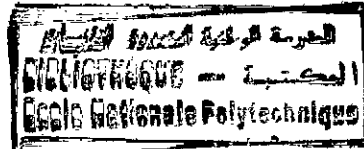


M0022/93B

ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT ELECTRONIQUE



THESE

présentée par M<sup>elle</sup> GAOUAR Ouahiba  
Ingénieur d'Etat en Electronique (E.N.P.)

pour l'obtention du Diplôme de

MAGISTER en Electronique

Option : Acquisition et Traitement de l'Information.

THEME

SYNTHESE D'ALGORITHMES  
DE COMMANDE ADAPTATIVE  
A MODELE DE REFERENCE

Généralisation et cas stochastique

soutenue le 10 novembre 1993, devant le jury composé de MM :

A. ZERGUERRAS	Maitre de Conférence	Président
F. CHÉGARA	Chargé de Cours	Rapporteur
M. S. AIT CHEIKH	Chargé de Cours	Examineur
F. BOUDJEMA	Chargé de Cours	Examineur
B. DERRAS	PhD.	Examineur

ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT ELECTRONIQUE

T H E S E

المدرسة الوطنية المتعددة التقنيات  
المكتبة — BIBLIOTHEQUE  
Ecole Nationale Polytechnique

présentée par M<sup>elle</sup> GAOUAR Ouahiba

Ingénieur d'Etat en Electronique (E.N.P.)

pour l'obtention du Diplôme de

M A G I S T E R en Electronique

Option : Acquisition et Traitement de l'Information.

T H E M E

SYNTHESE D'ALGORITHMES  
DE COMMANDE ADAPTATIVE  
A MODELE DE REFERENCE

Généralisation et cas stochastique

soutenue le 10 novembre 1993, devant le jury composé de MM :

A. ZERGUERRAS	Maitre de Conférence	Président
F. CHIGARA	Chargé de Cours	Rapporteur
M.S. AIT CHEIKH	Chargé de Cours	Examineur
F. BOUDJEMA	Chargé de Cours	Examineur
B. DERRAS	PhD.	Examineur

المدرسة الوطنية المتعددة التقنيات  
المكتبة — BIBLIOTHEQUE  
Ecole Nationale Polytechnique

A mes parents

## Remerciements

Je remercie vivement Monsieur le Professeur Zerguerras d'avoir accepté la présidence du jury.

Je remercie également mon directeur de thèse, Monsieur le Professeur Chigara, pour son aide, ses encouragements et sa disponibilité.

Que Messieurs les Professeurs Ait Cheikh et Derras trouvent ici l'expression de ma reconnaissance pour leur aide et leurs conseils.

Je remercie Messieurs les membres du jury.

Je remercie particulièrement les Professeurs de l'Université Technique et Technologique de Sofia et tout spécialement Monsieur le Professeur Christov pour son aide, ses conseils et l'intérêt qu'il a porté à mon travail.

Que Messieurs les Professeurs Petkov et Markov soient assurés de ma gratitude.

Un grand merci à ma famille d'Alger pour son accueil toujours chaleureux, ses encouragements et son soutien permanents.

Je remercie mes amis de Sofia, de l'E. N. P. d'Alger et assure mon amie Rabéa Dahmani de ma vive reconnaissance.

# SOMMAIRE

المدرسة الوطنية المتعددة التقنيات  
المكتبة — BIBLIOTHEQUE  
Ecole Nationale Polytechnique

Liste des abréviations .....	6
INTRODUCTION .....	7
PARTIE A : COMMANDE ADAPTATIVE DE SYSTEMES A PARAMETRES CONSTANTS	
Chapitre I : GENERALITES .....	11
1.1 Définition de la commande adaptative (CA) .....	11
1.2 Différentes approches de la CA .....	13
1.2.1 Systèmes adaptatifs en boucle ouverte ..	13
1.2.2 Systèmes adaptatifs en boucle fermée ...	13
a) Systèmes adaptatifs avec identification ou contrôleur auto-ajustable .....	14
b) Systèmes adaptatifs à modèle de référence	15
1.3 Classification des systèmes adaptatifs.....	15
1.4 Méthodes de synthèse des systèmes adaptatifs à modèle de référence (MR) .....	16
1.4.1 Formulation du problème de synthèse .....	17
1.4.2 Méthode de synthèse utilisant la théorie de l'hyperstabilité .....	18
1.5 Robustesse .....	21
Chapitre II : ETUDE DU MRAC (Model Reference Adaptive Control)	22
2.1 Généralités .....	22
2.2 Détermination de la partie linéaire de la commande .....	23
2.3 Détermination de la partie adaptative de la commande .....	23
2.4 Conclusion .....	26

