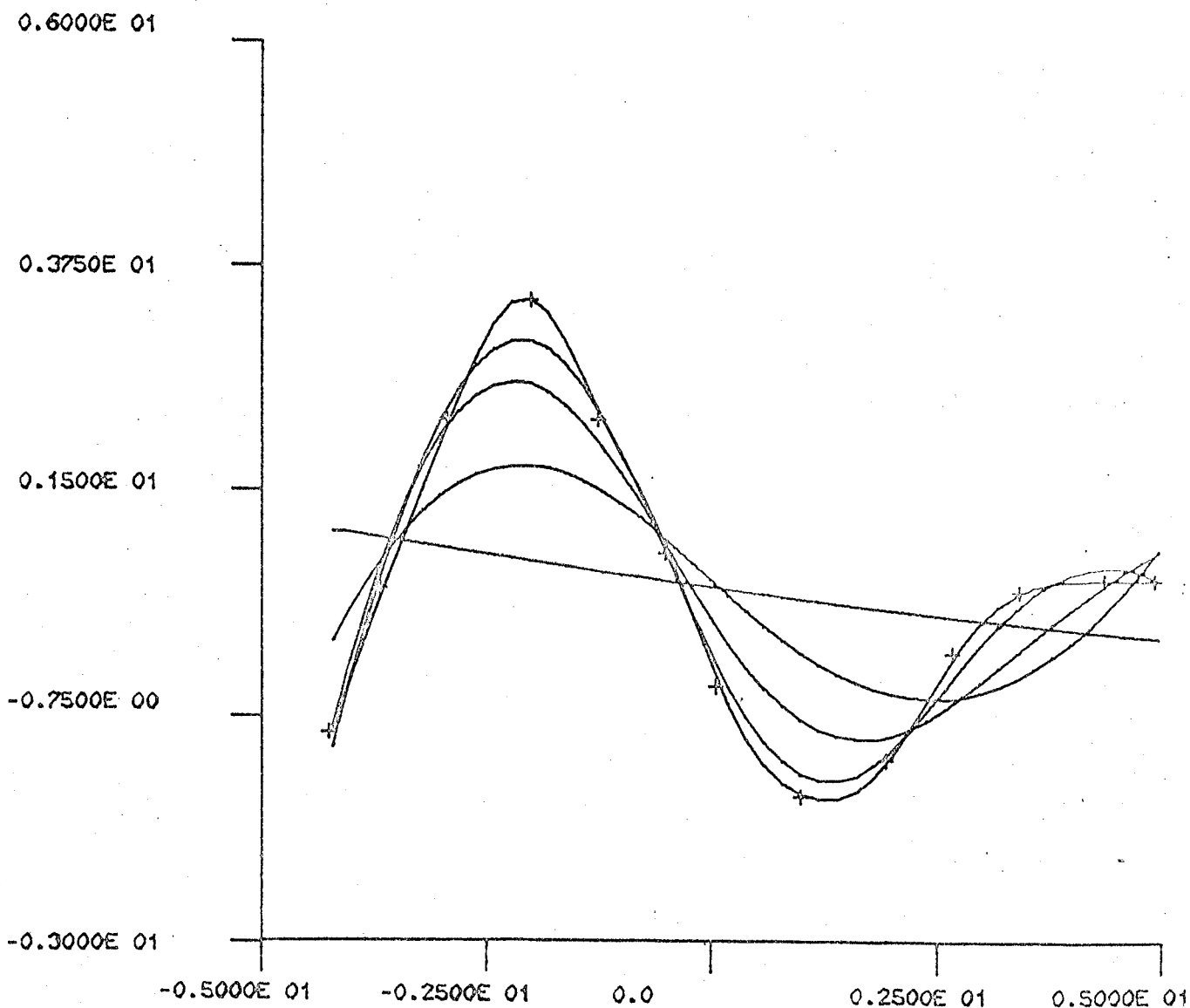
$$x = 0.2260739$$

$$x = 1.773926$$

C'est la commande AJUSP2 du système PGAN qui a permis de mettre en évidence sur la figure 9, les deux points fixes et l'invariance de l'abscisse des points fixes pour des  $y_i$  différents.



