

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA
RECHERCHE SCIENTIFIQUE

UNIVERSITE DE CONSTANTINE

INSTITUT D'ELECTRONIQUE

N° d'ordre:

Série:

MAN
3316

THESE

Présentée pour obtenir le diplôme de Docteur-d'Etat
En Electronique



**IDENTIFICATION DES PARAMETRES DANS LES PROBLEMES
NON LINEAIRES**

OPTION

CONTROLE

PAR

MANSOURI NOURA

THEME

SOUTENU LE: 08 Mars 1999

Devant le jury:

Président:

Directeur de thèse:

Rapporteur :

Examineurs:

A. SAID

J.P. KERNEVEZ

K. BELARBI

M. BOULEMDEN

N. BOUGUECHEL

K. BENMAHAMMED

Professeur

Professeur

M.C

Professeur

Professeur

Professeur

Université de Constantine

Université de Compiègne

Université de Constantine

Université de Batna

Université de Batna

Université de Sétif

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA
RECHERCHE SCIENTIFIQUE

UNIVERSITE DE CONSTANTINE

INSTITUT D'ELECTRONIQUE

N° d'ordre:

Série:

THESE

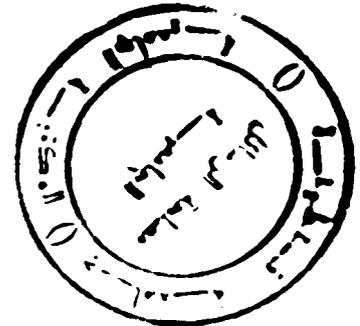
Présentée pour obtenir le diplôme de Docteur-d'Etat
En Electronique

***IDENTIFICATION DES PARAMETRES DANS LES PROBLEMES
NON LINEAIRES***

OPTION

CONTROLE

**PAR
MANSOURI NOURA**



THEME

SOUTENU LE: 08 Mars 1999

Devant le jury:

Président:

Directeur de thèse:

Rapporteur :

Examineurs:

A. SAID

J.P. KERNEVEZ

K. BELARBI

M. BOULEMDEN

N. BOUGUECHEL

K. BENMAHAMMED

Professeur

Professeur

M.C

Professeur

Professeur

Professeur

Université de Constantine

Université de Compiègne

Université de Constantine

Université de Batna

Université de Batna

Université de Sétif

RESUME

Le travail présenté dans cette thèse porte sur l'étude de différentes approches possibles, pour résoudre un problème d'identification des paramètres dans le cas non linéaire.

Chacune des méthodes présentées, est étudiée aussi bien du point de vue algorithmique que numérique. Les performances, les avantages et les inconvénients possibles, sont mis en évidence grâce à de nombreux exemples en relation avec des domaines multiples.

La méthode des moindres carrés est la méthode la plus couramment utilisée pour formuler un problème d'identification. On verra qu'il existe plusieurs possibilités pour minimiser un critère quadratique et que la mise au point d'algorithmes efficaces pour sa résolution revêt toujours une grande importance. La combinaison d'algorithmes déjà connus pour leurs performances, avec d'autres méthodes améliore encore plus, leur efficacité. Ceci permet de développer des algorithmes fonctionnant par application successive.

On traitera également des problèmes d'optimisation des systèmes, en utilisant un algorithme de continuation. Cet algorithme permet le suivi de l'évolution de la solution du point de vue trajectoire et stabilité. Il permet la détection des points singuliers, tels que les points de retournement et de bifurcation. Pour cela, les différentes méthodes de calcul du pas de déplacement et des directions de recherche sont proposées.

Lorsque le problème se rapporte à des systèmes modélisés par des équations différentielles avec conditions initiales inconnues ou mal définies, on montrera qu'il serait plus intéressant de procéder à une identification par étapes ou par application successive des algorithmes proposés. Ceci permet d'avoir des résultats d'une plus grande finesse et de gagner en temps de calcul.

ABSTRACT

In this work different approaches for non linear parameters identification are introduced and analysed

Numerical and algorithmic analysis are presented. The performances, advantages and possible shortcomings are highlighted by applying the approaches to a variety of test examples.

The least squares method has been widely used in system identification. Various efficient algorithms for minimising a quadratic cost are reviewed and analysed. It is shown that a combination of these algorithms with other methods can improve their efficiency. This leads to the so-called class of successive applications algorithms.

We also consider system optimisation using a continuation algorithm. This algorithm allows us to track the solution in terms of trajectory and stability. At the same time it can detect singular points such as limit and bifurcation points. Different methods for computing the direction of search and the displacement step are proposed.

On the other hand when the system is modelled with differential equations with unknown or badly defined initial conditions, it is shown that it would be more interesting to consider the identification with successive applications of the proposed algorithms. This gives finer results and decreases the run time.

ملخص

محتوى الأطروحة يخص العريف بمختلف الطرق التي يمكن إستعمالها لحل مسألة التعريف بالعوامل في حالة مسألة غير خطية كل من الطرق المقدمة معالجة سواء من الناحية اللوغاريتمية أو الرقمية لإيجابياتها و السلبيات الممكنة معرف بها بفضل الأمثلة المختلفة التي لها علاقة بمجالات عديدة سنرى أن هناك سبل كثيرة لتحديد القيمة الأدنى للعامل الرباعي ، وأن البحث على لوغاريتمات ناجعة لحله يشكل دائما مسألة هامة .

المزج بين عدد من اللوغاريتمات المعروفة بنجاحاتها مع طرق أخرى تحسن النتيجة . هذا يسمح بتشكيل لوغاريتمات تعمل بصفة متتالية.

سنعالج أيضا دراسة الأنظمة بإستعمال لوغاريتم لوغاريتماتها Continuation

نقط تغيير ، هذا اللوغاريتم يسمح بتتبع تطور الحل من ناحية المنحى و الاستقرار يسمح أيضا بكشف نقط تغيير إتجاه المنحى .

لهذا يعرض في هذه الأطروحة مختلف الطرق لحساب خطوة الإنتقال وإتجاهات البحث .

إذا تعلق الأمر بالأنظمة الممثلة بمعادلات تفاضلية مع شروط أولية غير معروفة أو غير محددة ، سوف نبرهن على أنه من الأفضل أن نعمل على التعريف بمراحل بتطبيق متتالي للوغاريتمات المستعملة.

النتائج المحصل عليها سوف تكون أكثر نجاعة وتسمح بربح وقت في الحساب .

REMERCIEMENTS

Je tiens à exprimer ma profonde gratitude et mes sincères remerciements à monsieur le professeur Jean-Pierre Kernévez dont les conseils et les encouragements ont été essentiels pour l'élaboration de cette thèse.

J'ai été très sensible à l'honneur que m'ont fait Messieurs les membres du jury, en acceptant de juger ce travail. Qu'ils trouvent ici l'expression de mes sincères remerciements

Je remercie également tous les membres du département génie informatique de l'Université de Technologie de Compiègne, en particulier ceux de la division mathématiques appliqués pour l'accueil chaleureux qu'il m'ont toujours réservé.

SOMMAIRE

INTRODUCTION

CHAPITRE I *Identification des paramètres dans les problèmes de moindres carrés non linéaires*

I- Introduction.....	4
II- Méthode des moindres carrés.....	4
II-1 : Problème sans contraintes	
II-1-1 : Conditions d'optimalité.....	5
II-1-2 : Les Méthodes de résolution.....	7
a Méthode du gradient simple.....	7
b Méthode du gradient conjugué.....	8
c Méthodes de Newton et Quasi-Newton.....	8
II-2 : Problèmes avec contraintes	
II-2-1 : Avec contraintes du type égalité.....	13
II-2-2 : Avec contraintes du type inégalité.....	15
II-2-3 : Méthodes de résolution.....	16
III- Performances des méthodes du type gradient.....	18
IV- Méthode de continuation.....	21.

CHAPITRE II *Identification des paramètres par les B-splines*

I- Introduction.....	27
II- Définitions des fonctions B-splines.....	27
III- Propriétés.....	28
IV- Les différents types de fonctions B-splines.....	29
V- Calcul des B-splines.....	30
VI- Approximation des données par des B-splines.....	31
VII- Identification des paramètres en utilisant les B-splines.....	35

<u>CHAPITRE III</u>	<i>Optimisation des systèmes par la méthode de continuation</i>	
I-	<i>Introduction.....</i>	50
II-	<i>Le principe de la continuation</i>	
	<i>II-1 : Méthodes de prédiction.....</i>	52
	<i>II-2 : Méthode de correction.....</i>	53
III-	<i>Les méthodes de continuation.</i>	
	<i>III-1 : Continuation par le paramètre λ.....</i>	53
	<i>III-2 : Continuation par la norme.....</i>	54
	<i>III-3 : Continuation par la pseudo-longueur d'arc.....</i>	55
IV-	<i>Points de retournement</i>	
	<i>IV-1 : Définition.....</i>	63
	<i>IV-2 : Stabilité de la solution.....</i>	64
V-	<i>Points de bifurcation.....</i>	64.
VI-	<i>Application de l'algorithme de continuation par la pseudo-longueur d'arc pour la résolution de différents problèmes</i>	
	<i>VI-1 : Identification de la courbe d'intersection entre deux surfaces....</i>	66
	<i>VI-2 : Etude du comportement d'un système dynamique simple.....</i>	70
VII-	<i>Optimisations des systèmes non linéaires</i>	
	<i>VII-1 : Introduction.....</i>	75
	<i>VII-2 : Conditions d'optimalité.....</i>	75
	<i>VII-3 : Détections des optimums.....</i>	75
	<i>VII-4 : Optimisation d'un système enzymatique à deux compartiments avec un seul substrat.....</i>	76
VIII-	<i>Identification des paramètres par continuation successive.....</i>	82
<u>CHAPITRE IV</u>	<i>Identification des paramètres par la méthode des sentinelles</i>	
I-	<i>Introduction.....</i>	87
II-	<i>Rappels sur la méthode des moindres carrés.....</i>	87
III-	<i>Minimisation de la fonction coût.....</i>	88

<i>IV- Conditions d'existence de la matrice W.....</i>	<i>89</i>
<i>V- Définitions des sentinelles.....</i>	<i>90</i>
<i>VI- Identifications des paramètres par la méthode des sentinelles pour un problème linéaire.....</i>	<i>90</i>
<i>VII- Identifications des paramètres par la méthode des sentinelles pour un problème non linéaire</i>	
<i>VII-1 : Système linéarisé.....</i>	<i>92</i>
<i>VII-2 : Sentinelles du système linéarisé.....</i>	<i>93</i>
<i>VIII- Application de la méthode pour l'identification des paramètres pour différents problèmes</i>	<i>94</i>
<i>IX- Identification des paramètres par application successive des algorithmes.....</i>	<i>102</i>

CONCLUSION

BIBLIOGRAPHIE

methodes iteratives, peuvent converger tres lentement ou tout simplement diverger, si les valeurs initiales de l'algorithme ne sont pas choisies dans le proche voisinage de la solution, on montrera alors que l'introduction d'une methode de continuation dans ce cas, peut resoudre ce type de problemes. Nous illustrerons ces ameliorations, par des exemples.

Dans le deuxieme chapitre, on introduira la methode des B-Splines. Cette methode surtout connue dans le domaine de la CAO, est egalement utilisee pour l'identification des parametres. Elle se base sur l'approximation des donnees relevees, par une courbe B-Splines. Apres, un rappel des definitions, des proprietes et des differents types de fonctions B-splines existantes, on detaillera l'algorithme de calcul de ces dernieres. Dans la deuxieme partie de ce chapitre, on developpera les etapes d'une identification des parametres par ces fonctions et on illustrera ceci par des exemples.

Le troisieme chapitre, est consacre a la methode de continuation. Cette methode est de plus en plus utilisee dans le domaine de l'ingenierie. Elle permet un suivi precis de la courbe solution, du systeme etudie. On peut ainsi determiner tous les points de bifurcations et de retournements. Ces derniers, representent soit des minima ou des maxima locaux. Pour cela, les methodes de continuation les plus connues sont presentees ainsi que les differentes etapes de calcul. On detaillera surtout la methode de continuation par la pseudo-longueur d'arc et le principe de la continuation successive. La aussi, des problemes relatifs a des domaines tres differents sont resolus par cette methode, ce qui montre l'efficacite de cette derniere.

Enfin le quatrieme chapitre, est reserve a la methode la plus recente, a savoir la methode des sentinelles. Apres quelques definitions preliminaires, on fera l'analogie avec la methode des moindres carrés et on definira ce qu'est une sentinelle, les conditions d'application de cette methode et son interet. Les resultats relatifs a certains problemes sont compares avec ceux obtenus avec la methode de continuation et celle des B-splines. Enfin, les resultats d'une identification des parametres en deux etapes pour des systemes definis par des equations differentielles y sont presentes. Cette facon de proceder semble etre assez interessante.

CHAPITRE I

IDENTIFICATION DES PARAMETRES DANS LES PROBLEMES DE MOINDRES CARRES NON LINEAIRES

I - Introduction

Les problèmes d'identification les plus couramment rencontrés, concernent l'identification de paramètres inconnus $\lambda = \{\lambda_1, \lambda_2, \dots, \lambda_m\}$, à partir de $Zd(t_j)$, les mesures effectuées sur le système, à différents instants t_j ($j = 1, \dots, m$)

La démarche habituellement suivie pour l'identification de ces paramètres est la suivante

1° - déterminer le modèle ce dernier consiste en une ou plusieurs équations mathématiques représentant le comportement d'un système ou d'une situation physique et faisant intervenir les paramètres inconnus, que l'on cherche à identifier

2° - définir un critère A cause des erreurs de mesures d'une part et l'inaptitude à trouver un modèle mathématique qui puisse rendre compte exactement de la réalité d'autre part, il est plus judicieux de prendre comme critère, $\varphi(\lambda)$, la somme des carrés des écarts entre la solution du modèle (que l'on notera $y(\lambda, t_j)$) et les observations $Zd(t_j)$, faites aux différents instants t_j ($j = 1, \dots, m$)

3° - déterminer le minimum du critère Cette opération revient à déterminer les paramètres λ^* pour lesquels les solutions du modèle approchent le mieux les observations. Les méthodes les plus utilisées pour cela, sont les méthodes d'optimisation.

Si on doit seulement, déterminer :

$$\min \varphi(\lambda) \quad (1.1)$$

on a un problème d'optimisation sans contraintes. Par contre, si on doit déterminer :

$$\min \varphi(\lambda) \quad (1.2)$$

avec

$$r(\lambda) = 0 \text{ et/ou } S(\lambda) \leq 0$$

On a un problème d'optimisation avec contraintes.

II- Méthode des moindres carrés

Dans le cas général, un critère de moindres carrés, est défini comme suit:

$$\varphi(\lambda) = \frac{1}{2} \sum_{k=1}^l \sum_{j=1}^m f_{kj}^2(\lambda) \quad (1.3)$$

avec:

$$f_{kj} = y_k(\lambda, t_j) - Zd_{kj} \quad (1.4)$$

k et m représente respectivement, le nombre de grandeurs modélisées et mesurées, et le nombre de mesures.

Cette forme simple, peut donner des résultats non satisfaisants, en particulier pour les raisons suivantes:

a/ les mesures Zd_{kj} peuvent représenter des grandeurs physiques diverses, mesurées sur des échelles différentes. Ceci peut entraîner la perte de certaines informations vu que certains résidus vont dominer

b/ Certaines observations sont plus importantes, pour l'estimation des paramètres que d'autres. Pour cela il faut mettre en évidence cette importance dans le critère à minimiser.

Une solution unique peut résoudre ces problèmes, à savoir

Attribuer un poids positif W_{k_l} différent à chacune des fonctions f_{k_l} . Le critère à minimiser devient alors

$$\varphi(\lambda) = \sum_{k=1}^l \sum_{l=1}^m W_{k_l} f_{k_l}^2(\lambda) \quad (1.5)$$

un poids, relativement plus faible, sera attribué aux observations mesurées sur une plus grande échelle ou jugées peu sûres, ainsi que qu'aux informations d'intérêt moindres pour l'estimation des paramètres.

Dans le cas où $k=1$, c'est à dire que l'identification des paramètres doit se faire, à partir des mesures relatives à une seule grandeur, on prend $w = 0.5$.

II-1: Pour un problème sans contraintes

II-1-1 : Conditions d'optimalité

a/ Définition

- Un point λ^* est dit minimum local d'une fonction $\varphi(\lambda)$, s'il existe $\delta > 0$ tel que, si $\forall \lambda$ satisfaisant $\|\lambda - \lambda^*\| < \delta$, on a $\varphi(\lambda^*) \leq \varphi(\lambda)$.
- Un point λ^* est dit minimum global d'une fonction $\varphi(\lambda)$, $\forall \lambda$, on a $\varphi(\lambda^*) \leq \varphi(\lambda)$.
- Tout minimum global est un minimum local.

b/ Conditions du 1^o ordre

Si $\varphi(\lambda)$ est continue et continûment dérivable; alors pour $\delta\lambda$ suffisamment petit, on peut écrire que:

$$\varphi(\lambda + \delta\lambda) \cong \varphi(\lambda) + g(\lambda) \delta\lambda + O(\|\delta\lambda\|^2) \quad (1.6)$$

et pour λ^* le minimum global on a:

$$\varphi(\lambda^* + \delta\lambda) \geq \varphi(\lambda^*) \quad \forall \lambda \quad (1.7)$$

soit:

$$g(\lambda^*) \delta\lambda \geq 0, \quad \forall \delta\lambda \quad (1.8)$$

cette expression ne peut être que nulle; car si elle est positive pour $\delta\lambda$, elle sera négative pour $-\delta\lambda$, d'où:

$$\Delta\varphi(\lambda^*) = g(\lambda^*) = 0, \quad (1.9)$$

cela signifie qu'au voisinage du minimum on a, au 1^o ordre

$$\varphi(\lambda + \delta\lambda) \cong \varphi(\lambda) \quad (1.10)$$

c'est à dire que la fonction objective est stationnaire au premier ordre pour un minimum sans contraintes

c/ Conditions du 2^o ordre

Si on suppose que $H(\lambda)$, la matrice constituée par les éléments $\frac{\partial^2 \varphi(\lambda)}{\partial \lambda_i \partial \lambda_j}$ existe partout et qu'elle est continue, on a alors pour $\delta\lambda$ suffisamment petit, le développement de Taylor jusqu'au 2^o ordre suivant:

$$\varphi(\lambda + \delta\lambda) \cong \varphi(\lambda) + g(\lambda)\delta\lambda + \frac{1}{2}\delta\lambda' H(\lambda)\delta\lambda + O(\|\delta\lambda\|^3) \quad (1.11)$$

et pour λ^* le minimum global on a:

$$\varphi(\lambda^* + \delta\lambda) \geq \varphi(\lambda^*) \quad \forall \lambda \quad (1.12)$$

et

$$g(\lambda^*) = 0, \quad (1.13)$$

ce qui implique que:

$$\frac{1}{2}\delta\lambda' H(\lambda^*) \delta\lambda \geq 0 \quad (1.14)$$

Cette inégalité signifie que la matrice des dérivées secondes est non négative. C'est la condition nécessaire de minimum.

Si en plus la matrice des dérivées secondes $H(\lambda^*)$ est définie positive alors $\delta\lambda' H(\lambda^*) \delta\lambda$ n'est jamais nul pour $\delta\lambda \neq 0$ et les termes d'ordre trois sont toujours négligeables pour $\delta\lambda$ suffisamment petit. Ce qui fait que, la condition suffisante pour que λ^* soit un minimum est:

$$g(\lambda^*) = 0 \text{ et } H(\lambda^*) \text{ définie positive.} \quad (1.15)$$

d/ Conditions d'optimalité

Si $\varphi(\lambda)$ est une fonction deux fois dérivable, la condition suffisante pour qu'un point λ^* soit un minimum local est que:

1/ - Le gradient $g(\lambda^*)$ soit nul: $g(\lambda^*) = 0$

2/ - la matrice des dérivées secondes ou Hessien $H(\lambda^*)$ soit définie positive:

Si $H(\lambda)$ est définie positive et $g(\lambda^*) = 0 \quad \forall \lambda$, alors λ^* est l'unique minimum global de φ .

II-1-2: Les méthodes de résolution

Les méthodes d'optimisation les plus utilisées sont des méthodes itératives de la forme

$$\lambda_{i+1} = \lambda_i + \rho_i d_i \quad (1.16)$$

où

ρ_i , d_i et d_i représentent respectivement le pas et la direction suivant laquelle on minimise.

Cette dernière doit être une direction de descente et le pas doit être choisi de telle manière à ce que l'on ait

$$\varphi(\lambda_{i+1}) < \varphi(\lambda_i) \quad (1.17)$$

Il existe une multitude de méthodes de ce type. Elles diffèrent les unes des autres par le choix de ρ et de d .

Ainsi les méthodes, dites de gradient, sont des méthodes qui vérifient la relation itérative décrite précédemment, avec:

$$d_i = -R_i g_i \quad (1.18)$$

où R_i représente une matrice définie positive et g_i le gradient de la fonction à minimiser.

Ce sont des algorithmes, du type:

$$\lambda_{i+1} = \lambda_i - \rho_i R_i g_i \quad (1.19)$$

a / Méthode du gradient simple ou méthode de descente

C'est la méthode d'optimisation, la plus ancienne et la plus connue. Elle a été introduite par Cauchy en 1848 et consiste à faire les itérations :

$$\lambda_{i+1} = \lambda_i - \rho_i g_i \quad (1.20)$$

(R_i étant prise égale à la matrice unité)

ρ_i est choisi de manière optimale; il doit donc, vérifier:

$$\varphi(\lambda_{i+1}) < \varphi(\lambda_i) \quad (1.21)$$

L'inconvénient majeur de cette méthode, c'est la vitesse de convergence. Cette dernière dépend essentiellement du rapport entre la plus grande et la plus petite valeur propre du Hessien, de la fonction à minimiser. Elle sera d'autant plus faible que le rapport est très petit devant 1.

C'est le cas des fonctions avec des contours étirés; la fonction décroît très rapidement au début, ensuite, la convergence devient très lente.

Des variantes de cette méthode existent dans la littérature [1-10][1-16], mais elles sont peu utilisées car, bien que meilleures du point de vue convergence, elles sont coûteuses en temps de calcul.

b/ Méthode du gradient conjugué

La méthode du gradient conjugué décrite initialement par Hestenes et Stiel (1952) est un algorithme qui utilise les directions conjuguées w_i pour la résolution des systèmes d'équations linéaires, ayant une matrice de coefficients, symétrique définie positive.

L'analogie entre ce genre de problèmes et la minimisation d'une fonction quadratique, est telle que Fletcher et Reeves[1-9] ont proposé d'utiliser les vecteurs de directions conjuguées w_i pour minimiser un critère quadratique.

Les directions conjuguées w_i sont définies par la relation:

$$d_i = w_i = -g_i + \alpha_i w_{i-1} \quad (1.22)$$

avec

$$w_0 = -g_0$$

α_i est choisi de telle manière à ce que, la relation :

$$(A w_i, w_{i-1}) = 0 \quad (1.23)$$

soit vérifiée

C'est à dire

$$\alpha_i = \frac{(A g_i, w_{i-1})}{(A w_{i-1}, w_{i-1})} \quad (1.24)$$

A étant une matrice définie positive

Les itérations de l'algorithme du gradient conjugué sont sous la forme:

$$\lambda_{i+1} = \lambda_i - \rho_i w_i \quad (1.25)$$

avec w_i calculé comme indiqué plus haut et ρ_i choisi de manière optimale.

Pour des fonctions quadratiques définies dans \mathbb{R}^N , la méthode converge en N itérations.

Pour des fonctions quelconques, le nombre d'itérations nécessaires dépend du choix des valeurs initiales et de la précision désirée mais l'algorithme doit être réinitialisé au bout de (N+1) itérations

Cette méthode est beaucoup plus rapide que la méthode du gradient simple présentée précédemment.

c/ Méthodes de Newton et Quasi-Newton

Pour cette méthode, les itérations sont de la forme:

$$\lambda_{i+1} = \lambda_i - H_i^{-1} g_i \quad (1.26)$$

c'est à dire que: $\rho_i = 1$ et $d_i = -H_i^{-1} g_i$; H_i étant la matrice des dérivées secondes ou Hessien.

- Pour une fonction quadratique, dont la matrice des dérivées secondes est définie positive, cette méthode converge en une seule itération

- Pour une fonction quelconque, la méthode ne converge pas en une seule itération, mais elle reste bonne tant que la matrice H est définie positive

Au voisinage du minimum, la convergence devient quadratique. Ceci explique pourquoi la méthode converge parfaitement, si les valeurs initiales du processus itératif, sont choisies dans un proche voisinage de la solution

Toutefois, Cette méthode très performante trouve une utilisation restreinte, et cela pour les raisons suivantes

- La condition nécessaire pour l'application de cette méthode, est que la matrice des dérivées secondes de la fonctionnelle soit définie positive.

- Le calcul des dérivées secondes est souvent, fastidieux, peu évident et cher en temps de calcul.

Pour remédier à ces inconvénients des méthodes dites du type Quasi-Newton, furent proposées Elles sont toutes basées sur le calcul approximatif du Hessien ou de son inverse.

1 - Méthode de Gauss-Newton

Soit à minimiser la fonction:

$$\varphi(\lambda) = \frac{1}{2} \sum_{j=1}^m f_j^2(\lambda) \quad (1.27)$$

le gradient d'une telle fonction est égale à:

$$g(\lambda) = J^T(\lambda) f(\lambda) \quad (1.28)$$

où

$J(\lambda)$ représente la matrice jacobienne dont les éléments sont : $\frac{\partial f_j}{\partial \lambda_i}$ ($j=1, \dots, m$ $i=1, \dots, n$)

et le hessien H est une matrice constituée par les éléments:

$$H_{\alpha, \beta}(\lambda) = \sum_{i=1}^m \frac{\partial f_i(\lambda)}{\partial \lambda_\alpha} \frac{\partial f_i(\lambda)}{\partial \lambda_\beta} + \sum_{i=1}^m \frac{\partial^2 f_i(\lambda)}{\partial \lambda_\alpha \partial \lambda_\beta} f_i(\lambda) \quad (1.29)$$

La simplification, introduite par Gauss, dans la formulation du Hessien, consiste à négliger dans l'expression de H, le terme $\sum_{i=1}^m \frac{\partial^2 f_i(\lambda)}{\partial \lambda_\alpha \partial \lambda_\beta} f_i(\lambda)$ devant $\sum_{i=1}^m \frac{\partial f_i(\lambda)}{\partial \lambda_\alpha} \frac{\partial f_i(\lambda)}{\partial \lambda_\beta}$.

Ce qui permettrait d'écrire que:

$$H_{\alpha, \beta}(\lambda) = \sum_{i=1}^m \frac{\partial f_i(\lambda)}{\partial \lambda_\alpha} \frac{\partial f_i(\lambda)}{\partial \lambda_\beta} \quad (1.30)$$

Soit

$$H(\lambda) = J'(\lambda) \cdot J(\lambda) \quad (1.31)$$

Par cette approximation, la formule itérative de Newton devient:

$$\lambda_{i+1} = \lambda_i - [J'(\lambda_i) \cdot J(\lambda_i)]^{-1} g(\lambda_i) \quad (1.32)$$

Cette simplification suppose qu'au voisinage du minimum λ^* , le résidu $f_i(\lambda)$ est très faible. C'est à dire $f_i(\lambda) \rightarrow 0$ quand $\lambda \rightarrow \lambda^*$. On évite ainsi, le calcul explicite des dérivées secondes, ce qui du point de vue temps de calcul, représente un gain appréciable par rapport à la méthode de Newton initiale.

Toutefois, cette simplification ne peut être faite que si l'hypothèse sur les résidus est vérifiée. Plusieurs variantes de cet algorithme existent également dans la littérature. On citera, à titre d'exemple ceux proposés par Gill et Murray dans les références [1-11] et [1-12].

2 - Méthode de Davidon-Fletcher-Powell (DFP)

L'algorithme DFP proposé par Fletcher-Powell[1-8], est une amélioration de l'algorithme initial décrit par Davidon[1-6]. La méthode utilisée est une méthode de quasi-Newton utilisant les directions conjuguées. L'inverse du Hessien $H(\lambda)$, est approximé par une matrice $G(\lambda)$ symétrique définie positive.

Les itérations étant de la forme:

$$\lambda_{i+1} = \lambda_i - \rho_i H_i^{-1} g_i \quad (1.33)$$

On désigne par:

$$- H_{k+1} = H(\lambda_{k+1}) \text{ et } H_k = H(\lambda_k)$$

$$- g_{k+1} = g(\lambda_{k+1}) \text{ et } g_k = g(\lambda_k)$$

et on pose:

$$- s_k = \lambda_{k+1} - \lambda_k$$

$$- y_k = g_{k+1} - g_k$$

le développement en série de Taylor de g , nous donne:

$$g_k = g_{k+1} - H_{k+1}(\lambda_{k+1} - \lambda_k) + \Delta^k = g_{k+1} - H_{k+1} s_k + \Delta^k \quad (1.34)$$

en négligeant le terme Δ^k dans l'expression précédente, on obtient:

$$y_k \approx H_{k+1} s_k \quad (1.35)$$

ou alors:

$$s_k \approx [H_{k+1}]^{-1} y_k \approx G_{k+1} y_k \quad (1.36)$$

G_{k+1} étant la matrice qui approxime $[H_{k+1}]^{-1}$. Elle est calculée itérativement, en ajoutant à chaque fois un terme de correction C_k à la matrice G_k .

$$G_{k+1} = G_k + C_k \quad (1.37)$$

Dans la méthode DFP, le terme de correction C_k est calculé grâce à la relation (1.38)

$$C_k = s_k u_k' - G_k y_k v_k' \quad (1.38)$$

avec u_k, v_k des vecteurs non nuls tels que:

$$u_k' y_k = 1 \quad \text{et} \quad v_k' y_k = 1 \quad (1.39)$$

pris égaux à

$$u_k = \frac{s_k}{s_k' y_k} \quad \text{et} \quad v_k = \frac{G_k y_k}{y_k' G_k y_k} \quad (1.40)$$

En remplaçant u_k, v_k et C_k par leur valeur dans la relation (1.38), on obtient la formule de calcul de la matrice approximant l'inverse du Hessien:

$$\begin{aligned} G_{k+1} &= G_k + s_k u_k' - G_k y_k v_k' \\ &= G_k + \frac{s_k s_k'}{s_k' y_k} - \frac{G_k y_k [G_k y_k]'}{y_k' G_k y_k} \end{aligned} \quad (1.41)$$

Au départ, on choisit une matrice symétrique définie positive quelconque; la matrice identité par exemple

Les itérations deviennent alors de la forme:

$$\lambda_{i+1} = \lambda_i - \rho_i G_i g_i \quad (1.42)$$

ρ_i étant calculé à chaque itération de manière optimale.

3- Méthode de Broyden-Fletcher - Goldfarb-Shanno (BFGS)

Cet algorithme, développé en 1970, consiste à faire les mêmes itérations que la méthode DFP, mais il utilise pour la construction d'une approximation du Hessien, une formule dérivée de la DFP.

Les auteurs ont montré que, si on interchangeait s_k et y_k dans l'expression (1.41), on obtiendrait la matrice M_{k+1} définie comme suit:

$$M_{k+1} = M_k + \frac{y_k y_k'}{y_k' s_k} - \frac{M_k s_k [M_k s_k]'}{s_k' M_k s_k} \quad (1.43)$$

et qui vérifie :

$$M_{k+1} s_k \approx y_k \quad (1.44)$$

avec M_0 une matrice symétrique définie positive arbitraire.

Cette matrice va approximer le Hessien. Pour obtenir l'inverse, il suffit de prendre l'inverse de la formule

$$[M_{k+1}]^{-1} = [M_k]^{-1} + \left[1 + \frac{y_k' [M_k]^{-1} y_k}{s_k' y_k} \right] \begin{bmatrix} s_k s_k' \\ s_k' y_k \end{bmatrix} - \left[\frac{s_k y_k' [M_k]^{-1} + [M_k]^{-1} y_k s_k'}{s_k' y_k} \right]$$

soit.

$$G_{k+1} = G_k + \left[1 + \frac{y_k' G_k y_k}{s_k' y_k} \right] \begin{bmatrix} s_k s_k' \\ s_k' y_k \end{bmatrix} - \left[\frac{s_k y_k' G_k + G_k y_k s_k'}{s_k' y_k} \right] \quad (1.45)$$

avec G_0 une matrice symétrique définie positive arbitraire et ρ_i calculé de manière optimale à chaque itération.

Comme pour la méthode du gradient conjugué, les méthodes BFGS et DFP doivent être réinitialisées périodiquement, si la fonction à minimiser n'est pas quadratique. Toutefois la méthode BFGS est beaucoup plus performante que la méthode DFP.

Remarque

Dans plusieurs algorithmes présentés précédemment, on doit calculer à chaque itération, le paramètre ρ_i de telle manière à minimiser la fonctionnelle dans la direction - d.

A cet effet, Fletcher et Powell[1-8] ont proposé un algorithme basé sur l'interpolation cubique, en utilisant les valeurs des fonctions et du gradient en deux points : x_i et z_i ; tels que :

$$z_i = x_i + \chi s_i \quad (1.46)$$

avec :

$$s_i = g_{i-1} - g_i \quad (1.47)$$

et $\chi > 0$ (par exemple = 1)

L'estimation de ρ_i , est donnée par la relation :

$$\rho_i = \chi \left(1 - \frac{(g_z' s_i) + \varpi - \beta}{(g_z' s_i) - (g_x' s_i) + 2\varpi} \right) \quad (1.48)$$

où

$$\varpi = \sqrt{\beta^2 - (g_x' s_i)(g_z' s_i)} \quad (1.49)$$

et

$$\beta = \frac{3}{\chi} (\varphi_x - \varphi_z) + (g_x' s_i)(g_z' s_i) \quad (1.50)$$

à l'itération 0, on choisi :

$$z_i = x_i + \eta s_i \quad (1.51)$$

avec

$$\eta_i = \min\left(1, \frac{-2(\varphi_x - \varphi_0)}{(g_x^T s_i)}\right) \quad (1.52)$$

φ_0 représente le minimum prédit pour la fonction à minimiser. Pour un problème de moindres carrés, $\varphi_0 = 0$.

Il est nécessaire de s'assurer que $\varphi(x_i + \alpha_i s_i)$ soit inférieure à φ_x et φ_0 . Dans le cas contraire, il faut reprendre l'interpolation.

4/ Méthode de Marquardt

L'algorithme proposé par Marquardt [1-18] se base sur la méthode de résolution des problèmes de moindres carrés non linéaire proposée par Levenberg [1-15]. Il consiste à faire les itérations suivantes

$$\lambda_{k+1} = \lambda_k - [J_k^T J_k + \mu_k D_k^2]^{-1} g_k \quad (1.53)$$

où J et g représentent respectivement le jacobien et le gradient.

- D^2 est une matrice diagonale dont les éléments sont égaux aux éléments de la diagonale principale de la matrice $J^T J$.

- μ est une variable positive que l'on calcule à chaque itération. Elle est choisie de telle manière à ce que l'on ait: $\varphi(\lambda_{k+1}) < \varphi(\lambda_k)$

Partant de valeurs initiales λ_k, μ_k on calcule $\varphi_k, \lambda_{k+1}, \varphi_{k+1}$

- si $\varphi(\lambda_{k+1}) < \varphi(\lambda_k)$, λ_k est remplacé par λ_{k+1} et μ_k est divisé par un pas de décrémentation μ_d constant ou variable.

- si $\varphi(\lambda_{k+1}) \geq \varphi(\lambda_k)$, λ_k est conservé et μ_k est multiplié par un pas d'incrément μ_i constant ou variable.

Cette méthode offre l'avantage de combiner deux méthodes:

- lorsque μ est très petit, μD^2 devient négligeable devant $J^T J$; on a une résolution par Gauss-Newton.

- lorsque μ est très grand, $J^T J$ devient négligeable devant μD^2 ; on a une résolution par la méthode du gradient.

II-2: Problème avec contraintes

II-2-1 : Avec contraintes de type égalité

Déterminer:

$$\min \varphi(\lambda) \quad (1.56)$$

avec

$$f_i(\lambda) = 0 \quad (i = 1, \dots, p)$$

Si $\varphi(\lambda)$ et les $r_i(\lambda)$ satisfont aux conditions de continuité, pour qu'un point régulier λ^* soit un minimum local, il est nécessaire qu'il existe des paramètres μ_i tels que la fonction

$$L(\lambda, \mu) = \varphi(\lambda) + \sum_{i=1}^p \mu_i r_i(\lambda) \quad (1.57)$$

soit stationnaire par rapport à λ et μ :

$$L(\lambda^*, \mu) = 0, \quad r_i(\lambda^*) = 0 \quad (i = 1, \dots, p) \quad (1.58)$$

Les μ_i sont appelés paramètres de Lagrange et la fonction $L(\lambda, \mu)$ le Lagrangien.

On ramène ainsi, Pour les conditions du 1^o ordre, le problème d'optimisation avec contraintes de type égalité à un problème sans contraintes. Ce raisonnement est évidemment valable seulement si les gradients des contraintes ne sont pas nuls. Dans le cas contraire, il faut développer à un ordre supérieur

Le développement du Lagrangien au 2^o ordre, par rapport à $\delta\lambda$ suffisamment petit donne:

$$L(\lambda + \delta\lambda, \mu) \cong L(\lambda, \mu) + L_{\lambda}(\lambda, \mu)\delta\lambda + \frac{1}{2}\delta\lambda' L_{\lambda\lambda}(\lambda, \mu)\delta\lambda + O(\|\delta\lambda\|^3) \quad (1.59)$$

et comme, pour λ^* le minimum global on a:

$$L_{\lambda}(\lambda^*, \mu) = 0, \quad (1.60)$$

on peut donc écrire:

$$L(\lambda^* + \delta\lambda, \mu) \cong L(\lambda^*, \mu) + \frac{1}{2}\delta\lambda' L_{\lambda\lambda}(\lambda^*, \mu)\delta\lambda + O(\|\delta\lambda\|^3) \quad (1.61)$$

D'autre part, on sait, par définition du Lagrangien que si $(\lambda + \delta\lambda)$ appartient à l'ensemble des solutions admissibles, alors:

$$L(\lambda + \delta\lambda, \mu) = \varphi(\lambda + \delta\lambda) \quad (R_i(\lambda) = 0) \quad (1.62)$$

En opérant ce remplacement dans l'équation (1.61), on arrive à:

$$\varphi(\lambda^* + \delta\lambda) \cong L(\lambda^*, \mu) + \frac{1}{2}\delta\lambda' L_{\lambda\lambda}(\lambda^*, \mu)\delta\lambda + O(\|\delta\lambda\|^3) \quad (1.63)$$

Ce qui nous permet de conclure que pour que λ^* soit un minimum local, il faut que :

1 - $L_{\lambda}(\lambda^*, \mu) = 0,$

2 - La matrice des dérivées secondes du Lagrangien $L_{\lambda\lambda}(\lambda^*, \mu)$, soit non négative dans l'ensemble \wp défini par $\wp = \{v / R_{\mu} v = 0\}$.

II-2-2 : Avec contraintes de type inégalité

Déterminer

$$\min \varphi(\lambda) \quad (1.64)$$

avec $S_j(\lambda) \leq 0 \quad (j = 1, \dots, q)$

Le nombre de contraintes peut être supérieur au nombre de paramètres à identifier, sans pour autant que le domaine des solutions admissibles soit vide. Cependant, en tout point de ce domaine, le nombre de contraintes qui peuvent être saturées simultanément est au plus égale au nombre de paramètres inconnus. Il est évident que ces contraintes doivent être indépendantes.

On dit que la contrainte $S_j(\lambda)$ est saturée lorsque on a $S_j(\lambda) = 0$ (ce qui n'est pas équivalent à une contrainte de type égalité).

Le Lagrangien s'écrit :

$$L(\lambda, \varepsilon) = \varphi(\lambda) + \sum_{j=1}^q \varepsilon_j S_j(\lambda) \quad (1.65)$$

et les conditions d'optimalité sont :

$$1 - L_{\lambda}(\lambda^*, \varepsilon) = 0,$$

$$2 - S_j(\lambda^*) = 0 \quad (j = 1, \dots, q)$$

avec $L_{\lambda\lambda}(\lambda^*, \varepsilon)$ non négative.

Pour tester si la fonction à minimiser, augmente vers l'intérieur du domaine admissible, il faut de plus écrire :

$$\varphi'_{\lambda} \partial \lambda \geq 0, \quad \forall \partial \lambda / S'_{j,\lambda} \partial \lambda < 0 \quad (1.66)$$

Comme :

$$\varphi_{\lambda}(\lambda^*) = -\varepsilon_j S'_{j,\lambda}(\lambda^*) \quad (1.67)$$

une nouvelle condition nécessaire de minimum, est que :

$$\varepsilon_j \geq 0 \quad (1.68)$$

Si - $S_j(\lambda^*) < 0$ alors $\varepsilon_j = 0$

- $S_j(\lambda^*) = 0$ alors $\varepsilon_j \geq 0$

Ce qui correspond à la condition:

$$S_j(\lambda^*) \varepsilon_j = 0 \quad (j=1, \dots, q) \quad (1.69)$$

La condition de signe distingue les paramètres ε_j , des paramètres de Lagrange; On parle alors de paramètres de **Khun et Tucker**.

Les conditions d'optimalités sont donc les suivantes:

$$1 - L_j(\lambda^*, \varepsilon) = 0,$$

$$2 - \varepsilon_j S_j(\lambda^*) = 0 \quad (j = 1, \dots, q)$$

$$3 - \varepsilon_j \geq 0 \quad (j=1, \dots, q)$$

avec $L_{\lambda, \lambda}(\lambda^*, \varepsilon)$ non négative.

Ces conditions peuvent être obtenues, en ramenant les contraintes de type inégalité à des contraintes de type égalité, par l'artifice suivant:

$$S_j(\lambda) < 0 \Leftrightarrow S_j(\lambda) + \tau_j^2 = 0 \quad (1.70)$$

Les paramètres τ_j étant à déterminer en même temps que les paramètres λ .

II-2-3 : Méthodes de résolution

Plusieurs méthodes ont été développées pour la résolution des problèmes de programmation non linéaire, avec contraintes [1-7][1-21]. Parmi tous ces algorithmes, les méthodes de pénalisation sont certainement celles qui sont les moins compliquées à programmer et en même temps, les plus efficaces.

Elles consistent à résoudre le problème par la construction d'une séquence de solutions d'un problème d'optimisation sans contraintes.

Le problème initial s'écrit alors sous la forme:

$$\min (\varphi(\lambda) + \alpha_j^I p_j^I(\lambda) + \alpha_j^{II} p_j^{II}(\lambda)) \quad (1.71)$$

avec $p_j^I(\lambda)$, $p_j^{II}(\lambda)$: les fonctions de pénalisations et α_j^I , α_j^{II} une suite croissante de nombres positifs.

Les fonctions de pénalisations sont en général définies par:

$$p_j^I(\lambda) = \sum_{i=1}^q (\max(S_i(\lambda), 0))^2 \quad (1.72)$$

et

$$p_j''(\lambda) = -\sum_{i=1}^q \frac{1}{S_i(\lambda)} \quad \text{ou} \quad p_j''(\lambda) = -\sum_{i=1}^q \log(-S_i(\lambda)) \quad (1.73)$$

Les algorithmes récemment développés utilisent plutôt la fonction exponentielle [1-24].

Soit:

$$S_{\text{ext}}(\lambda) = S_{\text{max}}(\lambda) = \max(S_i(\lambda)) \quad i = 1, \dots, q \quad (1.74)$$

on pose:

$$S_s(\lambda) = S_{\text{ext}}(\lambda) + \frac{1}{p} \log_a \left(\sum_{i=1}^q a^{p(S_i(\lambda) - S_{\text{ext}}(\lambda))} \right) \quad (1.75)$$

le problème revient ensuite, à résoudre le problème sans contraintes:

$$\min \varphi(\lambda) + \alpha' S_s(\lambda) + \alpha'' S_s(\lambda)^2 \quad (1.76)$$

en utilisant une méthode de gradient, par exemple.

On remarque que:

$$S_{\text{max}}(\lambda) \leq S_s(\lambda) \leq (S_{\text{max}}(\lambda) + \frac{1}{p} \log_a q) \quad (p > 0) \quad (1.77)$$

Si on accepte une erreur *tol*, telle que:

$$\frac{1}{|p|} \log_a q < \text{tol} \quad (1.78)$$

alors la base *a* doit vérifiée:

$$a > q^{\frac{1}{|p| \cdot \text{tol}}} \quad (1.79)$$

Dans l'algorithme, les valeurs de *a* (*a* > 1), α' , α'' sont mise à jour à chaque itération.

Dans certains algorithmes, on utilise directement la base *e* (népérien) au lieu de la base *a*.

III- Performances des méthodes du type gradient

Les méthodes d'optimisation du type gradient, sont des méthodes itératives. Leur efficacité dépend essentiellement des propriétés de la fonction à minimiser, du nombre de paramètres à estimer ainsi que du choix des valeurs initiales pour l'algorithme.

Les méthodes DFP et BFGS sont supérieures, du point de vue convergence, à la méthode du gradient conjugué mais elles ont l'inconvénient de nécessiter à chaque itération, un temps de calcul et un espace mémoire plus grands. C'est pour cette raison, qu'on évite de les utiliser pour la résolution des problèmes comportant un nombre de variables important.