Chapitre III : ETUDE DU MCS (Minimal Controller Synthesis)	27
3.1 Généralités .....	27
3.2 L'algorithme MCS .....	27
3.3 Hyperstabilité du MCS .....	28
3.4 Conclusion .....	30
Chapitre IV : EXEMPLES DE DERIVATION DES LOIS DE COMMANDE MRAC	
4.1 Cas d'un bras manipulateur .....	32
4.1.1 Variation des conditions initiales .....	38
4.1.2 Incertitudes paramétriques .....	41
4.2 Cas d'un servomécanisme .....	45
Chapitre V : EXEMPLES DE DERIVATION DES LOIS DE COMMANDE MCS	
5.1 Cas d'un bras manipulateur .....	49
5.1.1 Variation des conditions initiales .....	51
5.1.2 Incertitudes paramétriques .....	57
5.2 Cas d'un servomécanisme .....	60
Chapitre VI : ANALYSE COMPARATIVE DU MRAC ET DU MCS	63
6.1 Cas d'un bras manipulateur .....	63
6.2 Cas d'un servomécanisme .....	65
6.2.1 Lois PI et PID .....	65
6.2.2 Lois MRAC et MCS .....	67
6.3 Conclusion .....	68
Chapitre VII : ANALYSE DE LA CONVERGENCE DU MCS	70
7.1 Généralités .....	70
7.2 Influence des paramètres de convergence sur la commande et l'erreur du MCS .....	70
7.2.1 Effet de $\alpha$ sur la commande .....	71
7.2.2 Effet de $\beta$ sur la commande .....	71
7.2.3 Effet de $\alpha$ sur l'erreur de poursuite ...	71
7.2.4 Effet de $\beta$ sur l'erreur de poursuite ...	72
7.3 Etude de la convergence de l'erreur du MCS .....	72
7.3.1 Enveloppe de l'erreur du MCS .....	72

7.3.2	Dérivation du taux de convergence $\gamma$ . . . . .	80
7.3.3	Dérivation de l'amplitude K . . . . .	82
7.3.4	Résultats et simulation . . . . .	85
a)	Effet de $\alpha$ sur le taux de convergence $\gamma$ . . . . .	85
b)	Effet de $\beta$ sur le taux de convergence $\gamma$ . . . . .	85
c)	Effet de $\alpha$ sur l'amplitude K . . . . .	85
d)	Effet de $\beta$ sur l'amplitude K . . . . .	86
7.3.5	Recommandations sur le choix des paramètres de convergence . . . . .	86
7.4	Conclusion . . . . .	91

**PARTIE B : COMMANDE ADAPTATIVE DE SYSTEMES A PARAMETRES VARIABLES**

Chapitre I :	GENERALITES . . . . .	92
Chapitre II :	L'ALGORITHME MRACM (MRAC Modifié) . . . . .	93
2.1	Poursuite linéaire compensée . . . . .	94
2.2	Adaptation à MR . . . . .	95
Chapitre III :	IDENTIFICATION PARAMETRIQUE . . . . .	99
3.1	Généralités . . . . .	99
3.2	Algorithme MCR à trace constante (SISO) . . . . .	102
3.3	Algorithme MCR à trace constante (MIMO) . . . . .	104
3.3.1	Dérivation de l'algorithme . . . . .	104
3.3.2	Stabilité numérique . . . . .	108
3.4	Convergence paramétrique . . . . .	110
Chapitre IV :	DISCRETISATION DE LA COMMANDE . . . . .	112
Chapitre V :	TESTS ET SIMULATION DES ALGORITHMES :	
	MRACM, MRAC ET MCS . . . . .	115
5.1	Application à un système SISO . . . . .	115
5.1.1	Variations rectangulaires des paramètres . . . . .	116

5.1.2	Variations sinusoïdales des paramètres	122
5.2	Application à un système MIMO	124
5.2.1	Variations sinusoïdales des paramètres	125
a)	Variations lentes des paramètres	125
a-1)	Référence constante	125
a-2)	Référence carrée	126
b)	Variation en amplitude des paramètres de la matrice A	131
c)	Variation en amplitude des paramètres de la matrice B	141
d)	Variations rapides des paramètres	146
5.2.2	Variations rectangulaires des paramètres	151
Chapitre VI : CONCLUSION		158
PARTIE C : COMMANDE ADAPTATIVE DE SYSTEMES DANS UN ENVIRONNEMENT STOCHASTIQUE		
Chapitre I : GENERALITES		161
Chapitre II : TESTS ET SIMULATION DE L'ALGORITHME MCS		163
Chapitre III : DERIVATION DE L'ALGORITHME STOCHASTIQUE		169
3.1	Le filtre de Kalman étendu	169
3.2	Cas de systèmes SISO d'ordre 1	170
3.3	Cas de systèmes SISO d'ordre n	175
Chapitre IV : TESTS ET SIMULATION L'ALGORITHME STOCHASTIQUE		181
4.1	Application à un système SISO d'ordre 1	181
4.1.1	Bruits gaussiens	182
4.1.2	Bruits uniformément distribués	182
4.1.3	Bruits corrélés	182
4.1.4	Paramètre variable	187