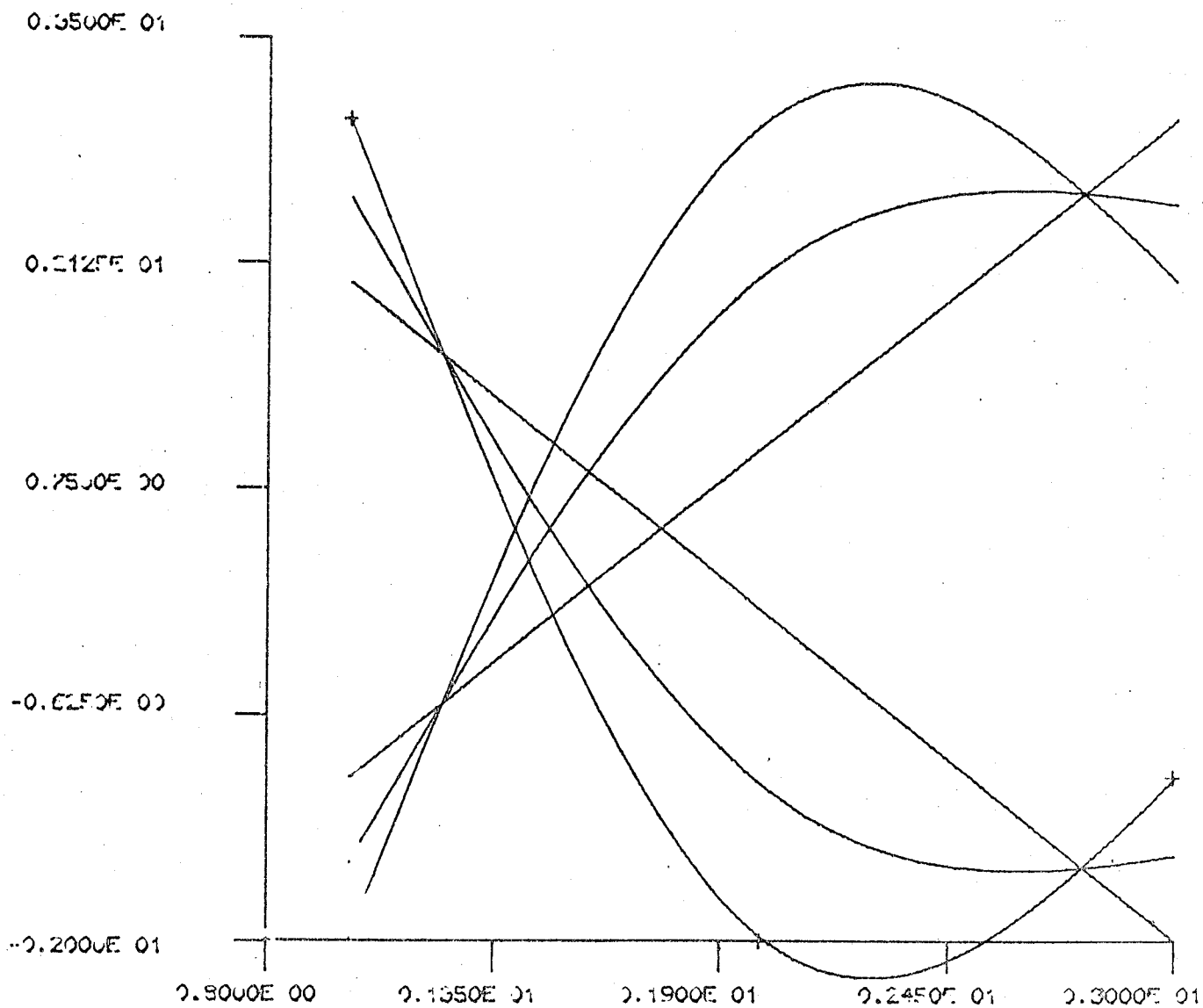
INSTITUT DE MATHÉMATIQUES APPLIQUÉES DE GRENOBLE  
AJUSTEMENT PAR DES SPLINES D'ORDRE Q



ECHELLE : 06/10

Figure 8- Montrant la tendance des courbes à passer par des points fixes pour des valeurs de RAU = 10000.0, 100.0, 1.0, 0.1, 0.000001 .

INSTITUT DE MATHÉMATIQUES APPLIQUÉES DE GRENOBLE  
AJUSTEMENT PAR DES SPLINES D'ORDRE 2



ECHELLE : 06,10

Figure 9- Mise en évidence des points fixes et de l'invariance des abscisses de ceux-ci.

#### 4.3 AUTRES RESULTATS.

Nous allons décrire dans ce paragraphe les quelques problèmes qui nous ont été soumis et pour lesquels nous pensons avoir donné une solution satisfaisante. De plus, chacun de ces exemples illustrera par ses résultats ce à quoi les commandes du système PGAN peuvent nous conduire.

On abordera dans l'ordre

- Le calcul du bilan de masse d'un glacier
- La résolution d'un système différentiel
- Des exemples divers.

##### 4.3.1 Calcul du bilan de masse d'un glacier [18]

Sur un glacier tempéré (masse de glace à 0°C), l'accumulation est principalement la mesure des précipitations solides tombées pendant une année. L'ablation mesure la fusion en surface de la neige ou de la glace. La différence entre l'accumulation et l'ablation est appelée: bilan. Sur la partie basse d'un glacier où le bilan est le plus souvent négatif, la surface est en glace à chaque fin d'été et la mesure du bilan est alors obtenue par différence annuelle d'émergence de balises articulées de 10 m de long et plantées par sondage à vapeur dans la glace. Les mesures sont faites au mois de septembre.

On appelle bilan spécifique, le bilan en un point donné du glacier. Le bilan de masse d'une zone est le bilan total intégré sur cette zone de glacier.

Le système d'axes local (x,y) est défini comme suit:

Ox: axe orienté suivant la ligne moyenne d'écoulement

Oy: axe perpendiculaire à Ox.

Les valeurs du bilan spécifique sont connues en une quinzaine de points irrégulièrement répartis sur la zone d'ablation. Nous voulons obtenir

les valeurs approximées du bilan spécifique aux mailles d'un réseau régulier (une centaine de points). Ceci nous permet de tracer les courbes d'égal bilan ou courbes 'équisomptuaires' (Prof. Lliboutry).