La méthode de Gauss-Newton est très performante, pourvu que la condition sur les résidus soit vérifiée.

La méthode de Marquardt est considérée comme l'une des méthodes de résolution de problèmes de moindres carrés non linéaires, les plus efficaces [1-1][1-2]. De plus, et contrairement à la méthode BFGS, elle ne nécessite pas des calculs longs ni d'espace mémoire important.

Pour mieux situer les performances des méthodes du type gradient, nous avons résolu différents problèmes de notre choix, en utilisant les deux méthodes les plus performantes (d'après la littérature), à savoir les méthodes de Marquardt et BFGS. Pour cela, on utilisera successivement chacune des deux algorithmes avec des valeurs initiales différentes.

Problème N°1

$$y(t, x) = x_1 e^{-(t x_2)} + x_3 e^{-(t x_4)}$$

les données y , sont calculées avec cette expression et pour $x = [2; 1; 1.5; 2]$

Le problème est résolu, en initialisant chacun des deux algorithmes avec différentes valeurs initiales. On remarque alors que pour certaines valeurs initiales, les algorithmes convergent vers x en un nombre d'itérations indiquées dans le tableau n°1; alors que pour d'autres, on a divergence.

Valeurs initiales	BFGS	Marquardt
[3; 1; 3; 2]	n. itérations = 10	n. itérations = 8
[3; 0; 3; 1]	n. itérations = 30	n. itérations = 10
[1; 1; 1; 1]	divergence	n. itérations = 8
[3; 0; 0; 1]	divergence	n. itérations = 9
[2; 0; 1; 1]	n. itérations = 16	n. itérations = 10
[2; 2; 2; 2]	divergence	n. itérations = 8

Tableau n°1

Problème N°2

$$y(t, x) = x_1 e^{-(t-x_2)^2} + x_3 e^{-(t-x_4)^2}$$

les données y_i sont calculées avec cette expression et pour $x = [-2; 2; 2.5; 4]$ et les résultats obtenus pour ce problème, sont représentés dans le tableau n°2.

Valeurs initiales	BFGS	Marquardt
[2, 1, 1, 2]	n. itérations = 9	n. itérations = 7
[2, 0, 1, 2]	n. itérations = 18	n. itérations = 10
[3, 0, 1, 1]	divergence	n. itérations = 11
[3, 0, 1, 0.5]	divergence	divergence
[3, 0, 3, 1]	n. itérations = 30	n. itérations = 16
[2, 0, 1, 1.5]	n. itérations = 22	n. itérations = 10

Tableau n°2

Problème N°3

$$y(t, x) = \frac{x_1(t^2 + x_2 t)}{(t^2 + x_3 t + x_4)}$$

les données y_i sont calculées avec cette expression et pour $x = [0.25; 0.4; 0.5; 0.4]$ et des valeurs de t comprises entre 0 et 10 avec un pas égale à 1 ($m=11$).

Les résultats obtenus pour ce problème, sont représentés dans le tableau n°3.

Valeurs initiales	BFGS	Marquardt
[0.1, 0.1, 0.1, 0.1]	n. itérations = 12	n. itérations = 12
[10, 1, 5, 1]	divergence	n. itérations = 36
[1, 1, 1, 1]	n. itérations = 20	n. itérations = 12
[10, 10, 10, 10]	n. itérations = 138	n. itérations = 41
[1, 100, 10, 1]	n. itérations = 185	n. itérations = 119
[0.1, 0.1, 0.5, 0.2]	n. itérations = 16	n. itérations = 12

Tableau n°3

Problème N°4

$$y(t, x) = x_1 + \frac{t}{x_2 v + x_3 w}; \quad \text{avec } v = 16 - t \quad \text{et } w = \min(v, t)$$

les données y_i sont calculées avec cette expression et pour $x = [0.1; 5; 0.84]$ et t prenant des valeurs variant de 1 à 15 avec un pas égale à 1 ($m=15$).

Les résultats obtenus pour ce problème, sont représentés dans le tableau n°4.

Valeurs initiales	BFGS	Marquardt
[1; 1; 1]	n. itérations = 12	n. itérations = 10
[10; 1; 1]	n. itérations = 13	n. itérations = 10
[0; 50; 0]	divergence	n. itérations = 17
[10; 1; 15]	divergence	divergence
[0; 0.1; 0]	n. itérations = 18	n. itérations = 12
[0; 50; 10]	divergence	n. itérations = 22

Tableau n°4

Problème N°5

$$y(t, x) = (x_1 + x_2 t - e^t)^2 + (x_3 + x_4 \sin(t) - \cos(t))^2$$

les données y_i sont calculées avec cette expression et pour $x = [-2; 25; 5; -5]$ et pour t variant entre 0 et 4, avec un pas de 0.2.

Les résultats obtenus pour ce problème, sont représentés dans le tableau n°5.

Valeurs initiales	BFGS	Marquardt
[1; 10; 1; 0]	n. itérations = 20	n. itérations = 12
[1; 20; 4; -2]	n. itérations = 14	n. itérations = 10
[0; 20; 0; -2]	divergence	n. itérations = 10
[1; 15; 10; -10]	n. itérations = 18	n. itérations = 9
[1; 40; 10; -10]	divergence	divergence

Tableau n°5

On remarque que le nombre d'itérations nécessaires pour la convergence de la méthode de Marquardt, est moins élevé ou au pire du même ordre que ceux nécessaires pour la méthode BFGS. Sachant qu'à chaque itération, la 2^o méthode nécessite plus de calcul. Ceci montre une supériorité de la méthode de Marquardt.

D'autre part, on remarque que pour certaines valeurs initiales l'un des deux algorithmes, voire les deux divergent.

La majorité des études faites sur les méthodes itératives, ont montré que ces dernières sont performantes, si les valeurs initiales sont choisies dans un proche voisinage de la solution. Dans le cas contraire, elles peuvent converger très lentement voire même diverger.

IV- Méthode de continuation

Dans la plus part des problèmes d'estimation des paramètres, on n'a pas une connaissance à priori des valeurs numériques des paramètres à estimer. Le choix des valeurs initiales, de l'algorithme de résolution se fait alors de façon arbitraire. On peut donc s'attendre à une divergence possible de l'algorithme.

Les méthodes de continuation telles que celles décrites dans [1-23][1-24] constituent une solution à ce type de problèmes. La convergence d'un l'algorithme itératif se trouve, très souvent améliorée, même pour un mauvais choix des valeurs initiales.

Le principe de ces méthodes consiste à modifier le problème initialement posé, par l'introduction d'un paramètre indépendant $k \in [0,1]$, dans la fonction coût initiale tel que pour:

- $k = 0$: on a une fonction coût qui est minimale pour les valeurs initiales choisies.
- $k = 1$: on retrouve la fonction coût initiale, que l'on cherche à minimiser.

Le problème revient donc à chercher les minimas des fonctions:

$$\varphi^{kj}(\lambda) = \sum_{j=1}^m (f_j^{kj})^2(\lambda) \quad (1.54)$$

avec

$$f_j^{kj}(\lambda) = f_j(\lambda) + (kj - 1)f_j(\lambda_0) \quad (1.55)$$

Pour chacune des valeurs intermédiaires k_j de k ($0 = k_0 < k_1 < \dots < k_l = 1$), on obtient une fonction coût intermédiaire dont on déterminera le minimum $(\lambda^l)^*$. Le minimum $(\lambda^l)^*$ de la fonction $\varphi^{k_l}(\lambda)$ sera également le minimum de la fonction $\varphi(\lambda)$.

Le pas Δk peut être pris constant ou variable.

Pour montrer les améliorations apportées par cette méthode. Nous avons résolu les problèmes précédents, avec les mêmes algorithmes combinés à la méthode de continuation

On s'est surtout intéressés aux cas des valeurs initiales pour lesquelles on a eu divergence auparavant. Le pas α a été pris constant, égale à 0.2 (sauf dans le cas du problème n°4 où on a eu convergence avec un pas de 0.1). Les résultats obtenus sont indiqués dans les tableaux numérotés de 6 à 15

Problème N°1

Méthode : BFGS

Valeurs initiales : [3, 0, 0, 1]

paramètre k	paramètres	nombre itérations
0.2	2.4445, 0.0042, 0.655, 1.46	15
0.4	1.8899, 0.0111, 1.309, 1.4609	12
0.6	1.3377, 0.0247, 1.9608, 1.4628	11
0.8	0.7955, 0.0639, 2.6026, 1.4683	12
1	1.9999, 1.0000, 1.500, 1.9999	21

Tableau n°6

Pour les valeurs initiales [1; 1; 1; 1] et [2; 2; 2; 2], on a toujours divergence de la méthode BFGS.

Problème N°2

Méthode : BFGS

Valeurs initiales : [3, 0, 1, 0.5]

paramètre k	paramètres	nombre itérations
0.2	3.0775, 0.226, 0.5318, 3.1924	15
0.4	2.3677, 0.3677, 1.0554, 3.2743	6
0.6	1.7361, 0.7336, 1.5644, 3.3906	6
0.8	1.6989, 1.6922, 1.9977, 3.8231	8
1	2.0000, 2.0000, 2.5000, 4.0000	7

Tableau n°7

Méthode : Marquardt

Valeurs initiales : [3, 0, 1, 0.5]

paramètre k	paramètres	nombre itérations
0.2	2.7912, 0.260, 0.6681, 2.1585	12
0.4	2.2296, 0.5358, 1.0675, 3.2051	8
0.6	1.7635, 0.9787, 1.5498, 3.4717	6
0.8	1.7651, 1.7061, 1.9971, 3.8385	7
1	2.0000, 2.0000, 2.5000, 4.0000	7

Tableau n°8

Problème N°3

Méthode : BFGS

Valeurs initiales : [10, 1.5, 1]

paramètre k	paramètres	nombre itérations
0.2	8.0512, 1.0466, 5.0259, 1.0685	15
0.4	6.1025, 1.1234, 5.0695, 1.1779	16
0.6	4.1541, 1.2747, 5.1581, 1.3810	16
0.8	2.2067, 1.7157, 5.4373, 1.8998	17
1	0.2499, 0.4000, 0.5000, 0.3999	36

Tableau n°9

Problème N°4

Méthode : BFGS

Valeurs initiales : [0, 50, 0]

paramètre k	paramètres	nombre itérations
0.2	0.0215, 17.9019, 2.0001	14
0.4	0.04, 10.8851, 1.5383	12
0.6	0.06, 7.8183, 1.2118	10
0.8	0.08, 6.0994, 0.9935	10
1	0.1, 5.00, 0.84	10

Tableau n°10

Méthode : Marquardt

Valeurs initiales : [10, 1, 15]

paramètre k	paramètres	nombre itérations
0.2	8.018, 2.0672, 9.6879	14
0.4	6.040, 3.529, 5.909	12
0.6	4.0608, 4.443, 3.3917	12
0.8	2.0807, 4.8718, 1.8216	12
1	0.1, 5, 0.84	16

Tableau n°11

Méthode : BFGS

Valeurs initiales : [1 ; 10; 15]

Pour ces valeurs initiales et un pas de continuation constant de 0.2, on a divergence. Par contre pour un pas égale à 0.1, on a convergence. Les valeurs obtenues, pour certaines valeurs de k, sont indiquées dans le tableau n°12.

Paramètre k	paramètres	nombre itérations
0.1	9.0088, 1.5228, 12.085	14
0.3	7.0294, 2.8741, 7.6350	12
0.6	4.0608, 4.4437, 3.3915	11
0.9	1.0904, 4.9631, 1.2748	9
1	0.0999, 5.0000, 0.8399	9

Tableau n°12

Problème N°5

Méthode : BFGS

Valeurs initiales : [0; 20; 0 ; -2]

paramètre k	paramètres	nombre itérations
0.2	0.6732, 21.0371, 4.01181, -4.8805	21
0.4	1.0355, 22.1196, 4.8091, -5.9696	10
0.6	1.3771, 23.1339, 5.1407, -6.2361	11
0.8	1.7015, 24.0903, 5.1764, -5.9106	9
1	2, 25, 5, -5	9

Tableau n°13

Méthode : Marquardt

Valeurs initiales : [1 , 30, 10, -1]

paramètre k	paramètres	nombre itérations
0.2	1.9696, 28.3471, 8.6821, -4.5067	14
0.4	2.2303, 27.4016, 7.8937, -5.2828	9
0.6	2.2231, 26.9307, 7.4591, -5.4716	10
0.8	2.0902, 25.4919, 5.7885, -5.322	9
1	1, 25, 5, -5	12

Tableau n°14

Méthode : BFGS

Valeurs initiales : [1 , 30, 10, -1]

paramètre k	paramètres	nombre itérations
0.2	1.996, 28.8351, 9.0671, -3.8463	14
0.4	2.20665, 27.8722, 8.2967, -4.9685	11
0.6	2.2231, 26.9607, 7.4591, -5.4716	13
0.8	2.1445, 25.9774, 6.4305, -5.4964	11
1	2, 25, 5, -5	11

Tableau n°15

Conclusion

La résolution des problèmes de moindres carrés non linéaires par les méthodes du type gradient, est une technique courante. Parmi les nombreuses méthodes existantes, les algorithmes de Marquardt et BFGS sont les plus efficaces.

La comparaison entre ces deux méthodes, a donné les résultats présentés dans ce chapitre. Ces derniers confirment la supériorité et l'efficacité de la méthode de Marquardt.

Le problème du choix des valeurs initiales de l'algorithme, peut être résolu en combinant l'algorithme à une méthode de continuation. Cette méthode peut être appliquée à n'importe quel algorithme itératif, même ceux utilisés pour la résolution des systèmes d'équations non linéaires.

CHAPITRE II

***IDENTIFICATION DES PARAMETRES
PAR LES B-SPLINES***

I- Introduction

Les fonctions splines introduites par Shoenberg[2-7] ont été surtout utilisées pour l'interpoler un ensemble de données. Il existe plusieurs présentations de ces fonctions, la plus connue et en même temps la plus utilisée, est celle exprimée comme une somme de fonctions spéciales appelées B-splines

L'identification des paramètres en utilisant les B-splines, est une méthode qui est surtout utilisée lorsque les données dont on dispose, représentent une grandeur dont le modèle est une équation (ou un système d'équations) différentielles de la forme $y'(t) = f(t, y, \lambda)$ ($y \in R^n, \lambda \in R^m$) ou aux dérivées partielles et que les conditions initiales ne soient pas définies. C'est à dire, dans le cas où l'expression de la fonction intervenant dans le critère à minimiser n'est pas connue explicitement[2-1][2-2][2-3]. Cette méthode se base sur l'interpolation des données par des fonctions B-splines. La dérivée de ces fonctions est ensuite, utilisée pour construire le critère à minimiser[2-5][2-10]

Cette méthode attrayante, est moins coûteuse en temps de calcul que les certaines méthodes itératives utilisées pour résoudre ce type de problèmes et décrites par Bard (1974), et Benson (1979). En général, ces méthodes procèdent comme suit : à partir d'une estimation initiale des paramètres, on intègre le système d'équations différentielles. L'écart, entre les solutions trouvées et les observations, est traité comme une fonction des paramètres que l'on minimise.

Parmi les inconvénients de ces méthodes itératives, on peut citer

- Le résultat de l'intégration numérique dépend des conditions initiales. Si ces dernières ne sont pas précises, on aura des résultats erronés.
- La procédure nécessite une assez bonne estimation initiale des paramètres; ce qui n'est pas le cas dans la plupart des problèmes d'identification.
- A chaque itération, on doit intégrer le système d'équations différentielles, en utilisant les nouveaux paramètres trouvés, et calculer le nouvel écart. Ceci nécessite beaucoup de calculs.

II- Définition des fonctions B-splines

Soit $\pi = (t_i)_{i \in \mathbb{Z}}$, une suite croissante de nombres réels.

Posons

$$g_m(s, t) = (s - t)_+^{m-1} = \begin{cases} (s - t)^{m-1} & \text{si } s \geq t \\ 0 & \text{ailleurs} \end{cases} \quad (2-1)$$

La B-spline de degré m , $M_i^m(t)$ est définie comme la différence divisée d'ordre m , de la fonction $g_m(s, t)$, basée sur les points: $t_i, t_{i+1}, \dots, t_{i+m}$.

$$* - [t_i] = g_m(t_i, t)$$

$$* - [t_i, t_j] = (g_m(t_i, t) - g_m(t_j, t)) / (t_i - t_j)$$

$$* - M_i^m(t) = d[t_i, \dots, t_{i+m}] = \frac{d[t_i, \dots, t_{i+m-1}] - d[t_i, \dots, t_{i+m-2}, t_{i+m}]}{(t_{i+m-1} - t_{i+m})}$$

La B-spline normalisée est définie par:

$$N_i^m(t) = (t_{i+m} - t_i) M_i^m(t) \quad (2-2)$$

III- Propriétés

Ces fonctions satisfont les propriétés suivantes:

$$\text{a / - } N_i^0(t) = \begin{cases} 1 & \text{si } t_i \leq t \leq t_{i+1} \\ 0 & \text{ailleurs} \end{cases} \quad (2-3)$$

$$\text{b / - } N_i^m(t) = \frac{t - t_i}{t_{i+m} - t_i} N_i^{m-1}(t) + \frac{t_{i+m+1} - t}{t_{i+m+1} - t_{i+1}} N_{i+1}^{m-1}(t) \quad (2-4)$$

Les termes $\frac{t - t_i}{t_{i+m} - t_i} N_i^{m-1}(t)$ et $\frac{t_{i+m+1} - t}{t_{i+m+1} - t_{i+1}} N_{i+1}^{m-1}(t)$ sont nuls, respectivement pour $t_i = t_{i+m}$ et $t_{i+1} = t_{i+m+1}$.

c / - $N_i^m(t)$ est une fonction polynomiale par morceaux.

d / - $N_i^m(t) = 0$ si $t \notin [t_i, t_{i+m+1}[$

e / - $N_i^m(t) > 0$ si $t \in [t_i, t_{i+m+1}[$

f / - $\sum N_i^m(t) = 1$

g / - Pour $m > 0$, et $t \in \mathbb{R}$, la dérivée de la B-spline $N_i^m(t)$ à droite est égale à:

$$N_i^{m'}(t) = m \left(\frac{N_i^{m-1}(t)}{t_{i+m} - t_i} - \frac{N_{i+1}^{m-1}(t)}{t_{i+m+1} - t_{i+1}} \right) \quad (2-5)$$

IV- Les différents types de fonctions B-splines

IV-1: B-spline linéaire (m =1)

$$N_i^1(t) = \frac{t-t_i}{t_{i+1}-t_i} N_i^0(t) + \frac{t_{i+2}-t}{t_{i+2}-t_{i+1}} N_{i+1}^0(t) \quad (2-6)$$

d'où:

$$N_i^1(t) = \begin{cases} \frac{t-t_i}{t_{i+1}-t_i} & \text{si } t_i \leq t \leq t_{i+1} \\ \frac{t_{i+2}-t}{t_{i+2}-t_{i+1}} & \text{si } t_{i+1} \leq t \leq t_{i+2} \end{cases}$$

IV-2: B-spline quadratique (m =2)

$$N_i^2(t) = \frac{t-t_i}{t_{i+2}-t_i} N_i^1(t) + \frac{t_{i+3}-t}{t_{i+3}-t_{i+1}} N_{i+1}^1(t) \quad (2-7)$$

d'où:

$$N_i^2(t) = \begin{cases} \frac{(t-t_i)^2}{(t_{i+2}-t_i)(t_{i+1}-t_i)} & \text{si } t_i \leq t \leq t_{i+1} \\ \frac{(t-t_i)(t_{i+2}-t)}{(t_{i+2}-t_i)(t_{i+2}-t_{i+1})} + \frac{(t-t_{i+1})(t_{i+3}-t)}{(t_{i+3}-t_{i+1})(t_{i+2}-t_{i+1})} & \text{si } t_{i+1} \leq t \leq t_{i+2} \\ \frac{(t_{i+3}-t)^2}{(t_{i+3}-t_{i+1})(t_{i+3}-t_{i+2})} & \text{si } t_{i+2} \leq t \leq t_{i+3} \end{cases}$$

IV-3 : B-spline cubique (m =3)

$$N_i^3(t) = \frac{t-t_i}{t_{i+3}-t_i} N_i^2(t) + \frac{t_{i+4}-t}{t_{i+4}-t_{i+1}} N_{i+1}^2(t) \quad (2-8)$$

d'où:

$$N_i^3(t) = \begin{cases} \frac{(t-t_i)^3}{(t_{i,3}-t_i)(t_{i,2}-t_i)(t_{i,1}-t_i)} & t_i \leq t \leq t_{i+1} \\ \frac{(t-t_i)^2(t_{i,2}-t)}{(t_{i,3}-t_i)(t_{i,2}-t_i)(t_{i,2}-t_{i+1})} + \frac{(t-t_{i,1})(t_{i,3}-t)(t-t_i)}{(t_{i,3}-t_i)(t_{i,3}-t_{i+1})(t_{i,2}-t_{i+1})} + \frac{(t-t_{i,1})^2(t_{i,4}-t)}{(t_{i,2}-t_i)(t_{i,3}-t_{i+1})(t_{i,2}-t_{i+1})} & \text{si } t_{i,1} \leq t \leq t_{i,2} \\ \frac{(t-t_i)(t_{i,3}-t)^2}{(t_{i,3}-t_i)(t_{i,3}-t_{i+1})(t_{i,3}-t_{i,2})} + \frac{(t_{i,4}-t)(t-t_{i,1})(t_{i,3}-t)}{(t_{i,4}-t_{i+1})(t_{i,3}-t_{i+1})(t_{i,3}-t_{i,2})} + \frac{(t-t_{i,2})(t_{i,4}-t)^2}{(t_{i,4}-t_{i+1})(t_{i,4}-t_{i,2})(t_{i,3}-t_{i,2})} & \text{si } t_{i,2} \leq t \leq t_{i,3} \\ \frac{(t_{i,4}-t)^3}{(t_{i,4}-t_{i,3})(t_{i,4}-t_{i,1})(t_{i,4}-t_{i,2})} & \text{si } t_{i,3} \leq t \leq t_{i+4} \end{cases}$$

V- Calcul des B-splines

Dans chaque intervalle, il y a seulement $(m + 1)$ B-splines de degré m , non nulles. $N_i^m(t)$ dépend seulement de $N_{i-1}^{m-1}(t)$, car $N_{i+1}^{m-1}(t)$ est nulle, alors que $N_{i-1}^{m-1}(t)$, $(0 < l \leq m)$ dépend aussi bien de $N_{i-1,1}^{m-1}(t)$ que de $N_{i-1}^{m-1}(t)$.

Ainsi, pour calculer les B-splines de degré m , on doit d'abord calculer les $(m-1)$ B-splines de degré inférieur.

La figure suivante montre l'interdépendance entre les fonctions B-splines de degré inférieur ou égale à 3.

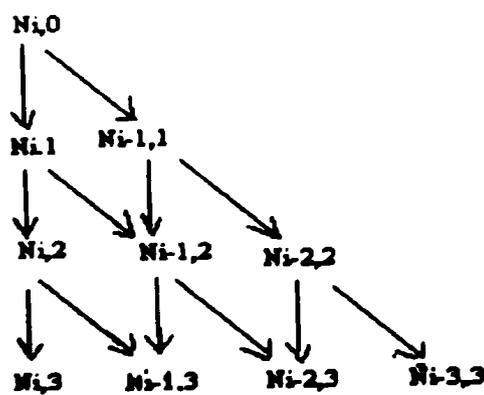


fig. 2.1 : Schéma de calcul des B-splines pour $m \leq 3$

Les flèches indiquent le sens de calcul, en tenant compte de l'indice i . La flèche verticale indique la multiplication par le 1^o terme de la relation (2-8) et la flèche oblique, la multiplication par le 2^o terme de la même relation. Les termes non représentés sont nuls.

VI- Approximations des données par des B-Splines

Soit $(y_1, t_1), (y_2, t_2), \dots, (y_N, t_N)$, un ensemble de données définies dans l'intervalle $[a, b]$. Si on divise cet intervalle en $(k-1)$ noeuds tels que : $a \leq t_1^* < t_2^* < \dots < t_{k-1}^* \leq b$, alors on peut approximer chaque donnée y_j , par une courbe spline $P(t)$, de la forme :

$$P(t) = \sum_{i=-m}^{k-1} a_i N_i^m(t) = y_j \quad j = 1, \dots, N \quad (2-9)$$

Pour chaque intervalle $[t_i^*, t_{i+1}^*]$, on a $(m+1)$ paramètres inconnus $a_{-m}, a_{-m+1}, \dots, a_{k-1}$, parmi lesquels au plus $(m+1)$ sont non nuls (vu les propriétés des B-splines).

La donnée, correspondant à $t \in [t_i^*, t_{i+1}^*]$ est donc égale à la somme de ces $(m+1)$ B-splines, ce qui donne :

$$P(t) = \sum_{i=-m}^i a_i N_i^m(t) = y_j \quad (2-10)$$

Pour une approximation par des B-splines cubiques, on a :

$$P(t) = \sum_{i=-3}^i a_i N_i^3(t) = y_j \quad (2-11)$$

avec :

$$N_{i-3}(t) = \frac{(t_{i+1} - t)^3}{(t_{i+1} - t_{i-2})(t_{i+1} - t_{i-1})(t_{i+1} - t_i)}$$

$$N_{i-2}(t) = \frac{(t_{i+1} - t)^2(t - t_{i-2})}{(t_{i+1} - t_{i-2})(t_{i+1} - t_{i-1})(t_{i+1} - t_i)} + \frac{(t_{i+2} - t)(t - t_{i-1})(t_{i+1} - t)}{(t_{i+2} - t_{i-1})(t_{i+1} - t_{i-1})(t_{i+1} - t_i)} + \frac{(t - t_{i-1})(t_{i+2} - t)^2}{(t_{i+2} - t_{i-1})(t_{i+2} - t_i)(t_{i+1} - t_i)}$$

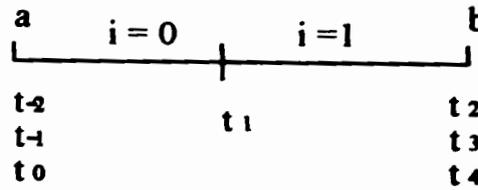
$$N_{i-1}(t) = \frac{(t_{i+1} - t)(t - t_{i-1})^2}{(t_{i+2} - t_{i-1})(t_{i+1} - t_{i-1})(t_{i+1} - t_i)} + \frac{(t_{i+2} - t)(t - t_{i-1})(t - t_i)}{(t_{i+2} - t_{i-1})(t_{i+2} - t_i)(t_{i+1} - t_i)} + \frac{(t - t_i)^2(t_{i+3} - t)}{(t_{i+3} - t_i)(t_{i+2} - t_i)(t_{i+1} - t_i)}$$

$$N_i(t) = \frac{(t - t_i)^3}{(t_{i+3} - t_i)(t_{i+2} - t_i)(t_{i+1} - t_i)}$$

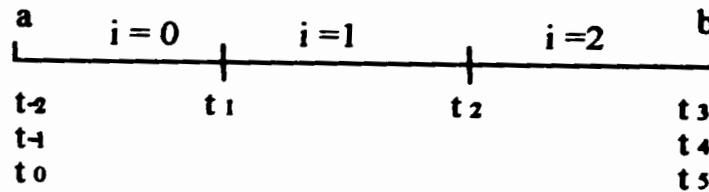
Les extrémités a et b , de l'intervalle, sont considérés comme des noeuds de multiplicité m .

a/ Valeur de i en fonction du nombre de noeuds

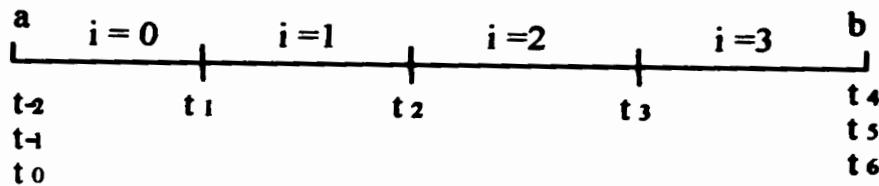
- 1 noeud



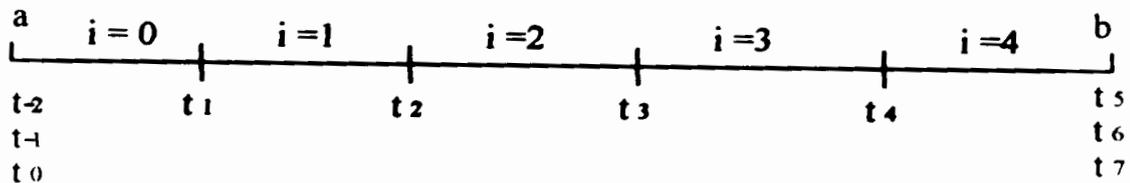
- 2 noeuds



- 3 noeuds



- 4 noeuds



Les noeuds peuvent coïncider ou alterner avec les abscisses des données. Leur nombre, ainsi que leur position sont des facteurs influents sur la qualité de l'approximation des données.

Chaque intervalle doit contenir au moins une donnée. S'il contient plus de $(m+1)$ données, on aura un système surdéterminé.

b/ Calcul des B-splines cubiques non nulles pour différentes valeur de i

*/ i=0

$$N_{-3}(t) = \frac{(t_1 - t)^3}{(t_1 - t_2)(t_1 - t_{-1})(t_1 - t_0)}$$

$$N_{-2}(t) = \frac{(t_1 - t)^2(t - t_2)}{(t_1 - t_2)(t_1 - t_{-1})(t_1 - t_0)} + \frac{(t_2 - t)(t - t_{-1})(t_1 - t)}{(t_2 - t_{-1})(t_1 - t_{-1})(t_1 - t_0)} + \frac{(t - t_0)(t_2 - t)^2}{(t_2 - t_{-1})(t_2 - t_0)(t_1 - t_0)}$$

$$N_{-1}(t) = \frac{(t_1 - t)(t - t_{-1})^2}{(t_2 - t_{-1})(t_1 - t_{-1})(t_1 - t_0)} + \frac{(t_2 - t)(t - t_{-1})(t - t_0)}{(t_2 - t_{-1})(t_2 - t_0)(t_1 - t_0)} + \frac{(t - t_0)^2(t_3 - t)}{(t_3 - t_0)(t_2 - t_0)(t_1 - t_0)}$$

$$N_0(t) = \frac{(t - t_0)^3}{(t_3 - t_0)(t_2 - t_0)(t_1 - t_0)}$$

*/ i=1

$$N_{-2}(t) = \frac{(t_2 - t)^3}{(t_2 - t_{-1})(t_2 - t_0)(t_2 - t_1)}$$

$$N_{-1}(t) = \frac{(t_2 - t)^2(t - t_{-1})}{(t_2 - t_{-1})(t_2 - t_0)(t_2 - t_1)} + \frac{(t_3 - t)(t - t_0)(t_2 - t)}{(t_3 - t_0)(t_2 - t_0)(t_2 - t_1)} + \frac{(t - t_1)(t_3 - t)^2}{(t_3 - t_0)(t_3 - t_1)(t_2 - t_1)}$$

$$N_0(t) = \frac{(t_2 - t)(t - t_0)^2}{(t_3 - t_0)(t_2 - t_0)(t_2 - t_1)} + \frac{(t_3 - t)(t - t_0)(t - t_1)}{(t_3 - t_0)(t_3 - t_1)(t_2 - t_1)} + \frac{(t - t_1)^2(t_4 - t)}{(t_4 - t_1)(t_3 - t_1)(t_2 - t_1)}$$

$$N_1(t) = \frac{(t - t_1)^3}{(t_4 - t_1)(t_3 - t_1)(t_2 - t_1)}$$

*/ i=2

$$N_{-1}(t) = \frac{(t_3 - t)^3}{(t_3 - t_0)(t_3 - t_1)(t_3 - t_2)}$$

$$N_0(t) = \frac{(t_3 - t)^2(t - t_0)}{(t_3 - t_0)(t_3 - t_1)(t_3 - t_2)} + \frac{(t_4 - t)(t - t_1)(t_3 - t)}{(t_4 - t_1)(t_3 - t_1)(t_3 - t_2)} + \frac{(t - t_2)(t_4 - t)^2}{(t_4 - t_1)(t_4 - t_2)(t_3 - t_2)}$$

$$N_1(t) = \frac{(t_3 - t)(t - t_1)^2}{(t_4 - t_1)(t_3 - t_1)(t_3 - t_1)} + \frac{(t_4 - t)(t - t_1)(t - t_2)}{(t_4 - t_1)(t_4 - t_2)(t_3 - t_2)} + \frac{(t - t_2)^2(t_5 - t)}{(t_5 - t_2)(t_4 - t_2)(t_3 - t_2)}$$

$$N_2(t) = \frac{(t - t_2)^3}{(t_5 - t_2)(t_4 - t_2)(t_3 - t_2)}$$

* / i=3

$$N_0(t) = \frac{(t_4 - t)^3}{(t_4 - t_1)(t_4 - t_2)(t_4 - t_3)}$$

$$N_1(t) = \frac{(t_4 - t)^2(t - t_1)}{(t_4 - t_1)(t_4 - t_2)(t_4 - t_3)} + \frac{(t_5 - t)(t - t_2)(t_4 - t)}{(t_5 - t_2)(t_4 - t_2)(t_4 - t_3)} + \frac{(t - t_3)(t_5 - t)^2}{(t_5 - t_2)(t_5 - t_3)(t_4 - t_3)}$$

$$N_2(t) = \frac{(t_4 - t)(t - t_2)^2}{(t_5 - t_2)(t_4 - t_2)(t_4 - t_2)} + \frac{(t_5 - t)(t - t_2)(t - t_3)}{(t_5 - t_2)(t_5 - t_3)(t_4 - t_3)} + \frac{(t - t_3)^2(t_6 - t)}{(t_6 - t_3)(t_5 - t_3)(t_4 - t_3)}$$

$$N_3(t) = \frac{(t - t_3)^3}{(t_6 - t_3)(t_5 - t_3)(t_4 - t_3)}$$

* / i=4

$$N_1(t) = \frac{(t_5 - t)^3}{(t_5 - t_2)(t_5 - t_3)(t_5 - t_4)}$$

$$N_2(t) = \frac{(t_5 - t)^2(t - t_2)}{(t_5 - t_2)(t_5 - t_3)(t_5 - t_4)} + \frac{(t_6 - t)(t - t_3)(t_5 - t)}{(t_6 - t_3)(t_5 - t_3)(t_5 - t_4)} + \frac{(t - t_4)(t_6 - t)^2}{(t_6 - t_3)(t_6 - t_4)(t_5 - t_4)}$$

$$N_3(t) = \frac{(t_5 - t)(t - t_3)^2}{(t_6 - t_3)(t_5 - t_3)(t_5 - t_3)} + \frac{(t_6 - t)(t - t_3)(t - t_4)}{(t_6 - t_3)(t_6 - t_4)(t_5 - t_4)} + \frac{(t - t_4)^2(t_7 - t)}{(t_7 - t_4)(t_6 - t_4)(t_5 - t_4)}$$

$$N_4(t) = \frac{(t - t_4)^3}{(t_7 - t_4)(t_6 - t_4)(t_5 - t_4)}$$

VIII- Identification des paramètres en utilisant les B-Splines

L'identification des paramètres en utilisant les B-splines, est une méthode qui est surtout utilisée lorsque les données dont on dispose, représentent une grandeur dont le modèle est une équation (ou un système d'équations) différentielles de la forme $y'(t) = f(t, y, \lambda)$ ($y \in R^n, \lambda \in R^{n\lambda}$) ou aux dérivées partielles. C'est à dire, dans le cas où on ne peut pas utiliser directement la méthode des moindres carrés.

L'identification des paramètres par la méthode des B-splines se fait en plusieurs étapes. En premier lieu, on détermine la courbe B-splines (en général des B-splines cubiques) qui approxime le mieux les données $y(t_j), (j = 1, \dots, m)$.

Pour cela, on utilise pour chaque donnée $y(t_j)$ la relation:

$$y(t_j) \cong \sum_{k=j-3}^j a_k N_k^3(t_j) \quad (j = 1, \dots, m) \quad (2-12)$$

On obtient ainsi un système d'équations linéaires, dont la résolution permet d'obtenir les coefficients a_k .

Ensuite, et dans le but d'avoir une courbe bien lisse, on calcule les ordonnées d'un plus grand nombre de points. Pour cela, on utilise la formule précédente avec des valeurs de t (autres que ceux correspondant aux données) très proches et situées dans l'intervalle considéré. Les coefficients a_k étant ceux déterminés précédemment.

La qualité de l'approximation dépend essentiellement du choix des noeuds et surtout de leur emplacement.

Le choix des noeuds, souvent arbitraire, ne donne pas obligatoirement une bonne approximation. Dans la plupart des cas, plusieurs essais sont nécessaires pour trouver l'emplacement des noeuds qui donnent la courbe qui approxime le mieux, les données.

Pour cela, à chaque choix de noeuds, il est important de visualiser la courbe obtenue, afin de voir laquelle est la plus proche des données. La sélection de la meilleure courbe, sera le résultat de l'appréciation visuelle de l'utilisateur.

L'étape suivante consiste à calculer:

$$\hat{y}(t_j) = y'(t_j) \cong \sum_{k=j-3}^j a_k (N_k^3(t_j))' \quad (j = 1, \dots, m) \quad (2-13)$$

où $(N_k^3(t_j))'$ représentent les dérivées des B-splines cubiques au point t_j .

Enfin, on procédera à l'identification des paramètres λ , en minimisant le critère quadratique:

$$\varphi(\lambda) \cong \sum_{j=1}^m (f(t_j, y, \lambda) - \hat{y}(t_j))^2 \quad (2-14)$$

Par un algorithme itératif, tel que l'algorithme de Marquardt ou BFGS.

Lors de la résolution de certains problèmes, par cette méthode on peut hésiter entre le choix de plusieurs courbes. Dans ce cas, on identifie les paramètres pour chaque courbe retenue. Ensuite, on intègre les équations modélisant le système, (par la méthode d'Euler ou de Runge-Kutta) en prenant les paramètres λ égaux à ceux trouvés. La valeur du résidu entre le résultat de l'intégration et les données $y(t_j), (j = 1, \dots, m)$ peut être un critère de choix entre les différentes solutions.

Pour mieux mettre en évidence ce procédé, nous présentons dans ce qui suit les résultats obtenus avec cette méthode, pour l'identification de paramètres intervenant dans un système modélisé par des équations différentielles avec des conditions initiales inconnues.

Pour chaque problème traité, les données sont approximées par des fonctions B-splines cubiques et le critère quadratique est minimiser par l'algorithme de Marquardt.

Les positions des nœuds ainsi que leur nombre, ont été choisis après examen des différentes courbes obtenues par approximation, en utilisant à chaque fois un nombre de nœuds et des positions différentes.

Problème n°1

Soit le système modélisé par l'équation différentielle suivante :

$$y_1' = p_1(126.2 - y_1)(91.2 - y_1)^2 - p_2 y_1^2$$

On dispose pour résoudre ce problème, également connu sous le nom de problème de Bellman, des données du tableau 2-1.

Les conditions initiales ne sont pas définies; il va falloir les identifier en même temps que les paramètres p_1 et p_2 .

t	1	2	3	4	5	6	7	8	10	12	15	20	25	30	40
y1	0	1.4	6.3	10.4	14.2	17.6	21.4	23	27	30.4	34.4	38.8	41.6	43.5	45.3

Tableau 2-1

1^oétape: détermination de la courbe qui approxime le mieux les données

Pour déterminer la meilleure courbe B-splines qui approxime les données, on essaie différentes combinaisons: positions-nombre de noeuds et pour chacune d'elle, on détermine les coefficients a_i ($i = k-m, \dots, k$). Les tableaux (2-2 à 2-6) indiquent les résultats de l'approximation des données du tableau 2-1, pour différentes combinaisons: nombre-position des noeuds.

Ces coefficients sont ensuite, utilisés pour calculer les ordonnées d'un plus grand nombre de points, appartenant à l'intervalle où ont été prélevés les données. Ce calcul va nous permettre d'obtenir un tracé plus lisse des courbes, ce qui facilitera l'appréciation de l'approximation.

Ainsi, chacune des courbes en trait plein, des figures (2-2 à 2-6), a été calculée pour 40 valeurs de t distinctes, appartenant à l'intervalle [1-40].