4.2	Application à un système SISO d'ordre 4	191
4.2.1	Bruits gaussiens	191
4.2.2	Bruits uniformément distribués	193
Chapitre V : CONCLUSION		202
CONCLUSIONS ET PERSPECTIVES		203
REFERENCES BIBLIOGRAPHIQUES		206
ANNEXE A :	Algorithme de Hooke et Jeeves	214
ANNEXE B :	Taux de convergence	217
ANNEXE C :	Stabilité au sens de Lyapunov	224
ANNEXE D :	Hyperstabilité	226
ANNEXE E :	Solutions de l'inégalité intégrale de Popov	229
ANNEXE F :	Filtrage de Kalman	231

LISTE DES ABREVIATIONS ET SYMBOLES

A : matrice d'évolution

B : matrice de commande

BLIT : bloc linéaire invariant dans le temps

BNLVT : bloc non linéaire et/ou variant dans le temps

CA : commande adaptative

CR : contre-réaction

IP : indice de performance

MIMO : multi-input multi-output

MCR : moindres carrés récurrents

MR : modèle de référence

MRAC : model reference adaptive control

MRACM : MRAC modifié

MCS : minimal controller synthesis

ODE : ordinary differential equation

SISO : single input single output

STR : self tuning regulator

المدرسة الوطنية المتعددة التقنيات  
BIBLIOTHEQUE — المكتبة  
Ecole Nationale Polytechnique

## INTRODUCTION

Les schémas de commande à contre-réaction (CR) conventionnels utilisent des régulateurs à paramètres fixes. Or, les performances de ces systèmes de commande sont affectées par des modifications de l'environnement ou par des variations non mesurables ou inconnues des paramètres du procédé. Lorsque la dégradation des performances est inacceptable, la commande adaptative (CA) devient nécessaire.

Un système de CA contient en plus d'une boucle de commande à CR ayant un contrôleur à paramètres ajustables, une boucle supplémentaire qui agit sur les paramètres de ce contrôleur (ou sur l'entrée de commande) afin de maintenir un certain niveau de performances du système lorsque les paramètres du procédé sont variables (et/ou inconnus).

La conception de systèmes de CA n'est pas une préoccupation récente. En 1958, Whitaker proposa une catégorie de structures dans laquelle la performance désirée du système commandé est spécifiée par un système prédéterminé appelé modèle de référence (MR) [17], [31]. La catégorie des systèmes adaptatifs à MR est caractéristique des problèmes d'asservissement formulés dans un cadre déterministe.

La même année, Kalman introduisit une classe de schémas d'adaptation dans lesquels les paramètres du procédé à commander sont identifiés puis les valeurs estimées sont utilisées pour déterminer les paramètres du régulateur. Ces structures ont conduit à la notion de

régulateur auto-ajustable (STR) [ 3] introduite dans le cadre de problèmes de régulation stochastique.

A cette époque, il n'y avait pas de résultat théorique fondamental assurant un fonctionnement stable de ces systèmes.

Dans les années 60, des résultats très importants pour la commande ont été obtenus : théories de la variable d'état, de la stabilité, de la commande stochastique. Ils ont permis une meilleure compréhension de la CA et un regain d'intérêt pour ce domaine.

A la fin des années 70 et au début des années 80, quelques démonstrations rigoureuses de stabilité de systèmes adaptatifs [17], [11], [27] ont été obtenues sous des hypothèses restrictives. Le désir de rendre ces hypothèses plus souples et d'étendre les méthodes de la CA à des systèmes plus complexes ( systèmes non linéaires, multivariables ...) [28], [13], [33] a suscité de nouvelles recherches.

Actuellement, des contrôleurs adaptatifs apparaissent sur le marché et la CA connaît un développement vigoureux. Les applications initialement confinées aux systèmes d'armes ou aux pilotes automatiques d'avions se sont répandues à des domaines variés [18], [ 3], [ 2] : fours de séchage, robots manipulateurs, régulation de pH; asservissements à moteurs électriques ...

Dans cette thèse, nous nous intéressons à la commande adaptative à MR et plus précisément à un algorithme décrit par Landau [16], [17]. La mise en oeuvre de cet algorithme nécessite une connaissance a priori du procédé à commander. Stoten et Béchoubane [36], [37] ont développé un algorithme basé sur celui de Landau mais ne demandant qu'une connaissance minimale du système à commander. Il s'agit de l'algorithme MCS.

Le travail que nous avons développé essaie de généraliser les algorithmes décrits et d'appliquer le MCS dans un environnement stochastique.

La thèse est présentée en trois parties.