En effet, on prouve expérimentalement que le bilan  $b(x,y,t)$  est fonction de l'exposition du point, de son altitude et des valeurs de températures et de précipitations pour chaque année, notée  $t$ .

$$b(x,y,t) = a_1(t) E(x,y) + a_2(t) \cdot Z + A(f(T), P)$$

où

$E(x,y)$  = énergie solaire potentielle reçue en un point

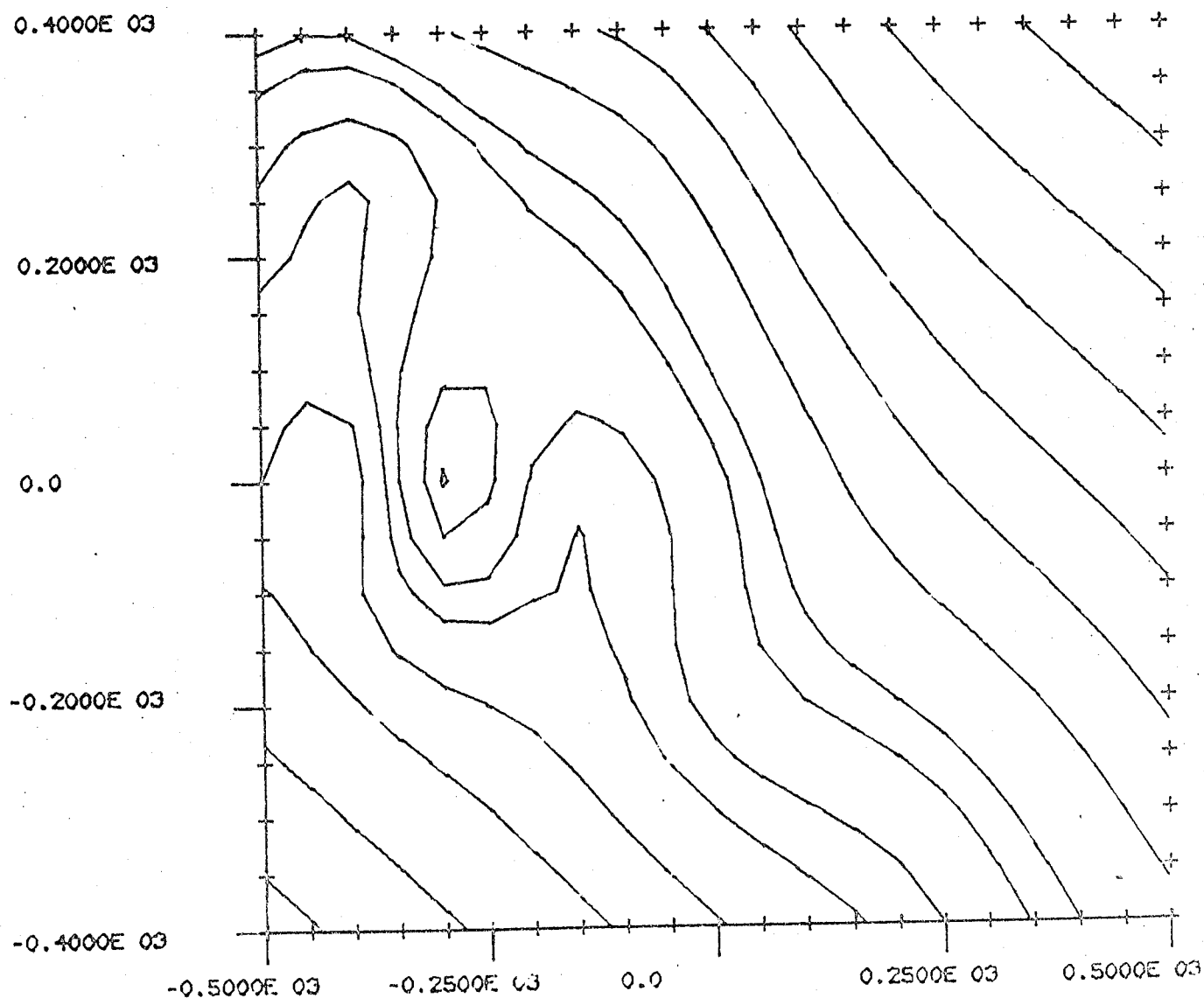
$Z$  = altitude de ce point

$T$  = températures relevées à une station proche du glacier

$P$  = quantités de précipitation relevées à une station proche du glacier.

Nous nous sommes intéressés au tracé des équisomptuaires; pour cela, on a divisé un réseau de mailles carrées de 100 m de côté, dont les côtés sont parallèles aux axes  $Ox$ ,  $Oy$ ; le nombre de points du réseau étant de l'ordre de la centaine. La valeur de la fonction  $b$  au noeud de ce réseau a été obtenue grâce à la commande TOMAN du système PGAN. Les données correspondant à ce problème sont décrites au paragraphe (2.2.5.1), en illustration de la commande TOMAN. Les résultats que donne cette commande ont servi de données pour la commande GRILLE qui a permis d'obtenir les équisomptuaires (figure 10) du glacier de Saint Sorlin, situé dans le massif des Grandes Rousses, pour la période 1969-1970. De plus, en interprétant l'allure des équisomptuaires, le glaciologue peut vérifier, à posteriori, si certains relevés de balises ne sont pas erronés. En effet, l'introduction d'un point aberrant, occasionne des erreurs importantes sur l'allure du réseau de courbes.

INSTITUT DE MATHÉMATIQUES APPLIQUÉES DE GRENOBLE  
CALCUL DE COURBES DE NIVEAUX A L'AIDE DE FONCTIONS SPLINES



ECHELLE : 05/10

Figure 10 - Equisomptuaires du glacier de Saint Sorlin pour les valeurs suivantes du bilan : -0.2, -0.4, -0.5, -0.75, -0.85, -1.0, -1.15, -1.25, -1.50, -1.75, -2.0, -2.25, -2.50, -2.75 ; en partant du coin inférieur gauche vers le coin supérieur droit.