Nombre de noeud: 1

position: 10

	a_3	a_2	a_1	a_0	a_1
$y_1(t)$	-1.6054	12.8628	48.1544	40.7945	45.4324

Tableau 2-2

Nombre de noeud: 1

position: 20

	a_3	a_2	a_1	a_0	a_1
$y_1(t)$	-1.5775	28.0068	41.5030	44.8644	45.2995

Tableau 2-3

Nombre de noeuds: 2

positions: 10, 20

	a_3	a_2	a_1	a_0	a_1	a_2
$y_1(t)$	-1.3273	11.8434	31.8879	41.2470	45.7290	45.2833

Tableau 2-4

Nombre de noeuds: 2

positions: 5, 15

	a_3	a_2	a_1	a_0	a_1	a_2
$y_1(t)$	-0.4062	2.6610	24.9005	41.5126	44.9852	45.2717

Tableau 2-5

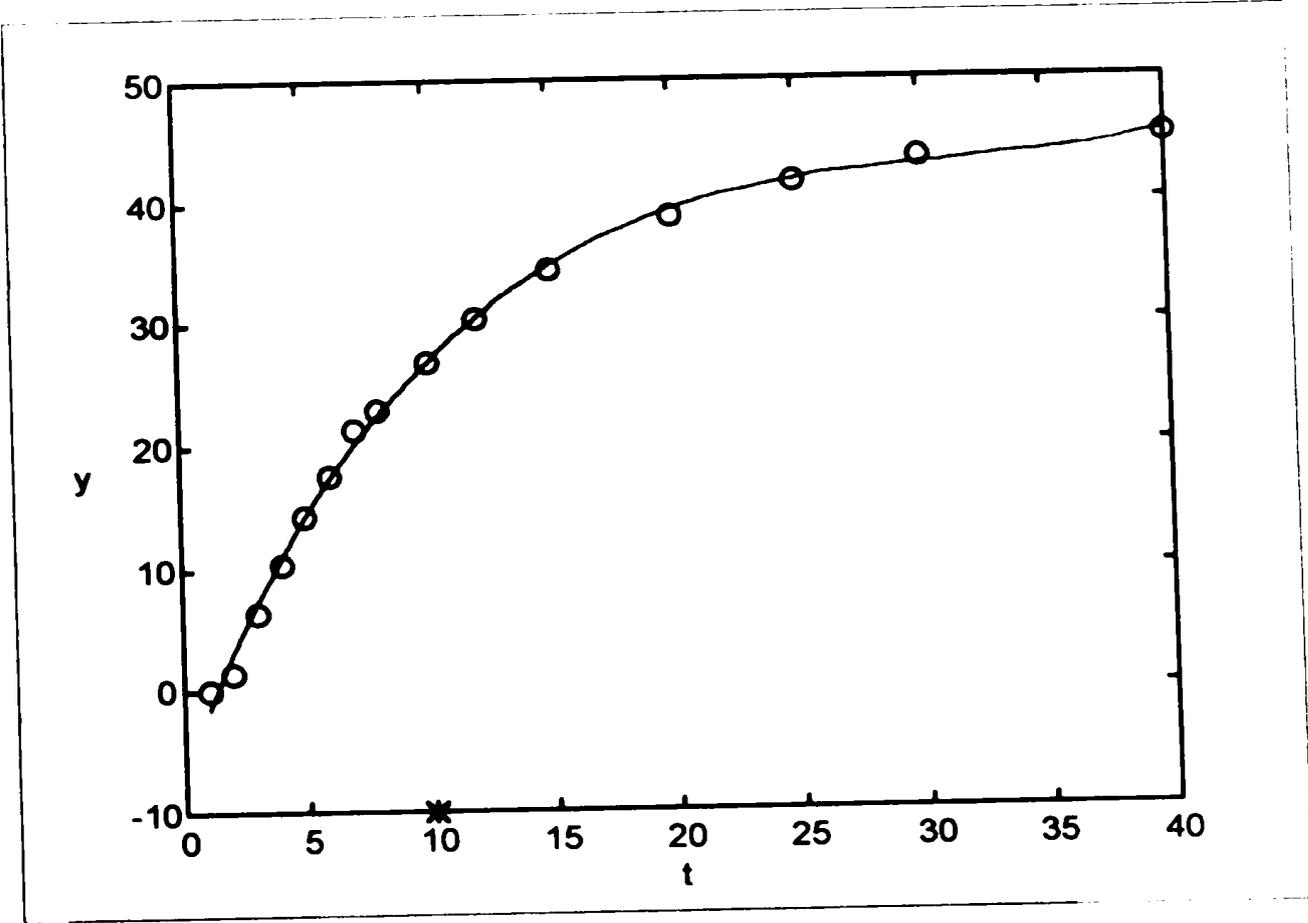


fig. 2-2: interpolation par des B-splines en utilisant 1 noeud (positions: 10)

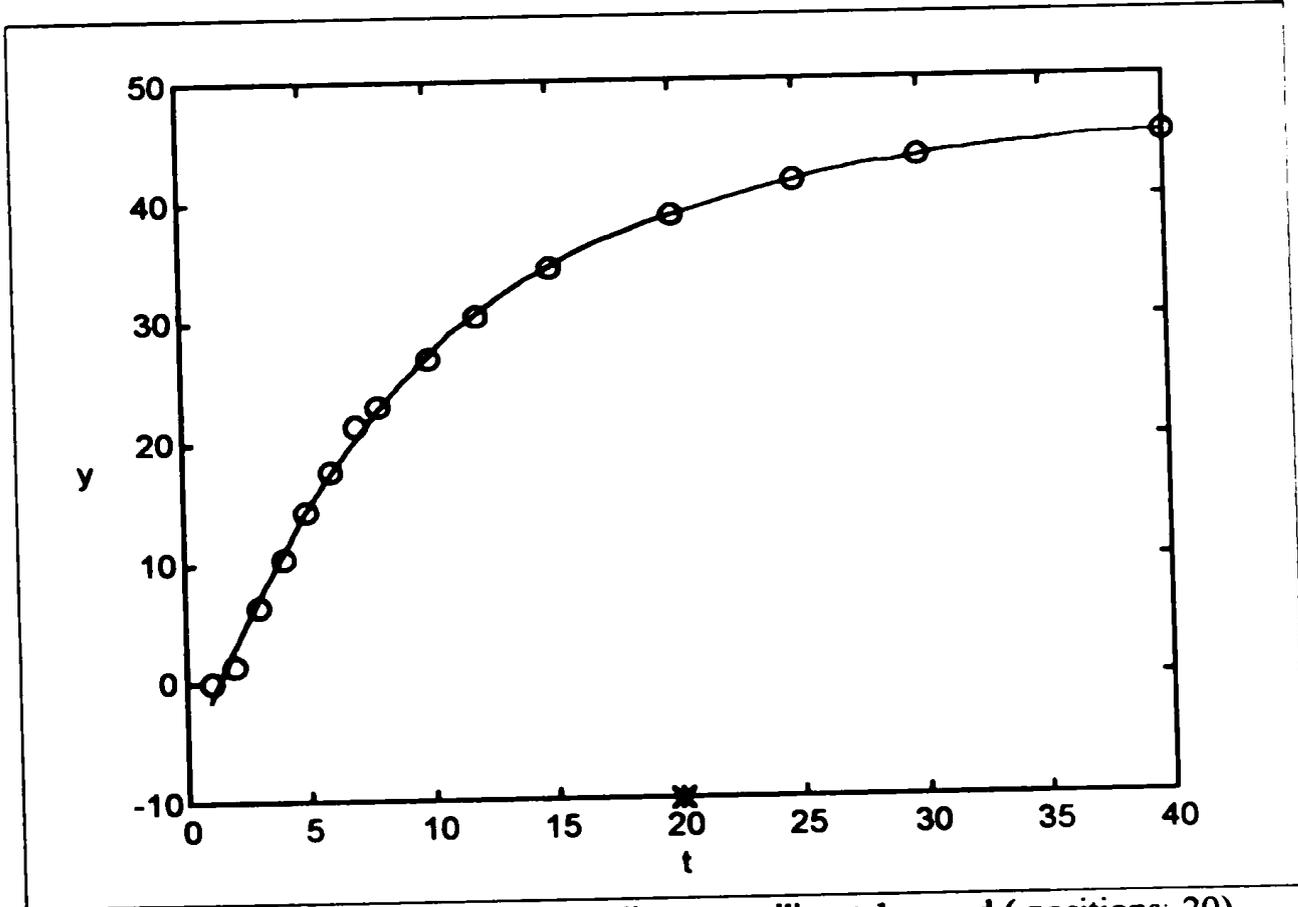


fig. 2-3: interpolation par des B-splines en utilisant 1 noeud (positions: 20)

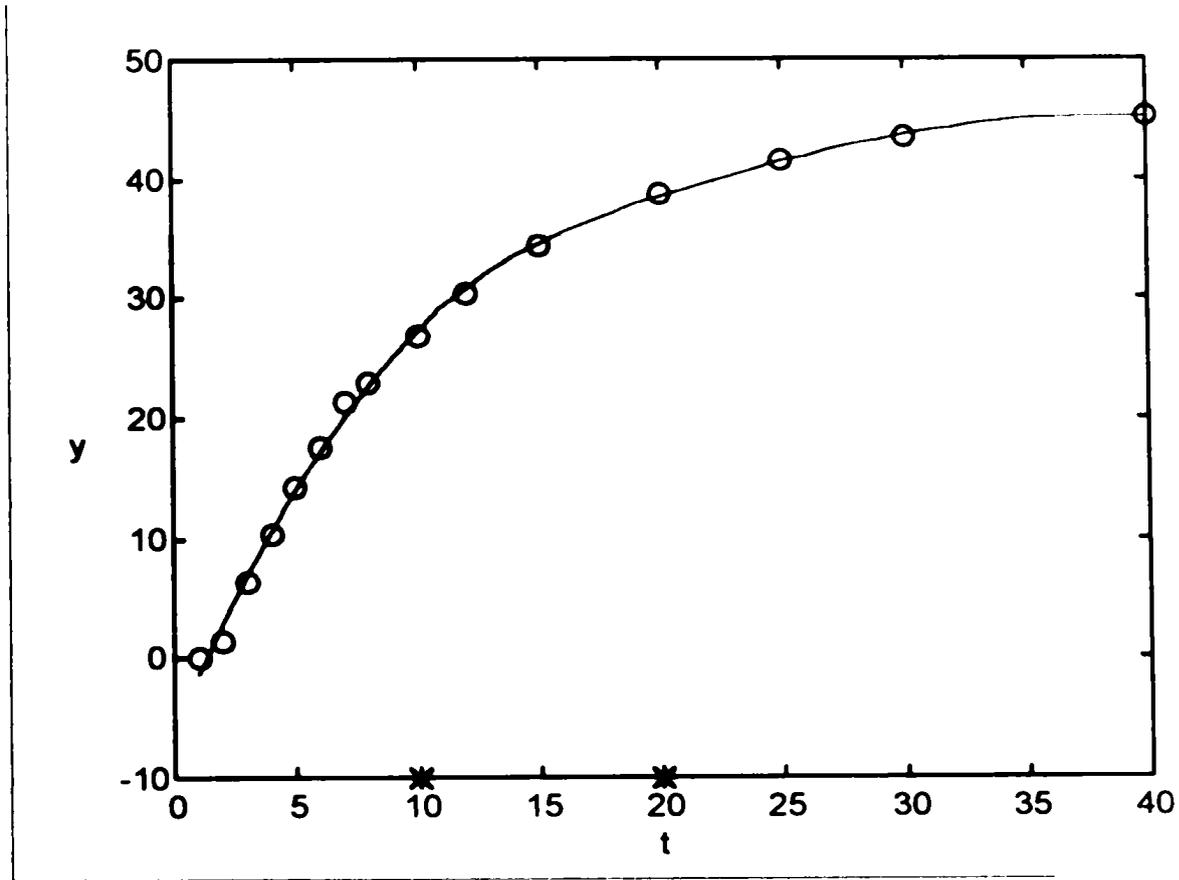


fig. 2-4: interpolation par des B-splines en utilisant 2 noeuds (positions: 10, 20)

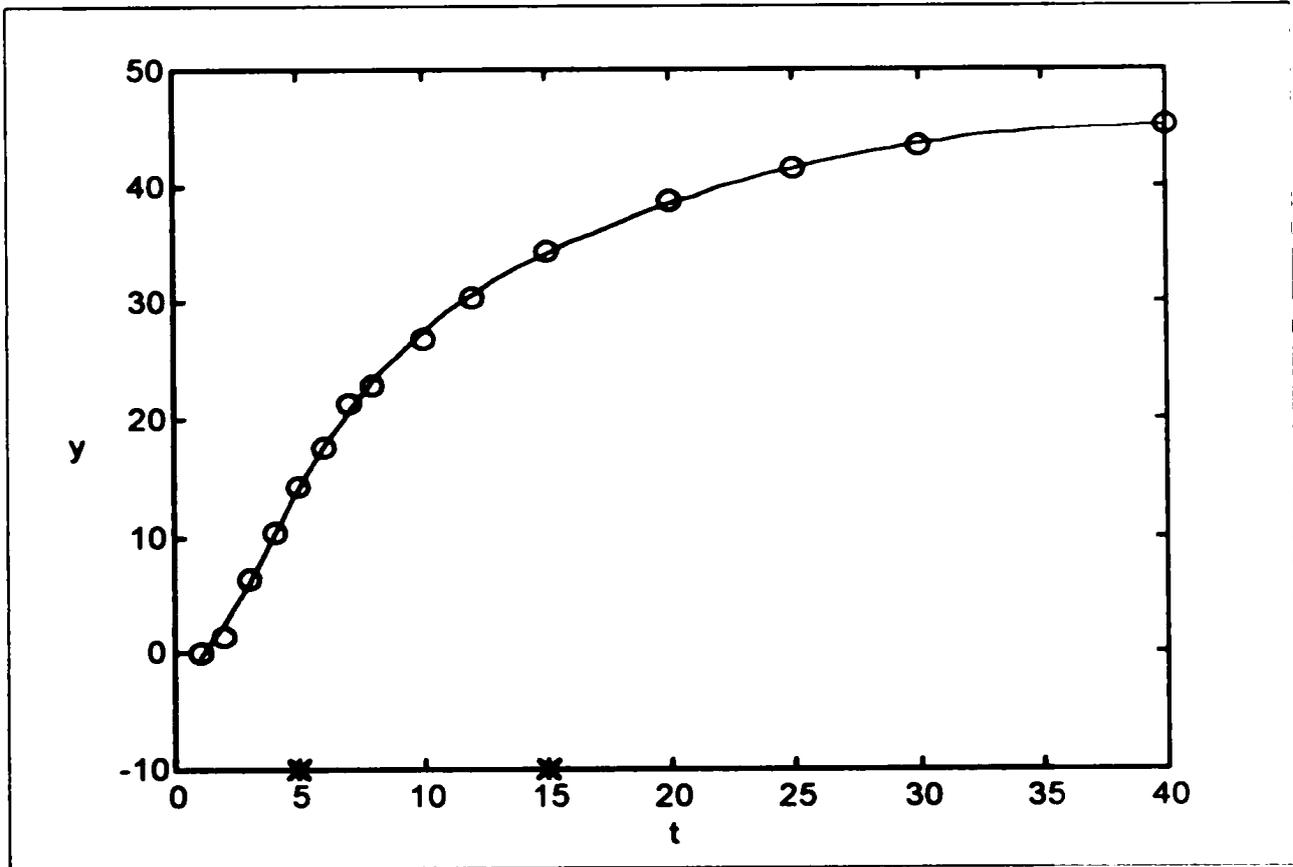


fig. 2-5: interpolation par des B-splines en utilisant 2 noeuds (positions: 5, 15)

Nombre de noeuds: 4

positions: 5, 10, 15, 25

	a_3	a_2	a_1	a_0	a_1	a_2	a_3	a_4
$y_1(t)$	-0.226	1.622	18.467	27.374	37.456	42.661	45.393	45.299

Tableau 2-6

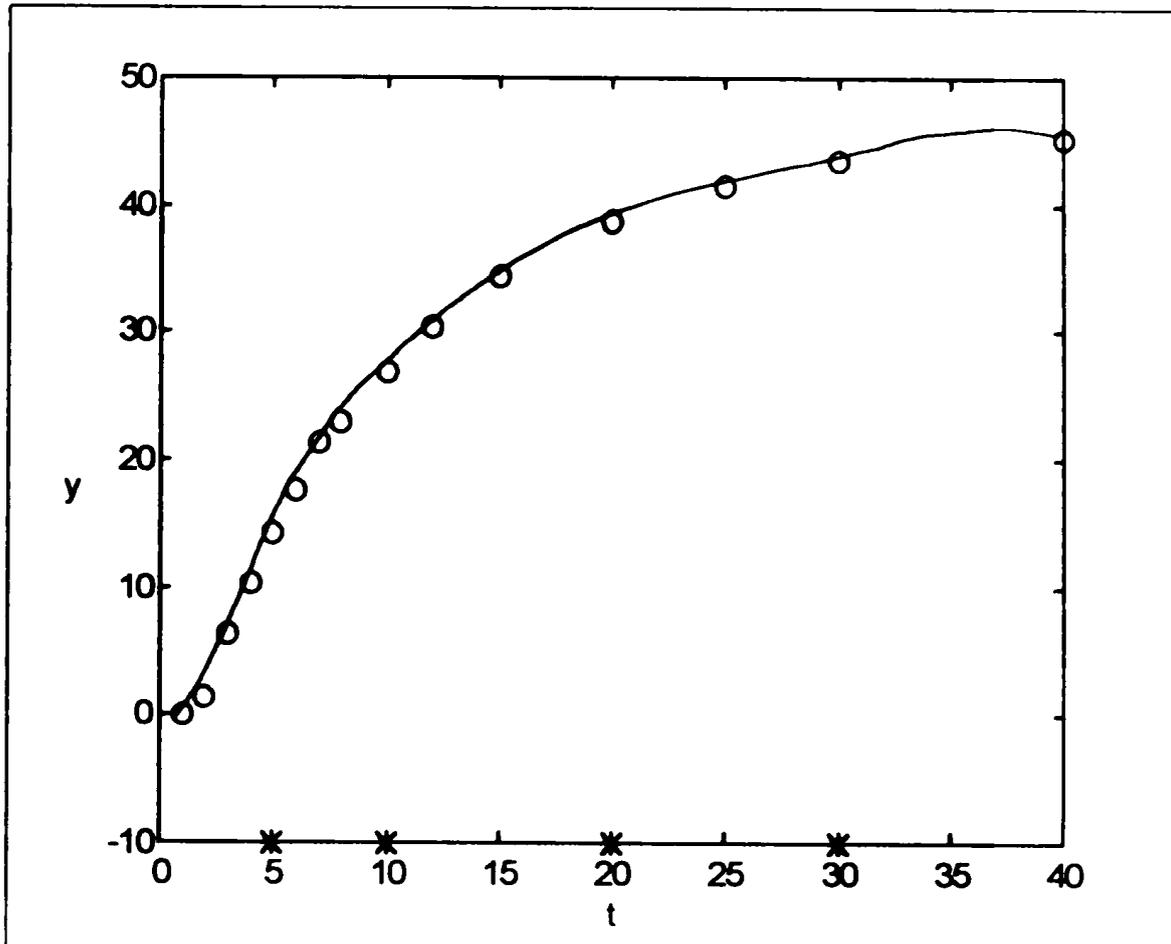


fig. 2-6: interpolation par des B-splines en utilisant 4 noeuds (positions: 5, 10, 15, 25)

L'examen des courbes obtenues pour les différentes combinaisons positions-nombre de noeuds montre l'efficacité de la méthode pour l'interpolation des données. On remarque, tout de même qu'on a une interpolation légèrement meilleure pour 1 et 2 noeuds que pour 4 noeuds

2°étape: Identification des paramètres p_1 et p_2 .

Les résultats de l'identification des paramètres p_1 et p_2 , et des conditions initiales $y(1)$ sont représentés dans le tableau 2-7. Les résultats obtenus avec 1 et 2 noeuds (positions 10-20) sont très proches, ils sont légèrement différents de ceux obtenus pour 2 noeuds (positions 5-15, surtout la condition initiale) et sensiblement différents des ceux relatifs à 4 noeuds.

Position des nœuds	p_1	p_2	$y(t)$
10	$0.463 \cdot 10^{-5}$	$0.2434 \cdot 10^{-3}$	-1.605
20	$0.462 \cdot 10^{-5}$	$0.2676 \cdot 10^{-3}$	-1.577
10,20	$0.4532 \cdot 10^{-5}$	$0.2536 \cdot 10^{-3}$	-1.327
5,15	$0.4138 \cdot 10^{-5}$	$0.2349 \cdot 10^{-3}$	-0.406
5, 10, 15,25	$0.3945 \cdot 10^{-5}$	$0.2118 \cdot 10^{-3}$	-0.226

tableau 2-7

L'intégration de l'équation différentielle, en utilisant les paramètres et les conditions initiales trouvés, permet l'obtention des courbes de la figure (2-7). Il apparaît clairement que le meilleur résultat correspond à l'utilisation d'un seul nœud (position 10), suivi de très près par 2 noeuds (positions 10-20).

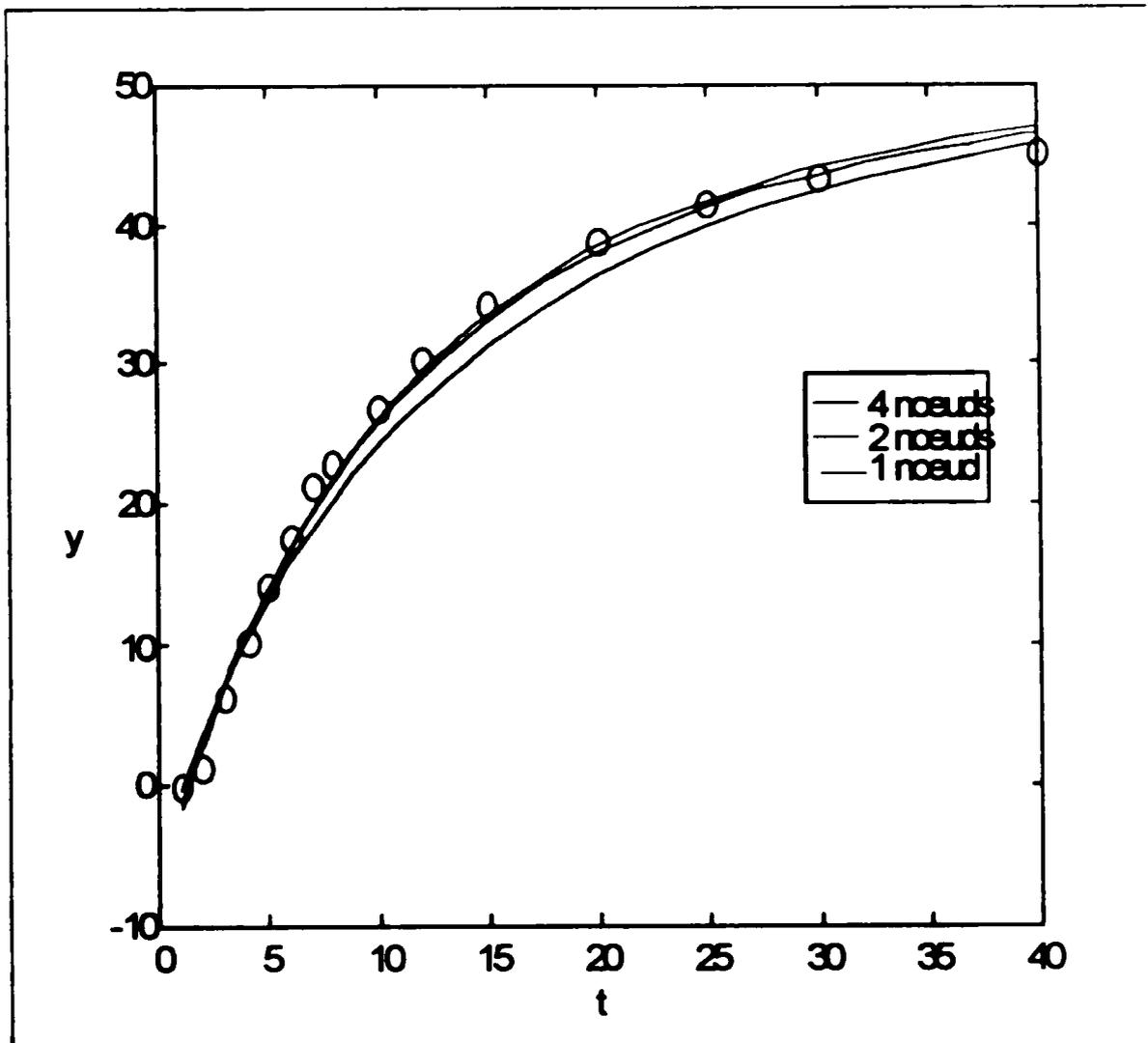


fig. 2-7: Les courbes obtenues après intégration comparées aux données

Problème n°2

On considère le système modélisé par les équations différentielles suivantes:

$$\begin{aligned} \dot{y}_1 &= p_1 y_1 - p_2 y_1 y_2 \\ \dot{y}_2 &= p_2 y_1 y_2 - p_3 y_2 \end{aligned}$$

où $p_1, p_2,$ et p_3 sont les paramètres que l'on veut identifier à partir des données du tableau 2-8. Les conditions initiales $y_1(0)$ et $y_2(0)$ ne sont pas précisées.

Ces équations représentent un système couplé avec des données fortement bruitées. Le problème est donc moins simple que le précédent.

t	0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5
y1	1	1.1	1.3	1.1	0.9	0.7	0.5	0.6	0.7	0.8	1
y2	0.3	0.35	0.4	0.5	0.5	0.4	0.3	0.25	0.25	0.3	0.35

Tableau 2-8

1°étape: détermination de la courbe qui approxime le mieux les données

Comme pour le problème précédent, on calcule d'abord les coefficients $a_i (i = k-m, \dots, k)$ pour différentes combinaisons positions-nombre de nœuds. On a essayé 1, 2 et 4 nœuds, avec des positions différentes. Pour chaque cas, on a calculé les coefficients. Les tableaux (2-9 à 2-13) représentent les résultats obtenus pour chaque cas.

Ces coefficients sont ensuite utilisés pour calculer les ordonnées d'un plus grand nombre de points appartenant à l'intervalle d'observation [0-5], pour obtenir une allure lisse des courbes interpolant les données. Des telles courbes sont nécessaires pour pouvoir apprécier la qualité de l'approximation et faire un choix.

A titre d'indication, chacune des courbes représentées sur les figures (2-8 à 2-12), a été tracée à partir de seulement 20 points différents.

Nombre de noeud: 1

position: 1.5

	a_3	a_2	a_1	a_0	a_1
y1(t)	0.9660	1.4130	0.8006	0.2518	1.0391
y2(t)	0.2995	0.3188	0.7662	-0.0171	0.3750

Tableau 2-9

Nombre de noeud: 1

position: 3

	a_3	a_2	a_1	a_0	a_1
y1(t)	0.9611	1.726 0	0.0005	0.7979	0.9909
y2(t)	0.2720	0.562 0	0.4013	0.1591	0.3660

Tableau 2-10

Nombre de noeuds: 2

positions: 1.5, 3

	a_3	a_2	a_1	a_0	a_1	a_2
y1(t)	0.9830	1.2404	1.3186	0.1229	0.8691	0.9815
y2(t)	0.3047	0.2659	0.6707	0.1687	0.2558	0.3533

Tableau 2-11

Nombre de noeuds: 2

positions: 1, 4

	a_3	a_2	a_1	a_0	a_1	a_2
y1(t)	0.9908	1.1734	1.3890	-0.0438	1.0223	0.9918
y2(t)	0.3032	0.2381	0.7967	0.0329	0.3435	0.3502

Tableau 2-12

Nombre de noeuds: 4

positions: 1, 2, 3, 4

	a_3	a_2	a_1	a_0	a_1	a_2	a_3	a_4
y1(t)	0.9980	0.9920	1.476	0.8749	0.4269	0.7252	0.8076	0.9997
y2(t)	0.300	0.3298	0.3875	0.6053	0.2489	0.2226	0.3238	0.3501

Tableau 2-13

L'examen des courbes de figures (2-8 à 2-12), fait ressortir que les meilleures approximations sont obtenues pour:

- 4 noeuds
- 1 noeud : position 3
- 2 noeuds: positions (1.5, 3)

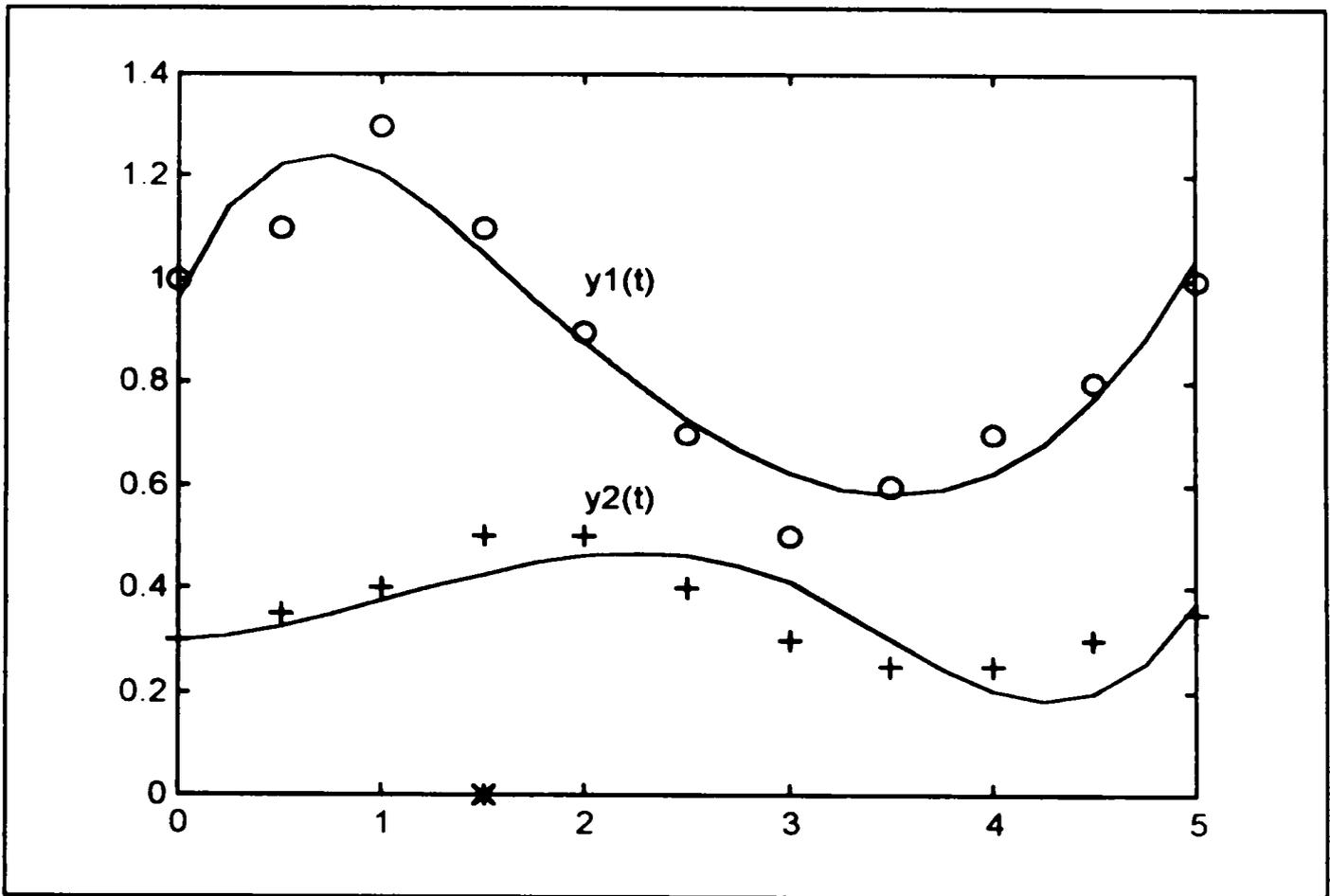


fig. 2-8: interpolation par des B-splines en utilisant 1 noeud (position: 1.5)

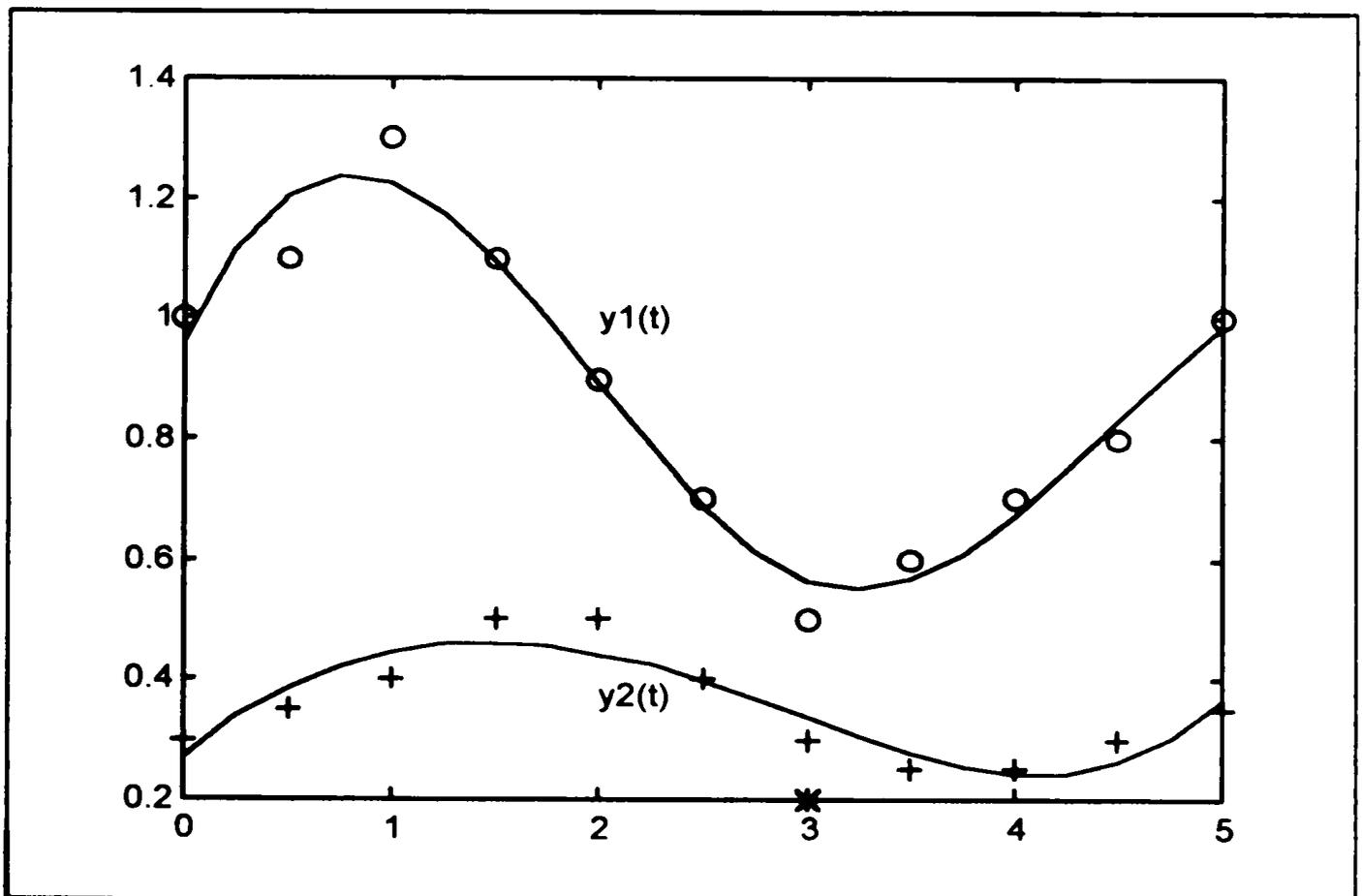


fig. 2-9: interpolation par des B-splines en utilisant 1 noeud (position: 3)

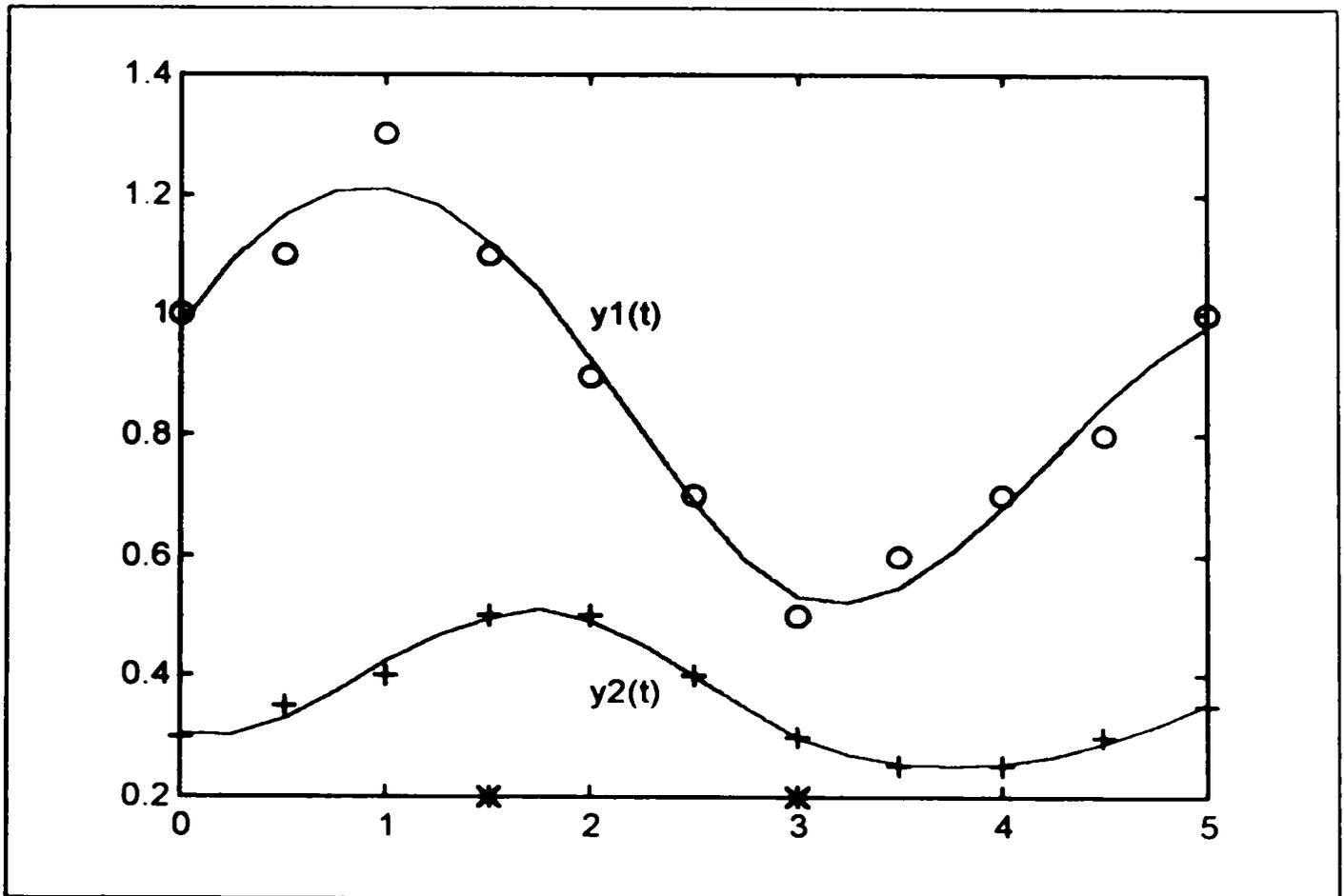


fig. 2-10: interpolation par des B-splines en utilisant 2 noeuds (positions: 1.5; 3)

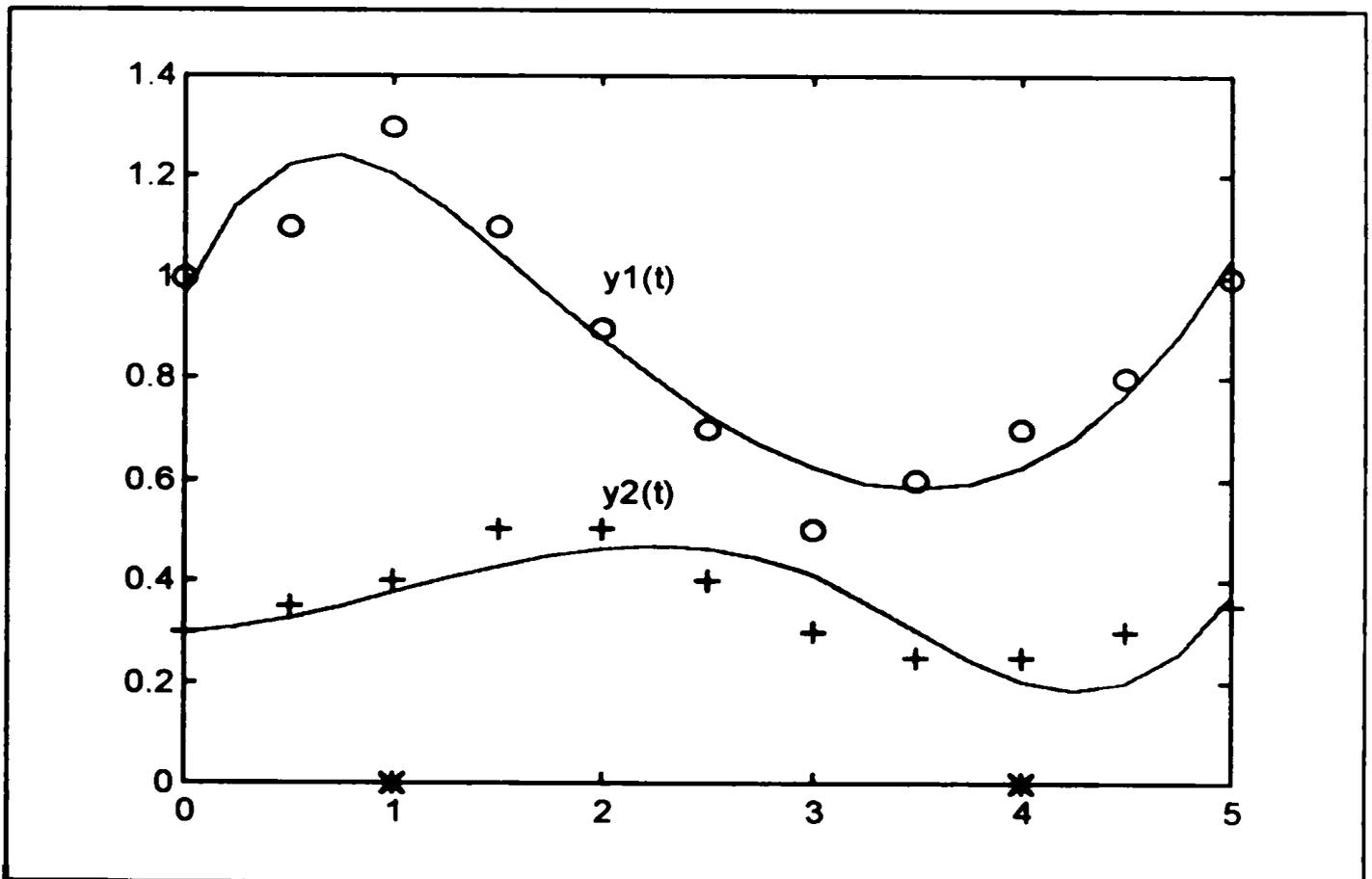


fig. 2-11: interpolation par des B-splines en utilisant 2 noeuds (positions: 1; 4)

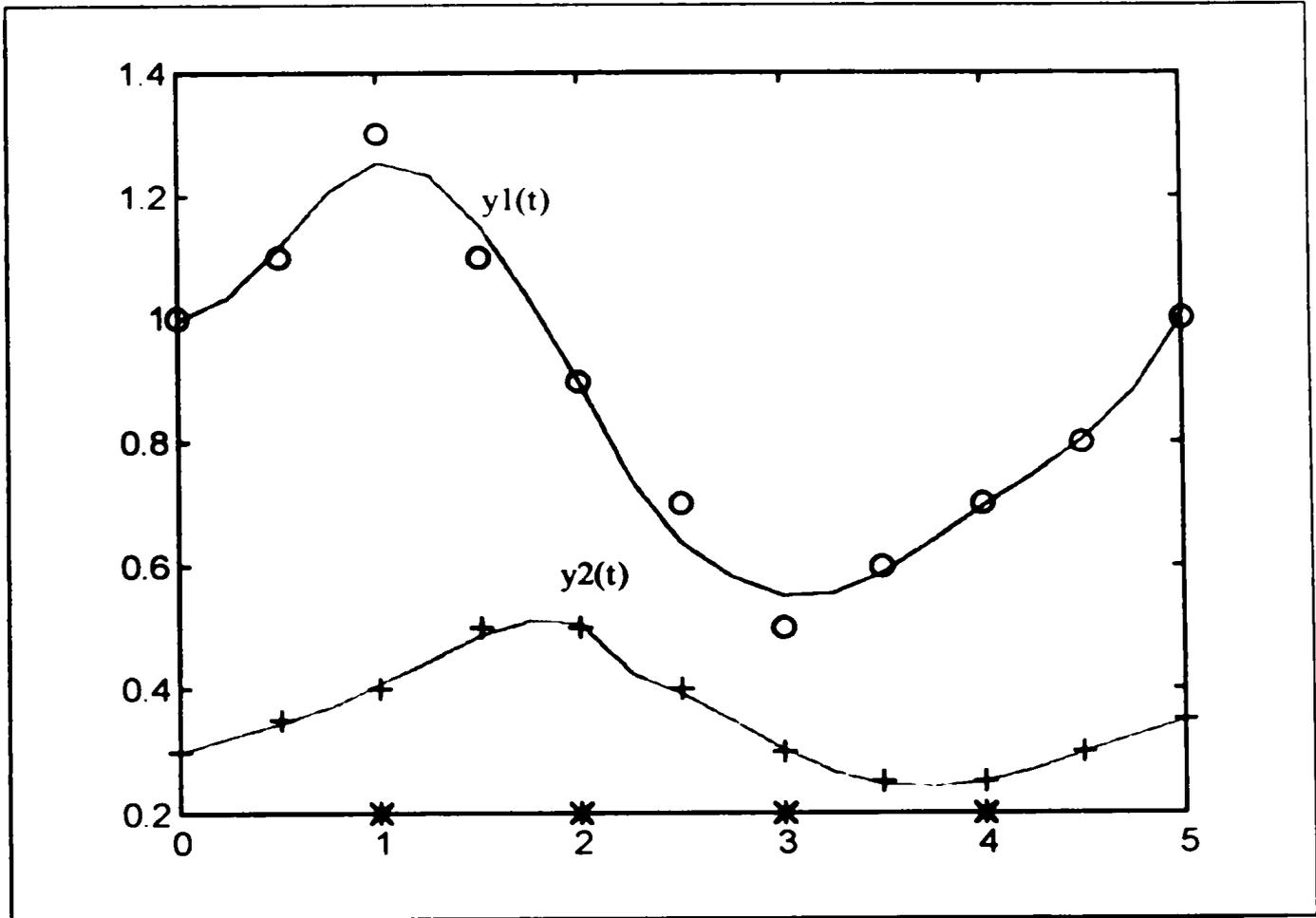


fig. 2-12: interpolation par des B-splines en utilisant 4 noeuds (positions: 1, 2, 3, 4)

2^oétape: Identification des paramètres $p_1, p_2,$ et p_3

On identifie les paramètres $p_1, p_2,$ et p_3 ainsi que les conditions initiales $y_1(0)$ et $y_2(0)$, pour chacun des 3 meilleurs cas d'approximation.

Les résultats relatifs aux différentes positions des noeuds sont représentés dans le tableau suivant:

position des noeuds	p_1	p_2	p_3	$y_1(0)$	$y_2(0)$	résidu
3	0.8161	2.2607	2.0625	1.058	0.2482	0.1742
1, 5, 3	0.8435	2.1981	2.0578	1.046	0.238	0.1690
1, 2, 3, 4	0.8492	2.2059	2.0236	1.053	0.246	0.1807

tableau 2-14

On remarque qu'il existe de légères différences entre les valeurs numériques des paramètres, surtout pour les paramètres p_1 et p_2 .