La première partie traite la CA de systèmes à paramètres constants. Dans le chapitre I, nous précisons la notion de CA, ses différentes approches et les techniques de synthèse de systèmes adaptatifs stables. Puis, nous reprenons l'algorithme de Landau dans le chapitre II. L'approche de Stoten et Benchoubane est étudiée dans le chapitre suivant. Nous dérivons et simulons les lois de commande du MRAC et celles du MCS, pour un bras manipulateur et pour un servomécanisme, dans les chapitres IV et V respectivement. Le chapitre VI renferme une étude comparative des résultats obtenus pour les deux algorithmes. Nous nous intéressons ensuite à l'influence des paramètres de la loi de commande sur le taux de convergence du MCS. Dans ce cadre, le chapitre VII propose une extension des travaux de Stoten et Benchoubane [39] à un système d'ordre deux. Les résultats obtenus nous permettent de proposer des recommandations pour le choix des paramètres de la loi de commande.

La deuxième partie de la thèse concerne les systèmes à paramètres variables. Après des généralités introduites dans le chapitre I, nous reprenons, dans le chapitre suivant, l'algorithme MRAC modifié [5]. Ce dernier incorpore un algorithme d'identification paramétrique introduit par Landau [19] pour des systèmes SISO. Dans le chapitre III, nous développons cet algorithme pour des systèmes MIMO. Le Chapitre IV indique la façon dont les lois de commande sont discrétisées. Puis, dans le chapitre V, nous testons et

simulons les algorithmes MRAC, MCS et MRACM pour des systèmes à paramètres variables. Nous considérons des paramètres sinusoidaux de différentes amplitudes et fréquences ainsi que des paramètres donnés par des fonctions rectangulaires. Les algorithmes sont appliqués à un système SISO et à un procédé MIMO. Le chapitre VI renferme une conclusion sur les résultats obtenus.

Enfin, la dernière partie de notre étude traite les systèmes dans un environnement stochastique. Dans le chapitre I, nous présentons des généralités sur la CA stochastique. Dans le chapitre suivant nous testons et simulons l'algorithme MCS dans un environnement stochastique. L'instabilité que nous observons, nous conduit à prendre une autre approche et l'algorithme stochastique qui en est déduit est présenté dans le chapitre III. Il est basé sur le filtrage de Kalman étendu. Nous procédons à des simulations de cet algorithme et les résultats obtenus sont rassemblés dans le chapitre IV. Les conclusions concernant cette nouvelle approche sont présentées dans le chapitre V.

PARTIE A

COMMANDE ADAPTATIVE  
DE SYSTEMES  
A PARAMETRES CONSTANTS



CHAPITRE IGENERALITES1.1 Définition de la CA

La CA rassemble en fait un ensemble de techniques utilisées pour l'ajustement automatique, en ligne et en temps réel des boucles de commande. Le but de ces techniques est de maintenir un certain niveau de performance lorsque les paramètres du procédé à commander sont inconnus et/ou variables dans le temps.

Pour atteindre cet objectif, il faut d'abord définir une mesure des performances du système que l'on appellera indice de performances (IP) [17]. Un système adaptatif mesure cet IP en utilisant les entrées, les sorties et les états du système à adapter (ou système ajustable). A partir de l'écart entre l'IP mesuré et l'IP désiré, un mécanisme d'adaptation modifie les paramètres du système ajustable ou génère un signal de commande supplémentaire afin de maintenir l'IP du système dans une plage de valeurs souhaitées.

Le schéma de principe d'un système adaptatif, donné en figure 1-1, fait apparaître :

- une boucle interne qui est une boucle de commande à CR comprenant le procédé et son contrôleur ajustable.
- une boucle externe qui agit sur le système ajustable afin de maintenir les performances du système. Elle a une structure à CR mais ici la variable à contrôler est la performance du système.

Cette boucle supplémentaire est spécifique à de nombreux schémas adaptatifs.