#### 4.3.2 Résolution d'un système différentiel du premier ordre.

Le problème posé est celui de l'énergie d'un électron dans un milieu polarisable. On constate qu'un électron libre, introduit dans un liquide isolant y est rapidement piégé. Le Professeur Felici, pour en éclaircir les raisons, a développé une théorie [19] permettant d'atteindre cette énergie lorsque l'interaction avec le milieu n'est due qu'à sa polarisation.

L'électron étant défini par sa fonction d'onde  $\Psi$ , son champ électrique  $E$  est celui de sa charge  $\rho = e|\Psi|^2$  et

$$\text{div } E = 4\pi\rho/\epsilon \quad (\epsilon \text{ primitivité})$$

La réaction du milieu sur l'électron est donnée par le champ électrique  $E'$  de la charge  $\rho'$  équivalente à la polarisation  $P$  et

$$\text{div } E' = 4\pi\rho' = -4\pi \text{ div } P.$$

Le potentiel où se meut l'électron vérifie:

$$\Delta V = -4\pi(\epsilon-1) e|\Psi|^2/\epsilon$$

L'équation d'onde (Schroedinger)

$$[(-\hbar^2/2m)\Delta - eV] \Psi = W \Psi$$

est transformée en divisant par  $\Psi$ , en prenant le laplacien, et en remplaçant  $\Delta V$  par sa valeur ci-dessus.

On obtient:

$$\Delta \left( \frac{\Delta \Psi}{\Psi} \right) = |\Psi|^2/a, \quad a = \frac{R_B \epsilon}{8\pi (\epsilon-1)}$$

où  $R_B$  est le rayon de Bohr (0.53 Å). D'autres considérations conduisant à l'équation:

$$y''/y = \int_0^x \frac{dx}{x^2} - \int_0^x y^2 dx \quad y(0) = y(+\infty) = 0$$

dont il faut déterminer  $y$  vérifiant  $y(0) = y(+\infty) = 0$ .

Pour résoudre ce problème, on est amené à faire les transformations suivantes:

On pose:

$$u(x) = \int_0^x y^2(t) dt$$

$$v(x) = \int_0^x \frac{u(t)}{t^2} dt$$

$$y'(x) = z(x)$$

La formule de départ se transforme alors pour donner le système différentiel du premier ordre:

$$\begin{cases} y'(x) = z(x) \\ z'(x) = y(x) (v(x) - 1) \\ u'(x) = y^2(x) \\ v'(x) = u(x)/x^2 \end{cases}$$

Pour déterminer les conditions initiales, on suppose que la solution  $y(x)$  est de la forme:

$$y(x) = a_1 x + a_2 x^2 + a_3 x^3 + \dots$$

en tenant compte du fait que  $\frac{y''}{y}$  doit être stable lorsque  $x \rightarrow \infty$ , on est amené à prendre  $a_2 = 0$ ; alors:

$$y(x) = a_1 x + a_3 x^3 + \dots$$

d'autre part, on montre que

$$\frac{y''(x)}{y(x)} \xrightarrow{x \rightarrow \infty} -1$$

on en déduit, en première approximation que:

$$a_3 = -\frac{a_1}{6}$$

donc

$$y(x) = a_1 x - \frac{a_1}{6} x^3 + \dots$$

et

$$y'(x) = a_1 - \frac{a_1}{2} x^2 + \dots$$

$$y''(x) = -a_1 + \dots$$

Nous prendrons pour  $y$ , en première approximation

$$y(x) = a_1 x$$

Pour établir les conditions initiales en zéro, on choisira  $x_0$  très près de l'origine; dans ce cas:

$$y(x_0) = a_1 x_0$$

-  $u(x_0)$  est obtenu par:

$$u(x_0) = \int_0^{x_0} u'(t) dt = \int_0^{x_0} y^2(t) dt = \int_0^{x_0} a_1^2 t^2 dt$$

$$u(x_0) = \frac{a_1^2}{3} x_0^3$$

-  $V(x_0)$  est obtenu par:

$$V(x_0) = \int_0^{x_0} V'(t) dt = \int_0^{x_0} \frac{u(t)}{t^2} dt = \int_0^{x_0} \frac{a_1^2 t^3}{t^2} dt$$

$$V(x_0) = \frac{a_1^2}{6} x_0^2$$



-  $z(x_0)$  est obtenu par:

$$z(x_0) = y'(x_0)$$

$$z(x_0) = a_1$$

En résumé on obtient le système:

$$\left\{ \begin{array}{l} y'(x) = z(x) \\ z'(x) = y(x)(v(x)-1) \\ u'(x) = y^2(x) \\ v'(x) = u(x)/x^2 \end{array} \right. \quad \text{avec} \quad \left\{ \begin{array}{l} y(x_0) = a_1 x_0 \\ z(x_0) = a_1 \\ u(x_0) = \frac{a_1^2}{3} x_0^3 \\ v(x_0) = u(x_0)/2 \end{array} \right.$$

Ce système est résolu par la commande SYSDIFRK du système PGAN. Le problème est de trouver la valeur du paramètre  $a_1$  pour lequel la solution vérifie:  $y(0) = 0$  et  $y(\infty) = 0$ , ou trouver une courbe passant par zéro et asymptote à l'axe Ox. La figure 11 montre un certain nombre d'essais effectués.