L'intégration des équations, en utilisant ces paramètres et ces conditions initiales, nous permet d'obtenir les courbes des figures (2-13) et (2-14)

Contrairement à l'exemple précédent, on ne peut pas dire quelle est la position des nœuds qui a donné le meilleur résultat, en se basant simplement sur l'appréciation visuelle de ces courbes. Pour cela, on a eu recours au calcul du résidu entre les données du tableau (2-8) et des points des courbes obtenues après intégration. Ce dernier est minimum pour 2 nœuds, suivi de 1 nœud puis de 4 nœuds, comme indiqué dans le tableau (2-14),

Toutefois, on peut remarquer de visu, que pour l'approximation de $y1(t)$, l'écart est plus petit pour 2 et 4 nœuds, alors que pour $y2(t)$ l'écart est plus petit pour 1 nœud

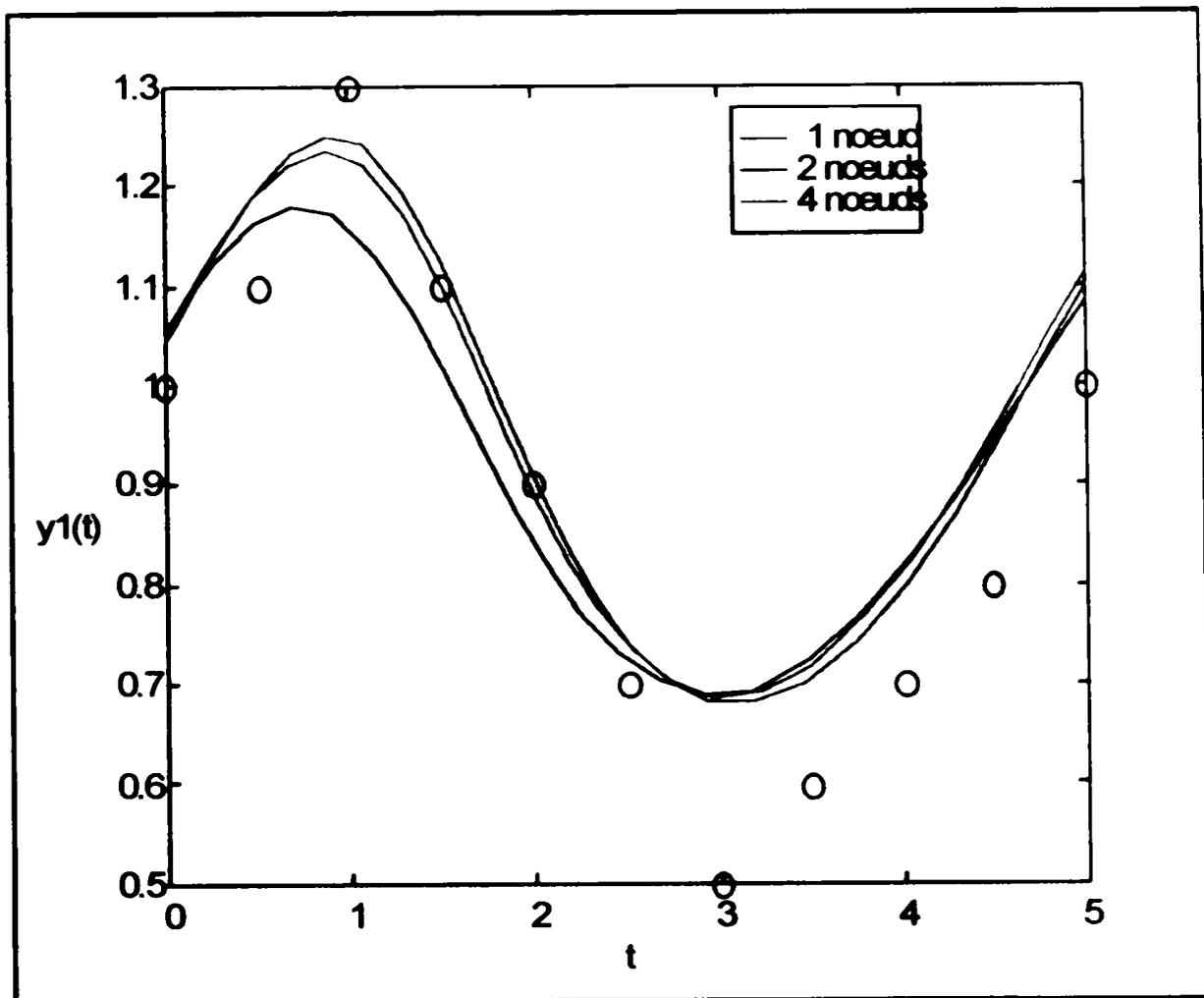


fig. 2-13 : $y1(t)$ obtenue après intégration, en comparaison avec les données

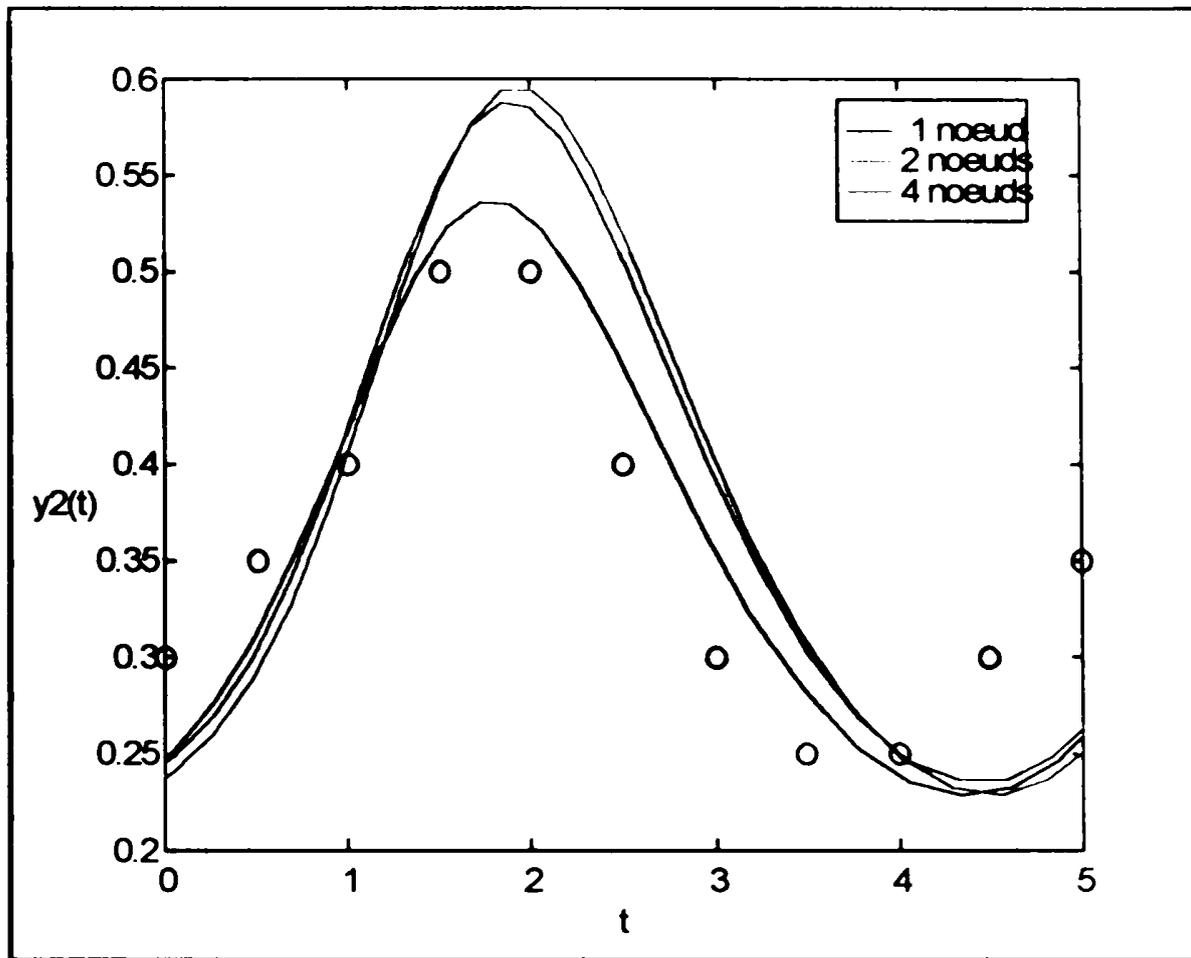


fig. 2-14 : $y_2(t)$ obtenue après intégration, en comparaison avec les données

L'identification des paramètres en utilisant les fonctions B-splines, est une technique qui utilise au préalable, une procédure d'interpolation des données dont dépend essentiellement la qualité du résultat.

Le choix de la courbe qui approxime le mieux les données est donc une étape très importante qui est quelquefois longue et nécessite plusieurs essais. Ceci se produit surtout, lorsque les données sont fortement bruitées.

Malgré cet inconvénient, l'identification des paramètres en utilisant les fonctions B-splines, reste attrayante et souvent utilisée dans les problèmes à données incomplètes.

CHAPITRE III

OPTIMISATION DES SYSTEMES PAR LA METHODE DE CONTINUATION

I- Introduction

Soit à déterminer, l'ensemble des solutions du système d'équations non linéaires de la forme

$$F(x, \lambda) = 0 \quad (3-1)$$

avec $x \in \mathbb{R}^n$ et $\lambda \in \mathbb{R}^m$

A partir d'un point initial (x_0, λ_0) , solution de (2-1).

Le développement de Taylor au 1^o ordre, de $F(x, \lambda)$ au voisinage de (x_0, λ_0) , donne:

$$F(x_0 + \Delta x, \lambda_0 + \Delta \lambda) = F(x_0, \lambda_0) + F_x(x_0, \lambda_0)\Delta x + F_\lambda(x_0, \lambda_0)\Delta \lambda$$

Si (x_0, λ_0) , est la dernière solution de (2-1) trouvée, et on veut également avoir:

$$F(x_0 + \Delta x, \lambda_0 + \Delta \lambda) = 0$$

alors :

$$F_x(x_0, \lambda_0)\Delta x + F_\lambda(x_0, \lambda_0)\Delta \lambda = 0$$

Ce qui implique:

$$\frac{\Delta x}{\Delta \lambda} = - [F_x(x_0, \lambda_0)]^{-1} \cdot F_\lambda(x_0, \lambda_0) \quad (3-2)$$

Pour construire ces solutions, il existe plusieurs méthodes, qui peuvent être classées en deux groupes:

1- Les méthodes qui utilisent le théorème des fonctions implicites, avec une forme de la méthode de Newton. Ce théorème assure l'existence d'une courbe unique $(x(\lambda), \lambda)$, au voisinage de (x_0, λ_0) , si la matrice jacobienne $F_x(x_0, \lambda_0)$ est régulière.

2- Les méthodes qui utilisent l'identité:

$$\frac{dx}{d\lambda} = - [F_x(x_0, \lambda_0)]^{-1} \cdot F_\lambda(x_0, \lambda_0) \quad (3-3)$$

avec la condition initiale $x(\lambda_0) = x_0$

pour former un système d'équations différentielles, que l'on résout par un des algorithmes connus

Toutefois, quelque soit le groupe auquel elles appartiennent, toutes ces méthodes sont performantes seulement si la matrice $F_x(x_0, \lambda_0)$ est régulière. A l'approche d'un point singulier (points de branchement, retournement ou bifurcation), ces méthodes deviennent inapplicables. A cet effet, des techniques particulières utilisant le principe de la continuation sont de plus en plus utilisées[3-1][3-4][3-11].

Ces méthodes indirectes fonctionnent suivant les étapes suivantes:

- Continuation d'une branche de solutions, en utilisant un procédé de *Prédiction-Correction*.
- A chaque point trouvé, tester si la matrice jacobienne est définie positive.

- Si la matrice perd sa régularité et devient singulière (ou l'inverse), ceci prouve que l'on se trouve au voisinage d'un point singulier que l'on peut déterminer avec exactitude en utilisant des méthodes telles que la bisection ou la sécante

II- Le principe de la Continuation

Afin de poursuivre le processus de détection des solutions, aux points singuliers et localiser ces points, on utilise une expression paramétrique de la courbe. Le problème (3-1) s'exprime alors, sous la forme de

Trouver $(x(s), \lambda(s))$ satisfaisant

$$F(x(s), \lambda(s)) = 0 \quad (3-4)$$

où s est un paramètre quelconque.

Partant d'un point initial $(x_0, \lambda_0) = (x(s_0), \lambda(s_0))$, satisfaisant $F(x(s), \lambda(s)) = 0$, les méthodes de continuation permettent de déterminer une branche entière de solutions $(x(s), \lambda(s))$, vérifiant (3-4), en suivant une courbe continue sur la surface définie par (3-1).

La condition nécessaire pour l'existence de cette branche de solutions à travers (x_0, λ_0) est obtenue par différentiation de (3-4) au point $s = s_0$.

$$F_x(x_0, \lambda_0) \dot{x} + F_\lambda(x_0, \lambda_0) \dot{\lambda} = 0 \quad (3-5)$$

avec

$$\dot{x} = \left. \frac{dx}{ds} \right|_{s=s_0} \quad \text{et} \quad \dot{\lambda} = \left. \frac{d\lambda}{ds} \right|_{\lambda=\lambda_0}$$

Le calcul de chaque solution (x_{k+1}, λ_{k+1}) à partir du point précédent (x_k, λ_k) , se fait en deux étapes:

- une première étape de prédiction $x_k, \lambda_k \rightarrow \hat{x}_{k+1}, \hat{\lambda}_{k+1}$
- une deuxième étape de correction $\hat{x}_{k+1}, \hat{\lambda}_{k+1} \rightarrow x_{k+1}, \lambda_{k+1}$

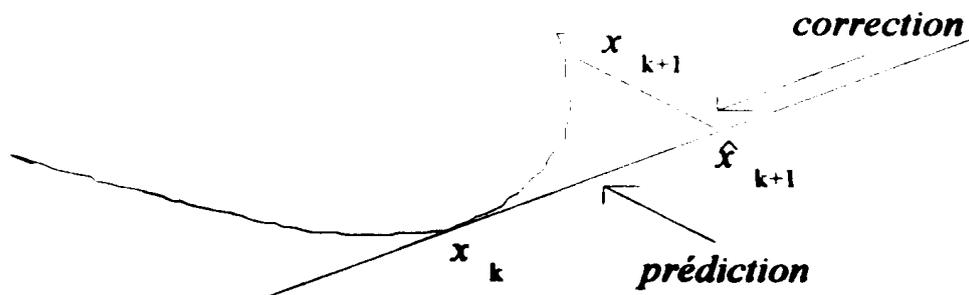


fig. 3-1: Etapes de prédiction et de correction de x , dans le procédé de continuation

Il est évident qu'à partir d'un seul point, on peut parcourir la courbe dans les deux sens, il suffit pour cela de définir le sens de direction IDIR en le mettant à ± 1 .

II-1 Méthodes de prédiction

a/ Prédiction d'Euler

La prédiction d'un point $(\hat{x}_{k+1}, \hat{\lambda}_{k+1})$ est obtenue en faisant:

$$\begin{cases} \hat{x}_{k+1} = x_k + IDIR \cdot \Delta s \cdot \dot{x}_k \\ \hat{\lambda}_{k+1} = \lambda_k + IDIR \cdot \Delta s \cdot \dot{\lambda}_k \end{cases} \quad k = 0, 1, \dots \quad (3.6)$$

Cette méthode, très couramment utilisée, nécessite la connaissance à priori du point précédent (x_k, λ_k) , des directions en ce point $(\dot{x}_k, \dot{\lambda}_k)$, du pas Δs et du sens de direction IDIR. Cette méthode peut être autodémarrée

Le pas Δs peut être pris variable ou constant. Son choix est important car de lui dépend le résultat de l'étape de correction. Les différentes procédures de calcul du pas seront exposées par la suite

b/ Prédiction par extrapolation du polynôme d'Hermite

La prédiction de (x_{k+1}, λ_{k+1}) par cette méthode nécessite, la connaissance à priori de: (x_{k-1}, λ_{k-1}) , $(\dot{x}_{k-1}, \dot{\lambda}_{k-1})$, (x_k, λ_k) et $(\dot{x}_k, \dot{\lambda}_k)$

Le polynôme d'interpolation d'Hermite s'écrit:

$$\begin{aligned} \hat{x}_{k+1}(s) &= x_{k-1}(s)p_{1,0}(s) + \dot{x}_{k-1}(s)p_{1,1}(s) + x_k(s)p_{2,0}(s) + \dot{x}_k(s)p_{2,1}(s) \\ &= x_{k-1}(s)p_{1,0}(\Delta s^{k-1}, \Delta s) + \dot{x}_{k-1}(s)p_{1,1}(\Delta s^{k-1}, \Delta s) + x_k(s)p_{2,0}(\Delta s^{k-1}, \Delta s) + \dot{x}_k(s)p_{2,1}(\Delta s^{k-1}, \Delta s) \end{aligned} \quad (3-7)$$

avec

$$\Delta s = s - s^k; \quad \Delta s^{k-1} = s^k - s^{k-1}$$

et où les polynômes de base $p_{i,j}(s)$ ($i=1, 2$ $j=0, 1$), sont définis par:

$$\begin{array}{cccc} p_{1,0}(s^{k-1}) = 1 & p_{1,0}(s^k) = 0 & \dot{p}_{1,0}(s^{k-1}) = 0 & \dot{p}_{1,0}(s^k) = 0 \\ p_{1,1}(s^{k-1}) = 0 & p_{1,1}(s^k) = 0 & \dot{p}_{1,1}(s^{k-1}) = 1 & \dot{p}_{1,1}(s^k) = 0 \\ p_{2,0}(s^{k-1}) = 0 & p_{2,0}(s^k) = 1 & \dot{p}_{2,0}(s^{k-1}) = 0 & \dot{p}_{2,0}(s^k) = 0 \\ p_{2,1}(s^{k-1}) = 0 & p_{2,1}(s^k) = 0 & \dot{p}_{2,1}(s^{k-1}) = 0 & \dot{p}_{2,1}(s^k) = 1 \end{array}$$

Cette méthode est certes plus précise que la méthode d'Euler, car le point prédit dépend du pas et des dérivées des deux points précédents, mais elle est plus longue et nécessite la connaissance à priori d'un nombre plus important d'éléments, pour fonctionner.

Si au départ, on ne connaît qu'un seul point; on peut calculer le point suivant en utilisant la méthode de prédiction d'Euler ensuite, on applique la méthode décrite dans ce paragraphe.

Comme pour la méthode de prédiction d'Euler, le pas Δs peut être pris constant ou variable.

II-2 : Methode de correction

La correction consiste à ramener le point prédit sur la courbe. La méthode la plus couramment utilisée est la méthode itérative de Newton-Raphson.

Le point prédit $(\hat{x}_{k+1}, \hat{\lambda}_{k+1})$ est utilisé comme point de départ pour les itérations de Newton.

Le point à l'itération ν est obtenu en faisant:

$$\begin{cases} x_{k+1}^{\nu+1} = x_{k+1}^{\nu} + \Delta x_{k+1}^{\nu} \\ \lambda_{k+1}^{\nu+1} = \lambda_{k+1}^{\nu} + \Delta \lambda_{k+1}^{\nu} \end{cases} \quad \nu = 0, 1, \dots \quad (3.8)$$

avec $x_{k+1}^0 = \hat{x}_{k+1}$ et $\lambda_{k+1}^0 = \hat{\lambda}_{k+1}$

et $\Delta x_{k+1}^{\nu}, \Delta \lambda_{k+1}^{\nu}$ obtenus après la résolution de :

$$F_x(x_{k+1}^{\nu}, \lambda_{k+1}^{\nu}) \Delta x_{k+1}^{\nu} + F_{\lambda}(x_{k+1}^{\nu}, \lambda_{k+1}^{\nu}) \Delta \lambda_{k+1}^{\nu} = -F(x_{k+1}^{\nu}, \lambda_{k+1}^{\nu}) \quad (3-9)$$

Cette méthode converge à condition que le point prédit soit suffisamment proche de la courbe. S'il ne l'est pas, l'algorithme peut diverger ou converger vers un point trop loin, ce qui peut "faire rater" des points importants de la courbe.

Dans ce cas, on redémarre l'algorithme avec un pas divisé par 2.

III- Les méthodes de Continuation

III-1 : Continuation par le paramètre λ

Le paramètre s est pris dans ce cas, égale à λ . On résout alors:

$$F(x(\lambda), \lambda) = 0 \quad (3-10)$$

et la différentiation de cette équation par rapport à λ , donne:

$$F_x(x(\lambda), \lambda) \dot{x} + F_{\lambda}(x(\lambda), \lambda) = 0 \quad (3-11)$$

avec $\dot{x} = \frac{dx}{d\lambda}$ et $\dot{\lambda} = 1$

Ce vecteur coïncide avec la direction; il est obtenu en résolvant:

$$F_x(x(\lambda), \lambda) \dot{x} = -F_{\lambda}(x(\lambda), \lambda) \quad (3-12)$$

Partant d'un point (x_k, λ_k) , et si on utilise la méthode d'Euler pour la prédiction du point suivant, on obtient:

$$\hat{x}_{k+1} = x_k + IDIR. \Delta s. \dot{x}_k \quad k = 0, 1, \dots \quad (3-13)$$

qui sera utilisée comme valeur initiale pour les iterations de Newton ($v=0$)

$$\begin{cases} F_x(x_{k+1}^v, \lambda_{k+1}^v) \Delta x_{k+1}^v = -F(x_{k+1}^v, \lambda_{k+1}^v) \\ x_{k+1}^{v+1} = x_{k+1}^v + \Delta x_{k+1}^v \end{cases} \quad v = 0, 1, \dots \quad (3-14)$$

Sachant que

$$\lambda_{k+1} = \lambda_k + \Delta \lambda \quad (3-15)$$

Cette méthode ne peut pas fonctionner pour des points de retournements quadratiques, sachant que pour ces derniers on doit avoir $\ddot{\lambda} \neq 0$ alors que dans le cas de la continuation par le paramètre λ , on a $\ddot{\lambda} = 0$.

III-2 : Continuation par la norme

Le paramètre s est choisi égale à la norme de x .

$$s = \|x\|_2 = \sqrt{x^t x}$$

et on résout:

$$\begin{cases} F(x(s), \lambda(s)) = 0 \\ x'(s)x(s) - s^2 = 0 \end{cases} \quad (3-16)$$

par différentiation de (3-16), on obtient:

$$\begin{cases} F_x(x(s), \lambda(s)) \dot{x} + F_\lambda(x(s), \lambda(s)) \dot{\lambda} = 0 \\ 2x'(s)\dot{x} - 2s = 0 \end{cases} \quad (3-17)$$

$$\dot{x} = \frac{dx}{ds} \quad \text{et} \quad \dot{\lambda} = \frac{d\lambda}{ds}$$

Si on utilise successivement, les méthodes d'Euler et de Newton, pour la prédiction et la correction du point (x_k, λ_k) .

$$\text{Prédiction:} \quad \begin{cases} \hat{x}_{k+1} = x_k + IDIR. \Delta s. \dot{x}_k \\ \hat{\lambda}_{k+1} = \lambda_k + IDIR. \Delta s. \dot{\lambda}_k \end{cases} \quad k = 0, 1, \dots \quad (3-18)$$

$$\text{Correction} \quad \begin{pmatrix} F_x(x_{k+1}^v, \lambda_{k+1}^v) & F_\lambda(x_{k+1}^v, \lambda_{k+1}^v) \\ (2x_{k+1}^v)' & 0 \end{pmatrix} \begin{pmatrix} \Delta x_{k+1}^v \\ \Delta \lambda_{k+1}^v \end{pmatrix} = - \begin{pmatrix} F(x_{k+1}^v, \lambda_{k+1}^v) \\ (x_{k+1}^v)'(x_{k+1}^v) - s^2 \end{pmatrix} \quad (3-19)$$

suivi de

$$\begin{cases} x_{k+1}^v = x_{k+1}^v + \Delta x_{k+1}^v \\ \lambda_{k+1}^v = \lambda_{k+1}^v + \Delta \lambda_{k+1}^v \end{cases} \quad v = 0, 1, \dots \quad (3.20)$$

Les nouvelles directions $(\dot{x}_{k+1}, \dot{\lambda}_{k+1})$ sont obtenues en résolvant:

$$\begin{pmatrix} F_x(x_{k+1}, \lambda_{k+1}) & F_\lambda(x_{k+1}, \lambda_{k+1}) \\ (2x_{k+1}^t) & 0 \end{pmatrix} \begin{pmatrix} \Delta \dot{x}_{k+1} \\ \Delta \dot{\lambda}_{k+1} \end{pmatrix} = - \begin{pmatrix} 0 \\ 2s \end{pmatrix} \quad (3.21)$$

avec une équation de normalisation:

$$(\dot{x}_{k+1})^t (\dot{x}_{k+1}) + \dot{\lambda}^2 = 1 \quad (3.22)$$

III-3 : Continuation par la pseudo-longueur d'arc

a/ Présentation de la méthode

C'est la méthode de continuation, la plus connue et la plus utilisée [3-2][3-3][3-5]. Elle consiste à prendre le paramètre s égale à la longueur d'arc.

du point de vue géométrique, cette méthode permet de trouver une solution (x_{k+1}, λ_{k+1}) de $F(x, \lambda) = 0$, dans un hyperplan se trouvant à une distance Δs de la solution précédente (x_k, λ_k) , perpendiculairement à la direction $(\dot{x}_k, \dot{\lambda}_k)$

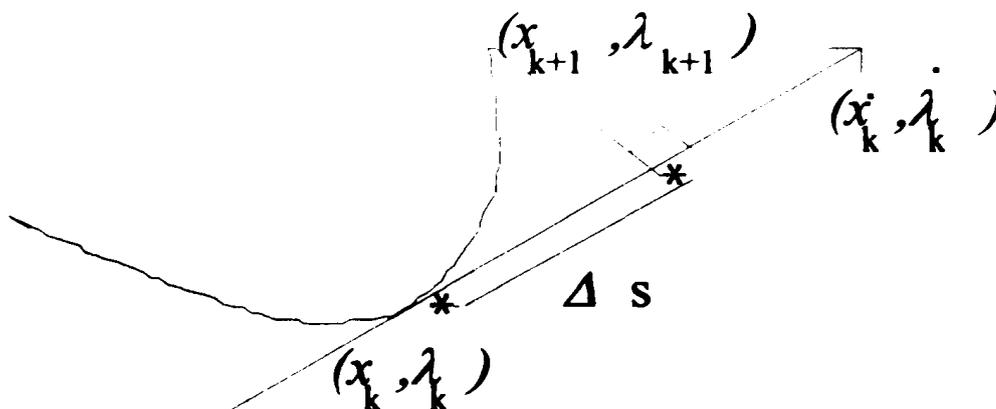


fig. 3-2: Continuation par la pseudo-longueur d'arc

Ce qui revient à résoudre pour un point (x_{k+1}, λ_{k+1}) :

$$\begin{cases} F(x_{k+1}^v, \lambda_{k+1}^v) = 0 \\ (x_{k+1}^v - x_k)' \dot{x}_k + (\lambda_{k+1}^v - \lambda_k) \dot{\lambda}_k - \Delta s = 0 \end{cases} \quad (3-23)$$

La méthode de Newton consiste alors à résoudre:

$$\begin{pmatrix} F_x(x_{k+1}^v, \lambda_{k+1}^v) & F_\lambda(x_{k+1}^v, \lambda_{k+1}^v) \\ \dot{x}_k' & \dot{\lambda}_k \end{pmatrix} \begin{pmatrix} \Delta x_{k+1}^v \\ \Delta \lambda_{k+1}^v \end{pmatrix} = - \begin{pmatrix} F(x_{k+1}^v, \lambda_{k+1}^v) \\ (x_{k+1}^v - x_k)' \dot{x}_k + (\lambda_{k+1}^v - \lambda_k) \dot{\lambda}_k - \Delta s \end{pmatrix} \quad (3-24)$$

avec les itérations:

$$\begin{cases} x_{k+1}^{v+1} = x_{k+1}^v + \Delta x_{k+1}^v \\ \lambda_{k+1}^{v+1} = \lambda_{k+1}^v + \Delta \lambda_{k+1}^v \end{cases} \quad v = 0, 1, \dots \quad (3-25)$$

Pour $v=0$, on prend les valeurs trouvées lors de l'étape de prédiction.

Après l'étape de correction, la nouvelle direction $(\dot{x}_{k+1}, \dot{\lambda}_{k+1})$ est calculée par différentiation de (3-23), suivie d'une normalisation:

$$- \begin{pmatrix} F_x(x_{k+1}^v, \lambda_{k+1}^v) & F_\lambda(x_{k+1}^v, \lambda_{k+1}^v) \\ \dot{x}_k' & \dot{\lambda}_k \end{pmatrix} \begin{pmatrix} \dot{x}_{k+1} \\ \dot{\lambda}_{k+1} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad (3-26)$$

$$- (\dot{x}_{k+1})' (\dot{x}_{k+1}) + \dot{\lambda}_{k+1}^2 = 1 \quad (3-27)$$

Remarque:

La matrice :

$$\tilde{A} = \begin{pmatrix} F_x & F_\lambda \\ \dot{x}' & \dot{\lambda} \end{pmatrix}$$

est régulière pour tous les points réguliers (F_x régulière) ainsi qu'au points de retournements (F_x singulière).

Pour tous ces points, l'espace nul de la matrice F_x est de dimension 1 ($\dim N(F_x) = 1$), c'est à dire il existe des vecteurs ϕ et ψ , tels que

$$\begin{cases} F_x \phi = 0 \\ F_x' \psi = 0 \end{cases} \quad (3-28)$$

La direction $(\dot{x}_k, \dot{\lambda}_k)$ en un point régulier appartenant à la branche de solutions, coïncide avec le vecteur nul ϕ de la matrice F_x .

Pour les points de bifurcations, la matrice \tilde{A} est singulière. L'espace nul est d'au moins dimension 2

Un point de bifurcation ne peut pas être pris comme point de départ pour la continuation d'une branche

b/ Méthode de résolution

Le système à résoudre par Newton est de la forme:

$$\begin{pmatrix} F_x(x_{k+1}^v, \lambda_{k+1}^v) & F_\lambda(x_{k+1}^v, \lambda_{k+1}^v) \\ \dot{x}_k & \dot{\lambda}_k \end{pmatrix} \begin{pmatrix} y \\ z \end{pmatrix} = - \begin{pmatrix} f \\ h \end{pmatrix} \quad (3-29)$$

$$y = \Delta x_{k+1}^v, \quad z = \Delta \lambda_{k+1}^v, \quad f = F(x_{k+1}^v, \lambda_{k+1}^v), \quad h = (x_{k+1}^v - x_k)' \dot{x}_k + (\lambda_{k+1}^v - \lambda_k) \dot{\lambda}_k - \Delta s \quad (3-30)$$

La méthode de Newton, est applicable à ce système si la matrice $\tilde{A} = \begin{pmatrix} F_x & F_\lambda \\ \dot{x}' & \dot{\lambda} \end{pmatrix}$ est régulière.

La décomposition de la matrice \tilde{A} en deux matrices telles que:

$$\tilde{A} = \begin{pmatrix} F_x & F_\lambda \\ \dot{x}' & \dot{\lambda} \end{pmatrix} = \begin{pmatrix} F_x & 0 \\ \dot{x}' & 1 \end{pmatrix} \begin{pmatrix} I & F_x^{-1} F_\lambda \\ 0 & \delta \end{pmatrix} \quad (3-31)$$

implique que \tilde{A} est non singulière si on a $\delta \neq 0$. En d'autres termes, Si F_x est régulière, alors \tilde{A} est non singulière si

$$\delta = \dot{\lambda} - \dot{x}' (F_x^{-1} F_\lambda) \neq 0 \quad (3-32)$$

Cas où F_x est régulière

La matrice F_x est décomposée en L.U, où L est une matrice triangulaire inférieure avec tous les éléments de la diagonale principale égaux à 1; et U une matrice triangulaire supérieure. Cette décomposition est obtenue grâce à l'algorithme de Gauss avec la stratégie du pivot total.

On calcule ensuite, β , γ , et δ définis par:

$$\gamma = L^{-1} F_\lambda \quad \beta = (U')^{-1} \dot{x} \quad \delta = \dot{\lambda} - \beta' \gamma$$

ce qui permet d'écrire:

$$\tilde{A} = \begin{pmatrix} F_x & F_\lambda \\ \dot{x}' & \dot{\lambda} \end{pmatrix} = \begin{pmatrix} L & 0 \\ \beta' & 1 \end{pmatrix} \begin{pmatrix} U & \gamma \\ 0 & \delta \end{pmatrix} \quad (3-33)$$

on pose:

$$\begin{pmatrix} U & \gamma \\ 0 & \delta \end{pmatrix} \begin{pmatrix} y \\ z \end{pmatrix} = \begin{pmatrix} \hat{f} \\ \hat{h} \end{pmatrix} \quad (3-34)$$

on résout d'abord:

$$\begin{pmatrix} L & 0 \\ \beta' & 1 \end{pmatrix} \begin{pmatrix} \hat{f} \\ \hat{h} \end{pmatrix} = \begin{pmatrix} f \\ h \end{pmatrix} \quad (3-35)$$

Ce qui donne:

$$\hat{f} = L^{-1} f \quad \text{et} \quad \hat{h} = h - \beta' \hat{f}$$

Ensuite, on résout (3-34), et on obtient:

$$z = \hat{h} \delta \quad \text{et} \quad y = U^{-1}(\hat{f} - \gamma z)$$

Cas où F_x est singulière

On considère le cas où F_x est singulière et \tilde{A} régulière.

La résolution du système (3-29) revient à résoudre successivement (3-36) et (3-37):

$$F_x y + F_\lambda z = -f \quad (3-36)$$

$$\dot{x}' y + \dot{\lambda} z = -h \quad (3-37)$$

La résolution de (3-36) nous permet d'obtenir z .

$$F_\lambda z = -(f + F_x y) \quad (3-38)$$

$$\psi' (F_\lambda z) = -(\psi' f + \psi' F_x y) \quad (3-39)$$

$$z = -\frac{\psi' f}{\psi' F_\lambda} \quad (3-40)$$

sachant qu'on a: $\psi' F_\lambda \neq 0$.

La solution y doit donc vérifiée:

$$F_x y = -(f + F_\lambda z) = -\left(f + \frac{\psi' f}{\psi' F_\lambda} F_\lambda\right) = \hat{f} \quad (3-41)$$

La matrice F_x étant singulière, son inverse n'existe pas. On pose donc, que la solution est de la forme

$$y = y_p + \alpha \phi \quad (3-42)$$

où y_p est une solution particulière.

***- Calcul de la solution particulière**

$$F_x y = \hat{f} = F_x y_p + \alpha F_x \phi = F_x y_p \quad (3-43)$$

La matrice F_x est décomposée en LU; Puisque c'est une matrice singulière on aura $\delta=0$.

La résolution de (3-42) revient donc à résoudre:

$$F_x y_p = \begin{pmatrix} L & 0 \\ \beta' & 1 \end{pmatrix} \begin{pmatrix} U & \gamma \\ 0 & 0 \end{pmatrix} y_p = \hat{f} \quad (3-44)$$

On résout d'abord:

$$\begin{pmatrix} L & 0 \\ \beta' & 1 \end{pmatrix} \begin{pmatrix} \tilde{f} \\ 0 \end{pmatrix} = \hat{f} \quad (3-45)$$

ensuite:

$$\begin{pmatrix} U & \gamma \\ 0 & 0 \end{pmatrix} y_p = \begin{pmatrix} \tilde{f} \\ 0 \end{pmatrix} \quad (3-46)$$

Si on pose $y_p = \begin{pmatrix} y_{1p} \\ y_{2p} \end{pmatrix}$; y_{2p} peut prendre n'importe quelle valeur, en particulier $y_{2p}=0$. Dans ce cas, la solution particulière est égale à:

$$y_p = U^{-1} \tilde{f} \quad (3-47)$$

***- Calcul de α**

Pour le calcul de α , on considère l'équation (3-37). On y remplaçant y , par son expression définie par (3-42), on obtient alors:

$$\dot{x}' (y_p + \alpha \phi) + \dot{\lambda} z = h \quad \Rightarrow \quad \alpha = - \frac{h + \dot{\lambda} \frac{\psi' f}{\psi' F_\lambda} + \dot{x}' y_p}{\dot{x}' \phi} \quad (3-48)$$

- Calcul des vecteurs ϕ et ψ .*1°/ calcul de ϕ**

Le calcul de ce vecteur revient à trouver la solution de:

$$F_x \phi = \begin{pmatrix} L & 0 \\ \beta' & 1 \end{pmatrix} \begin{pmatrix} U & \gamma \\ 0 & 0 \end{pmatrix} \phi = 0 \quad (3-49)$$

on pose $\phi = \begin{pmatrix} \mu \\ -1 \end{pmatrix}$, avec:

$$\mu = U^{-1} \gamma \quad (3-50)$$

2°/ calcul de ψ

Comme pour le calcul précédent; on pose $\psi = \begin{pmatrix} \omega \\ -1 \end{pmatrix}$, et on résout:

$$F_x' \psi = \begin{pmatrix} U' & 0 \\ \gamma' & 0 \end{pmatrix} \begin{pmatrix} L' & \beta \\ 0 & 1 \end{pmatrix} \psi = 0 \quad (3-51)$$

On obtient alors:

$$\omega = (L')^{-1} \beta \quad (3-52)$$

c/ Calcul de la direction

La méthode de continuation, nécessite le calcul, pour chaque point $u_k = (x_k, \lambda_k)$, de la direction $\dot{u}_k = (\dot{x}_k, \dot{\lambda}_k)$.

Pour un point régulier, on peut prendre cette direction, comme étant le vecteur nul de la matrice F_u (voir section suivante).

Une autre méthode consiste à prendre :

$$\dot{u}_k \approx \frac{1}{\Delta s} (u_k - u_{k-1})$$

Une troisième méthode de calcul, serait de résoudre:

$$F_u \dot{u}_k = 0 \quad (3-53)$$

Soient deux matrices de permutations P et Q non singulières, telles que:

$$PF_u Q = [I \quad -\Gamma] \quad (3-54)$$

on réécrit (3-53), sous la forme:

$$F_u Q Q^{-1} \dot{u}'_k = 0 \quad (3-55)$$

et on multiplie cette expression par P :

$$P F_u Q Q^{-1} \dot{u}'_k = 0 \quad (3-56)$$

on arrive alors à:

$$[I \quad -\Gamma] Q^{-1} \dot{u}'_k = 0 \quad (3-57)$$

d'où

$$\dot{u}'_k = Q \begin{pmatrix} -\Gamma \\ 1 \end{pmatrix} \quad (3-58)$$

Le vecteur obtenu est ensuite normalisé.

Pour suivre le tracé de la courbe, dans un intervalle où F_u est régulière, \dot{u}'_k doit garder un signe constant. Si on suppose que \dot{u}'_{k-1} est dans cet intervalle; on doit avoir:

$$\dot{u}'_k \dot{u}'_{k-1} > 0 \quad (3-59)$$

Dans le cas contraire, on a un point de retournement.

d/ Contrôle du pas de déplacement

Le choix du pas de déplacement a une importance capitale pour la méthode de continuation. Un pas trop petit entraîne une perte en temps de calcul; alors qu'un pas trop grand peut entraîner le saut d'une branche à une autre. Le pas peut être choisi fixe ou variable. plusieurs méthodes de calcul peuvent être proposées

*** 1° méthode de calcul du pas**

C'est la méthode la plus simple, elle consiste à attribuer une valeur de départ au pas Δs . Ensuite, procéder comme suit:

- Si la méthode de correction converge très rapidement; ce pas est incrémenté.
- Si la méthode de correction converge très lentement ou diverge; les calculs sont repris avec un pas plus petit (par exemple: divisé par 2).

Il est évident, que l'on doit toujours vérifier que la valeur prise par le pas, doit être inférieure à Δs_{max} et supérieur à Δs_{min} , définis au départ.

*** 2° méthode de calcul du pas**

Cette méthode consiste à calculer le pas par prédiction.

Soit le développement de Taylor d'ordre 2, autour du point $x_k(s)$:

$$x_{k+1}(s) = x_k(s) + \Delta s_k \dot{x}_k(s) + \frac{\Delta s_k^2}{2} \ddot{x}_k(s) + O(\Delta s_k^3) \quad (3-60)$$

Si on pose que:

$$\frac{\Delta s_k^2}{2} \ddot{x}_k(s) = C = \text{constante} \quad (3-61)$$

(C : constante positive, inférieure à 1).

alors la valeur prédite du pas peut être prise égale à:

$$\Delta \hat{s}_k \approx \sqrt{\frac{2C}{\|\ddot{x}_k(s)\|}} \quad (3-62)$$

Ce qui revient à dire, que la valeur prédite du pas est proportionnelle à la racine carrée du rayon de courbure. Ce qui réalise le maximum de déplacement, tout en minimisant l'éloignement par rapport à la courbe solution.

En fait, on prend:

$$\Delta \hat{s}_k = \min(\Delta \hat{s}_k, 2\Delta s_{k-1}, \Delta s_{max}) \quad (3-63)$$

Δs_{max} étant le déplacement maximale autorisé.

Lorsque la courbe est assez régulière, la valeur prédite du pas est une bonne approximation de Δs_k ; Par contre lorsque la variation de la courbe, au voisinage du point considéré, est importante, cette prédiction peut s'écarter de la valeur Δs_k , et dans ce cas là, il faut le calculer suivant la formule (3-66).

On pose:

$$x_{k+1}(s) = x_k(s) + \Delta s_k \dot{x}_k(s) + O(\Delta s_k^2) \quad (3-64)$$

$$x_k(s) = x_{k+1}(s) - \Delta s_k \dot{x}_{k+1}(s) + O(\Delta s_k^2) \quad (3-65)$$

En soustrayant (3-65) à (3-64), on obtient:

$$\Delta s_k \approx \frac{2\|x_{k+1}(s) - x_k(s)\|}{\|\dot{x}_{k+1}(s) + \dot{x}_k(s)\|} \quad (3-66)$$

III- Points de retournement**III-1 : Définition**

Un simple point de retournement est tel que:

$$\dim N(F_x)=1 \quad \text{et} \quad F_\lambda \notin R(F_x)$$

c'est à dire, il vérifie:

$$F_x \varphi = 0, \quad F_x' \psi = 0, \quad F_\lambda' \psi \neq 0$$

Par différentiation de:

$$F(x(s), \lambda(s)) = 0 \quad (3-67)$$

on obtient:

$$F_x(x(s), \lambda(s)) \dot{x}(s) + F_\lambda(x(s), \lambda(s)) \dot{\lambda}(s) = 0 \quad (3-68)$$

on multiplie (3-68) à gauche, par ψ'

$$\psi' F_x(x(s), \lambda(s)) \dot{x}(s) + \psi' F_\lambda(x(s), \lambda(s)) \dot{\lambda}(s) = 0 \quad (3-69)$$

Sachant que:

$$\psi' F_x = 0 \quad \text{et que} \quad \psi' F_\lambda \neq 0 \quad (3-70)$$

(3-69) se réduit à:

$$\psi' F_\lambda(x(s), \lambda(s)) \dot{\lambda}(s) = 0 \quad (3-71)$$

ce qui implique que :

$$\dot{\lambda}(s) = 0 \quad (3-72)$$

D'autre part, si on différentie (3-68), on arrive à:

$$F_x \ddot{x}(s) + F_{xx} \dot{x}(s) \dot{x}(s) + F_{x\lambda} \dot{\lambda}(s) \dot{\lambda}(s) + F_\lambda \ddot{\lambda}(s) + 2F_{x\lambda} \dot{x}(s) \dot{\lambda}(s) = 0 \quad (3-73)$$

En tenant compte de (3-72), l'expression (3-73) se simplifie, ce qui donne en la multipliant par ψ' :

$$\psi' F_x \ddot{x}(s) + \psi' F_{xx} \dot{x}(s) \dot{x}(s) + \psi' F_\lambda \ddot{\lambda}(s) = 0 \quad (3-74)$$

d'où, sachant (3-70) :

$$\ddot{\lambda}(s) = -\frac{\psi' F_{xx} \dot{x}(s)\dot{x}(s)}{\psi' F_{\lambda}} \quad (3-75)$$

Lorsqu'on a $\ddot{\lambda} \neq 0$, on dit qu'on a un point de retournement quadratique. Ce qui correspond à :

$$\psi' F_{xx} \dot{x}(s)\dot{x}(s) \neq 0 \quad (3-76)$$

Remarque:

Pour un point de retournement, on a $\dot{\lambda}(s) = 0$, d'où :

$$F_x(x(s), \lambda(s)) \dot{x}(s) = 0 \quad (3-78)$$

d'autre part on sait que :

$$\dim N(F_x) = 1 \text{ et que } F_x \phi = 0$$

On en déduit que :

$$\dot{x}(s) = \phi \quad (3-79)$$

Ce qui montre que la direction coïncide avec le vecteur nul de F_x .

III-2 : Stabilité de la solution

Un point de retournement est un point où la branche de solutions, change de direction et la matrice F_x est singulière. Ceci implique que nécessairement, au moins une des valeurs propres de F_x est nulle de positive cette dernière va devenir négative ou inversement.

On a alors un changement de stabilité: Si la branche de solutions étaient stable avant le point de retournement, elle va devenir instable après et inversement.

IV- Points de bifurcation

Un point de bifurcation correspond à l'intersection d'au moins deux branches de solutions, dont l'une est la branche actuelle.

Pour ces points la matrice \tilde{A} est singulière et l'espace nul de la matrice $F_u = [F_x \quad F_{\lambda}]$ est d'au moins dimension 2; c'est à dire qu'il existe ϕ_1 , ϕ_2 et ψ tels que :

$$\begin{cases} F_u \phi_1 = F_u \phi_2 = 0 \\ F_u' \psi = 0 \end{cases} \quad (3-80)$$

L'équation (3-81) est différenciée deux fois, le résultat est ensuite, multiplié à droite par ψ' , on obtient alors l'équation (3-82).

$$F(x(s), \lambda(s)) = F(u(s)) = 0 \quad (3-81)$$

$$\psi' F_{uu}(u(s)) \dot{u}(s) \dot{u}(s) = 0 \quad (3-82)$$

Pour un point de bifurcation, on a :

$$\dot{u}(s) = \alpha \phi_1 + \theta \phi_2 \quad (3-83)$$

En remplaçant $\dot{u}(s)$ par son expression, dans (3-82), on obtient l'équation (3-84), communément appelée Equation Algébrique de Bifurcation (ABE).

$$\psi' F_{uu}(u(s)) (\alpha \phi_1 + \theta \phi_2) (\alpha \phi_1 + \theta \phi_2) = 0$$

$$\psi' F_{uu}(u(s)) (\alpha^2 \phi_1 \phi_1 + 2\alpha \theta \phi_1 \phi_2 + \theta^2 \phi_2 \phi_2) = 0$$

$$\alpha^2 c_{11} + 2\alpha \theta c_{12} + \theta^2 c_{22} = 0 \quad (3-84)$$

avec

$$c_{11} = \psi' F_{uu}(u(s)) \phi_1 \phi_1; \quad c_{12} = \psi' F_{uu}(u(s)) \phi_1 \phi_2; \quad c_{22} = \psi' F_{uu}(u(s)) \phi_2 \phi_2;$$

Les paires de solutions (α, θ) doivent être différentes de $(0,0)$ et on suppose qu'on a $\alpha \neq 0$.