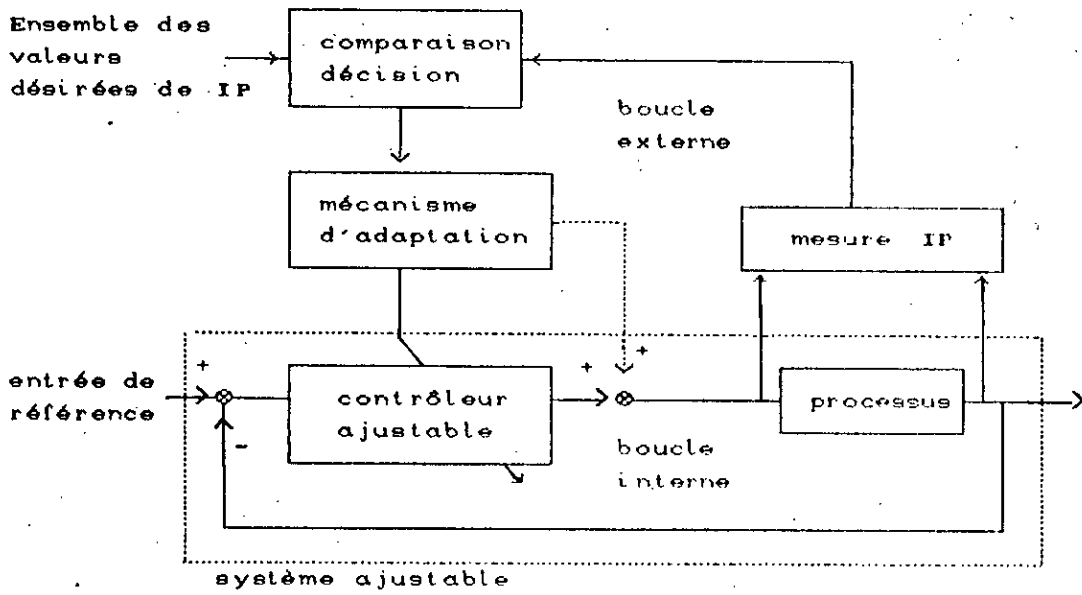


Fig. 1-1 Schéma de principe d'un système adaptatif

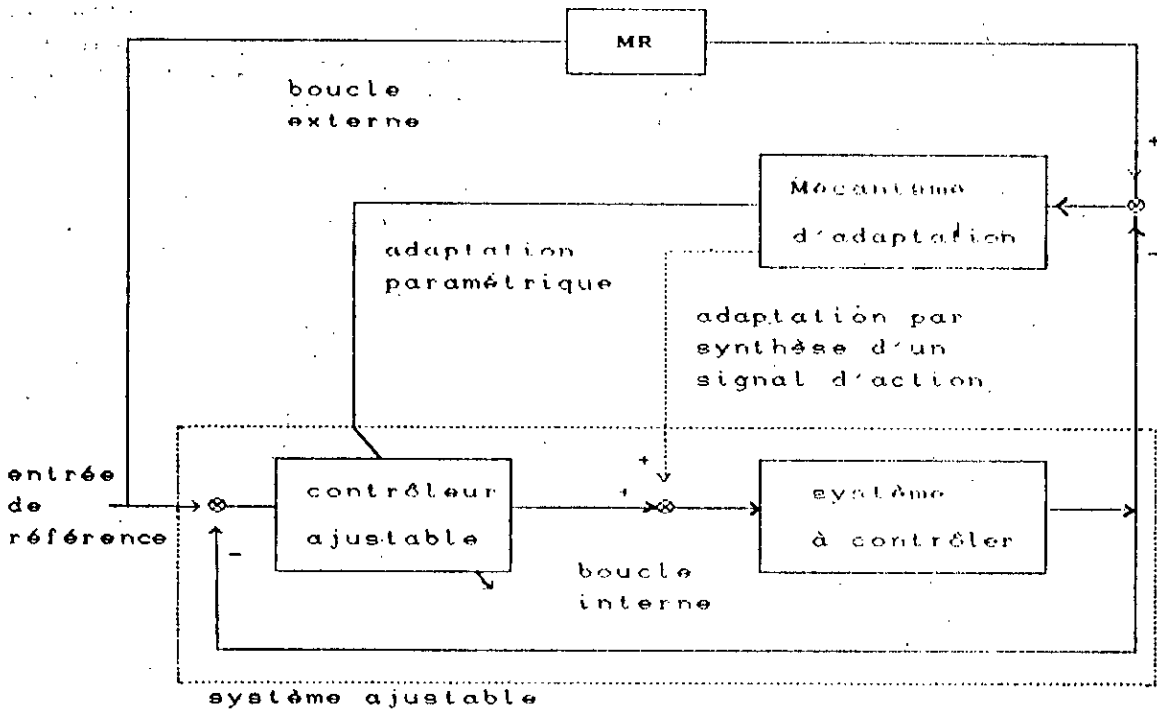


Fig. 1-2 Système adaptatif à modèle de référence

## 1.2 Différentes approches de la CA

On distingue deux grandes classes de systèmes adaptatifs :

- systèmes adaptatifs en boucle ouverte,
- systèmes adaptatifs en boucle fermée pour lesquels il existe essentiellement deux approches :

- les systèmes adaptatifs auto-ajustables avec identification

- les systèmes adaptatifs à modèle de référence (MR).

### 1.2.1 Systèmes adaptatifs en boucle ouverte

Dans ces schémas, on suppose connue une relation entre certaines variables de l'environnement (variables auxiliaires) et les paramètres du procédé. Si ces variables sont mesurables, elles peuvent être utilisées pour modifier les paramètres du contrôleur [3]. La solution d'adaptation la plus simple consiste à définir à l'avance plusieurs modes de fonctionnement (correspondant à différents ensembles de valeurs des variables auxiliaires) et à associer à chacun d'eux un réglage du contrôleur. Les systèmes ainsi obtenus sont dits à gains préprogrammés (gain scheduling). Dans cette approche, les modifications des performances ne sont ni mesurées, ni comparées avec les performances désirées. Si les relations entre mesures de l'environnement et paramètres dynamiques du procédé changent, les performances de ce dernier peuvent devenir catastrophiques. D'autre part, bien que le principe semble simple, sa mise en oeuvre peut être coûteuse car elle peut requérir l'introduction de capteurs additionnels.