INSTITUT DE MATHÉMATIQUES APPLIQUÉES DE GRENOBLE  
RÉSOLUTION D'UN SYSTÈME DIFFÉRENTIEL PAR UNE MÉTHODE DE RUNGE-KUTTA

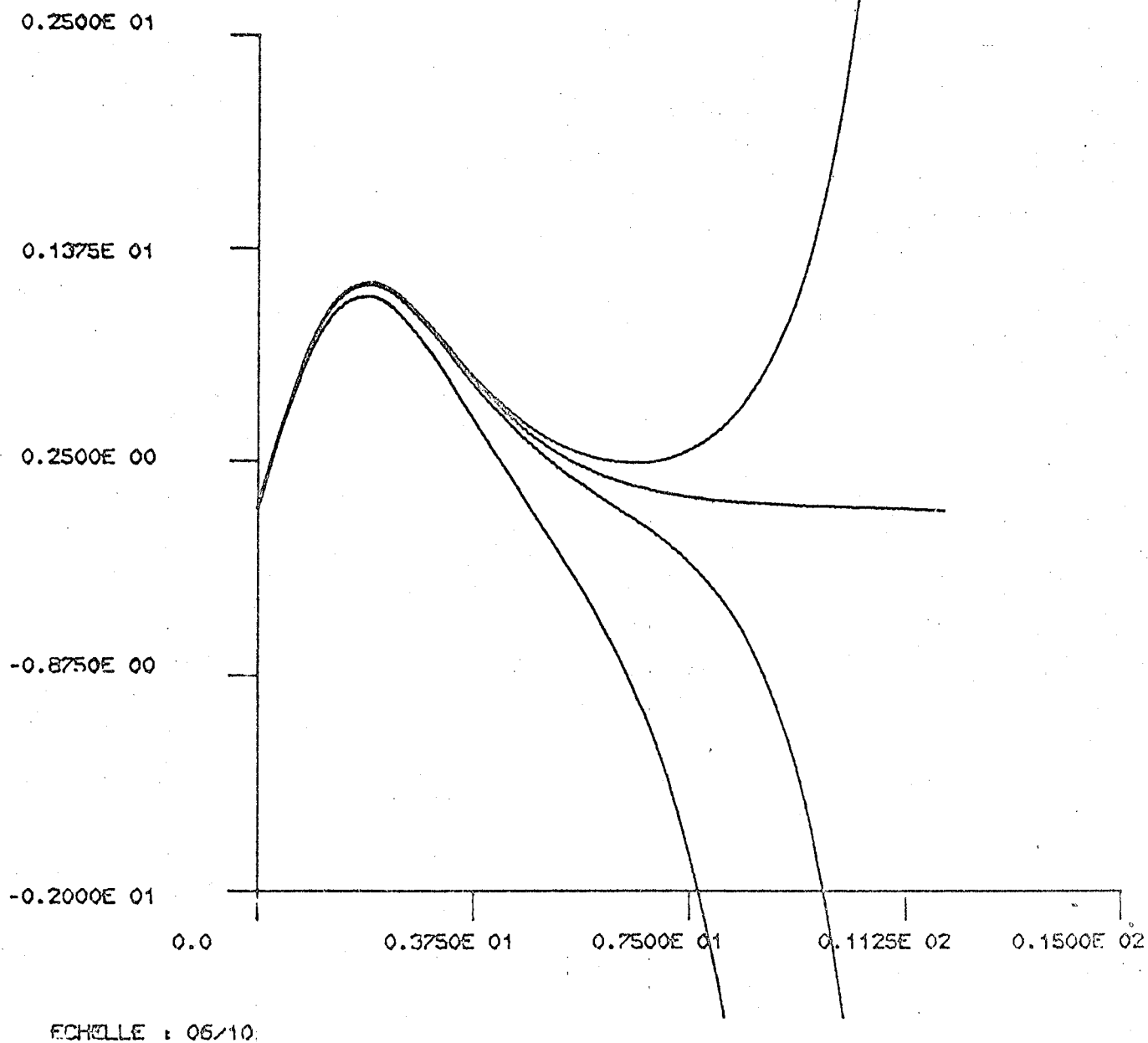


Figure 11 - Différents graphes de  $y(x)$  correspondant aux valeurs 1.085 , 1.081 , 1.070 , 1.030 du paramètre.