La relation (3-84) est un trinôme du second degré en θ/α ,

$$c_{11} + 2 \frac{\theta}{\alpha} c_{12} + \frac{\theta^2}{\alpha^2} c_{22} = 0$$

Le discriminant $\Delta = c_{12}^2 - c_{11} c_{22}$ est positive, ce qui correspond à deux solutions réelles dont l'une égale, à $(\alpha_1, \theta_1) = (1, 0)$ correspond à la direction de la branche actuelle \dot{u}_1 . La direction de la deuxième branche \dot{u}_2 , sera définie par (α_2, θ_2) .

$$\begin{cases} \dot{u}_1 = \alpha_1 \phi_1 + \theta_1 \phi_2 \\ \dot{u}_2 = \alpha_2 \phi_1 + \theta_2 \phi_2 \end{cases} \quad (3-85)$$

Puisque une des deux directions est \dot{u}_1 , alors:

$$\psi' F_{uu}(u(s)) \dot{u}(s) \dot{u}(s) = \psi' F_{uu}(u(s)) \phi_1 \phi_1 = 0 \quad (3-86)$$

ce qui implique $c_{11}=0$, d'où $\Delta = c_{12}^2$.

Les solutions sont donc égales à:

$$\begin{aligned}
 * \frac{\theta_1}{\alpha_1} &= \frac{-c_{12} + \sqrt{\Delta}}{c_{22}} = 0 \\
 * \frac{\theta_2}{\alpha_2} &= \frac{-c_{12} - \sqrt{\Delta}}{c_{22}} = -2 \frac{c_{12}}{c_{22}}
 \end{aligned} \tag{3-87}$$

La première solution correspond à $(\alpha_1, \theta_1) = (1, 0)$, alors que la deuxième solution nécessite le calcul des coefficients c_{12} et c_{22} .

En utilisant, la méthode des différences divisées, on peut approximer ces coefficients par l'expression (3-88).

$$F''_{uu} \phi_i \phi_j = \frac{F(u + eh(\phi_i + \phi_j)) - F(u + eh\phi_i) - F(u + eh\phi_j) + F(x)}{2eh^2} \tag{3-88}$$

Les vecteurs ϕ_1 et ψ , sont calculés suivant la procédure présentée précédemment. Pour le calcul de ϕ_2 , on choisit $\phi_2 \perp \phi_1$. Il va donc correspondre au vecteur nul de la matrice $\begin{bmatrix} F'_u \\ \phi'_1 \end{bmatrix}$.

Remarque: ϕ_2 est un vecteur nul de F'_u et $\begin{bmatrix} F'_u \\ \phi'_1 \end{bmatrix}$, par contre ϕ_1 est un vecteur nul de F'_u mais pas de $\begin{bmatrix} F'_u \\ \phi'_1 \end{bmatrix}$.

V- Application de l'algorithme de continuation par la pseudo-longueur d'arc pour la résolution de différents problèmes.

Problème 1: Identification de la courbe d'intersection, entre deux surfaces.

La courbe d'intersection entre, une surface $P(u,v)$ du plan $u-v$, et une surface $Q(s,t)$ du plan $s-t$, est définie par l'équation:

$$F(u,v,s,t) = P(u,v) - Q(s,t) = 0 \tag{3-89}$$

où:

$$\left. \begin{aligned} P(u, v) &= \{X_p(u, v), Y_p(u, v), Z_p(u, v)\} \\ Q(s, t) &= \{X_q(s, t), Y_q(s, t), Z_q(s, t)\} \end{aligned} \right\} \quad (3-90)$$

avec X , Y et Z représentent les coordonnées cartésiennes de chaque point.

Un point d'intersection est point qui appartient aux deux surfaces, par conséquent ses coordonnées cartésiennes vont satisfaire les équations:

$$\left. \begin{aligned} X_p(u, v) - X_q(s, t) &= 0 \\ Y_p(u, v) - Y_q(s, t) &= 0 \\ Z_p(u, v) - Z_q(s, t) &= 0 \end{aligned} \right\} \quad (3-91)$$

Ces équations définissent un système de trois équations à quatre inconnues; une équation additionnelle est nécessaire pour sa résolution. Pour cela, on utilise la méthode de continuation par la pseudo-longueur d'arc.

Si pose $W = \{u, v, s\}$

L'équation à rajouter est:

$$(W_k - W_{k-1})W_{k-1} + (t_k - t_{k-1})t_{k-1} - \Delta s^2 = 0 \quad (3-92)$$

L'identification de la courbe d'intersection, va donc consister à continuer la courbe de solutions du système complet:

$$\left. \begin{aligned} F(W_k, t_k) &= 0 \quad k = 1, \dots \\ (W_k - W_{k-1})W_{k-1} &+ (t_k - t_{k-1})t_{k-1} - \Delta s^2 = 0 \end{aligned} \right\} \quad (3-93)$$

A titre d'exemple, on considère le cas d'une intersection entre deux cylindres $C1(u, v)$ et $C2(s, t)$, pour lesquels les coordonnées cartésiennes sont définies respectivement par:

$$\begin{aligned} X_{c1}(u, v) &= 1 - u \\ Y_{c1}(u, v) &= 2Rv / (1 + v^2) \\ Z_{c1}(u, v) &= 2Rv^2 / (1 + v^2) \end{aligned}$$

et
(3-94)

$$\begin{aligned} X_{c2}(s, t) &= (1 - s^2) / (1 + s^2) \\ Y_{c2}(s, t) &= 2s / (1 + s^2) \\ Z_{c2}(s, t) &= t \end{aligned}$$

avec R le rayon du cylindre.

L'algorithme est testé pour différentes valeurs de R ; et pour chaque cas, les points de départ de la branche sont calculés de telle manière à ce qu'ils correspondent aux valeurs limites de l'espace considéré

Pour savoir si toute la branche de solutions a été déterminée, on fait deux tests.

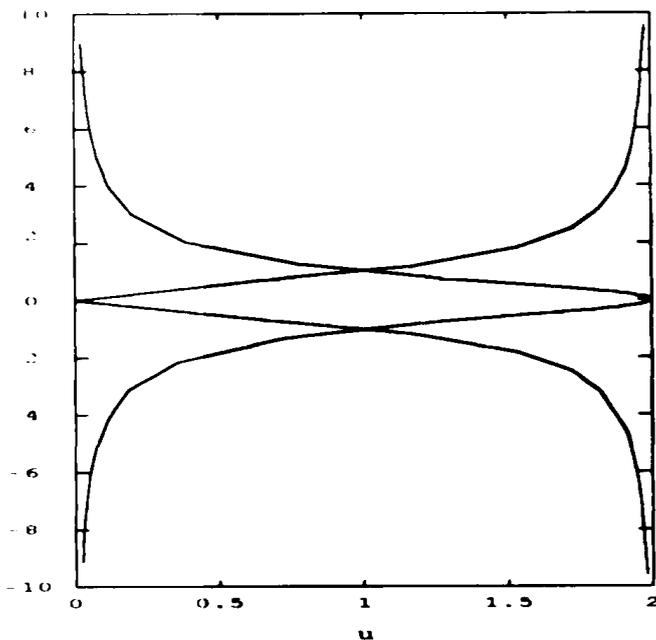
- Le premier test, sert à savoir si le point prédit se situe entre les deux premiers de départ de la branche ou non. Si le test est positif, on a une courbe fermée, donc toute la branche a été calculée
- Le deuxième test permet de vérifier si les paramètres trouvés se situent en dehors du domaine considéré ou s'ils prennent des valeurs égales aux valeurs limites. Dans ce cas, on conclue que la courbe d'intersection est une courbe ouverte

Après le calcul de la branche entière de solutions, on considère les points de bifurcations possibles et on procède à un changement de branche en ces points

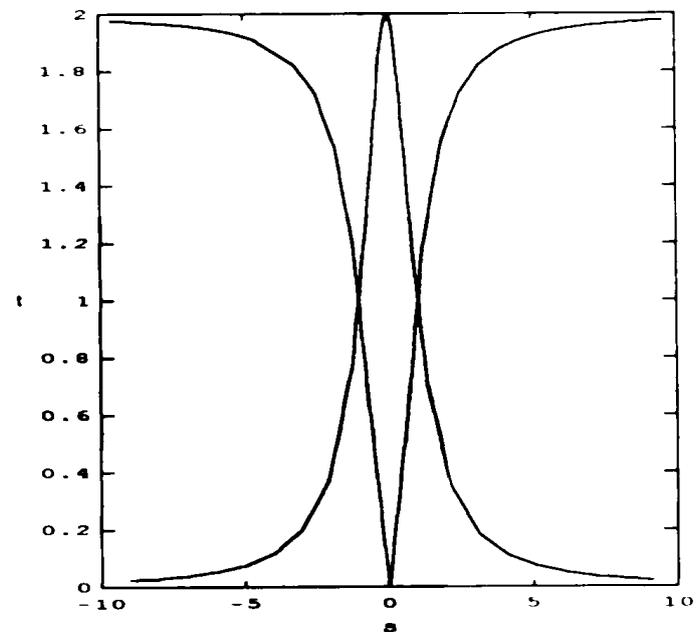
Pour le cas où les deux cylindres ont des rayons égaux ($R=1$), La courbe d'intersection obtenue par continuation est représentée sur les figures (3-1-a) et (3-2-b). Ces figures montrent clairement qu'il existe deux points bifurcation $(u,v,s,t)=(1,1,1,1)$ et $(u,v,s,t)=(1,-1,-1,1)$

Partant de $(u,v,s,t) = (2,10,10,2)$, la branche de solutions est continuée jusqu'au point $(1.9801, -9.9852, -9.9851, 1.9801)$, en passant par le point de retournement: $(0,0,0,0)$

Chaque point de bifurcation est ensuite considéré, et un changement de branche (positive et négative) est effectué, dans la direction du deuxième vecteur nul du jacobien en ces points



a)



b)

Fig.3-1. Représentation de la courbe d'intersection ($R=1$)

a) dans le plan $u-v$

b) dans le plan $s-t$

Pour $R=2$, on obtient les courbes des figures (3-2-a; 3-2-b). Pour ce cas on, a seulement des points de retournements. Partant du bord correspondant au point $(u,v,s,t)=(2,0,5,0)$, La courbe est tracée jusqu'à ce que les conditions d'arrêt soient vérifiées. Cette branche contient trois points de retournement: $(u,v,s,t) = (0.9999, 0.2679, 0.9999, 0.2679)$, $(0,0,0,0)$, et $(0.9999, -0.2679, 0.9999, 0.2679)$.

Les deux autres branches sont continuées, partant respectivement de $(u,v,s,t) = (2, 10, 5,4)$ et $(u,v,s,t) = (2,-10,-5,4)$, Chacune d'elles contient un point de retournement $(u,v,s,t)=(0.9999, 3.7320, 0.9999, 3.7320)$ pour la première et $(u,v,s,t)=(0.9999, -3.7320, 0.9999, 3.7320)$ pour la deuxième

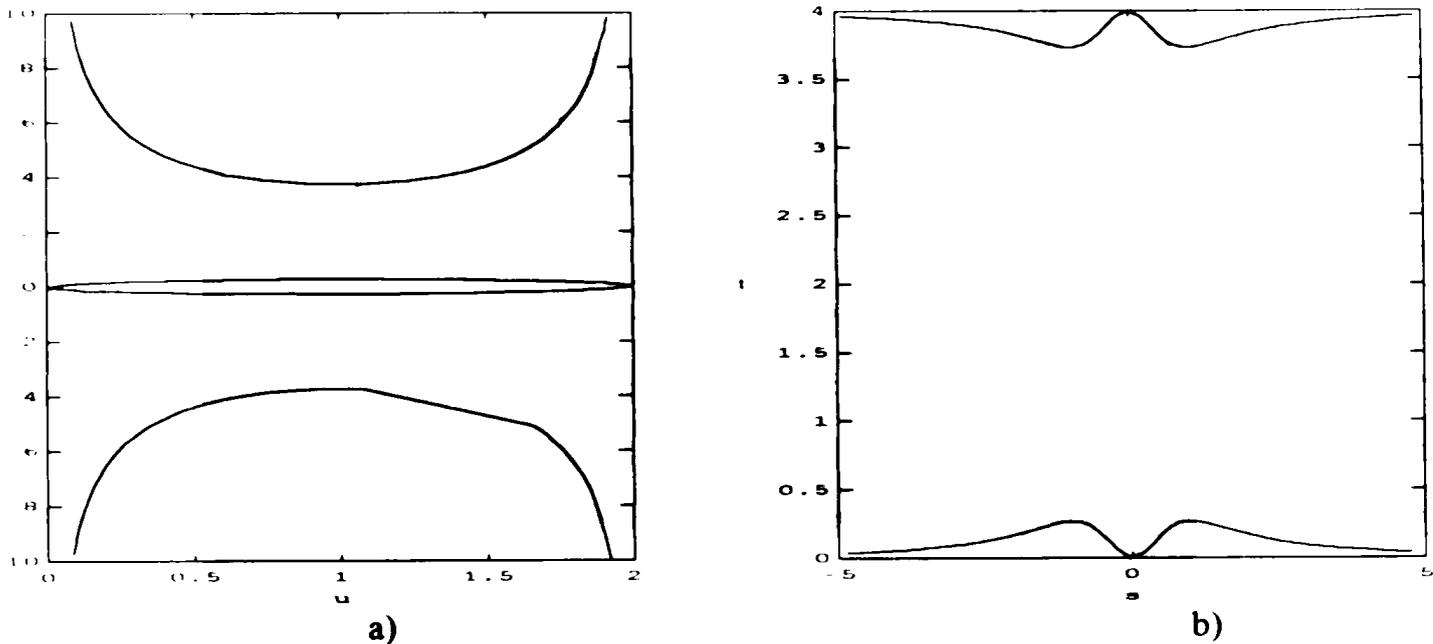


Fig.3-2. : Représentation de la courbe d'intersection ($R=2$)

a) dans le plan $u-v$ b) dans le plan $s-t$

Les figures (3-3-a, 3-3-b) représentent la courbe d'intersection obtenue par continuation pour le cas $R=0.5$. Le procédé de calcul est identique à celui du cas précédent. Il y a trois branches de solutions. La première branche est continuée partant du point $(0,-5,0,1)$, elle contient trois points de retournement: $(u,v,s,t) = (1.8660, -0.9999, -3.7320, 0.4999)$, $(0, 0, 0, 0)$, et $(1.8660, 0.9999, 3.7320, 0.4999)$.

Chacune des deux autres branches contient un point de retournement respectivement: $(0.13364, -0.9999, -0.2676, 0.4999)$ et $(0.13364, 0.9999, 0.2676, 0.4999)$

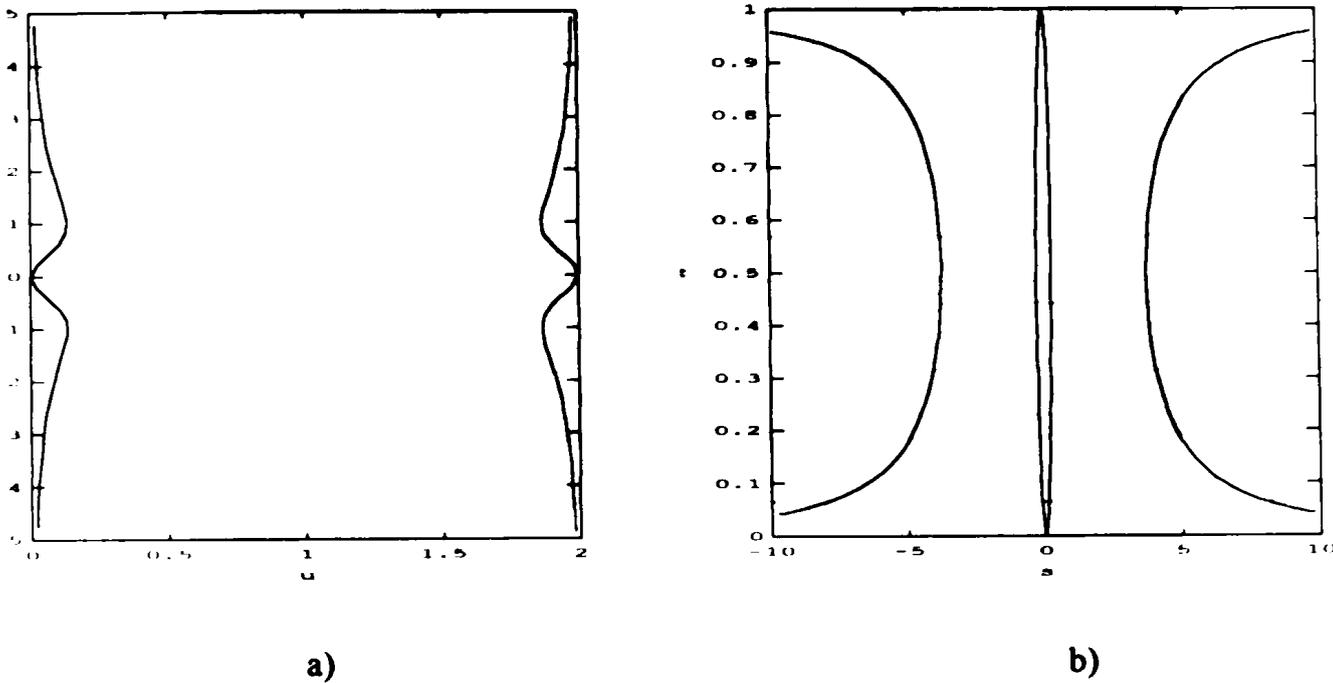


Fig.3-3. : Représentation de la courbe d'intersection ($R=0.5$)
 a) dans le plan $u-v$ b) dans le plan $s-t$

Problème 2: Etude du comportement d'un système dynamique simple

Soit le système dynamique modélisé par le système d'équations non linéaire de la forme:

$$\dot{x} = f(x, \lambda) \quad (3-$$

$$x \in R^n, \lambda \in R$$

95)

L'étape préliminaire dans l'étude d'un tel système est la détermination des points d'équilibre. Ces points sont les solutions du système équations:

$$f(x, \lambda) = 0 \quad (3-96)$$

Si pour certaines valeurs prises par le paramètre λ , le comportement de la solution du système est différent, ce système présente alors des points singuliers à l'équilibre (des points de bifurcation ou des points de retournement par exemple). La frontière entre deux comportements représente un point singulier et correspond le plus souvent à un changement du type de stabilité.

Le système considéré est un système de n équations à $(n+1)$ inconnues qui ne peut être résolu directement, une équation supplémentaire est nécessaire.

Comme pour l'exemple précédent, on complète la définition du problème en utilisant la technique de la pseudo-longueur d'arc et on détermine les branches de solutions correspondant au système étudié en utilisant la méthode de Continuation.

Le système complet est alors résolu pour $k=1$ jusqu'à ce que toute la branche de solution soit déterminée

$$\left. \begin{aligned} f(x, \lambda) = 0 \quad k = 1, \dots \\ (x_k - x_{k-1})\dot{x}_{k-1} + (\lambda_k - \lambda_{k-1})\dot{\lambda}_{k-1} - \Delta s^2 = 0 \end{aligned} \right\} \quad (3-97)$$

L'exemple que l'on va présenter pour illustrer ceci, représente un système électrique constitué de deux génératrices identiques (G1 et G2), montées en parallèle qui déchargent dans une charge R.

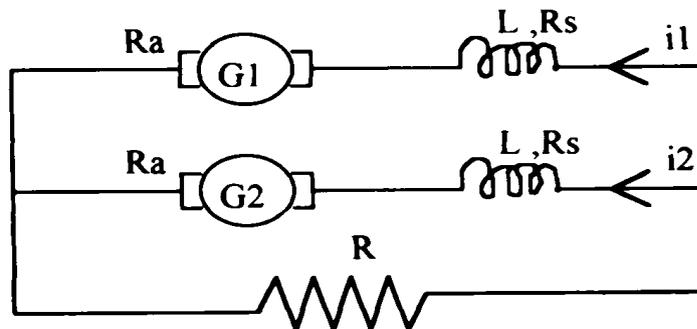


fig. 3-3 : représentation du système étudié

Les équations régissant ce système sont :

$$\begin{aligned} L \frac{di_1}{dt} &= E(i_1) - (Ra + Rs)i_1 - R(i_1 + i_2) = f_1(i_1, i_2, R) \\ L \frac{di_2}{dt} &= E(i_2) - (Ra + Rs)i_2 - R(i_1 + i_2) = f_2(i_1, i_2, R) \end{aligned} \quad (3-98)$$

avec

$Ra \cdot Rs = 0.3\Omega$ et $E(i)$ est une fonction non linéaire de i .

Afin de déterminer les solutions à l'équilibre, l'algorithme décrit précédemment est utilisé pour la résolution du système

$$\begin{cases} f_1(i_1, i_2, R) = 0 \\ f_2(i_1, i_2, R) = 0 \end{cases} \quad (3-99)$$

L'étude mathématique du système dans tout l'espace, fait ressortir l'existence de 4 branches de solutions comportant plusieurs points de retournement et 3 points de bifurcation qui correspondent au cas où $R=0$; c'est à dire lorsqu'on a un court-circuit.

Les points de retournement correspondent à des changements dans le type de stabilité. Les parties en rouge des figures (3-4a), (3-4b) et (3-4c) représentent les zones de fonctionnement stable.

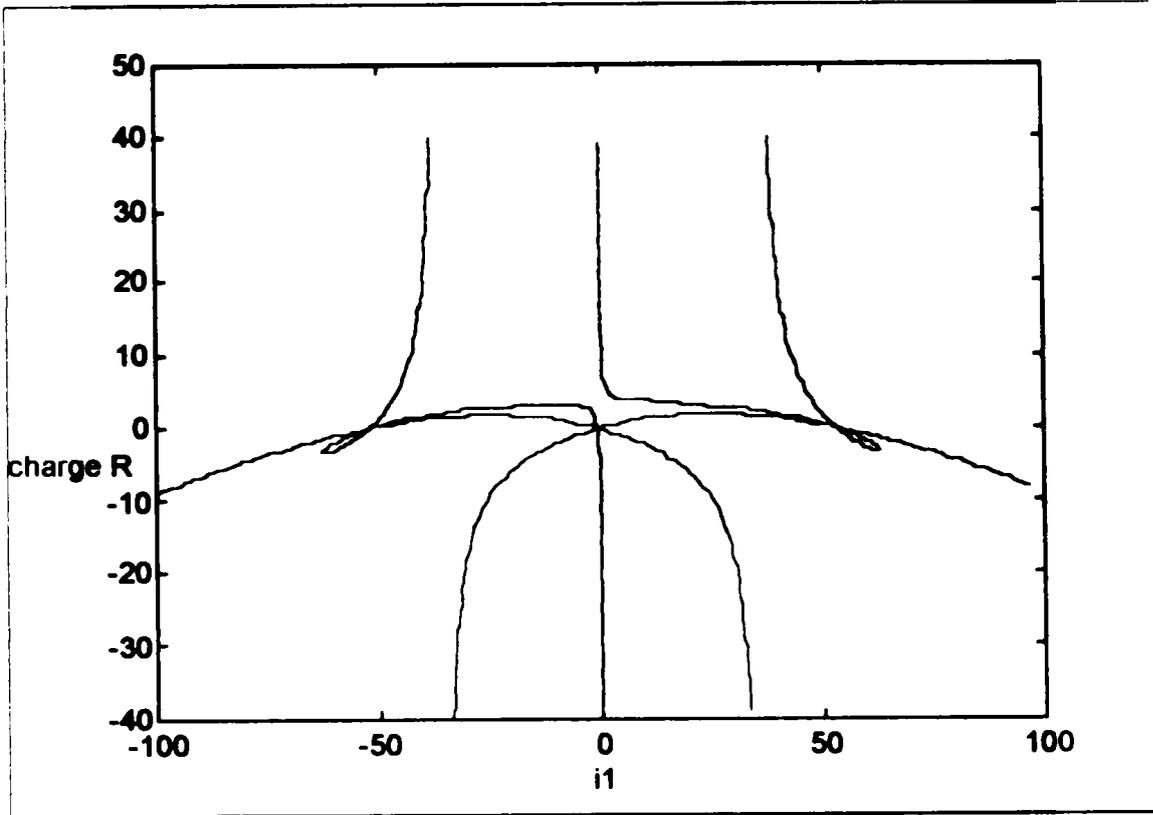


fig. 3-4 a : courant i_1 en fonction de la charge R

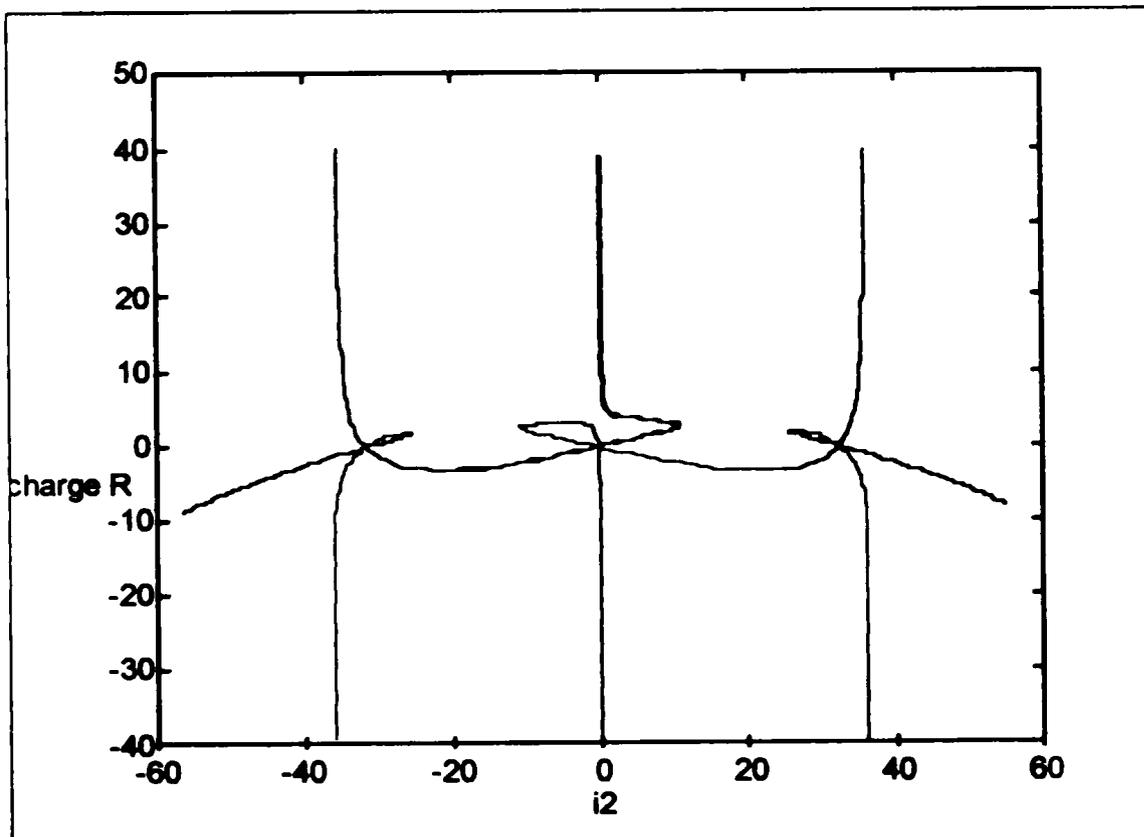


fig. 3-4 b : courant i_2 en fonction de la charge R

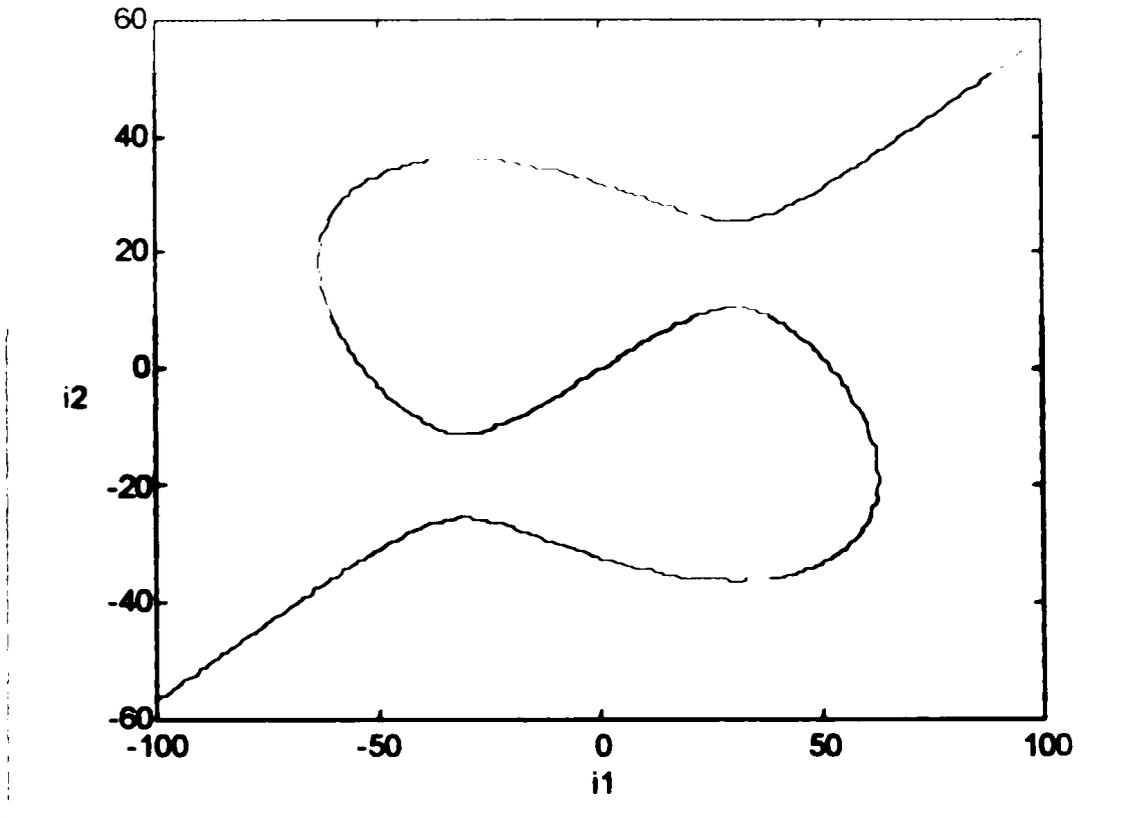


fig. 3-4 c : courant i_1 en fonction du courant i_2

En réalité, les courants i_1 , i_2 et la charge R sont positifs; Les figures (3-5a), (3-5b) et (3-5c) représentent les parties des branches de solutions présentées sur les figures précédentes, qui vérifient ces considérations.

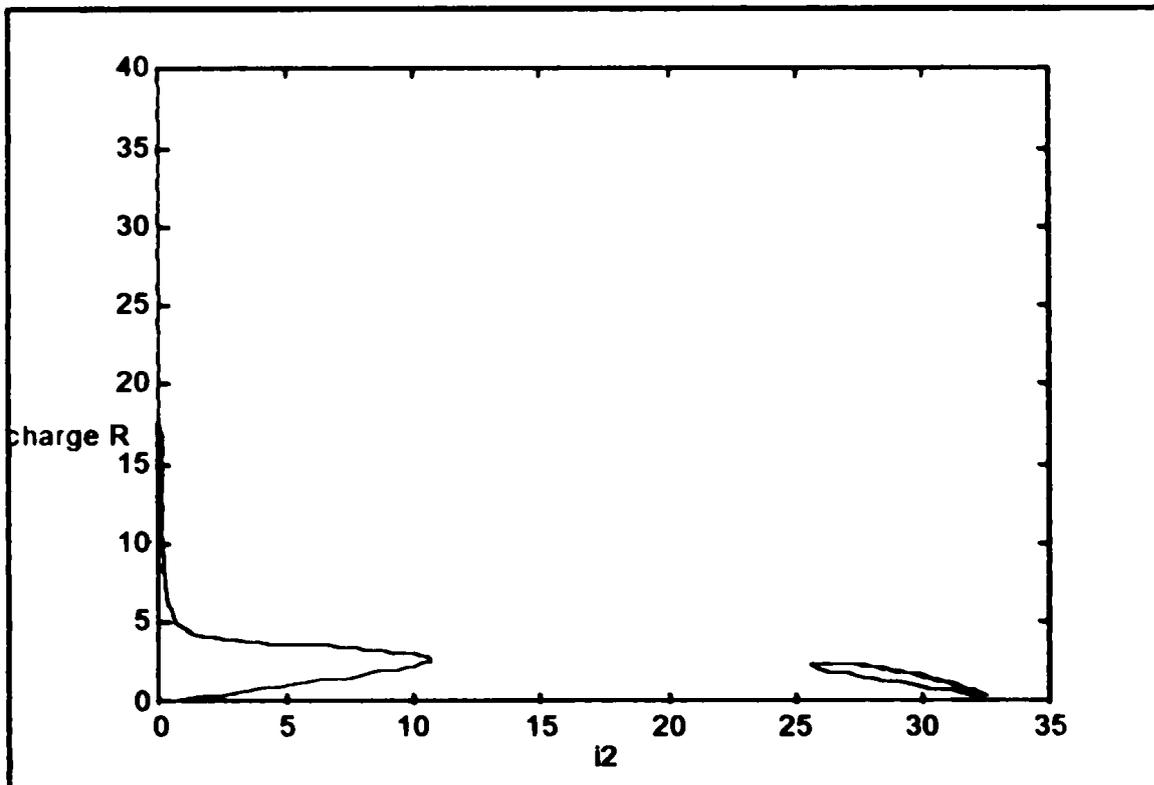


fig. 3-5 a : courant i_2 en fonction de la charge R, pour un fonctionnement réel

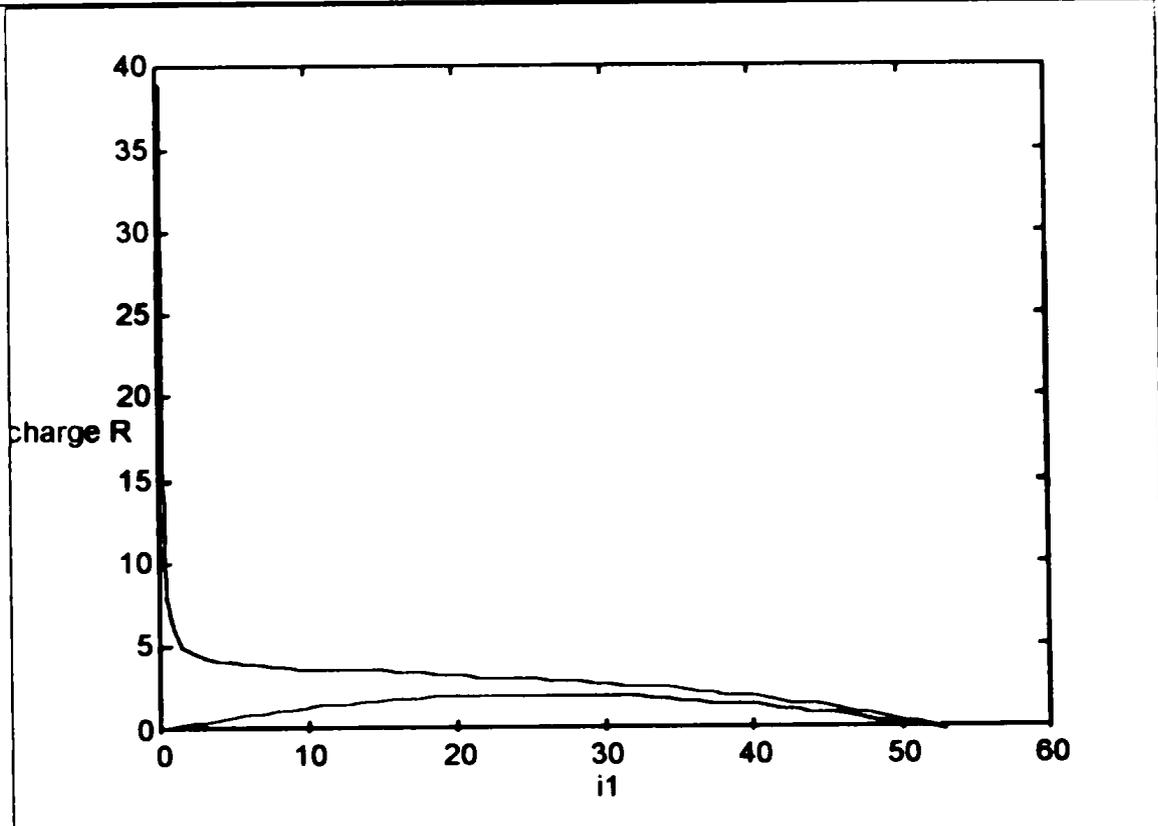


fig. 3-5 b : courant i_1 en fonction de la charge R , pour un fonctionnement réel

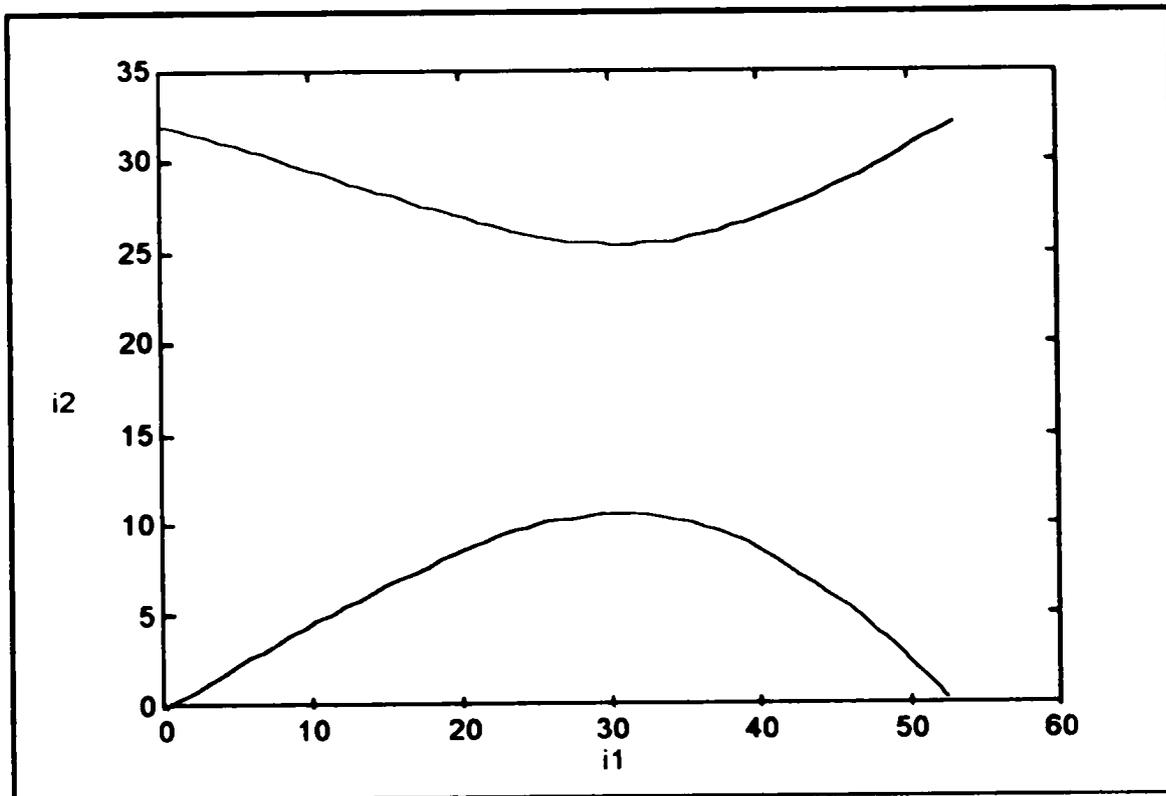


fig. 3-5 c : courant i_1 en fonction du courant i_2 , pour un fonctionnement réel

L'examen des résultats relatifs à ces dernières courbes, fait ressortir l'existence de deux zones de fonctionnement stable. La première zone correspond à : $i_1 > 30.2$ A, $i_2 > 25.5$ A et la charge $R < 2$ Ω . La deuxième zone correspond à : $i_1 < 30.2$ A, $i_2 < 10.6$ A et la charge $R > 2.7$ Ω .

VI- Optimisation des systèmes non linéaires**VI-1 : Introduction**

Considérons le problème

$$\min J(x, \lambda) \quad (3-100)$$

avec

$$f(x, \lambda) = 0 \quad (3-101)$$

$$x \in R^n, \lambda \in R^{n\lambda}, f: R^{n+n\lambda} \rightarrow R^n, J: R^{n+n\lambda} \rightarrow R$$

Ceci est problème de programmation non linéaire avec contraintes d'égalité. La différence avec le traitement standard du problème, est que dans notre étude, on supposera que ces équations définissent une surface unique dans l'espace $R^{n+n\lambda}$.

L'objectif est donc de trouver au moins un point (x^*, λ^*) , solution de (3-101) qui minimise localement la fonction coût (3-100). Pour cela, on doit déterminer une courbe $(x(s), \lambda(s))$ sur la surface $f(x, \lambda) = 0$, qui passe par au moins un minimum local.

Si on désigne par ω le minimum de $J(x, \lambda)$, alors le problème revient à résoudre simultanément les équations

$$\begin{cases} J(x, \lambda) - \omega = 0 \\ f(x, \lambda) = 0 \end{cases} \quad (3-102)$$

VI-2: Conditions d'optimalité

La condition nécessaire pour qu'un point $u_0 = (x_0, \lambda_0)$ solution de $f(u) = 0$, minimise localement la fonction coût $J(u)$, est qu'il existent $p \in R^n$ et $q \in R^{n\lambda}$, tels que:

$$\begin{cases} f'_u p + J'_u q = 0 \\ p' p + q' q - 1 = 0 \end{cases} \quad (3-103)$$

VI-3 : Détection des optimums

On pose

$$F(u, \omega) = \begin{cases} f(u) \\ J(u) - \omega \end{cases} \quad (3-104)$$

L'algorithme de continuation va permettre de tracer les branches de solutions de:

$$F(u, \omega) = 0 \quad (3-105)$$

et de déterminer ainsi, les points de retournements par rapport à ω . Ces derniers correspondent aux différents optimums.

Pour cela on doit résoudre le système étendu (3-106)

$$\begin{cases} F(u, \omega) = 0 \\ F'_u \psi = 0 \\ \psi' \psi - 1 = 0 \end{cases} \quad (3-106)$$

avec $\psi = \{p, q\} \in R^{n-1}$

Par analogie, à ce qui a été dit précédemment, un point de retournement va être caractérisé par :

$$\dot{\omega} = 0 \quad (3-107)$$

Si de plus ce point est quadratique, on aura :

$$\ddot{\omega} \neq 0 \quad (3-108)$$

VI-4 : Optimisation d'un système enzymatique à deux compartiments avec un seul substrat.

Le système également étudié dans Doedel[3-2], peut être schématisé comme suit :



Les deux compartiments où ont lieu les réactions, (grâce à la présence de l'enzyme E) sont séparés les uns des autres par une membrane, à travers laquelle peut diffuser le substrat.

- s1 et s2 représentent les concentrations du substrat à l'intérieur, respectivement du compartiment 1 et du compartiment 2.
- s0 c'est la concentration du substrat extérieur au compartiment 1.
- s0+μ c'est la concentration du substrat à l'extérieur du compartiment 2; μ peut être assimilé à une perturbation

D'après Kernévez[3-8], Ce procédé est modélisé par le système d'équations différentielles suivant :

$$\begin{aligned} \frac{ds1}{dt} &= (s0 - s1) + (s2 - s1) - \rho R(s1) \\ \frac{ds2}{dt} &= (s0 + \mu - s2) + (s1 - s2) - \rho R(s2) \end{aligned}$$

avec $R(s) = \frac{s}{1 + s + s^2}$;

Dans notre étude on va considérer le système à l'équilibre, c'est à dire:

$$\begin{aligned} f_1(s1, s2, s0) &= (s0 - s1) + (s2 - s1) - 100 R(s1) = 0 \\ f_2(s1, s2, s0) &= (s0 + \mu - s2) + (s1 - s2) - 100 R(s2) = 0 \end{aligned}$$

et on veut chercher les extremums de l'activité définie par

$$J(s_1, s_2, s_0) = 100 (R(s_1) + R(s_2))$$

On applique l'algorithme décrit précédemment, à ce problème pour différentes valeurs de μ . On continue alors, les branches de solutions pour

$$\begin{cases} f(s_1, s_2, s_0) = 0 = \begin{cases} f_1(s_1, s_2, s_0) \\ f_2(s_1, s_2, s_0) \end{cases} \\ J(s_1, s_2, s_0) - \omega = 0 \end{cases}$$

Les résultats obtenus pour chaque cas, sont représentés dans ce qui suit.