### 1.2.2 Systèmes adaptatifs en boucle fermée

Contrairement aux structures précédentes, les techniques de CA en boucle fermée ne demandent pas de mesures

supplémentaires mais elles requièrent une plus grande puissance de traitement de l'information. Les systèmes de CA en boucle fermée sont fondés sur les hypothèses suivantes [18] :

Hypothèse 1 :

Le procédé à contrôler est représenté par un modèle mathématique dont la structure est invariante mais dont les valeurs des paramètres peuvent changer dans le temps.

Hypothèse 2 :

Pour toutes les valeurs possibles de cette structure, il existe un contrôleur qui permet de réaliser les performances voulues. Le rôle de la boucle d'adaptation est de trouver les bonnes valeurs des paramètres de ce contrôleur.

a) Systèmes adaptatifs avec identification ou contrôleur

auto-ajustable

Dans ces schémas, pour ajuster les paramètres du contrôleur, on utilise l'estimation paramétrique du système à contrôler [ 3]. Cette estimation est donnée par un identificateur en temps réel.

Le contrôleur adaptatif avec identification est très souple par rapport à la méthode de calcul du contrôleur et par rapport au choix de l'identificateur. Toute méthode conduisant à un système bouclé "identificateur + régulateur " stable convient, en principe.

Cependant, l'analyse de la stabilité de ces systèmes est plus complexe que celle des systèmes adaptatifs à MR. Ceci est principalement dû à la transformation généralement non linéaire entre paramètres estimés et paramètres du contrôleur.

## b) Systèmes adaptatifs à modèle de référence

Dans un système adaptatif à MR [16], [17], [31], l'ensemble des IP désirés (cf. fig 1-1) est remplacé par un indice de performance généré par un système dynamique appelé modèle de référence (fig. 1-2). Le modèle de référence (MR) est un système auxiliaire excité par les mêmes entrées que le système ajustable. Dans ce cas, l'écart entre IP désiré -spécifié par le MR- et IP mesuré est directement obtenu en comparant les sorties (ou les états) du MR et du système ajustable. Le mécanisme d'adaptation utilise la différence entre ces sorties (ou états) pour modifier les paramètres du système ajustable ou pour générer un signal d'entrée auxiliaire afin de minimiser l'écart entre les deux IP.

Un avantage de ce type de systèmes est sa rapidité d'adaptation due au fait que la comparaison des IP est directe. En contrepartie, une certaine connaissance a priori des structures (du modèle de procédé ou du système ajustable) est nécessaire pour l'implémentation de ce type de systèmes adaptatifs.

Les propriétés de stabilité des systèmes adaptatifs à MR les rendent particulièrement intéressants.

### 1.3 Classification des systèmes adaptatifs

Il existe de nombreux critères de classification des systèmes adaptatifs [15], [17].

Ainsi, les structures des systèmes adaptatifs à MR peuvent différer par les positions relatives du procédé et du MR. Nous étudierons des systèmes dits à structure parallèle dont le schéma de principe est représenté en figure 1-3.

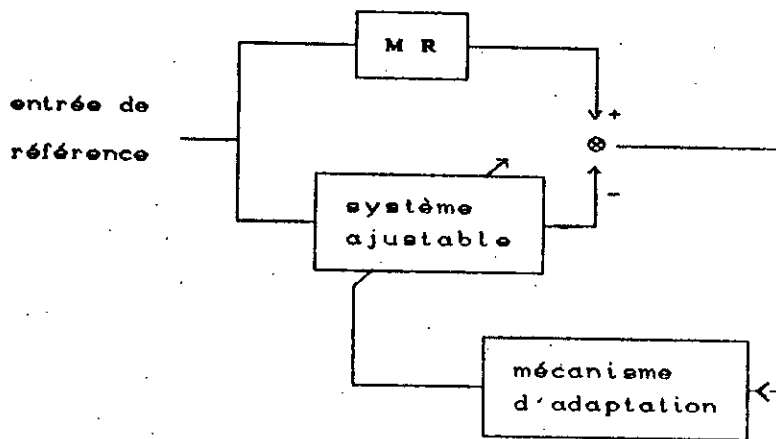


Fig. 1-3 Système adaptatif à MR à structure parallèle

#### 1.4 Méthodes de synthèse des systèmes adaptatifs à MR

Les premiers algorithmes étaient construits à partir des méthodes du gradient. Malheureusement les systèmes ainsi obtenus peuvent devenir instables. Actuellement, la synthèse des systèmes adaptatifs à MR utilise principalement des méthodes basées sur des théories de stabilité :

- la deuxième méthode de Lyapunov [44] qui suppose la construction de fonctions dites de Lyapunov. Or, il n'y a pas de procédure générale de construction de ces fonctions.