#### 4.3.3 Autres exemples.

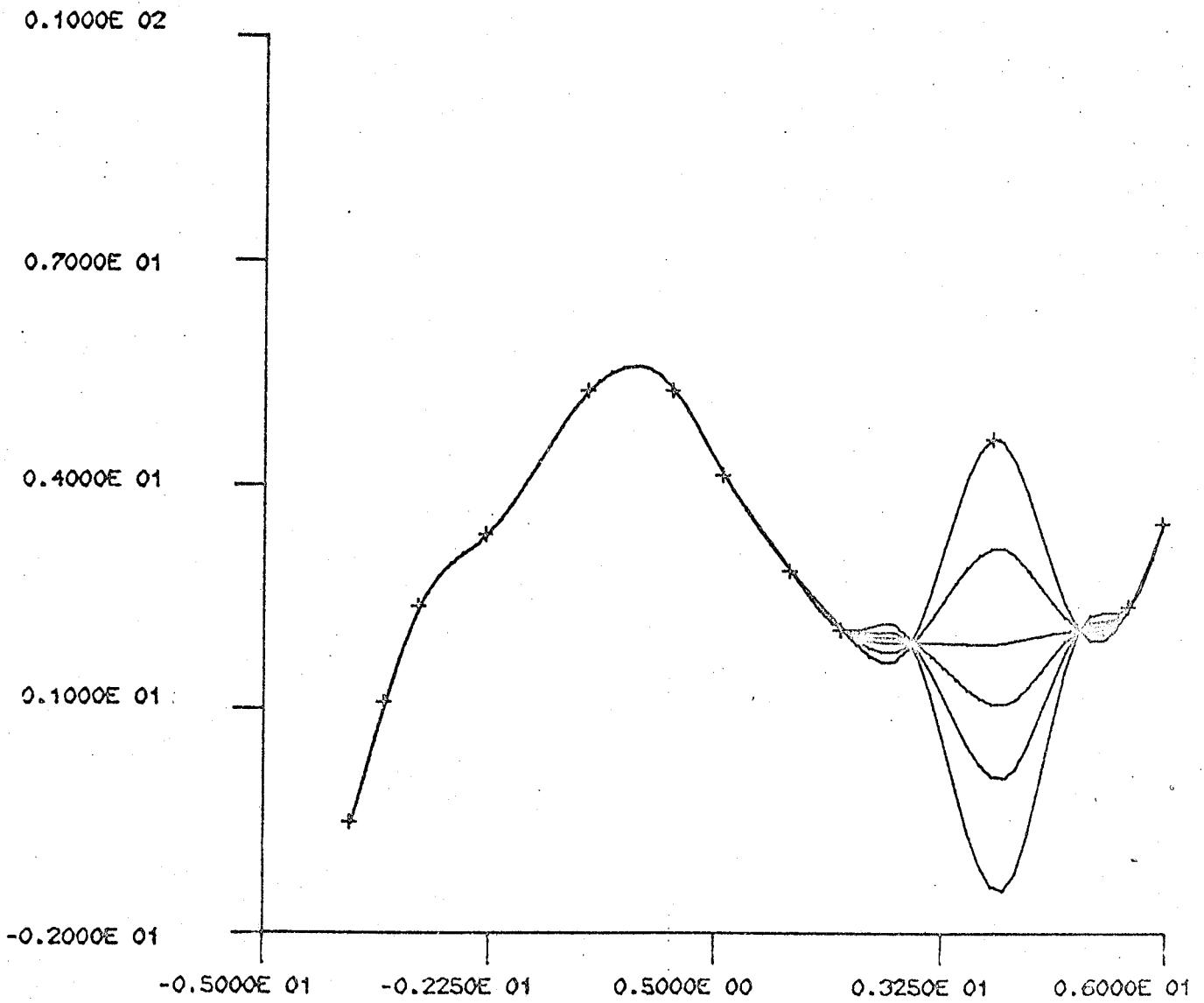
La figure 12 a pour but d'illustrer ce que l'on peut obtenir par la commande INTERSP2 du système PGAN. On a voulu montrer la possibilité offerte à l'utilisateur, d'observer les perturbations causées par la présence d'un point aberrant dans les données, et d'en voir les répercussions sur l'ensemble du graphe. Les graphes de la figure 12 ont été obtenus en déplaçant le point aberrant, sensiblement sur une verticale.

La figures 13 et 14 illustrent sur le même exemple, commandes INTERSPQ et AJUSMDR du système PGAN.

La figure 13 correspond à la commande INTERSPQ pour les ordres :  $q=3,4,5,6$  de la fonction spline d'interpolation. On peut remarquer que pour l'ordre 6, le graphe obtenu n'est pas très bon. En effet, plus la répartition des points est irrégulière, plus l'ordre de la fonction spline doit être élevé pour en épouser les contours, ce qui n'est pas le cas de cet exemple.

La figure 14, donnée par la commande AJUSMDR du système PGAN, donne pour les degrés: 2,3,4,5 du polynôme d'ajustement au sens des moindres carrés, les graphes correspondants.

INSTITUT DE MATHEMATIQUES APPLIQUEES DE GRENOBLE  
INTERPOLATION PAR DES SPLINES D'ORDRE 2



ECHELLE : 06/10

Figure 12 - Perturbations entraînée par le déplacement d'un point dans un plan vertical.