1° Cas : $\mu = 1$

Partant du point $(s_1, s_2, s_0, \omega) = (0, 0, 0, 0)$, on continue les branches de solutions relatives à ce problème. On détecte alors l'existence de quatre optimums locaux: deux maximums et deux minimums, comme indiqué dans le tableau n°3-1 et montré par les figures (3-6.a), (3-6.b), (3-6.c) et (3-6.d)

Optimum n°	s1	s2	s0	ω
1	0.8334	1.0558	33.5787	66.2682
2	0.6215	7.3419	24.8572	42.7511
3	0.8956	8.9979	25.9916	43.0909
4	8.7753	9.2662	18.3964	19.7512

Tableau n° 3-1

Il apparaît clairement, que pour une perturbation μ égale à 1, l'activité est maximale pour une concentration extérieure égale à 33.5787

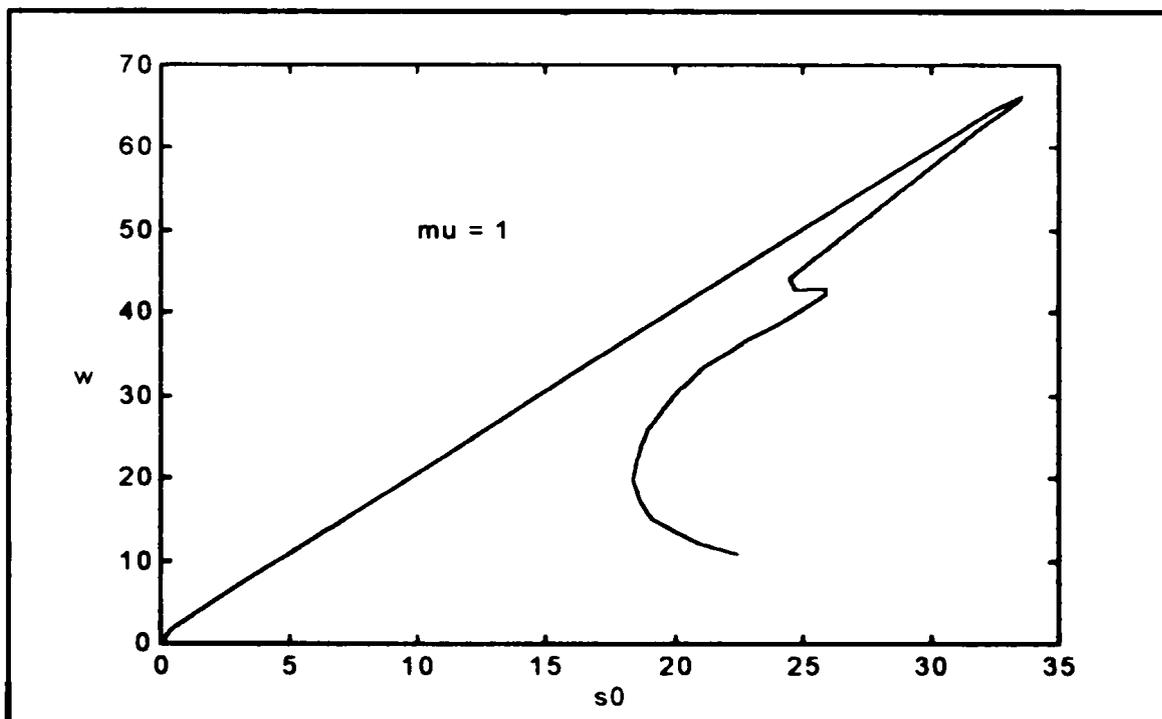


fig. 3-6.a : L'activité ω en fonction du paramètre s_0

Concernant la stabilite du systeme, on a un changement du type de stabilite apres chaque optimum. En effet, le systeme est stable sur la premiere branche ($0 \leq s_0 \leq 33.5787$), ensuite il devient instable (s_0 variant de 66.2682 à 42.7511 et s_0 variant de 33.5787 à 24.8572), puis de nouveau stable, instable et enfin stable.

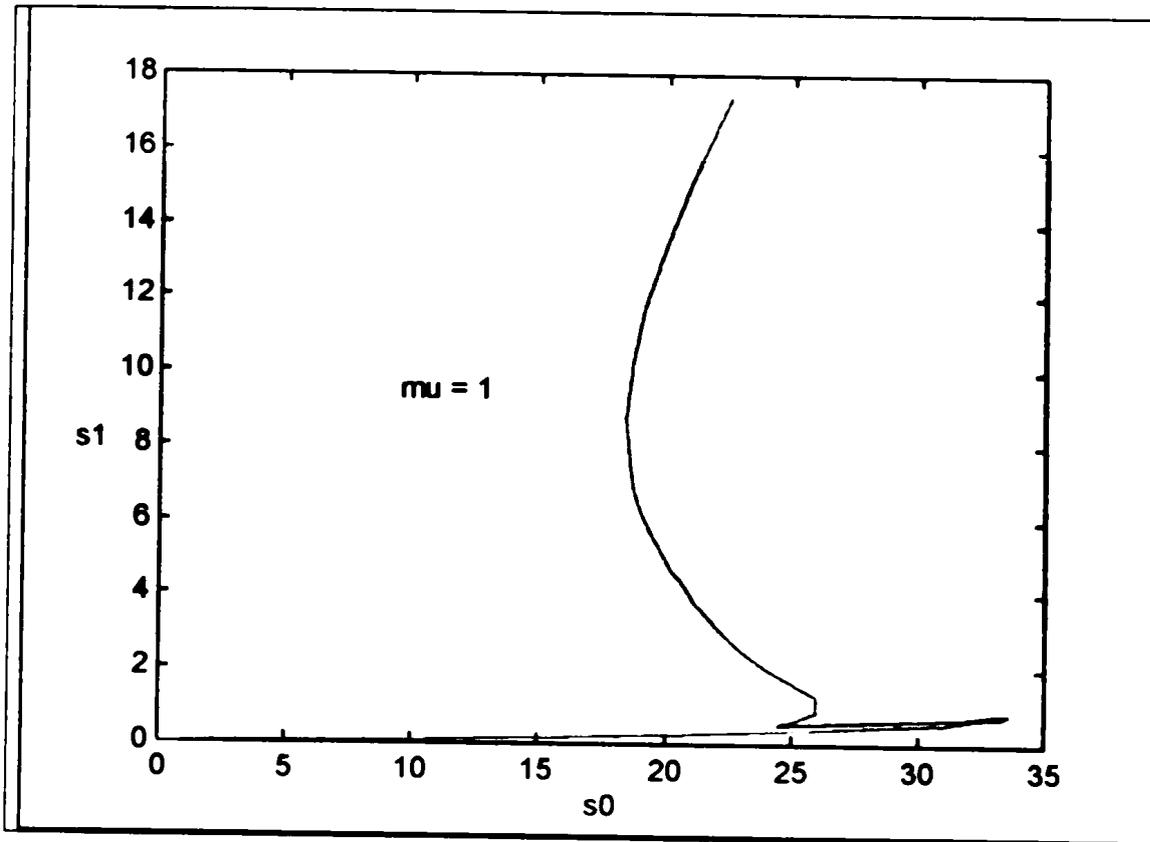


fig. 3-6.b : La concentration s_1 en fonction du paramètre s_0

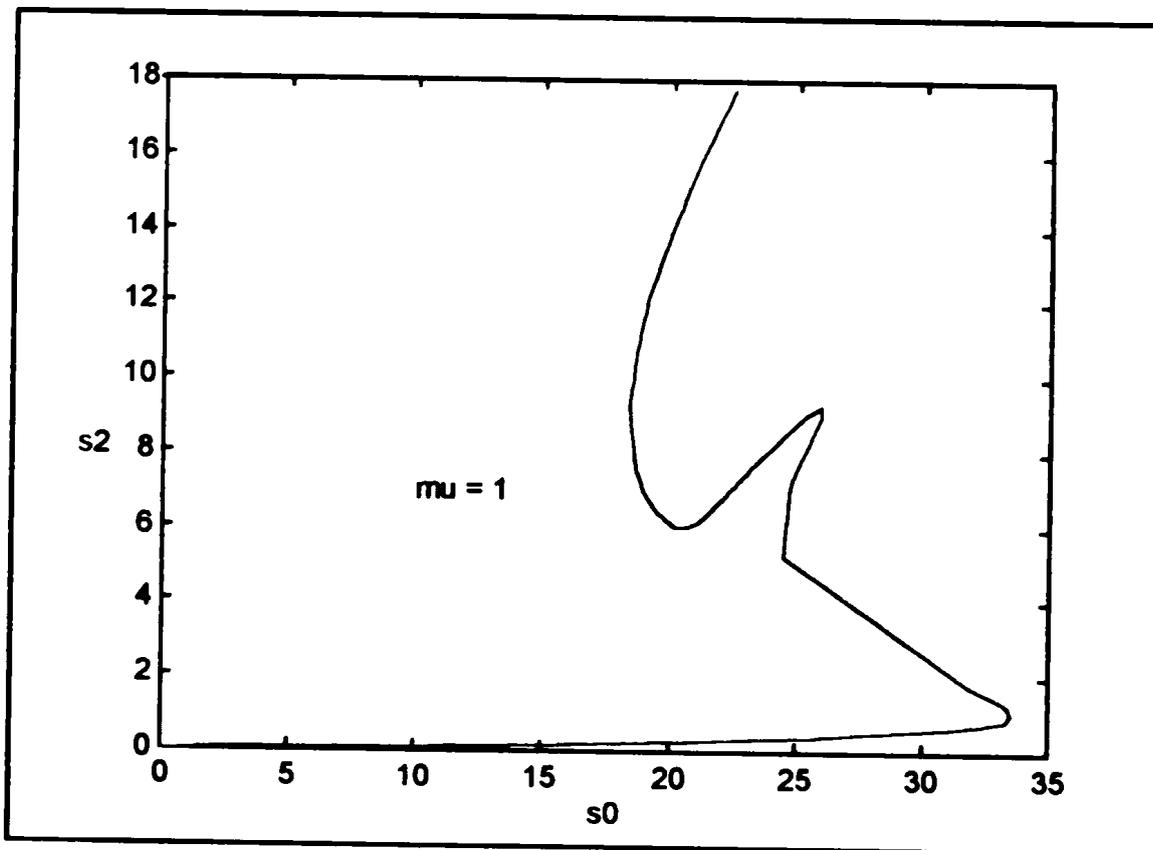


fig. 3-6.b : La concentration s_2 en fonction du paramètre s_0

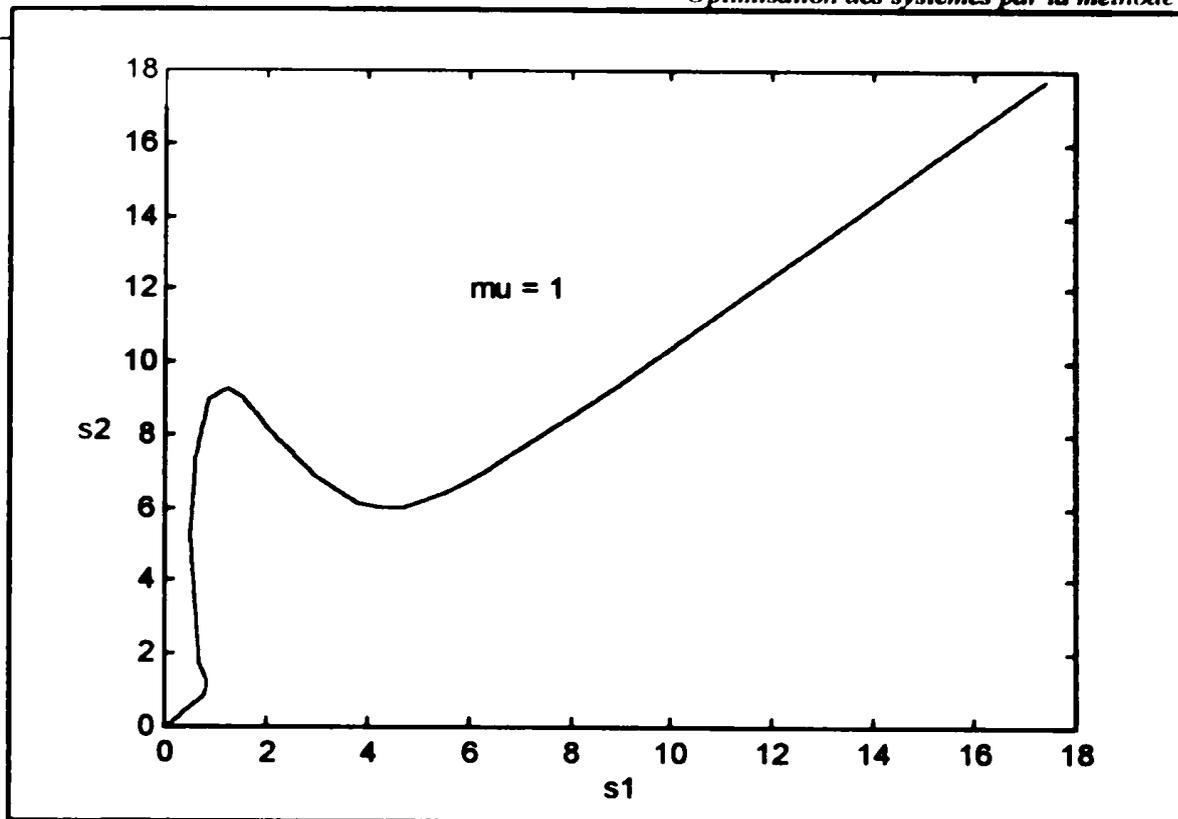


fig 3-6 d . La concentration dans le compartiment2 en fonction de la concentration dans le compartiment 1

2° Cas : $\mu = 5$

Comme pour le cas précédent, on part du point $(s_1, s_2, s_0, \omega) = (0, 0, 0, 0)$, et on continue les branches de solutions relatives à ce problème. Là aussi, on détecte l'existence de quatre optimums locaux deux maximums et deux minimums, comme indiqué dans le tableau n°3-2 et montré par les figures (3-7 a), (3-7 b), (3-7 c) et (3-7 e)

On remarque que l'activité maximale, est plus petite dans ce cas que dans le cas précédent ; ce qui est normale puisque la perturbation est plus importante

Optimum n°	s1	s2	s0	ω
1	0.5566	1.0695	29.8651	63.1051
2	0.4478	7.1482	20.9142	39.2322
3	0.9476	10.9454	24.2507	41.6090
4	8.4000	10.7550	16.5503	18.9454

Tableau n° 3-2

Pour ce qui est de la stabilité du système, on obtient la même chose que pour le cas $\mu = 1$, c'est à dire qu'on a une succession d'état stable et d'état instable, après chaque optimum.

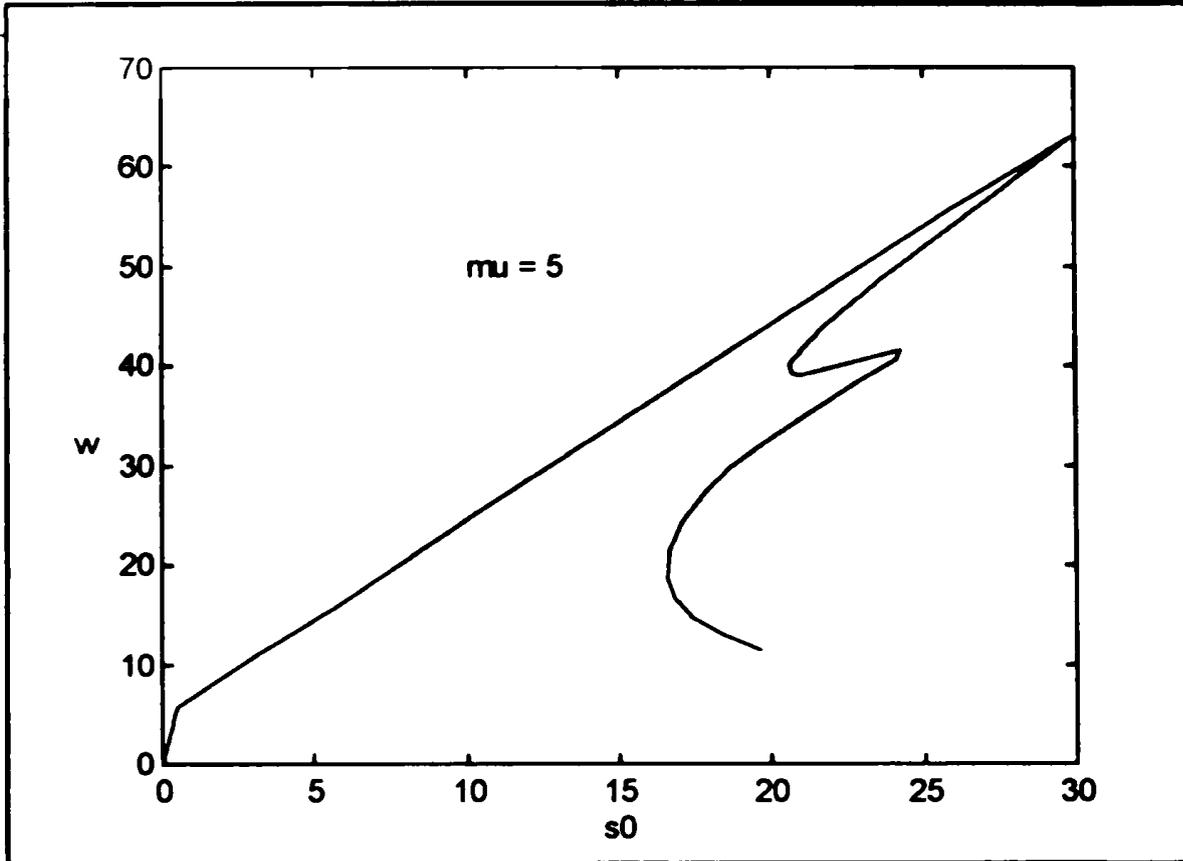


fig. 3-7.a : L'activité ω en fonction du paramètre s_0

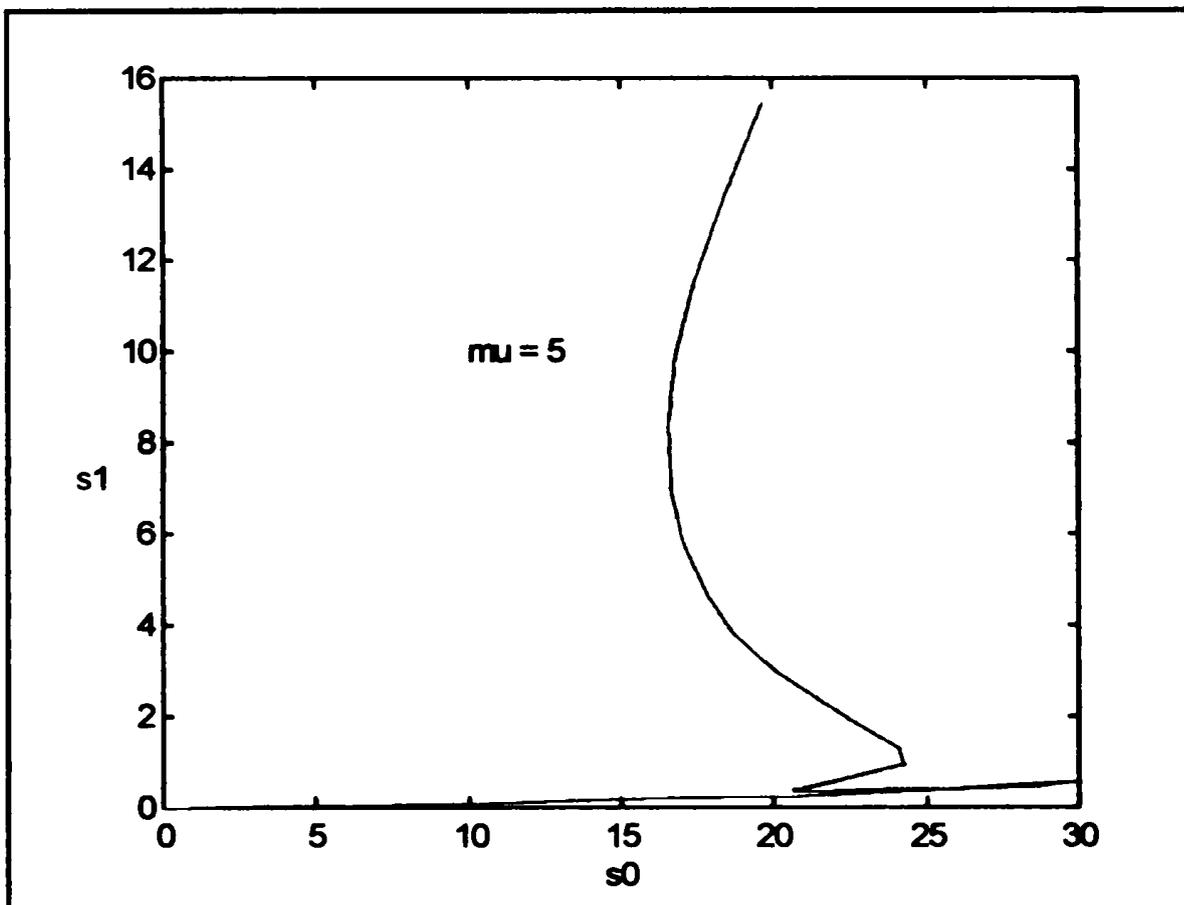


fig. 3-7.b : La concentration s_1 en fonction du paramètre s_0

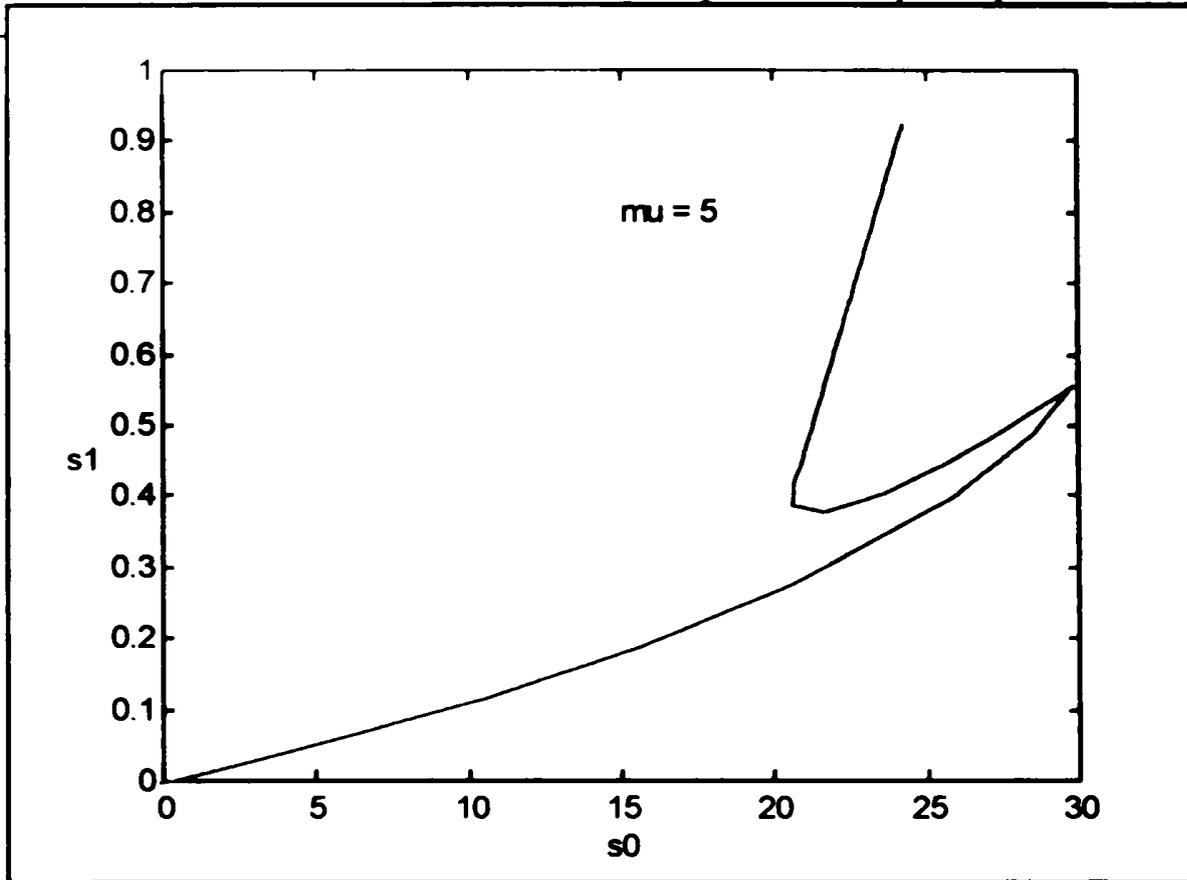


fig. 3-7.c : Agrandissement de la figure(3-7.b) pour $0 < s_1 < 1$

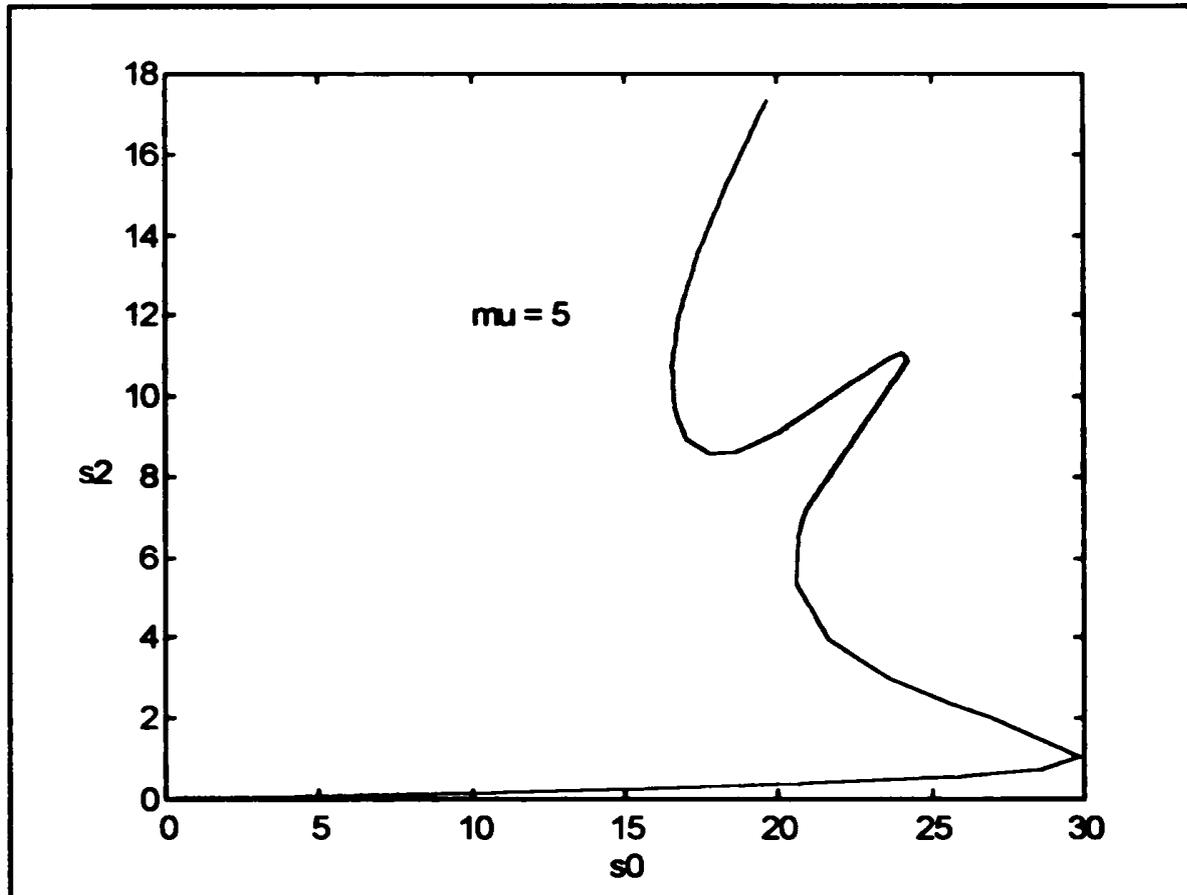


fig. 3-7.d : La concentration s_2 en fonction du paramètre s_0

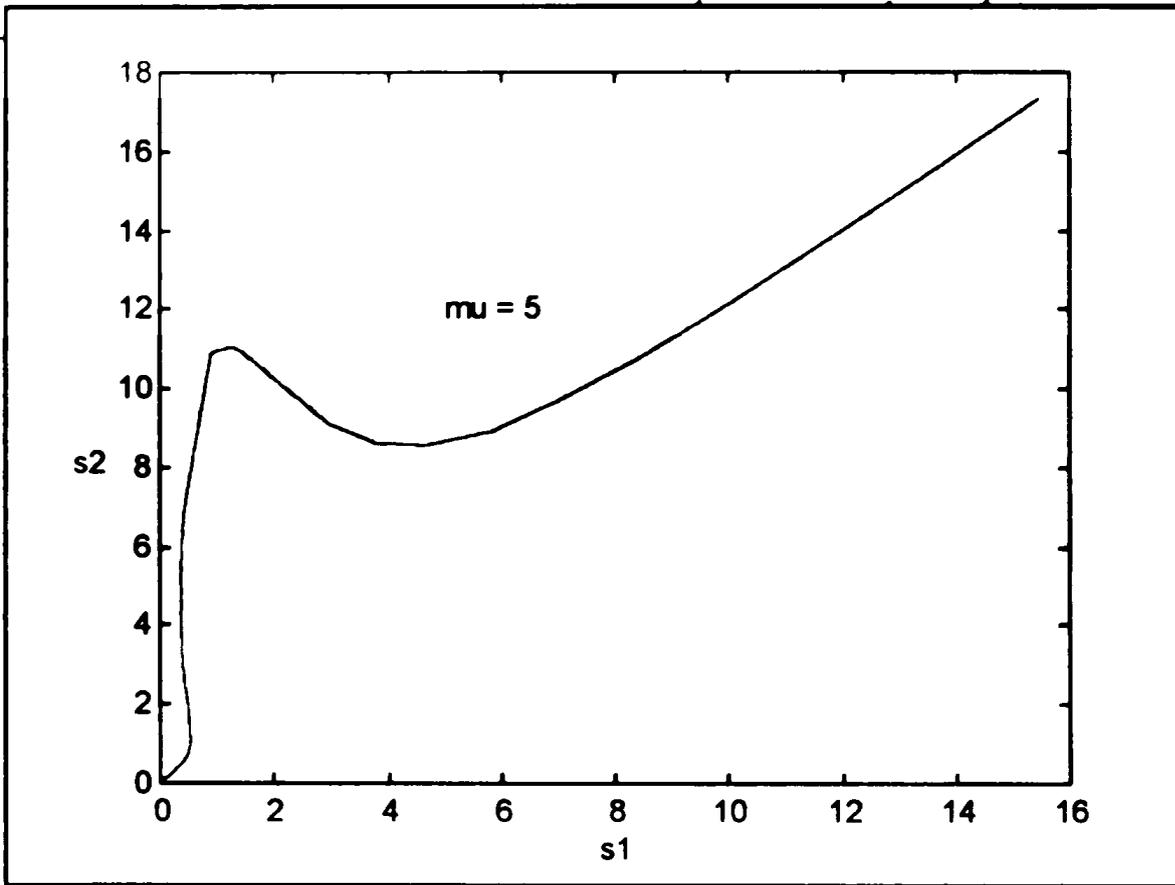


fig. 3-6 e : La concentration dans le compartiment 2 en fonction de la concentration dans le compartiment 1

Les résultats obtenus avec ces deux exemples mettent en évidence l'influence du paramètre μ sur l'activité ω . Pour trouver la valeur de μ pour laquelle on a la plus grande activité, on doit donc considérer ce paramètre comme inconnu dans le problème d'optimisation posé et l'identifier en même temps que les autres paramètres. Ce sera le sujet du paragraphe suivant.

VII- Identification des paramètres par continuation successive

L'algorithme de continuation présenté précédemment, est directement applicable lorsque le nombre de paramètre λ est égale à 1. Si ce nombre est supérieure, on procède par continuation successive.

Les étapes de l'algorithme sont alors les suivantes :

- 1/ Continuer une branche de solutions pour le système (3-102), avec tous les paramètres λ_i , fixes, sauf un paramètre λ_k qui lui reste libre. On détermine l'extremum quadratique ω . Le système à résoudre est alors le suivant :

$$\bullet \begin{cases} f(x, \lambda) = 0 \\ J(x, \lambda) - \omega = 0 \\ f_x(x, \lambda) p + J_x(x, \lambda) q = 0 \\ f_\lambda(x, \lambda) p + J_\lambda(x, \lambda) q = \Gamma \\ p' p + q^2 - 1 = 0 \end{cases} \quad \Gamma = \{\Gamma_i; i = 1, \dots, n_\lambda\} \quad (3-109)$$

- 2/ Pour l'extremum trouvé, définir p et q qui sont tous deux non nuls en même temps et qui vérifient les trois dernières équations de (3-109). Le Γ_i , correspondant au paramètre λ_i , libre, est nul. Les autres Γ_i , ne sont pas nécessairement nuls.
- 3/ Libérer un autre paramètre λ_k et calculer la nouvelle solution pour (3-109). Pour chaque solution trouvée, recommencer la procédure de continuation, en libérant à chaque fois un des paramètres restants jusqu'à ce qu'il n'y ait plus de paramètres fixes. Le Γ_i , correspondant au paramètre libéré est mis à zéro. Ce qui donne, lorsque tous les paramètres sont libérés, un vecteur Γ nul.

Le système peut être réécrit sous la forme :

$$\left\{ \begin{array}{l} f(x, \lambda_i) = 0 \\ J(x, \lambda_i) - \omega = 0 \\ f_x(x, \lambda_i) p + J_x(x, \lambda_i) q = 0 \\ f_{\lambda_k}(x, \lambda_i) p + J_{\lambda_k}(x, \lambda_i) q = 0 \quad k = 1, \dots, \nu\lambda \\ p' p + q^2 - 1 = 0 \\ f_{\lambda_i}(x, \lambda_i) p + J_{\lambda_i}(x, \lambda_i) q - \Gamma_i = 0 \quad i = 1, \dots, \nu\mu \end{array} \right. \quad (3-110)$$

$\nu\lambda$ représente le nombre de paramètres libres et $\nu\mu$ le nombre de paramètres restants, c'est à dire on a $\nu\lambda + \nu\mu = n\lambda$.

Les quatre premières équations de (3-110) forment un système d'équations non linéaires que l'on résout par la méthode de Newton- Raphson; ensuite on calcule Γ par la dernière équation de (3-110).

*- Application de L'algorithme au problème décrit précédemment

Dans le cas de l'exemple précédent, on a $\lambda = \{s_0, \mu\}$. Les résultats présentés correspondent à la continuation de la branche de solutions pour $\lambda_1 = s_0$ libre et $\lambda_2 = \mu$ fixe, égale à 1 dans le premier cas et à 5, dans le deuxième cas. Pour chaque cas on a également :

- 1°cas : $\mu = 1$

Pour le maximum $\omega = 66.2682$, on a :

- $\Gamma_{s_0} = f_{s_0}(s_1, s_2, s_0) p + J_{s_0}(s_1, s_2, s_0) q = F_{s_0}' \psi = 0$
- $\Gamma_{\mu} = f_{\mu}(s_1, s_2, s_0) p + J_{\mu}(s_1, s_2, s_0) q = F_{\mu}' \psi = -0.46372$

- 2°cas : $\mu = 5$

Pour le maximum $\omega = 63.1051$, on a :

- $\Gamma_{s_0} = f_{s_0}(s_1, s_2, s_0) p + J_{s_0}(s_1, s_2, s_0) q = F_{s_0}' \psi = 0$
- $\Gamma_{\mu} = f_{\mu}(s_1, s_2, s_0) p + J_{\mu}(s_1, s_2, s_0) q = F_{\mu}' \psi = 0.5483$

On libère le deuxième paramètre μ , on obtient alors :

- 1° cas : $\mu = 1$

Partant de s_1, s_2, s_0, μ et ω correspond à l'activité maximale, c'est à dire le point $(s_1, s_2, s_0, \mu, \omega) = (0.8334, 1.0558, 33.5787, 1, 66.2682)$, le système est résolu au bout de 6 itérations. Le tableau suivant montrent les résultats intermédiaires pour chaque itération (iter. n° indique le numéro de l'iteration)

iter n°	s1	s2	s0	μ	ω	$\ F\ $
1	1.3863	0.9344	36.8632	2.7708	68.8632	4.5417
2	0.7614	0.9812	35.8240	1.2654	68.6399	3.9226
3	0.9279	0.9987	34.6569	0.2504	67.1367	0.7499
4	0.9928	0.9999	34.3726	0.0332	66.7204	0.0768
5	0.9999	0.9999	34.3337	$3.34 \cdot 10^{-4}$	66.6672	$8.06 \cdot 10^{-3}$
6	0.9999	0.9999	34.3333	$0.337 \cdot 10^{-7}$	66.6666	$0.806 \cdot 10^{-7}$

On calcule le vecteur Γ pour le maximum trouvé, on trouve :

$$-\Gamma_{s_0} = 0 \quad \text{et} \quad \Gamma_{\mu} = 0$$

par conséquent le vecteur Γ est bien nul.

- 2° cas : $\mu = 5$

Partant de $(s_1, s_2, s_0, \mu, \omega) = (0.5565, 1.0695, 29.8656, 5, 63.1051)$, le système est résolu au bout de 6 itérations. Le tableau suivant montrent les résultats intermédiaires pour chaque itération.

iter n°	s1	s2	s0	μ	ω	$\ F\ $
1	0.9441	1.1423	32.0625	4.0778	64.5386	0.1354
2	1.0714	0.9694	34.6484	0.8624	65.0522	0.07323
3	1.0027	0.9998	34.3358	$1.95 \cdot 10^{-5}$	66.6670	$8.06 \cdot 10^{-3}$
4	0.9999	1.0000	34.3333	$0.413 \cdot 10^{-7}$	66.6666	$0.806 \cdot 10^{-8}$

Comme pour le cas précédent, le vecteur Γ pour le maximum trouvé, est nul:

$$-\Gamma_{s_0} = 0 \quad \text{et} \quad \Gamma_{\mu} = 0$$

Il en ressort que l'activité maximale est $\omega = 66\,6666$, elle correspond à une solution symétrique $s_1 = s_2 = 1$ avec $s_0 = 34\,3333$ et une perturbation nulle $\mu = 0$

*- Application de L'algorithme à un autre problème

Soit à résoudre : $\min J(u, \lambda_1, \lambda_2, \lambda_3, \lambda_4) = u + \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4$

avec $f(u, \lambda_1, \lambda_2, \lambda_3, \lambda_4) = u^2 + \lambda_1^2 + \lambda_2^2 + \lambda_3^2 + \lambda_4^2 - 1$

1^oétape

λ_1 libre et $\lambda_2 = \lambda_3 = \lambda_4 = 0$ (fixes)

Partant de $(u, \lambda_1, \omega) = (0, 0, 0)$, on obtient :

- $(u, \lambda_1, \omega) = (0.7071, 0.77071, 1.4142)$

- $\Gamma_1 = 0.155 \cdot 10^{-7}$

- Nombre d'itérations : 4

2^oétape

on libère λ_2 et $\lambda_3 = \lambda_4 = 0$

Partant de $(u, \lambda_1, \lambda_2, \omega) = (0.7071, 0.77071, 0, 1.4142)$, on obtient :

- $(u, \lambda_1, \lambda_2, \omega) = (0.5773, 0.5773, 0.5773, 1.7320)$

- $\Gamma_1 = 0.0435 \cdot 10^{-9}$, $\Gamma_2 = 2.3188 \cdot 10^{-9}$

- Nombre d'itérations : 3

3^oétape

on libère λ_3 et $\lambda_4 = 0$

Partant de $(u, \lambda_1, \lambda_2, \lambda_3, \omega) = (0.5773, 0.5773, 0.5773, 0, 1.7320)$, on obtient :

- $(u, \lambda_1, \lambda_2, \lambda_3, \omega) = (0.5000, 0.5000, 0.5000, 0.5000, 2.000)$

- $\Gamma_1 = -0.557 \cdot 10^{-7}$, $\Gamma_2 = -0.557 \cdot 10^{-7}$, $\Gamma_3 = 0.265 \cdot 10^{-7}$

- Nombre d'itérations : 4

4^oétape

on libère λ_4

Partant de $(u, \lambda_1, \lambda_2, \lambda_3, \lambda_4, \omega) = (0.5000, 0.5000, 0.5000, 0.5000, 0, 2.000)$,

on obtient :

- $(u, \lambda_1, \lambda_2, \lambda_3, \lambda_4, \omega) = (0.4884, 0.4884, 0.4884, 0.4884, 0.4884, 2.4440)$

- $\Gamma_1 = 0.154 \cdot 10^{-6}$, $\Gamma_2 = 0.154 \cdot 10^{-6}$, $\Gamma_3 = 0.154 \cdot 10^{-6}$, $\Gamma_4 = 0.102 \cdot 10^{-6}$

- Nombre d'itérations : 4

CHAPITRE IV

IDENTIFICATION DES PARAMETRES PAR LA METHODE DES SENTINELLES

I. Introduction

Comme il a été présenté dans le premier chapitre, les problèmes d'identification les plus couramment rencontrés en mathématiques appliquées sont les problèmes de moindres carrés qui consistent à minimiser un critère quadratique représentant l'écart entre $y(t, \lambda)$ la solution du modèle mathématique représentant le système et $Zd(t)$, les mesures relevées sur un intervalle de temps $[t_0, t_1] \subset]0, T[$

Ces mesures sont en général bruitées et vérifient la relation:

$$Zd(t) = y(t, \lambda) + \text{erreur}(t) \quad (4-1)$$

La méthode des sentinelles pour l'identification des paramètres, se base sur la théorie des Sentinelles présentée par J L Lions [4-3]

Elle consiste à construire, pour chaque paramètre inconnu, une sentinelle indépendante des mesures, qui va nous indiquer la sensibilité de ce paramètre par rapport au bruit. En effet, et d'après le "Shadow theorem", il existe un « agent secret » dans l'ombre de « l'observateur », cet agent se trouve dans l'attracteur (étrange ou non) et a une trajectoire très proche de celle des mesures Zd . L'identification des paramètres λ fournit l'identité de l'agent secret $y(t, \lambda)$.

Cette méthode est donc très intéressante pour la résolution des problèmes mal définis où lorsque les observations sont fortement bruitées. D'ailleurs les résultats obtenus lors de la résolution de différents problèmes réels par cette méthode (détaillés dans la référence [4-2]), sont une preuve de l'efficacité de la méthode.

II- Rappel de la méthode des moindres carrés

Le problème de moindre carré revient à minimiser une fonction coût:

$$\varphi(\lambda) = \frac{1}{2} \|y(t, \lambda) - Zd(t)\|_H^2 \quad (4-2)$$

Sachant que:

$\|\cdot\|_H$ représente la norme associée au produit scalaire dans H. et que $\|f\|_H^2 = \int_{t_0}^{t_1} f^2(t) dt$

En développant (4-2), on obtient :

$$\begin{aligned} \varphi(\lambda) &= ((y(t, \lambda) - Zd(t)), (y(t, \lambda) - Zd(t))) \\ &= \frac{1}{2} (y'(t, \lambda), y(t, \lambda)) - (y(t, \lambda), Zd(t)) + \frac{1}{2} \|Zd(t)\|^2 \end{aligned} \quad (4-3)$$

Si on pose que la solution $y(t, \lambda)$ est de la forme :

$$y(t, \lambda) = B \lambda \quad (4-4)$$

B étant un opérateur, alors (4-3) se réécrit comme suit :

$$\begin{aligned}\varphi(\lambda) &= \frac{1}{2}((B\lambda)'(B\lambda)) - (B\lambda, Z_d(t)) + \frac{1}{2}|Z_d(t)|^2 \\ &= \frac{1}{2}(\lambda' B' B \lambda) - (\lambda' B' Z_d(t)) + \frac{1}{2}|Z_d(t)|^2 \\ &= \frac{1}{2}(\lambda' A \lambda) - (\lambda' b) + \frac{1}{2}|Z_d(t)|^2\end{aligned}\quad (4-5)$$

Avec :

$$A = B' B \quad \text{et} \quad b = B' Z_d(t) \quad (4-6)$$

III- Minimisation de la fonction coût

La condition pour que le vecteur λ^* soit un optimum est :

$$\frac{\partial \varphi(\lambda)}{\partial \lambda} = 0 \quad (4-7)$$

ceci implique :

$$A \lambda - b = 0 \quad (4-8)$$

Si la matrice A est définie positive, alors :

$$\lambda = A^{-1} b = A^{-1} B' Z_d \quad (4-9)$$

On pose :

$$W = A^{-1} B' \quad (4-10)$$

l'expression (4-9) devient :

$$\lambda = W Z_d \quad (4-11)$$

Si de plus, l'opérateur B est injectif, alors pour tout ensemble de données Z_d , il existe un vecteur de paramètre unique λ^* tel que :

$$\varphi(\lambda) > \varphi(\lambda^*) \quad \forall \lambda \quad (4-12)$$

Proposition

En multipliant l'expression (4-10) par B , on obtient :

$$W B = A^{-1} B' B \quad (4-13)$$

Sachant que :

$$A = B' B$$

alors

$$W B = A^{-1} B' B = A^{-1} A = I \quad (4-14)$$

(I étant la matrice identité).

W est donc l'inverse généralisé de B .