- la théorie de l'hyperstabilité [29] qui est plus systématique. Elle est utilisée dans les algorithmes MRAC [16], [17] et MCS [36], [37] auxquels nous nous intéressons.

Dans le cas idéal les hypothèses suivantes sont considérées :

- a) le MR est linéaire et invariant,
- b) le MR et le système ajustable ont la même dimension,
- c) tous les paramètres du système ajustable

sont accessibles pour l'adaptation,  
d) en dehors du vecteur d'entrée, aucun autre  
signal extérieur n'est appliqué.

#### 1.4.1 Formulation du problème de synthèse

Il s'agit, dans notre cas, de synthétiser un mécanisme  
d'adaptation afin

- qu'en l'absence de toute perturbation, l'erreur  
généralisée tende asymptotiquement vers zéro quelle que soit  
l'entrée de référence<sup>1)</sup> et quels que soient les paramètres  
du système ajustable<sup>2)</sup>.

ou

- que l'erreur généralisée reste bornée quelle que  
soit l'entrée de référence<sup>1)</sup> et quels que soient les  
paramètres du système ajustable<sup>2)</sup> et ceci malgré la présence  
éventuelle de perturbations.

Dans le cas idéal, on peut arriver à synthétiser un  
mécanisme d'adaptation qui satisfait la première formulation.  
La deuxième formulation est moins restrictive que la première.  
Elle convient mieux pour la synthèse des systèmes adaptatifs  
dans le cas réel.

---

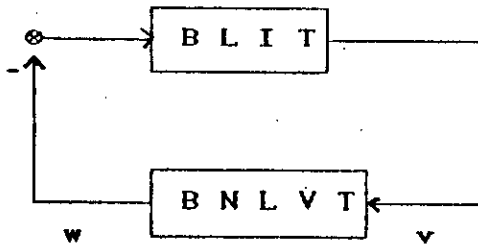
<sup>1)</sup> appartenant à l'espace des fonctions d'entrée.

<sup>2)</sup> appartenant à l'espace des paramètres.

### 1.4.2 Méthode de synthèse utilisant la théorie

#### de l'hyperstabilité

La théorie de l'hyperstabilité [29] étudie la stabilité des types de systèmes à contre-réaction représentés en figure 1-4.



BLIT : bloc linéaire invariant dans le temps

BNLVT : bloc non linéaire et/ou variant dans le temps

Fig. 1-4 Système à contre-réaction

La relation (1-1) ci-dessous est appelée inégalité intégrale de Popov [16], [17].

$$\int_{t_0}^t w^T v dt \geq -\gamma_0^2 \quad (1-1)$$

$\gamma_0^2$  : constante finie positive.

$v$  : entrée du BNLVT.

$w$  : sortie du BNLVT.

Le BLIT est dit (asymptotiquement) hyperstable par rapport à l'ensemble des blocs de contre-réaction définis par la relation (1-1), si et seulement si sa matrice de transfert est (strictement) réelle positive.

Si le BLIT est (asymptotiquement) hyperstable alors le système à contre-réaction de la figure 1-4 est (asymptotiquement) stable globalement.

La méthode de synthèse fondée sur la théorie de



l'hyperstabilité comporte trois étapes résumées en figure 1-5.

Dans un premier temps, le système adaptatif est ramené, par une transformation linéaire, à un système équivalent à contre-réaction représenté en figure 1-5b.

Dans un deuxième temps, les blocs inconnus L2 et NL2 de la figure 1-5b sont déterminés en utilisant la théorie de l'hyperstabilité. Ces calculs présentés en annexe consistent en

- la détermination de L2 tel que la fonction de transfert du bloc linéaire (L1, L2) soit (strictement) réelle positive

- la résolution d'une inégalité intégrale de la forme (1-1), (Fig.1-5c).

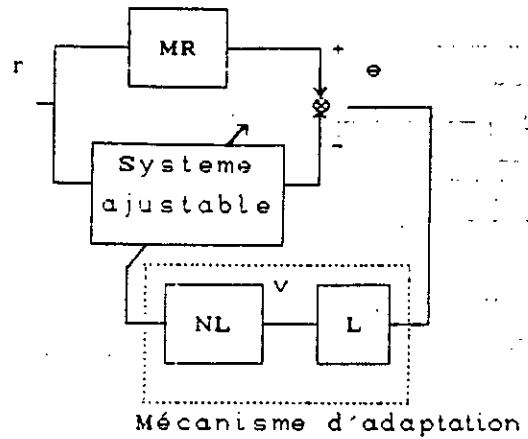
Enfin, le système hyperstable obtenu est ramené si nécessaire à la forme initiale, par une transformation linéaire inverse. La forme finale est présentée en figure 1-5d.