INSTITUT DE MATHÉMATIQUES APPLIQUÉES DE GRENOBLE  
INTERPOLATION PAR DES SPLINES D'ORDRE Q

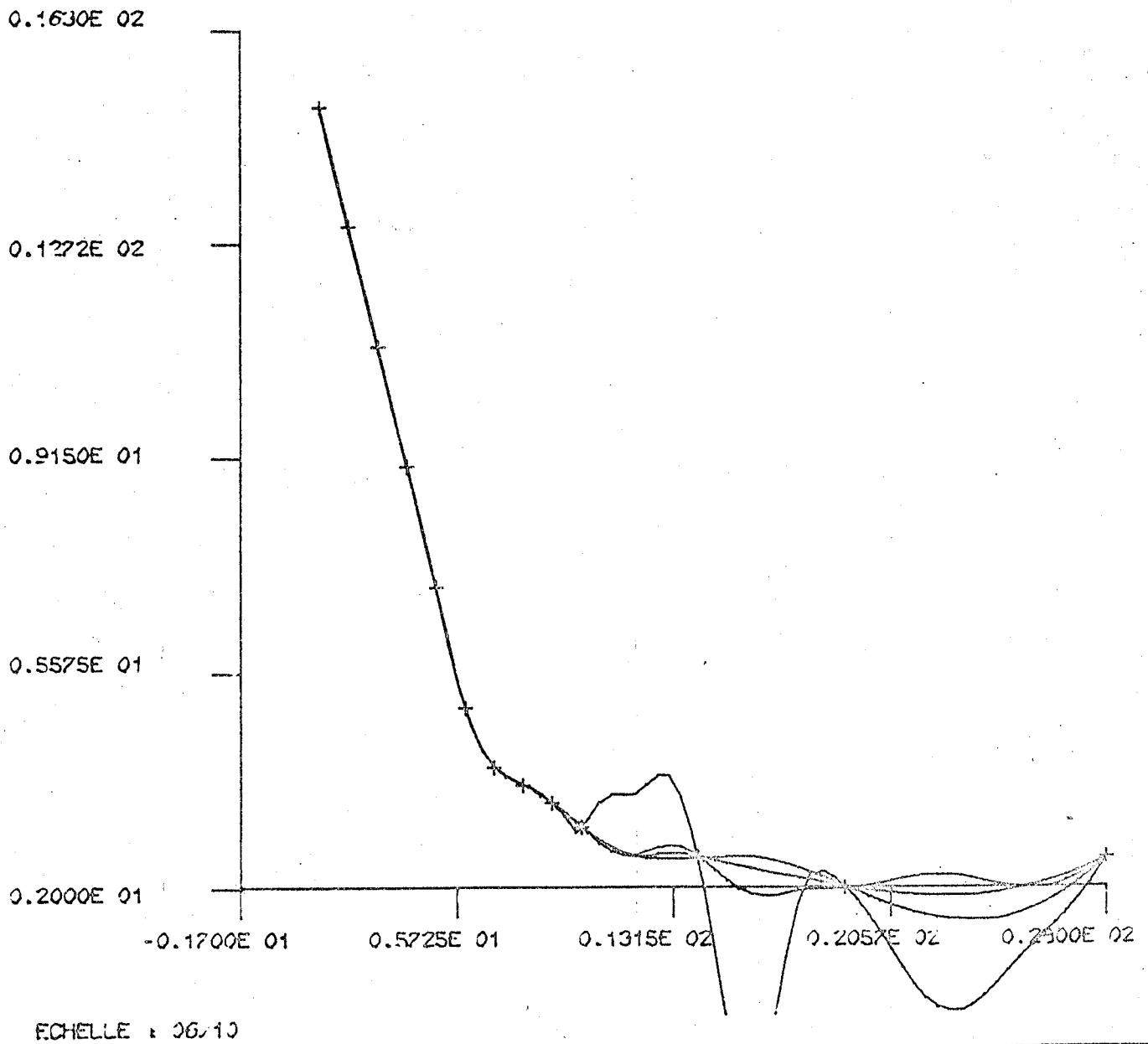


Figure 13 - Graphes correspondant à des fonctions splines d'ordre: 3,4,5,6 .