IV- Conditions d'existence de la matrice W

Pour que la matrice W existe il faut que les deux conditions suivantes soient vérifiées:

- L'opérateur B doit être injectif.
- La matrice A doit être définie positive.

a/ Opérateur B injectif

Pour que B soit injectif, il doit vérifier :

- $B \lambda = 0 \Rightarrow \lambda = 0$
- $B \lambda_1 = B \lambda_2 \Rightarrow \lambda_1 = \lambda_2$

Ce qui veut dire que pour un ensemble de données $Zd(i)$, correspond un vecteur de paramètre unique λ .

b/ La matrice A définie positive

On a :

$$\forall \lambda, (A \lambda, \lambda) = (B \lambda, B \lambda) = |B \lambda|_H^2 \geq 0$$

De plus, on sait que :

$$(A \lambda, \lambda) = 0 \Rightarrow |B \lambda| = 0$$

Puisque B est injectif, alors :

$$B \lambda = 0 \Rightarrow \lambda = 0$$

Par conséquent :

$$\forall \lambda \neq 0, (A \lambda, \lambda) > 0$$

Ce qui veut dire que la matrice symétrique A est définie positive si B est injectif.

V- Définition des Sentinelles

Si on désigne par e^i , le $i^{\text{ème}}$ vecteur de la base canonique considérée, alors le $i^{\text{ème}}$ composant du vecteur λ est donné par la relation :

$$\begin{aligned}\lambda_i &= (e^i, \lambda) = (e^i, WZd) = (W^i e^i, Zd) \\ &= (B A^{-1} e^i, Zd) \\ &= (W_i, Zd)\end{aligned}\tag{4-15}$$

et W_i est appelé sentinelle associée au paramètre λ_i .

Ainsi, pour un ensemble de données Zd , le $i^{\text{ème}}$ paramètre λ_i , est égale au produit de la sentinelle associée au paramètre et des données Zd . Pour chaque paramètre λ_i ($i=1, \dots, n_\lambda$), on associe une sentinelle W_i .

L'identification des paramètres va s'effectuer en deux étapes : lors de la première étape on détermine les sentinelles associées à chacun des paramètres. Ensuite, on fait le produit avec le vecteur des données. Cette façon de procéder, représente un avantage certain par rapport aux autres méthodes.

En effet, pour un problème donné, les sentinelles sont calculés une fois pour toute indépendamment des données. Ces dernières interviennent seulement dans la dernière étape de calcul des paramètres.

Le calcul explicite de la sentinelle W_i , se fait comme suit :

1. - On détermine γ_i tel que :

$$A\gamma_i = e^i$$

2. - On calcule W_i par :

$$W_i = B\gamma_i$$

La méthode des sentinelles permet également de mesurer la sensibilité de chaque paramètre au bruit. En effet, $|W_i|$ est une indication relative à la perturbation par le bruit, de l'estimation.

$$\lambda_i = (W_i, Zd) \quad \Rightarrow \quad |\lambda_i| \leq |W_i| \cdot |Zd|\tag{4-16}$$

VI- Identification des paramètres par la méthode des sentinelles, pour un problème linéaire

La solution $y(t)$, d'un problème linéaire peut être exprimé sous la forme d'une combinaison linéaire de solutions particulières $\psi_i(t)$ ($i = 1, \dots, n_\lambda$), obtenues en prenant à chaque fois un seul paramètre λ_i égale à 1 et tous les autres nuls.

$$y(t) = \sum_{i=1}^{n_\lambda} \psi_i(t) \cdot \lambda_i\tag{4-17}$$

Dans ce cas, et par analogie à ce qui a été décrit précédemment, l'opérateur B sera égale à :

$$B = \begin{bmatrix} \psi_1(t) & \psi_2(t) & \dots & \psi_{n_\lambda}(t) \end{bmatrix} \quad (4-18)$$

et la matrice A est telle que :

$$\begin{aligned} A = B' B &= \begin{bmatrix} \psi_1(t) \\ \psi_2(t) \\ \vdots \\ \psi_{n_\lambda}(t) \end{bmatrix} \begin{bmatrix} \psi_1(t) & \psi_2(t) & \dots & \psi_{n_\lambda}(t) \end{bmatrix} \\ &= \begin{bmatrix} (\psi_1(t), \psi_1(t)) & \dots & \dots & (\psi_1(t), \psi_{n_\lambda}(t)) \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ (\psi_{n_\lambda}(t), \psi_1(t)) & \dots & \dots & (\psi_{n_\lambda}(t), \psi_{n_\lambda}(t)) \end{bmatrix} \end{aligned} \quad (4-19)$$

La matrice A est donc une matrice symétrique de dimension (n_λ, n_λ) , dont les éléments $a_{i,j}$ sont égaux à :

$$a_{i,j} = \int_{t_0}^{t_1} (\psi_i(t), \psi_j(t)) dt \quad 1 \leq i, j \leq n_\lambda \quad (4-20)$$

et le vecteur :

$$b = B' Z d = \begin{bmatrix} \psi_1(t) \\ (\psi_2(t), Z d) \\ \vdots \\ \vdots \\ (\psi_{n_\lambda}(t), Z d) \end{bmatrix}$$

est constitué d'éléments b_i , tels que :

$$b_i = \int_{t_0}^{t_1} (\psi_i(t), Z d) dt \quad 1 \leq i \leq n_\lambda \quad (4-21)$$

L'identification des paramètres par la méthode des sentinelles, pour un problème linéaire, se résume aux étapes suivantes :

1. Déterminer $\{\psi_i(t) \quad (i = 1, \dots, n_\lambda)\}$, l'ensemble des solutions particulières du problème. Elles sont obtenues en résolvant $n\lambda$ fois le problème, en prenant à chaque fois un seul λ_i égale à 1 et tous les autres égaux à 0.
2. Construire le vecteur B
3. Calculer la matrice A en utilisant la relation (4-19).
4. Calculer la matrice γ , en résolvant : $A\gamma = I$ (identité).
5. Calculer les sentinelles W , par la relation $W = \gamma \cdot B'$.
6. En déduire les paramètres à identifier, en utilisant la relation (4-11).

VII- Identification des paramètres par la méthode des sentinelles, pour un problème non linéaire

Pour ce type de problèmes, l'identification se fera de manière itérative par une méthode du type gradient, en utilisant les sentinelles calculées pour le problème linéarisé.

La première étape de l'identification des paramètres, pour un problème non linéaire est donc la définition du modèle linéarisé relatif au problème non linéaire étudié

1- Système linéarisé

Soit $y(t)$ la solution du problème non linéaire, pour les paramètres λ et $y(t) + \hat{y}(t)$ la solution du même problème pour des paramètres égaux à $\lambda + \hat{\lambda}$; $\hat{\lambda}$ étant une petite variation de λ .

Le système linéarisé c'est le système dont la solution $\hat{y}(t)$ représente la perturbation intervenant sur la solution $y(t)$ pour une variation $\hat{\lambda}$ de λ .

Sa solution est une combinaison linéaire des solutions particulières, $\psi_i(t)$ obtenues en prenant à chaque fois un seul élément de $\hat{\lambda}$ égale à 1 et tous les autres nuls.

1° cas : $y(t) = f(t, \lambda)$

pour des paramètres λ , on a :

$$y(t) = f(t, \lambda) \quad (4-21)$$

et pour $\lambda + \hat{\lambda}$ on a :

$$y(t) + \hat{y}(t) = f(t, \lambda + \hat{\lambda}) \quad (4-22)$$

En soustrayant (4-22) à (4-21),

$$y(t) + \hat{y}(t) - y(t) = f(t, \lambda + \hat{\lambda}) - f(t, \lambda) \quad (4-23)$$

on obtient .

$$\hat{y}(t) \cong f_{\lambda}(t, \lambda) \hat{\lambda} \quad (4-24)$$

2° cas : $y'(t) = f(t, y, \lambda) \quad y(0) = \tau$

Si τ est connu, on écrit l'équation pour respectivement, des paramètres λ et $(\lambda + \hat{\lambda})$

$$\lambda \rightarrow y'(t) = f(t, y, \lambda) \quad y(0) = \tau$$

$$\lambda + \hat{\lambda} \rightarrow y'(t) + \hat{y}'(t) = f(t, y + \hat{y}, \lambda + \hat{\lambda}) \quad y(0) + \hat{y}(0) = \tau + \hat{\tau}$$

la soustraction des deux équations donne :

$$\hat{y}'(t) = f(t, y + \hat{y}, \lambda + \hat{\lambda}) - f(t, y, \lambda) \quad \hat{y}(0) = \hat{\tau} \quad (4-25)$$

d'où :

$$\hat{y}'(t) \cong f_y(t, y, \lambda) \hat{y} - f_{\lambda}(t, y, \lambda) \hat{\lambda} \quad \hat{y}(0) = \hat{\tau} \quad (4-26)$$

Les expressions (4.24) et (4.26) représentent les systèmes linéarisés dans chaque cas. Les solutions $\hat{y}(t)$ sont une combinaison linéaire de toutes les fonctions $\psi_i(t, \lambda)$, obtenues pour:

$$\hat{\lambda}_i = e^i \quad i = 1, \dots, n_{\lambda} \quad (4-27)$$

$$\hat{y}(t) = \sum_{i=1}^{n_{\lambda}} \hat{\lambda}_i \psi_i(t, \lambda) \quad (4-28)$$

VII-2 : Sentinelles du système linéarisé

On définit la matrice $A(\lambda)$, dont les éléments:

$$a_{i,j}(\lambda) = \int_{t_0}^{t_1} \psi_i(t, \lambda) \cdot \psi_j(t, \lambda) dt \quad 1 \leq i, j \leq n_{\lambda} \quad (4-29)$$

La sentinelle W associée aux paramètres inconnus est calculée en résolvant successivement les équations (4-30) et (4-31).

$$A(\lambda) \gamma(\lambda) = e^k \quad (4-30)$$

$$W(\lambda) = \psi(\lambda) \gamma(\lambda) \quad (4-31)$$

e^k : k-ème vecteur unité.

Ensuite, les paramètres sont ensuite mis à jour en faisant:

$$\lambda^{k+1} = \lambda^k - W(\lambda^k) (\gamma(\lambda^k) - Z d) \quad k = 1, \dots \quad (4-32)$$

Lorsque τ est inconnu, le vecteur des paramètres à identifier n'est plus λ mais $v = \{ \lambda, \tau \}$ et $\psi(t, \lambda)$ est un vecteur de dimension $(n + n_r)$.

L'algorithme est identique au précédent, il suffit de remplacer λ et $\hat{\lambda}$ par v et \hat{v} .

VIII- Application de la méthode des sentinelles pour l'identification des paramètres pour différents problèmes

On résout les problèmes suivants, successivement par la méthode des sentinelles puis la méthode de continuation, et cela pour comparer les performances des deux méthodes.

Problème n°1

$$y(t) = m_1 e^{-\frac{t-\theta_1}{\sigma_1}} + m_2 e^{-\frac{t-\theta_2}{\sigma_2}}$$

$y(t)$ représente la somme de deux gaussiennes que l'on désire identifier. Pour cela, on dispose des données du tableau n° (4-1)

t	284	284.25	284.5	284.75	285	285.25	285.5	285.75	286	286.25	286.5	286.75	287
y(t)	0.07	0.15	0.3	0.6	1.1	1.7	2.4	3.2	3.8	4.1	4	3.7	3.1

t	287.25	287.5	287.75	288	288.25	288.5	288.75	289	289.25	289.5	289.75	290
y(t)	2.5	1.9	1.4	1	0.7	0.6	0.4	0.3	0.15	0.1	0.05	0.03

t	290.25	290.5	290.75	291
y(t)	0.002	0.0013	0	0

tableau n° 4-1

Partant des valeurs initiales $[m_1, \theta_1, \sigma_1, m_2, \theta_2, \sigma_2]^0 = [4, 286, 1, 1, 288, 1]$, on obtient les résultats représentés dans le tableau n°(4-2). On remarque que la méthode des sentinelles converge plus rapidement et donne un écart quadratique plus faible que la méthode de continuation.

Algorithme	m_1	θ_1	σ_1	m_2	θ_2	σ_2	Niter	$\ \varphi\ $
Continuation	3.9176	286.312	1.1590	0.507	287.703	1.350	18	0.0134
Sentinelles	3.9436	286.298	1.1302	0.641	287.750	1.410	10	0.0074

tableau n° 4-2

Avec les paramètres obtenus par la méthode des sentinelles, on calcule les deux courbes déconvoluées : $y_1(t) = m_1 e^{-\left|\frac{t-\theta_1}{\sigma_1}\right|^2}$ et $y_2(t) = m_2 e^{-\left|\frac{t-\theta_2}{\sigma_2}\right|^2}$ que l'on représente sur la figure n°(4-1). Les données sont représentées sur la même figure par des (*).

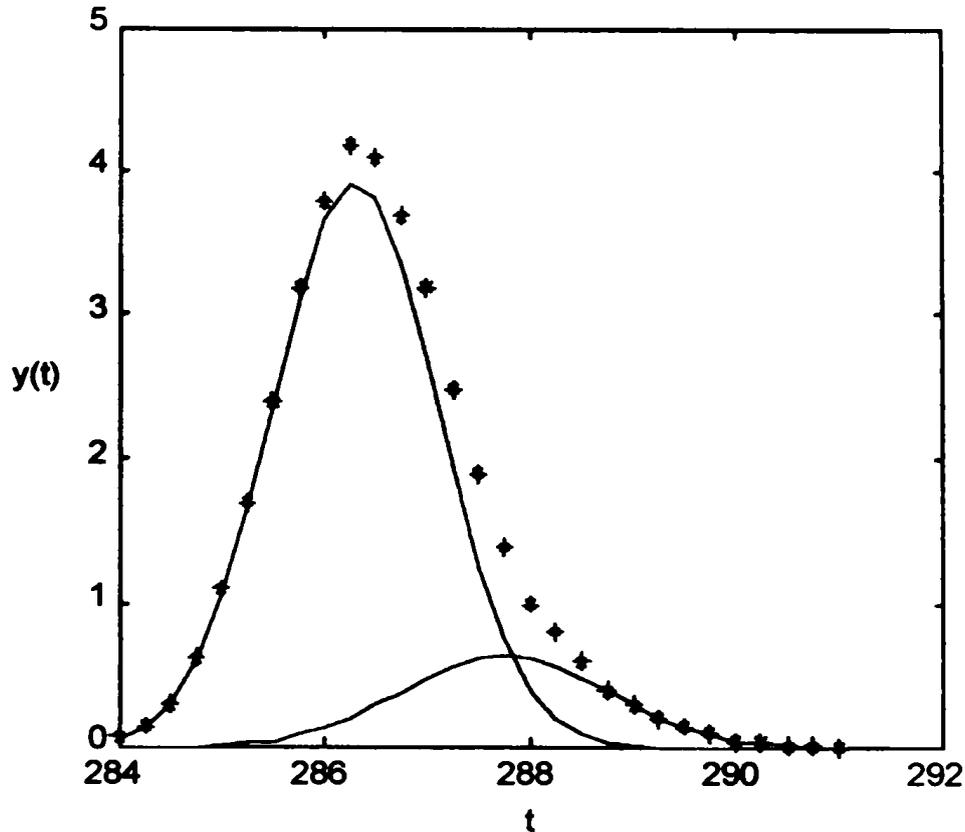


fig. n°4-1 : Courbes déconvoluées obtenues avec la méthode des sentinelles

Problème n°2:

$$y_1' = y_2$$

$$y_2' = \varepsilon(1 - y_1^2)y_2 - y_1$$

Ces équations, plus connues, sous le nom d'équations de Van-der-pol, sont souvent rencontrées en électronique. Le paramètre à identifier ε , désigne la caractéristique non linéaire du circuit.

t	0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5
$y_1(t)$	1.9	1.5	0.9	0.2	-0.5	-1.1	-1.4	-1.2	-0.8	-0.3	0.2
$y_2(t)$	0.1	-0.8	-1.3	-1.5	-1.4	-1	-0.25	0.3	0.8	1.1	1.2

tableau n° 4-3

Pour les données du tableau n°(4-3) et en prenant comme valeur initiale $\varepsilon^0 = 0.5$, on obtient les résultats représentés dans le tableau n°(4-4). Les conditions initiales $y_1(0)$ et $y_2(0)$ sont considérées connues, égales respectivement à 1.9 et 0.1.

Algorithme	ϵ	$y1(0)$	$y2(0)$	Niter	$\ \varphi\ $
continuation	0.0341	1.9	0.1	3	0.108
sentinelles	0.0337	1.9	0.1	3	0.108

tableau n° 4-4

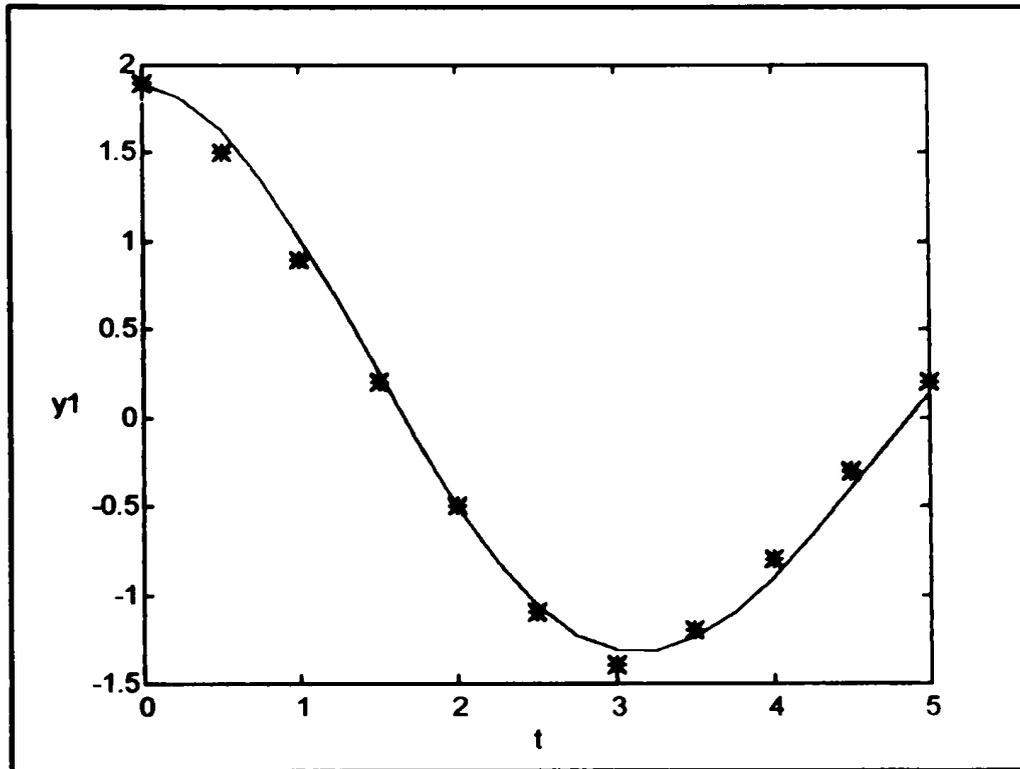


fig. 4-2. $y_1(t)$ obtenue après identification, en comparaison avec les données.

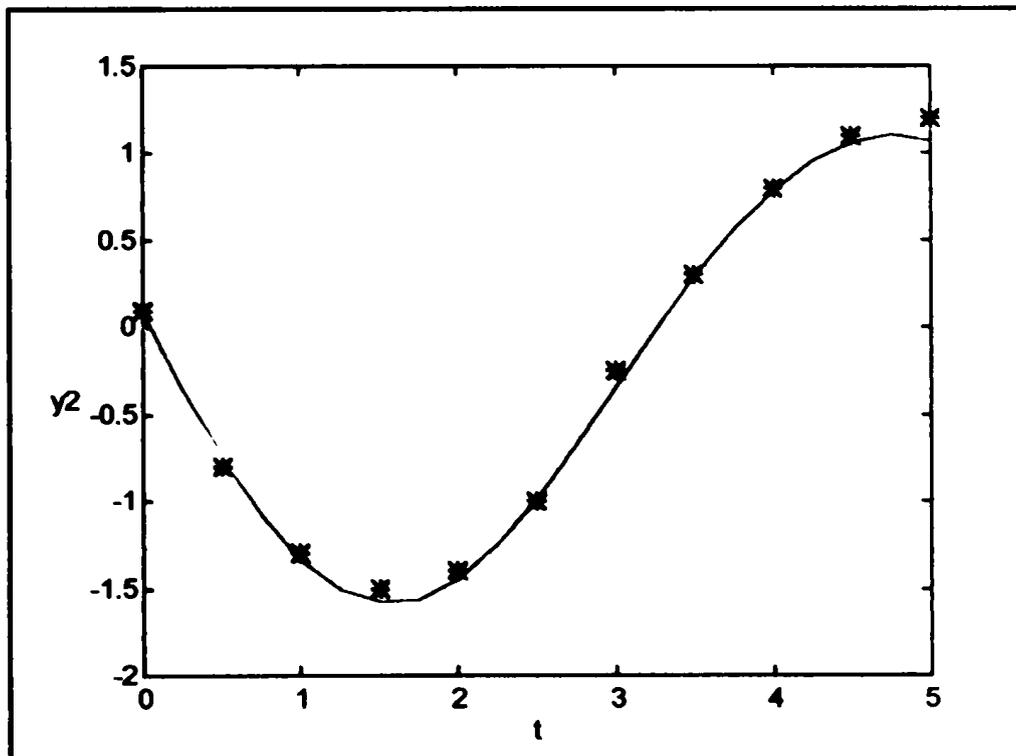


fig. 4-2 : $y_2(t)$ obtenue après identification en comparaison, avec les données.

Les résultats obtenus avec chacune des deux méthodes sont comparables. Les paramètres obtenus ainsi que le nombre d'itérations nécessaires à la convergence des algorithmes, sont les mêmes

Les courbes obtenues avec les algorithmes de continuation et des sentinelles sont représentées en trait plein sur les figures (4-2-a) et (4-2-b) alors que les données sont représentées par des (*) sur les mêmes figures.

Problème n°3:

$$y_1' = p_1 (126.2 - y_1)(91.9 - y_1)^2 - p_2 y_1^2$$

Ce problème également traité dans le chapitre 2, a été résolu par la méthode de continuation puis la méthode des sentinelles. La méthode de continuation et pour $[p_1, p_2]^0 = [0.1, 0.1]$, a donné les résultats représentés dans le tableau n°(4-5), alors que la méthode des sentinelles diverge quelque soit le choix des valeurs initiales.

La condition initiale de y_1 est prise égale à la première donnée relevée c'est à dire $y_1(1)$.

Algorithme	p_1	p_2	$y_1(1)$	Niter	$\ \varphi\ $
continuation	$0.4797 \cdot 10^{-5}$	$0.2591 \cdot 10^{-3}$	0	5	19.8158

tableau n° 4-5

Lorsqu'on remplace, dans l'expression de y_1' , p_1 par $10^{-5} pp_1$ et p_2 par $10^{-3} pp_2$, les deux méthodes convergent (tableau n° 4-6). Ceci laisse supposer que la méthode des sentinelles est très sensible à l'échelle de mesure des paramètres, alors que la méthode de continuation l'est moins.

Algorithme	pp_1	pp_2	$y_1(1)$	Niter	$\ \varphi\ $
continuation	0.4797	0.2591	0	5	19.8158
sentinelles	0.4982	0.3228	0	8	24.1851

tableau n° 4-6

On remarque également, que la méthode de continuation donne les mêmes résultats que précédemment et que ces derniers sont meilleurs que ceux obtenus avec la méthode des sentinelles.

D'ailleurs les valeurs numériques des paramètres obtenus par la méthode de continuation, sont très proches de ceux obtenus par la méthode des B-splines, dans le chapitre 2.

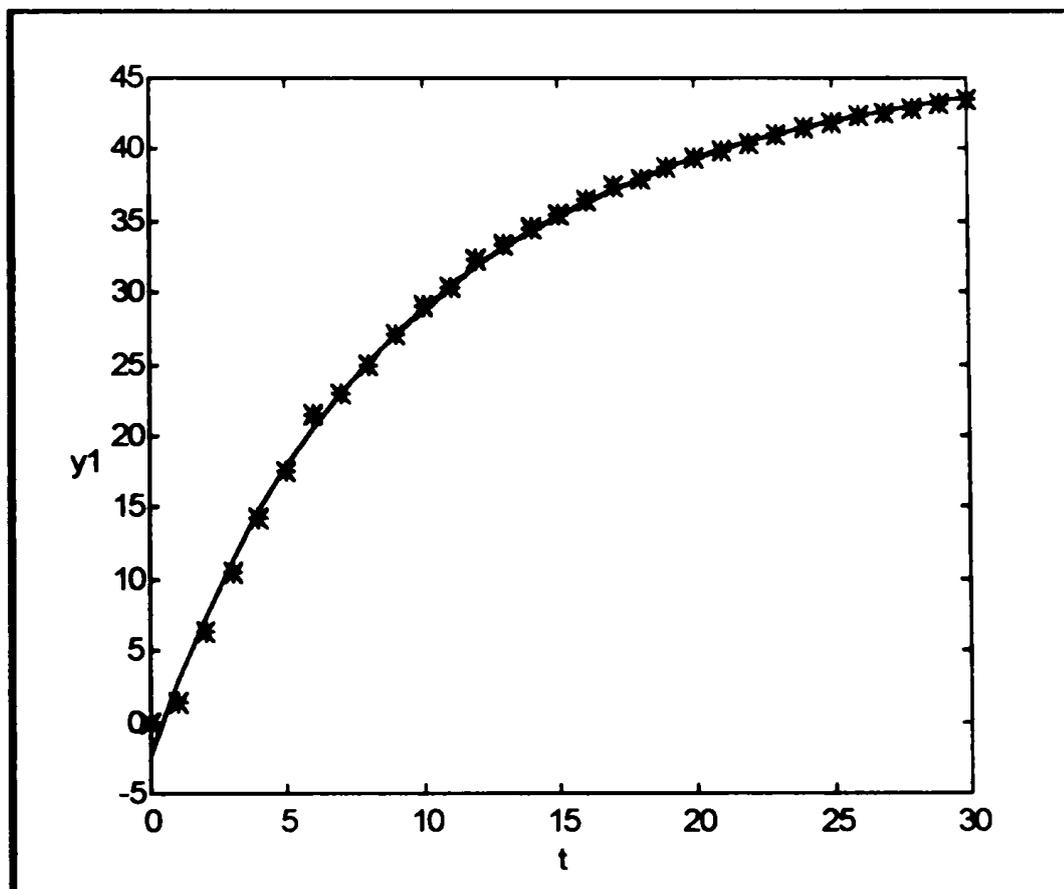


fig. n°4-3-a: Courbes obtenues après identification par la méthode des sentinelles

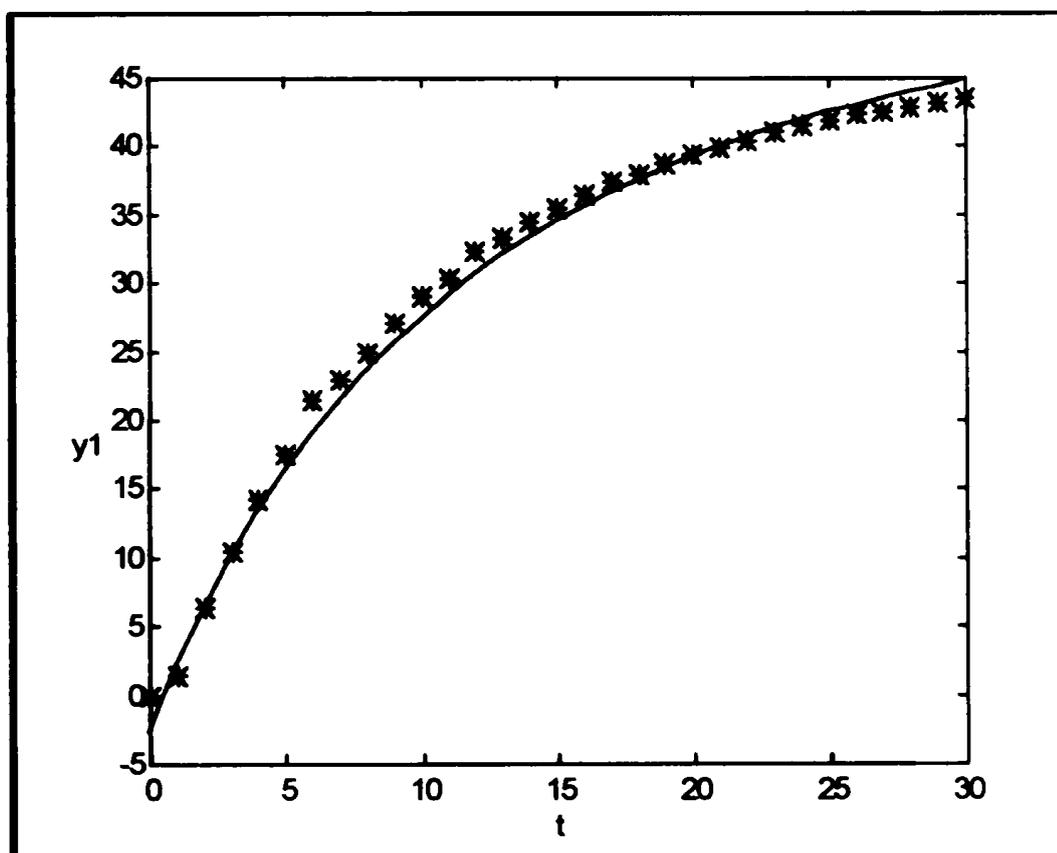


fig. n°4-3-b: Courbes obtenues après identification par la méthode de continuation

Problème n°4:

$$y_1' = p_1 y_1 - p_2 y_1 y_2$$

$$y_2' = p_2 y_1 y_2 - p_3 y_2$$

Ce problème a également été traité dans le chapitre 2, avec des données légèrement différentes de celle que l'on a utilisées ici, et qui sont représentées dans le tableau n°(4-7).

Le nombre de paramètres à identifier dans ce cas, est plus important que dans les exemples précédents (p_1, p_2, p_3).

t	0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5
y1(t)	1	1.1	1.2	1.1	0.9	0.7	0.6	0.6	0.7	0.8	1
y2(t)	0.2	0.25	0.4	0.5	0.5	0.45	0.35	0.3	0.25	0.25	0.3

tableau n° 4-7

Les résultats obtenus, avec chacune des deux méthodes (tableau n°4-8) sont pratiquement les mêmes. Les écarts quadratiques correspondant sont également assez proches.

$$[p_1, p_2, p_3]^0 = [1, 2, 1.5]$$

Algorithme	p_1	p_2	p_3	$y_1(0)$	$y_2(0)$	Niter	$\ \varphi\ $
méthode de continuation	0.8066	2.2652	1.9573	1	0.2	3	0.0304
méthode des sentinelles	0.8071	2.2662	1.9598	1	0.2	3	0.0314

tableau n° 4-8

En comparaison, avec les résultats obtenus avec les B-splines, on remarque que les valeurs des paramètres sont très voisins de ceux obtenus avec 1noeud (position 3).

Les figures (4-4) et (4-5) montrent les courbes $y_1(t)$ et $y_2(t)$ obtenues par identification avec chacune des deux méthodes et leur représentation par rapport aux données.

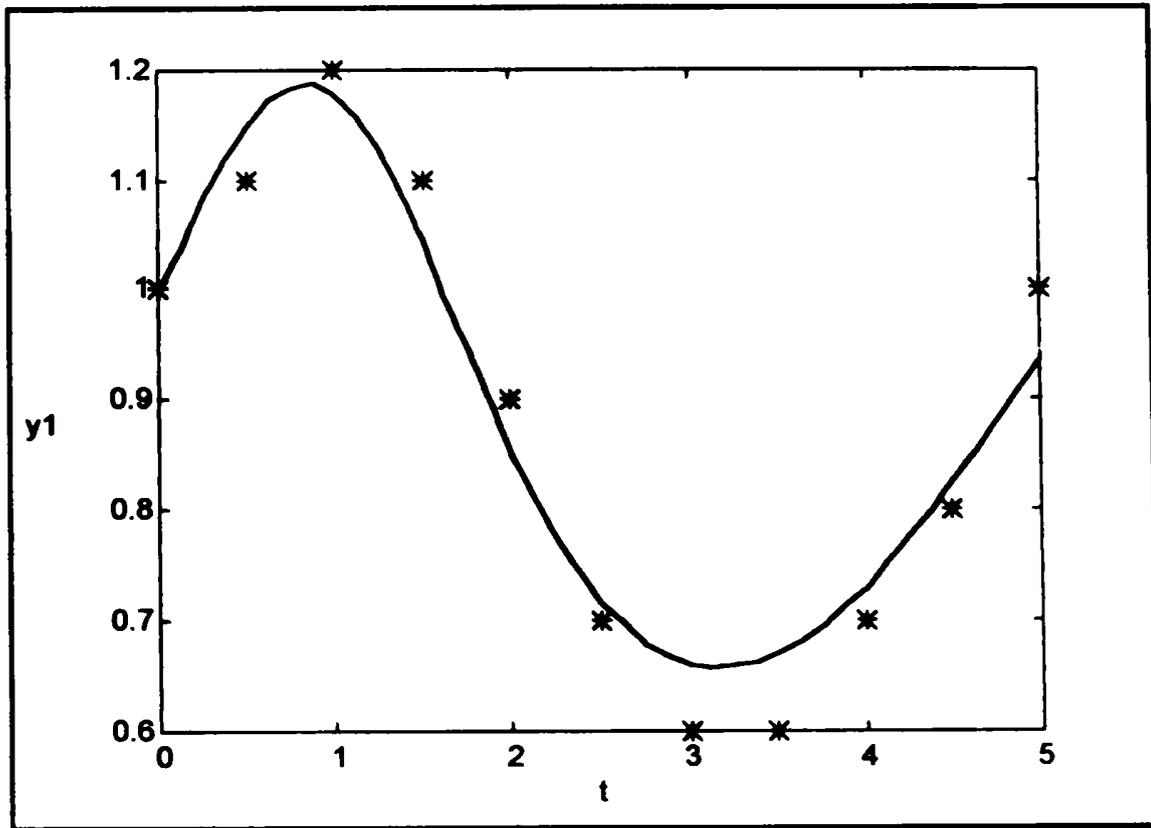


fig. n°4-4-a : $y_1(t)$ obtenue après identification par la méthode de continuation en comparaison avec les données

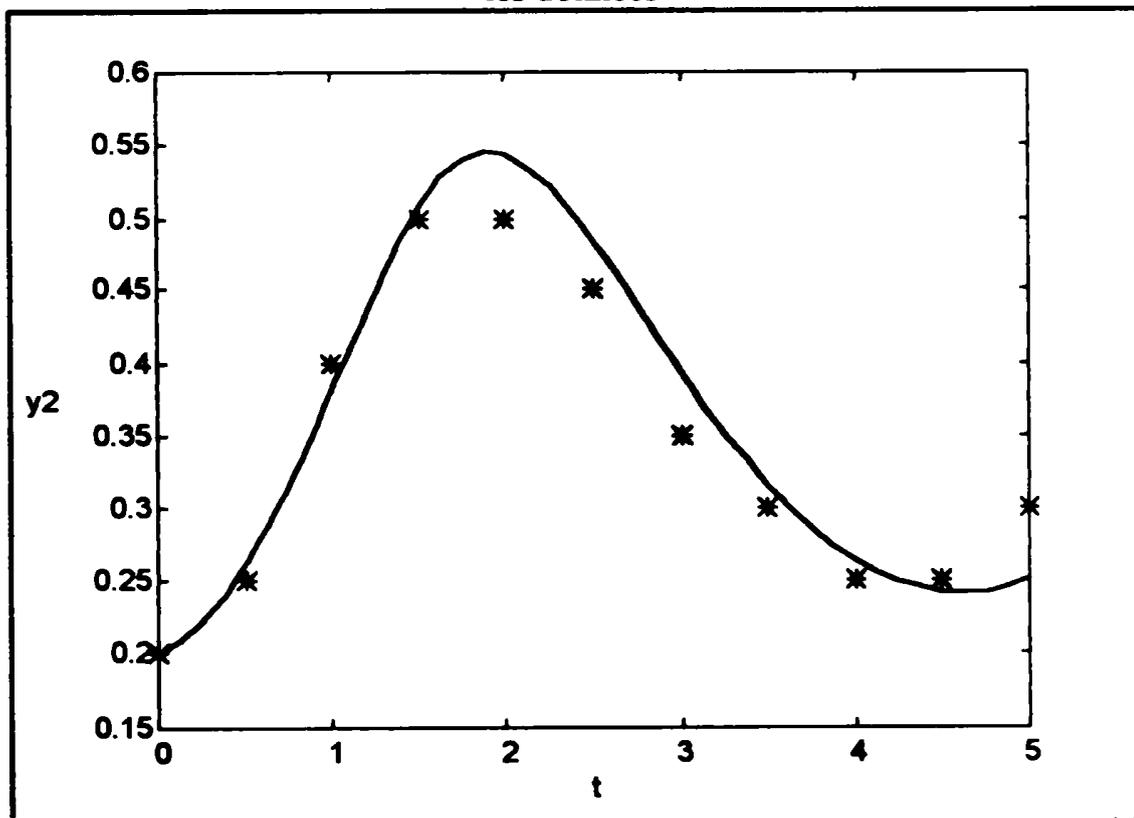


fig. n°4-4-b : $y_2(t)$ obtenue après identification par la méthode de continuation en comparaison avec les données

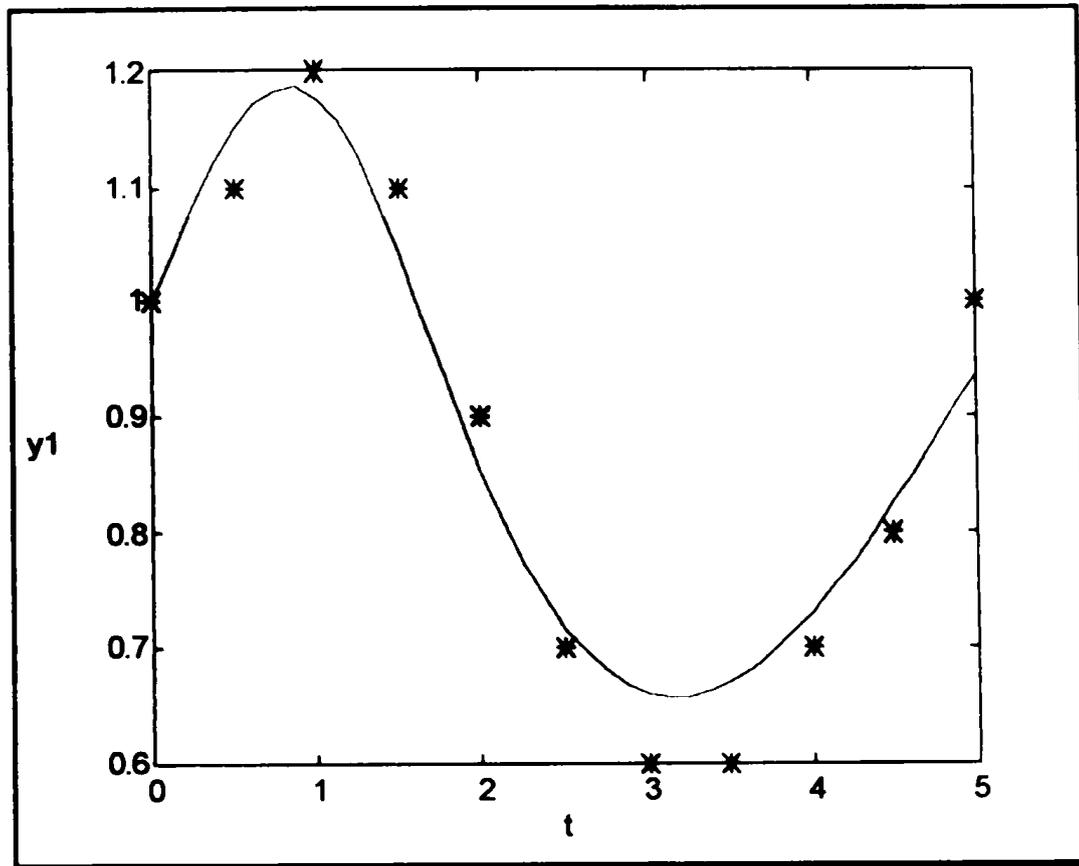


fig. n°4-5-a : $y_1(t)$ obtenue après identification par la méthode des sentinelles en comparaison avec les données

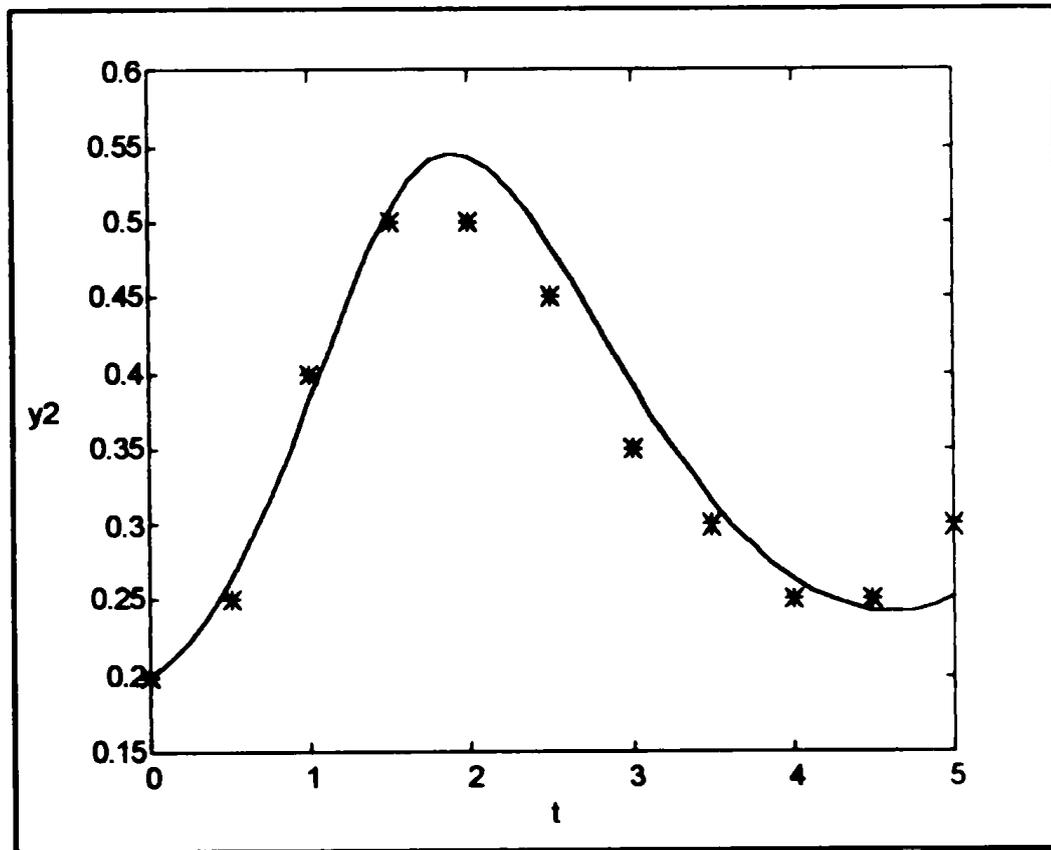


fig. n°4-5-b : $y_2(t)$ obtenue après identification par la méthode des sentinelles en comparaison avec les données

IX- Identification des paramètres par application successive des algorithmes

Lorsque le problème d'identification porte sur un système, modélisé par un système d'équations différentielles de la forme:

$$\begin{aligned} y_1'(t) &= f_1(t, y, \lambda) \\ &\vdots \\ y_n'(t) &= f_n(t, y, \lambda) \end{aligned}$$

Les conditions initiales $\{y_1(0), y_2(0), \dots, y_n(0)\}$ peuvent être considérées comme :

- égales aux mesures $Zd(t_1)$, prises à $t=t_1$. Ce cas peut être envisagé lorsque que les données sont très fiables et représentent parfaitement le modèle.
- ou alors inconnues. Là on est sûr d'avoir des résultats plus exacts ; le temps de calcul sera plus long puisque, le nombre de paramètres à identifier est plus important.

En réalité, il est très rare que l'on connaisse, à l'avance, la part de l'erreur sur la mesure. Par conséquent, la formulation du problème, en considérant systématiquement les conditions initiales comme inconnues, entraîne des calculs supplémentaires qui ne sont peut être pas nécessaires, alors que dans le cas contraire les résultats peuvent être incohérents du fait de l'erreur sur la mesure.

L'alternative pour résoudre ce problème, serait de procéder à une identification en deux étapes : dans un premier temps, on identifie seulement les paramètres $\lambda = \{\lambda_1, \lambda_2, \dots, \lambda_{n_i}\}$, les conditions initiales sont prises égales à $Zd(t_1)$; ensuite, les conditions initiales sont supposées inconnues et l'identification est reprise pour déterminer aussi bien $\lambda = \{\lambda_1, \lambda_2, \dots, \lambda_{n_i}\}$ que $\{y_1(0), y_2(0), \dots, y_n(0)\}$. Si les données sont fortement bruitées, les paramètres obtenus lors de la première étape sont alors corrigés.

Dans ce qui suit, nous mettons en évidence l'intérêt d'une identification en deux étapes, lorsque le système est modélisé par des équations différentielles en comparant les performances des méthodes de continuation et des sentinelles. Pour cela, on a repris l'identification des paramètres pour les problèmes précédents:

- 1°- en considérant au départ, que les conditions initiales sont inconnues
- 2°- en procédant en deux étapes

Les résultats obtenus sont présentés dans ce qui suit. On désignera par MC1 l'algorithme de continuation pour le premier cas (conditions initiales inconnues du départ) et par MC2 le même algorithme pour le deuxième cas (identification en deux étapes).