INSTITUT DE MATHÉMATIQUES APPLIQUÉES DE GRENOBLE  
AJUSTEMENT AU SENS DES MOINDRES CARRÉS

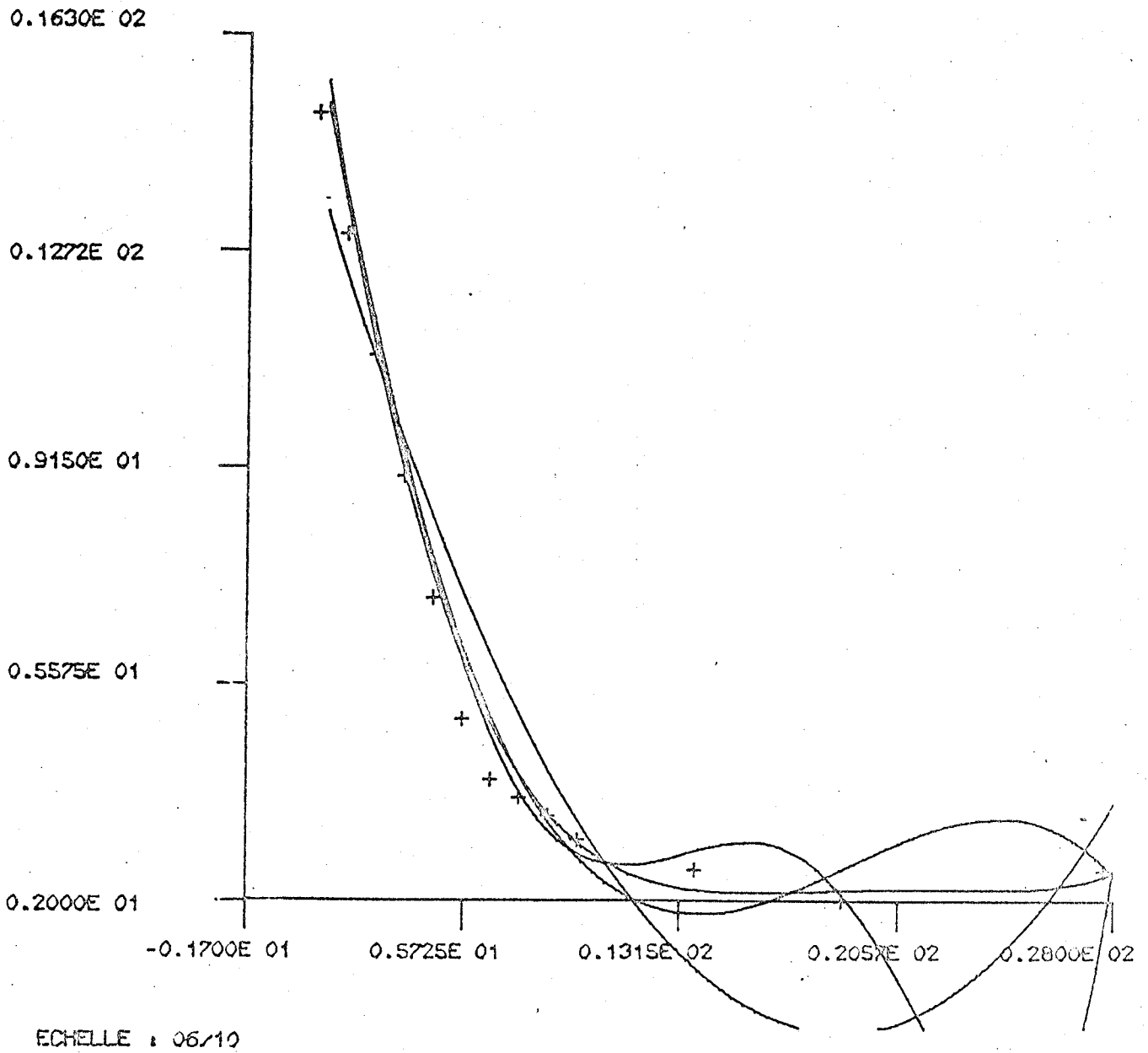


Figure 14 - Graphes pour les degrés: 2,3,4,5 du polynome d'ajustement.



## C O N C L U S I O N

Par ce travail nous avons voulu atteindre plusieurs objectifs:

- . Réaliser un ensemble de programmes interactifs (PGAN) dont le but est de permettre la visualisation et la conversation à l'aide d'un terminal graphique, pour traiter certains problèmes de base de l'analyse numérique.
- . Faire de PGAN un outil facile à utiliser (simplicité des commandes et des fonctions).
- . Permettre à l'utilisateur d'adapter PGAN à ses besoins par l'addition de nouveaux programmes (commande ADD).
- . Enfin, banaliser certains nouveaux processus de calcul (interpolation et ajustement par des fonctions splines).

Nous pensons avoir atteint ces objectifs, peut-être pas toujours de façon parfaite mais dans un contexte qui est opérationnel et déjà utilisé par certains physiciens.

Par sa relative indépendance machine (principalement écrit en Fortran) PGAN peut facilement se transposer sur un autre matériel à condition d'y disposer d'un outil analogue à GSP.

Les possibilités d'extension sont nombreuses tant par la commande ADD qui permet d'ajouter de nouvelles méthodes de calcul que par la création de nouvelles fonctions telles que:



- Effet de loupe
- Création d'une Trace
- D'autres représentations graphiques.

Enfin, nous pensons avoir créé avec PGAN un nouvel outil de calcul simple et efficace dans le domaine du Traitement graphique. Je terminerai en souhaitant que PGAN serve le plus grand nombre d'utilisateurs et incite de nombreux chercheurs à exploiter cette nouvelle voie.

REFERENCES  
BIBLIOGRAPHIQUES

- [1] IBM - System/360 model 67. Functional characteristic. Form A27- 2719.
- [2] IBM - System/360. Component Description.  
IBM 2250 Display Unit model 1. Form A27-2701.
- [3] Control Program-67/Cambridge Monitor System.  
USER'S GUIDE. Program number: 360D-05-2-005
- [4] IBM - System/360. Fortran IV Language. Form GC28-6515
- [5] IBM - System 360. Graphic Programming Services for Fortran IV.  
Form C27-6932.
- [6] IBM - System/360. Operating System. Assembler Language. Form C28-6314.
- [7] C.HANS et O.LECARME  
Support graphique pour un terminal graphique 2250 sous CMS. Note  
technique IMAG.
- [8] O.LECARME et J.C.SAILLARD  
Manuel d'utilisation de G.S.P sous CMS. Note technique IMAG.
- [9] M.DUC-JACQUET  
Cours d'analyse numérique, ch 3, IMAG.
- [10] P.J.LAURENT  
Approximation et Optimisation, ch 4, Hermann, à paraître.

- [11] M.DUC-JACQUET  
Cours d'analyse numérique, ch 2, IMAG.
  
- [12] P.J.LAURENT  
Approximation et Optimisation, ch 4, Hermann, à paraître.
  
- [13] M.DUC-JACQUET  
Cours d'analyse numérique, ch 1, IMAG.
  
- [14] N.GASTINEL  
Analyse numérique linéaire, ch 4, Hermann.
  
- [15] J.THOMANN  
Détermination et construction de fonctions spline à deux variables  
définies sur un domaine rectangulaire ou circulaire. Thèse 3ième Cycle,  
Février 1970. LILLE.
  
- [16] Procédures Algol en Analyse Numérique. Tome 2, page 83.  
Edition du CNRS.
  
- [17] Procédures Algol en Analyse Numérique. Tome 1, Page 178.  
Edition du CNRS.
  
- [18] C.GARCIN  
Etude de la zone d'ablation du glacier de Saint Sorlin. Thèse de  
3ième Cycle. Laboratoire de glaciologie Alpine du CNRS. GRENOBLE.
  
- [19] N.J.FELICI - CRAS,  
t.270, p.1449-1452 (8 juin 1970).