De même, nous noterons l'algorithme des sentinelles par SENT1 pour l'identification des conditions initiales du départ, et par SENT2 pour l'identification en deux étapes.

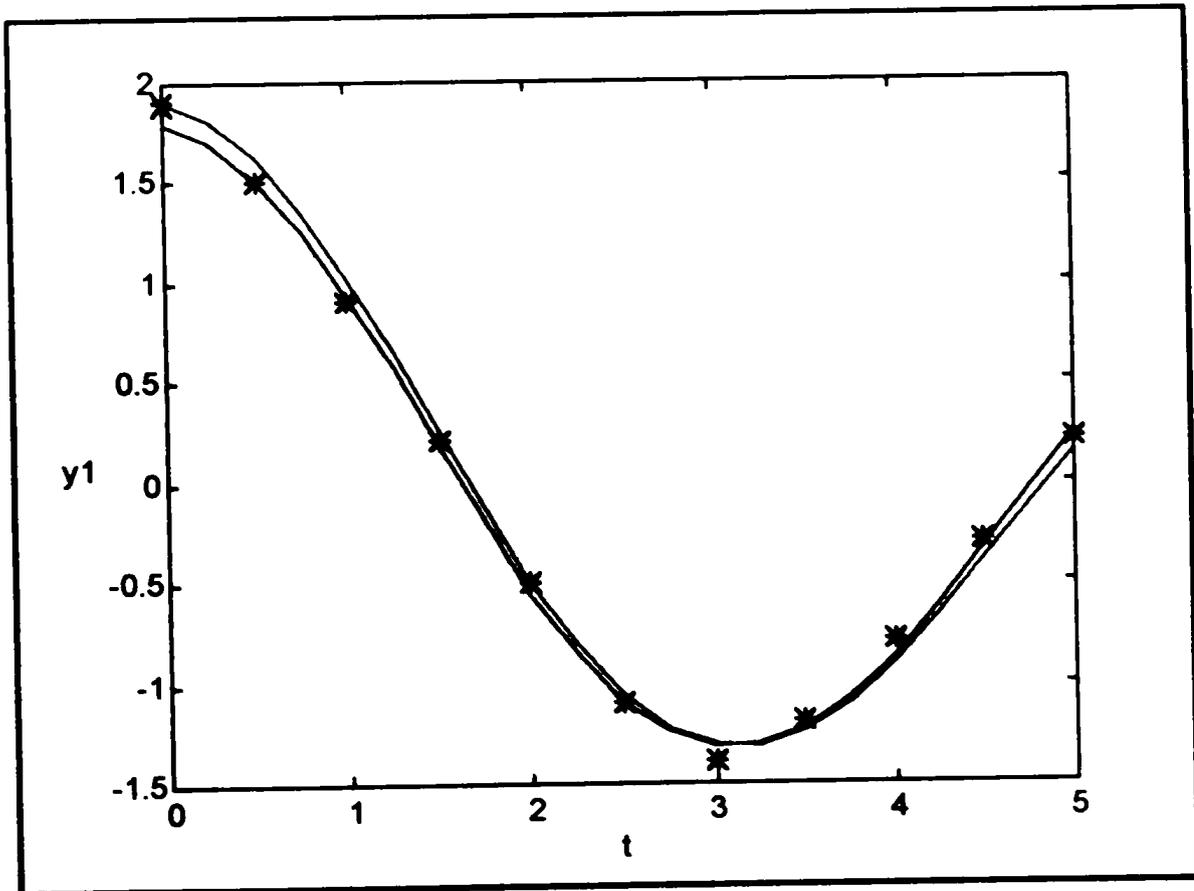
Problème n°2:

Pour les données du tableau n°4-3 et en prenant comme valeur initiale $\varepsilon^0 = 0.5$, on obtient les résultats représentés dans le tableau suivant :

	Algorithme	ε	$y_1(0)$	$y_2(0)$	Niter	$\ \varphi\ $
méthode de continuation	MC1	0.0805	1.8372	0.0279	4	0.0364
	MC2	0.0793	1.8375	0.0282	(3)+2	0.0360
méthode des sentinelles	SENT1	0.0802	1.8296	0.028	5	0.0365
	SENT2	0.0801	1.8298	0.028	(3)+3	0.0365

tableau n°4-9

Les résultats obtenus avec les deux méthodes sont comparables. Les paramètres obtenus, en considérant du départ les conditions initiales comme inconnues, sont très proches de ceux obtenus avec une identification en deux étapes. Le plus petit écart quadratique est tout de même obtenu avec l'algorithme MC2.


 fig. 4-5-a : $y_1(t)$ obtenue après identification en comparaison avec les données.

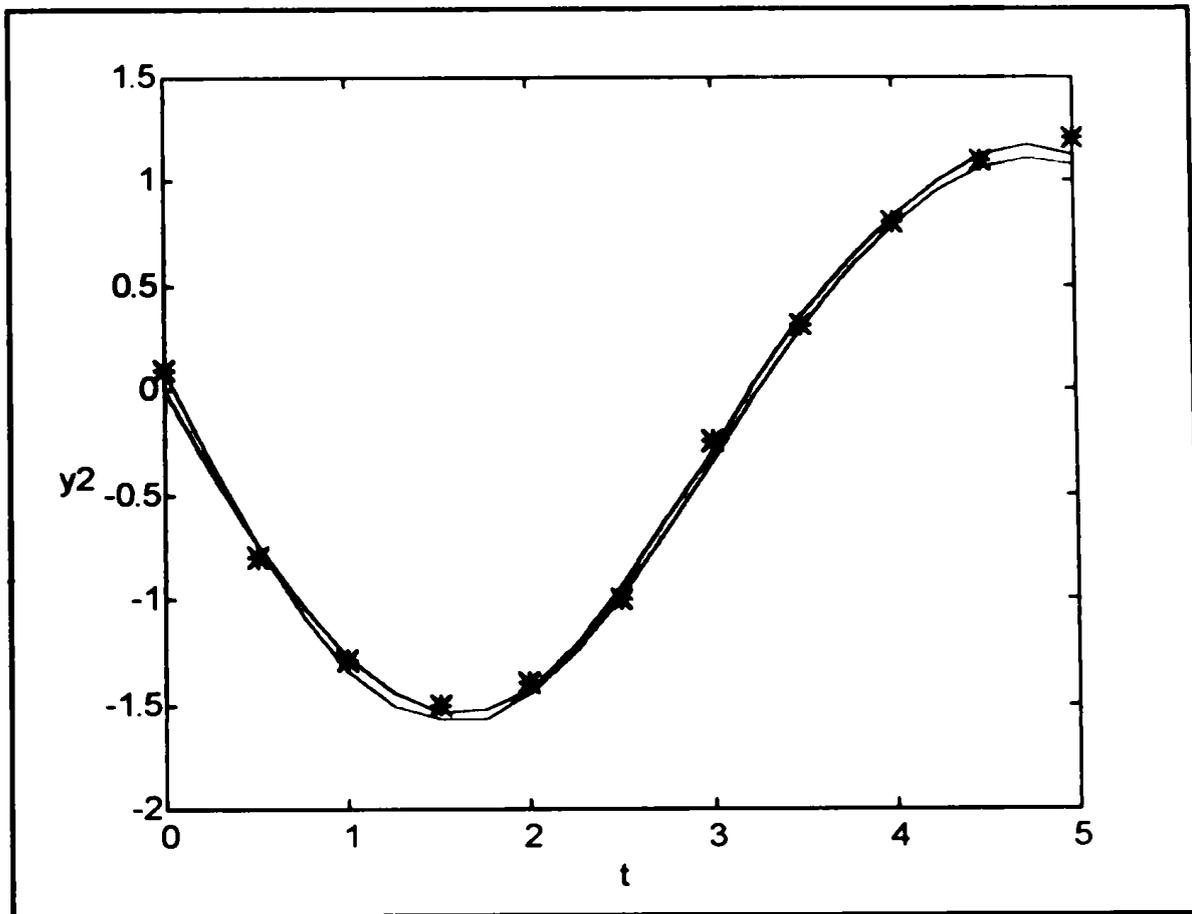


fig. 4-5-b : $y_2(t)$ obtenue après identification en comparaison avec les données.

Les courbes en bleu, des figure (4-5-a) et (4-5-b), sont celles obtenues en considérant les conditions initiales égales aux premières mesures; alors que celles qui sont en traits pleins correspondent aux résultats des algorithmes MC1 et SENT1. Ces dernières sont confondues avec les courbes obtenues en faisant l'identification en deux étapes (MC2 et SENT2) Les données sont représentées par des (*) sur les figures.

Problème n° 3:

Comme précisé précédemment la méthode des sentinelles diverge pour ce problème, si on considère directement p_1 et p_2 . La méthode de continuation pour sa part, donne les résultats suivants :

	Algorithme	p_1	p_2	$y_1(1)$	Niter	$\ \varphi\ $
méthode de continuation	MC1	$0.5216 \cdot 10^{-5}$	$0.3493 \cdot 10^{-3}$	-2.1379	6	10.3834
	MC2	$0.5210 \cdot 10^{-5}$	$0.3482 \cdot 10^{-3}$	-2.1179	(5)+3	10.3730

tableau n°4-10

Par contre, lorsqu'on remplace, p_1 par $10^{-5} pp_1$ et p_2 par $10^{-3} pp_2$, on obtient

	Algorithme	pp_1	pp_2	$y_1(t)$	Niter	$\ \varphi\ $
méthode de continuation	MC1	0.5216	0.3493	-2.1379	6	10.3834
	MC2	0.5210	0.3482	-2.1179	(5)+3	10.3730
méthode des sentinelles	SENT1	0.5348	0.3953	-2.3759	9	11.6240
	SENT2	0.5349	0.3954	-2.3747	(8)+3	11.6536

tableau n°4-11

On remarque que la méthode de continuation donne de meilleurs résultats et le fait de considérer les conditions initiales inconnues permet d'affiner la qualité de l'identification.

L'écart quadratique le plus faible est obtenu avec l'algorithme MC2, et les résultats obtenus dans ce cas (valeurs des paramètres et condition initiales), sont plus précis que ceux obtenus par la méthode des B-splines.

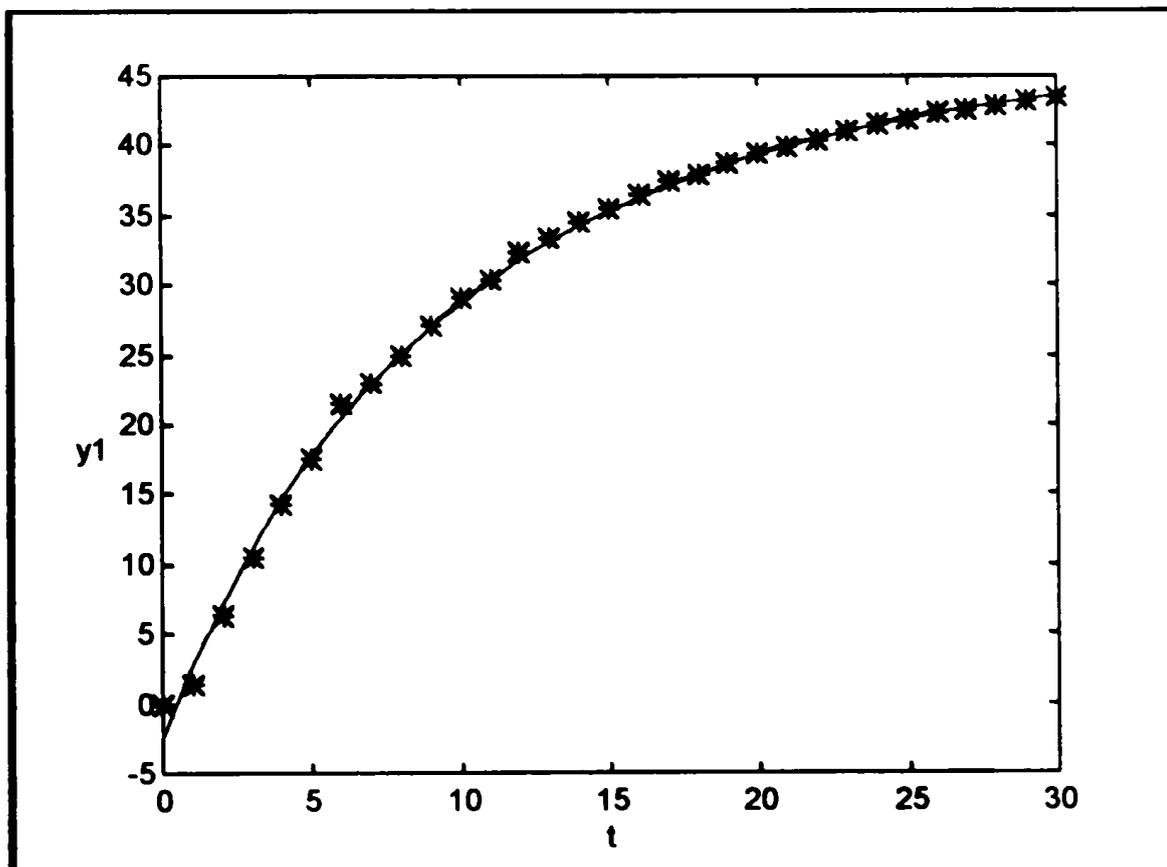


fig. n°4-5-a : Courbes obtenues après identification par la méthode des sentinelles

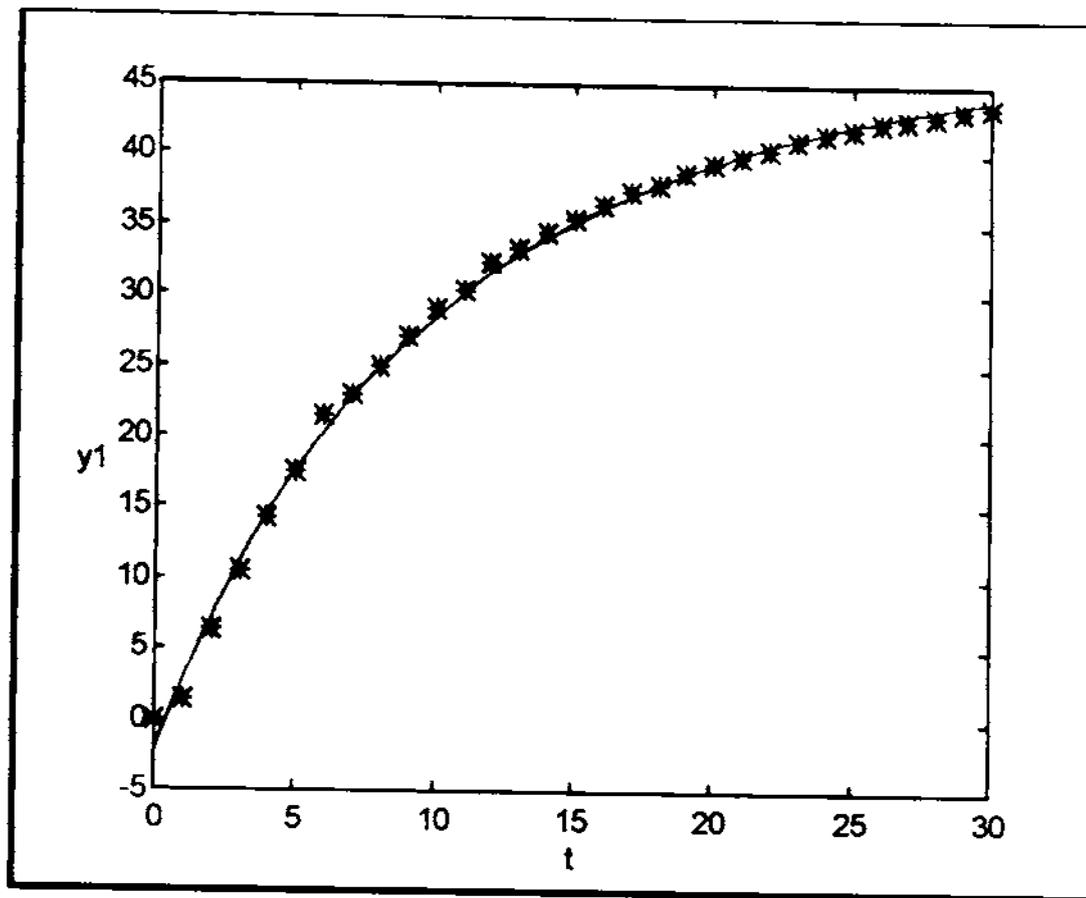


fig. n°4-6-b : Courbes obtenues après identification par la méthode de continuation

Problème n°4

L'examen des résultats obtenus, avec chaque méthode (tableau n° 4-12) fait ressortir une légère différence entre les paramètres obtenus avec les algorithmes MC1 et SENT1 alors que les écarts quadratiques correspondant sont assez proches. Les autres résultats sont assez proches

L'écart quadratique le plus faible, comme pour les exemples précédents, correspond à l'algorithme en deux étapes MC2.

	Algorithme	p_1	p_2	p_3	$y_1(0)$	$y_2(0)$	Niter	$\ J\ $
méthode de continuation	MC1	0.9068	2.5476	2.1223	0.8838	0.1698	5	0.0236
	MC2	0.8462	2.4270	2.0609	0.9345	0.1746	(3)+3	0.0216
méthode des sentinelles	SENT1	0.8627	2.4687	2.0917	0.9191	0.1738	4	0.0230
	SENT2	0.8627	2.4688	2.0917	0.9191	0.1738	(3)+2	0.0230

tableau n°4-12

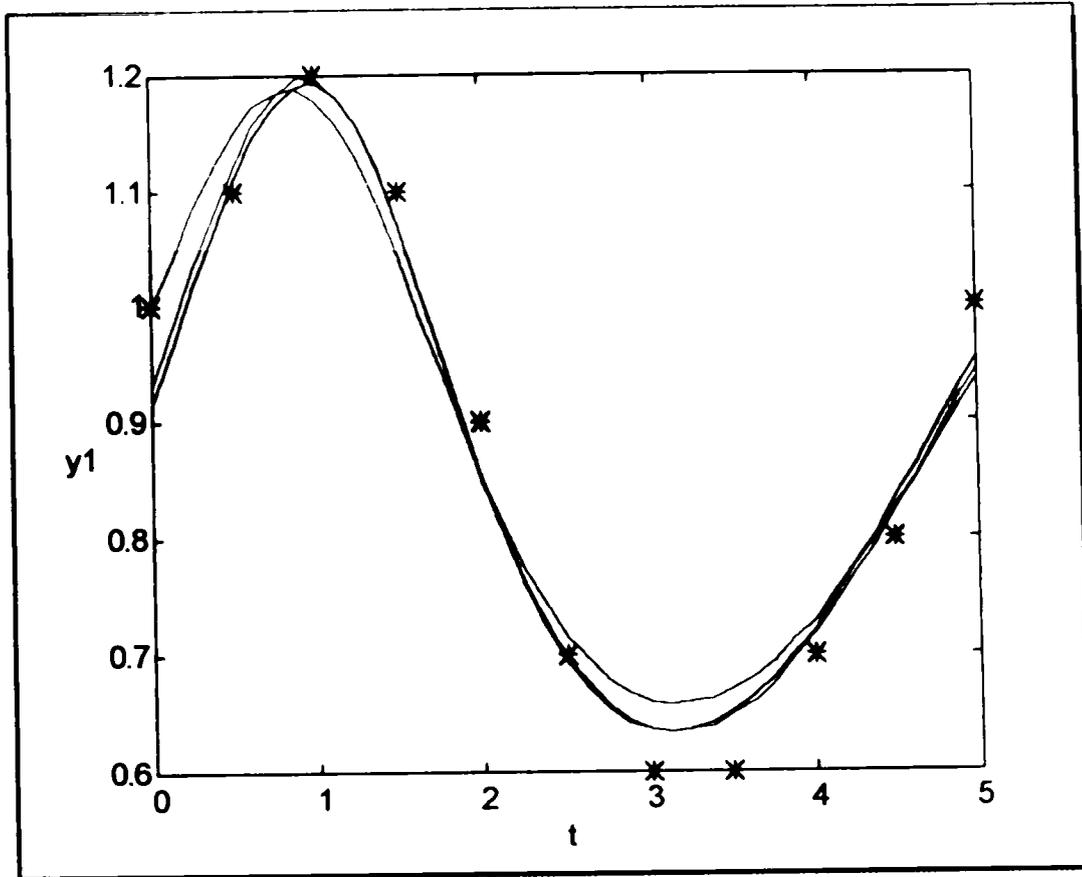


fig. n°4-7-a. Courbe $y_1(t)$ obtenue après identification par la méthode de continuation en comparaison avec les données

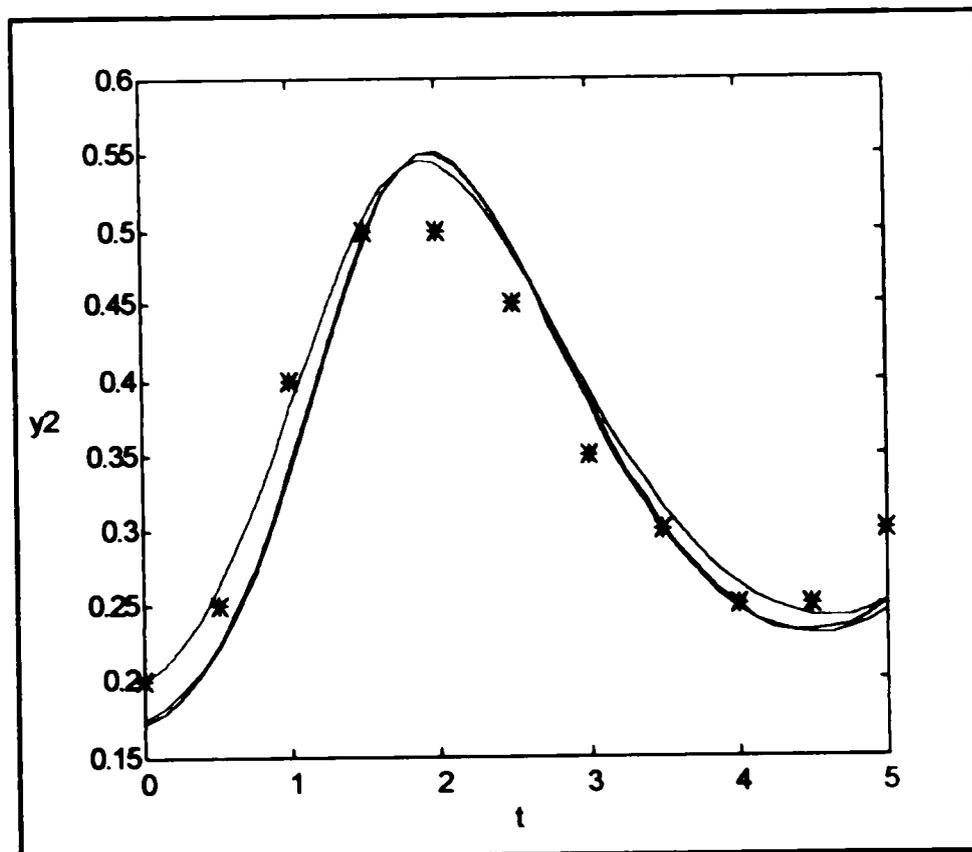


fig. n°4-7-b. Courbe $y_2(t)$ obtenue après identification par la méthode de continuation en comparaison avec les données

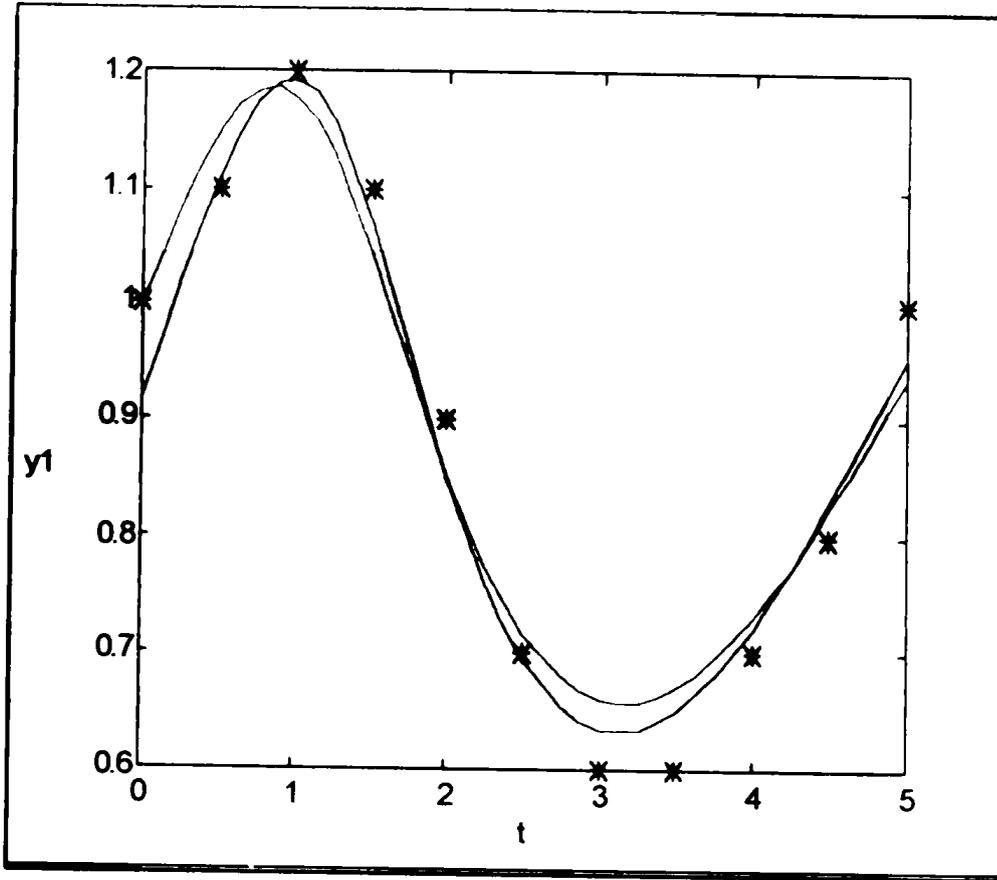


fig. n°4-8-a : Courbe $y_1(t)$ obtenue après identification par la méthode des sentinelles en comparaison avec les données

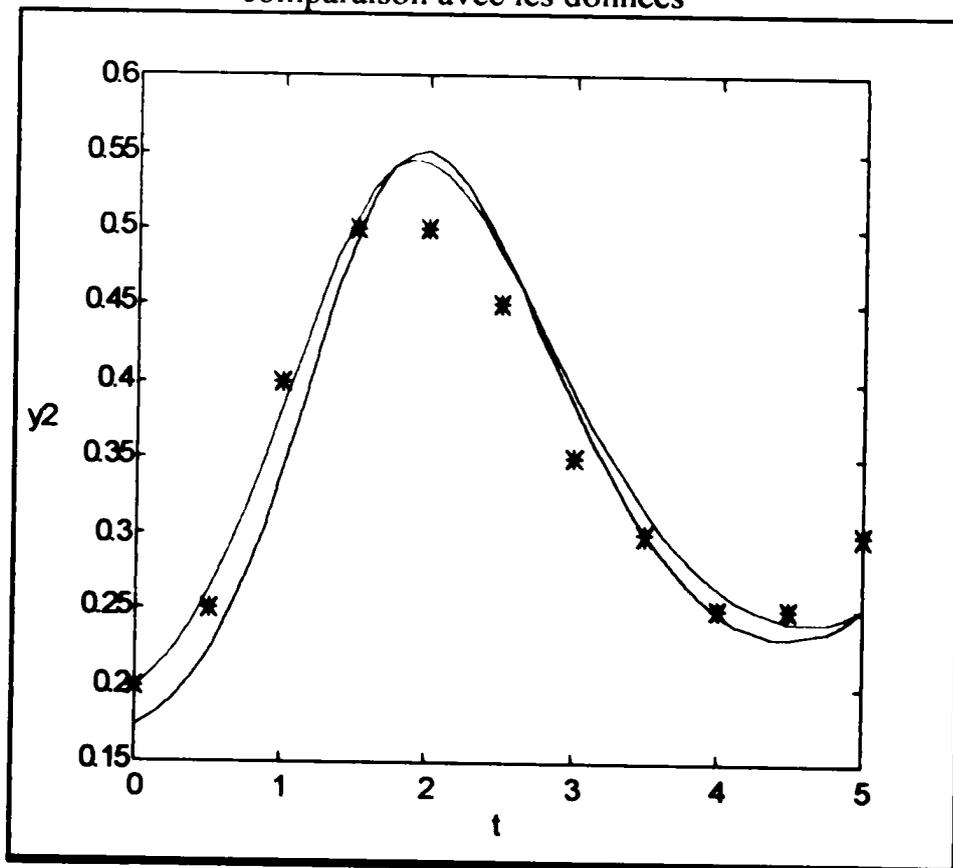


fig. n°4-8-b: Courbe $y_2(t)$ obtenue après identification par la méthode des sentinelles en comparaison avec les données

Les courbes en rouge et en vert, des figures (4-7-a) et (4-7-b) représentent $y_1(t)$ et $y_2(t)$ obtenues par identification avec chacune des deux algorithmes MC1 et MC2.

Pour les algorithmes SENT1 et SENT2, ces deux courbes sont confondues, elles sont représentées en rouge sur les figures (4-8-a) et (4-8-b)

Toutes les courbes sont représentées par rapport aux données et aux courbes obtenues en considérant les conditions initiales connues (courbes en bleu).

On remarque que la courbe $y_1(t)$ est beaucoup plus proche des données, que la courbe $y_2(t)$ et ceci quelque soit la méthode ou la formulation utilisée

En conclusion, on peut dire que la méthode des sentinelles est très performante pour l'identification des paramètres dans les problèmes de moindres carrés, lorsque le modèle est une équation non linéaire. Par contre lorsqu'on a à faire à un système d'équations différentielles ordinaires la méthode de continuation et la méthode des sentinelles ont des performances très proches (à l'exception du premier cas de l'exemple 3). On remarquera tout de même que l'écart quadratique le plus faible est dans les trois cas obtenu avec l'algorithme MC2, c'est à dire en faisant une identification par étapes et en utilisant la méthode de continuation.

De plus, le nombre d'itérations nécessaires pour une identification en deux étapes n'est pas beaucoup plus élevé que dans le cas d'une formulation directe. La première étape nécessite la majorité des itérations, cette étape est justement celle qui manipule le moins de paramètres, donc la moins coûteuse en temps calcul.

CONCLUSION GENERALE

Le travail développé dans cette thèse comporte plusieurs parties qui gravitent toutes autour des problèmes d'identification. Pour chacune d'elles, on a étudié les aspects algorithmiques et numériques des différentes méthodes que l'on propose d'utiliser. Ces méthodes générales peuvent être utilisées pour la résolution d'un grand nombre de problèmes d'identification, relatifs à des domaines différents. Elles ont l'avantage de ne pas être compliquées du point de vue algorithmique et d'être en même temps efficaces. Elles peuvent fonctionner avec des moyens informatiques de capacité moyenne, car l'espace mémoire requis par la majorité des algorithmes présentes est relativement faible. Les temps de calcul nécessaires sont également réduits, même lorsqu'on procède par étapes ou par application successive des algorithmes.

La méthode des moindres carrés est la formulation la plus courante d'un problème d'identification. Les méthodes présentées dans cette thèse représentent les différentes approches que l'on peut utiliser pour résoudre ce type de problèmes.

Les méthodes de résolution du type gradient, sont les plus connues et les plus anciennes. Nous avons montré que d'une part, l'algorithme de Marquardt était le plus performant et le plus indiqué pour ces problèmes et que d'autre part, le problème du choix des valeurs initiales de tout algorithme itératif peut être résolu, en combinant l'algorithme utilisé à une méthode de continuation.

L'interpolation des données par des fonctions B-splines cubiques, est également une méthode que l'on peut utiliser pour identifier aussi bien les paramètres inconnus que les conditions initiales d'un système. Toutefois, il n'existe pas de technique rigoureuse susceptible de donner la meilleure interpolation. Plusieurs essais sont souvent nécessaires, avant de faire un choix. Ce dernier se base, essentiellement sur l'appréciation visuelle des différentes courbes obtenues. La détermination, de la position des noeuds ainsi que leur nombre en utilisant une procédure d'optimisation, donne de mauvais résultats, surtout lorsque les données sont bruitées.

La méthode de continuation est une autre méthode que l'on peut utiliser pour l'identification des paramètres puisque comme nous l'avons montré, un lien existe entre l'optimisation d'un système et la continuation de ses solutions. Cette méthode très intéressante, permet d'obtenir avec précision le comportement d'un système en fonction des paramètres inconnus. On peut également procéder par continuation successive. Cette technique permet un suivi de l'évolution de la solution ainsi qu'un gain en temps de calcul. La méthode de continuation ne s'applique pas seulement aux problèmes de moindres carrés. La diversité des problèmes présentés dans cette thèse, montrent que les applications sont nombreuses.

En fin, la méthode des sentinelles présente l'intérêt d'une méthode qui nécessite un seul calcul des sentinelles pour un problème donné et pour différents jeux de données. Ces sentinelles permettent de mesurer la sensibilité de chaque paramètre au bruit dont sont entachées les données. Toutefois, pour l'application de cette méthode, on doit obligatoirement vérifier les conditions de bijection et de régularité énoncées.

BIBLIOGRAPHIE

- [1-1] - **Y. Bard**, " Comparison of gradient methods for the solution of non linear parameter estimation problems", SIAM J Numer. Anal., Vol 7, p 157, 1970
- [1-2] - **Y. Bard**, " Non linear parameter estimation", Academic Press, New York, 1974
- [1-3] - **M. Benson**, " parameter fitting in dynamic models", Ecol. Mod., N°6, p 97, 1979
- [1-4] - **K.M Brown, J.E Jr Dennis**, " Derivative free analogues of the Levenberg-Marquardt and Gauss algorithms for non linear least squares approximation, Numer. Math., Vol 18, p 289, 1972
- [1-5] - **C.G Broyden**, " A class of methods for solving nonlinear simultaneous equations", Math. Comput., Vol 19, N°92, p 577, 1965
- [1-6]- **W.C Davidon**, " Variable metric methods for minimization", AEC Research and Development Rept., ANL 5990 (Rev.), 1959
- [1-7] - **A.V Fiacco, G.P, Mac Cormick**, " Non linear programming sequential unconstrained minimization techniques", Wiley, New york 1968
- [1-8] - **R. Fletcher, M.J.D. Powell**, " A rapidly convergent descent method for minimization", Comp J., Vol 6, p 163, 1963
- [1-9] - **R. Fletcher, C.M. Reeves**, " Function minimization by conjugate gradients", Comp J., Vol 7, p 149, 1964
- [1-10] - **G.E Forsythe**, "An asymptotic directions of the S-dimensional optimum gradient method, Numerische Mathematik, Vol 11, p. 57, 1968
- [1-11] - **P.E. Gill, W. Murray**, " Quasi-Newton methods for unconstrained optimization", J Inst Math Appl., Vol 9, p. 91, 1972
- [1-12] - **P.E. Gill, W. Murray**, " An algorithm for least squares estimation of nonlinear parameters", SIAM J. Num. Anal., Vol 15, p. 977, 1978
- [1-13] - **M.R Hestenes, E. Stiefel**, " Methods of conjugate gradients for solving linear equations", J. Res. Nat. Bur. Stan., Vol 49, p. 409, 1952
- [1-14] - **L. Kaufman, G.S. Sylvester and M.H. Wright**, " Structured linear least squares problems in system identification and separable non linear data fitting", SIAM J. OP., Vol 4, N°4, p 847, 1994
- [1-15] - **K. Levenberg**, " A method for the solution of certain non linear problems in least squares", Quart. Appli. Math., Vol 2, p. 164, 1944
- [1-16] - **D.G Luenberger**, " Introduction to linear and non linear programming", Addison-Wesley, 1973

- [1-17]- **N. Mansouri, J.P Kernévez**, "Identification des paramètres par une méthode combinée Marquardt-Continuation", ICEA 92, Vol 3, Tizi-Ouzou 4-6 mai 1992
- [1-18] - **D.W. Marquardt**, " An algorithm for least squares estimation of non linear parameters", SIAM J Appli Math , Vol 11, p 431, 1963
- [1-19] - **M. Minoux**, " Mathematical programming : theory and algorithms", John Wiley, 1986
- [1-20]- **J.J. Moré**, " The Levenberg- Marquardt algorithm : implementation and theory " , Lecture notes in mathematics 630, Num Anal., p 105, 1977
- [1-21]- **E. Polak**, " Computational methods in optimization", Academic Press, New-York, 1971
- [1-22]- **M.J.D. Powell**, " An efficient method of finding the minimum of a function of several variables without calculating derivatives", Comp J , Vol 7, p 155, 1964
- [1-23]- **D.E. Salane**, " A continuation approach for solving large residual nonlinear least squares problems", Scient Stat Comp , Vol 8, N°4, p 655, 1987
- [1-24]- **J. Qin, D.T. Nguyen**, " Generalized exponential penalty function for non linear programming " , Comp Struct , Vol 50, N° 4, p 509, 1994
- [1-25]- **J.M.Varah**, "Relative sizes of the hessian terms in nonlinear estimation", Siam J Sci Stat Comp , Vol 11, N° 1, p 174, 1990
- [1-26]- **N. De Villiers and D.Glasser**, " A continuation method for nonlinear regression", Siam J Numer Anal , 18, p 1139, 1981
- [2-1]- **H.T Banks, J.M. Growley, K. Kunish**, " Cubic spline approximation techniques for parameter estimation in distributed system " , IEEE Trans. Auto. Cont., Vol Ac-28, p 773, 1983
- [2-2]- **H.T Banks, F. Kappel**, " Spline approximation for functional differential equations " , J Differential Equations, Vol 34, p 496, 1979
- [2-3]- **H.T Banks, P.D. Lamm**, " Estimation of variable coefficients in parabolic distributed systems", IEEE Trans Auto Cont , Vol Ac-30, p 386, 1985
- [2-4]- **C. DE Boor** , " A practical guide to splines " , Heidelberg Springer Verlag, 1978
- [2-5]- **J.N. Holt**, " Nonlinear least squares inversion of an integral equation using free knot cubic splines", Lecture notes in control and information sciences, Proceeding of the 8th IFIP conference on optimization techniques (part 2), Wursburg, sept 5-9, 1977
- [2-6]- **G. Nurnberger, M. Sommer**, " Alternation for best spline approximations", Num Math , Vol 41, p 207, 1983
- [2-7]- **I.J Shoenberg, A. Whitney**, " On polya frequency functions III", Trans Amer. Mathem Soc , Vol 74, p. 246, 1953
- [2-8] - **M.H. Shultz**, " Spline analysis", Englewood Cliffs, Prentice Hall, 1973

- [2-9]- **L.L. Shumaker**, "Spline functions - basic theory", John Wiley, New-York, 1981
- [2-10]- **J.M. Varah**, "A spline least square method for numerical parameter estimation in differential equations", SIAM J. Sci. Stat. Comp., Vol 3, N°1, p. 28, 1982
- [3-1]- **C.A Canizares, F.L Alvarado**, "Point of collapse and continuation methods for large AC/DC systems", IEEE Trans in Power systems, Vol 8, N°1, p. 1, 1993
- [3-2]- **E.J. Doedel, J.P. Kernévez**, "AUTO software for continuation and bifurcation problems in ordinary differential equations", Applied mathematics report, California Institute of technology, 1986
- [3-3]- **E. Carrera**, "A study on arc-length type methods and their operation failures illustrated by a simple model", Comp. & struct., Vol 50, N°2, p. 21, 1994
- [3-4]- **P.G. Drazin, Y. Tourigny**, "Numerical study of bifurcations by analytic continuation of a function defined by a power series", SIAM J. Appl. Math., Vol 56, N°1, p. 1, 1996
- [3-5]- **M. Fafard, B. Massicotte**, "Geometrical interpretation of the arc-length method", Comp & Struct., Vol 46, N°4, p. 603, 1993
- [3-6]- **A. Jepson, A. Spence**, "Folds in solutions of two parameter systems and their calculation - part I", SIAM J. Num. Anal., Vol 22, N°2, p. 347, 1985
- [3-7]- **M. Kubicek**, "Algorithm for evaluation of complex bifurcation points in differential equation", SIAM J. Appl. Math., Vol 38, N°1, 1980
- [3-8]- **J.P Kernévez**, "Enzyme mathematics", North Holland Press, 1980
- [3-9]- **N. Mansouri, J.P Kernévez**, "Application de la méthode de continuation par la pseudo-longueur d'arc à l'étude des systèmes non-linéaires", COMAEI 96, Tlemcen 3-5 dec. 1996, Vol 1, p. 144
- [3-10]- **N. Mansouri, J.P Kernévez**, "Estimation of parameters in nonlinear problems", Num. Algorith., Vol 18, p. 333, 1998
- [3-11]- **G. Moore, A. Spence**, "The calculation of turning points of non-linear equations", SIAM Journal Numer. Anal., Vol 17, N°4, p. 567, 1980
- [3-12]- **W.C Rheinboldt**, "Solutions fields of non linear equations and continuation methods", SIAM J. Num. Anal., Vol 17, N°2, 1980
- [3-13]- **D.H. Sattinger**, "Topics in stability and bifurcation theory", Lecture Notes in Mathematics, Springer-Verlag, Berlin, 1973
- [3-14]- **N.C. Rheinboldt, J.V. Burkardt**, "A locally parametrized continuation", ACM Trans. Math. Softw., Vol 9, N°2, p. 215, 1983
- [3-15]- **D. Roose, B. De Dier**, "Numerical determination of an emanating branch of Hopf bifurcation points of two parameter problems", SIAM J. Sci. Stat. Comp., Vol 10, N°4, p. 671, 1989

[3-16]- **R. Seydel**, " From equilibrium to chaos Practical bifurcation and stability analysis", Elsevier Science Publishers, North- Holland, 1988

[3-17]- **R. Seydel**, " Numerical computation of branch points in non linear equations", Numer Math , N°33, p 339-352, 1979

[3-18]- **J. Shi and M.A. Crisfield**, " A semi-direct approach for the computation of singular points", Comp & struct , Vol 51, N° 1, p 107, 1994

[3-19] - **P. Wriggers, J.C. Simo**, , "A general procedure for direct computation of turning and bifurcation points", Int Numer Meth Engng , Vol 30, p. 339, 1990

[4-1] - **B.E. Ainseba, J.P. Kernévez, R. Luce**, " Identification des paramètres dans les problèmes non linéaires à données incomplètes " , Math Mod Num. Anal , Vol 28, N°3, p 313, 1994

[4-2] - **J.P Kernévez**, " The sentinel method and its application to environmental pollution problems", CRC Press, New-York, 1997

[4-3] - **J.L.Lions**, " Sentinelles pour les systèmes distribués", Masson, 1992

[4-4] - **J.L.Lions**, " Sur les sentinelles des systèmes distribués ; cas des conditions initiales incomplètes, conditions frontières, termes sources, coefficients incomplètement connus", CRAS, Vol 307, p. 819 et p. 865, 1988

RESUME

Le travail présenté dans cette thèse porte sur l'étude de différentes approches possibles, pour résoudre un problème d'identification des paramètres dans le cas non linéaire

Chacune des méthodes présentées, est étudiée aussi bien du point de vue algorithmique que numérique. Les performances, les avantages et les inconvénients possibles, sont mis en évidence grâce à de nombreux exemples en relation avec des domaines multiples.

La méthode des moindres carrés est la méthode la plus couramment utilisée pour formuler un problème d'identification. On verra qu'il existe plusieurs possibilités pour minimiser un critère quadratique et que la mise au point d'algorithmes efficaces pour sa résolution revêt toujours une grande importance. La combinaison d'algorithmes déjà connus pour leurs performances, avec d'autres méthodes améliore encore plus, leur efficacité. Ceci permet de développer des algorithmes fonctionnant par application successive.

On traitera également des problèmes d'optimisation des systèmes, en utilisant un algorithme de continuation. Cet algorithme permet le suivi de l'évolution de la solution du point de vue trajectoire et stabilité. Il permet la détection des points singuliers, tels que les points de retournement et de bifurcation. Pour cela, les différentes méthodes de calcul du pas de déplacement et des directions de recherche sont proposées.

Lorsque le problème se rapporte à des systèmes modélisés par des équations différentielles avec conditions initiales inconnues ou mal définies, on montrera qu'il serait plus intéressant de procéder à une identification par étapes ou par application successive des algorithmes proposés. Ceci permet d'avoir des résultats d'une plus grande finesse et de gagner en temps de calcul.

Mots clés: B-splines, Identification, Non linéaire, Paramètres, Sentinelles,

ABSTRACT

In this work different approaches for non linear parameters identification are introduced and analysed. For that, Numerical and algorithmic analysis are presented. The performances, advantages and possible shortcomings are highlighted by applying the approaches to a variety of test examples.

The least squares method has been widely used in system identification. Various efficient algorithms for minimising a quadratic cost are reviewed and analysed. It is shown that a combination of these algorithms with other methods can improve their efficiency. This leads to the so-called class of successive applications algorithms.

We also consider system optimisation using a continuation algorithm. This algorithm allows us to track the solution in terms of trajectory and stability. At the same time it can detect singular points such as limit and bifurcation points. Different methods for computing the direction of search and the displacement step are proposed.

On the other hand when the system is modelled with differential equations with unknown or badly defined initial conditions, it is shown that it would be more interesting to consider the identification with successive applications of the proposed algorithms. This gives finer results and decreases the run time.