En général, dans la première étape, on cherche à obtenir un système à CR équivalent qui régit l'évolution de l'erreur d'état généralisée. Dans ce cas, si le système à CR est asymptotiquement hyperstable alors il est asymptotiquement stable globalement et donc l'erreur généralisée tend asymptotiquement vers zéro.

Si le système à CR est hyperstable alors il est stable et l'erreur généralisée reste bornée.

Les systèmes hyperstables possèdent plusieurs propriétés [29] parmi lesquelles la propriété d'entrée-bornée / sortie-bornée. Cette propriété permet la synthèse de systèmes adaptatifs dans des situations plus réalistes. Pour cela, on ramène le système adaptatif à un système à CR équivalent possédant une entrée externe  $d(t)$  représentant l'effet des modifications des hypothèses du cas idéal (perturbations...). (Fig.1-6). Puis on applique la méthode de synthèse.

METHODE DE SYNTHESE FONDEE SUR LA THEORIE DE L'HYPERSTABILITE

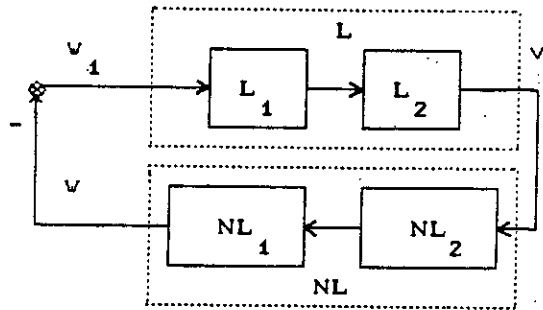


a) Système adaptatif initial

L : bloc linéaire  
NL : bloc non linéaire

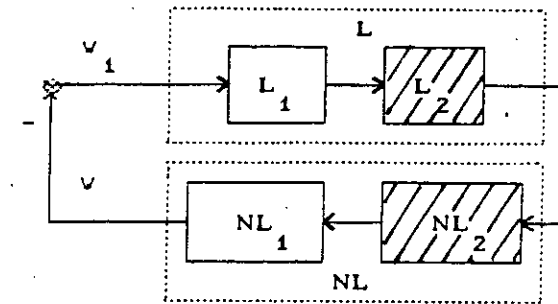
20

Résolution  
du problème  
d'hyperstabilité



c) Système hyperstable

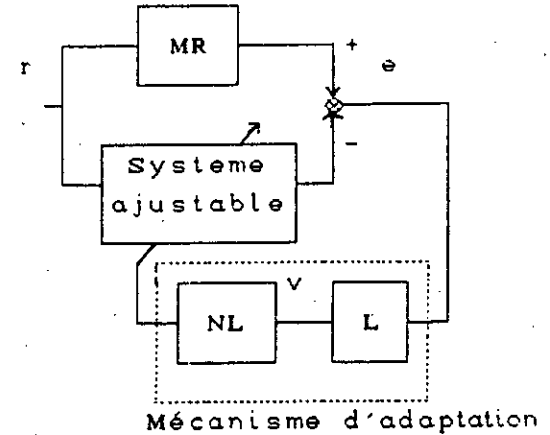
transformation  
linéaire



b) Système à CR équivalent

Les blocs hachurés sont les blocs à déterminer

transformation  
linéaire inverse



d) Système adaptatif ramené  
à sa forme initiale

Fig 1-5 Méthode de synthèse fondée sur la théorie de l'hyperstabilité

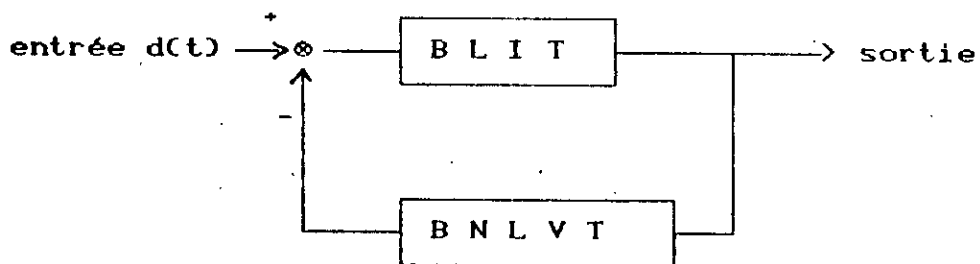


Fig. 1-6 - Système équivalent à CR avec une entrée externe

### 1.5 Robustesse

Les processus à contrôler sont généralement complexes et possèdent une dynamique mal connue. Le concepteur de systèmes de commande ne dispose pas alors d'un modèle détaillé du processus. L'objectif premier étant de faire coïncider le mieux possible le processus et le MR, il est en outre souhaitable que le système adaptatif soit capable de maintenir une certaine stabilité en présence d'effets perturbateurs (dynamiques non modélisées, perturbations de sortie...). Cette propriété est appelée robustesse de la stabilité vis-à-vis de l'effet perturbateur considéré.

Plusieurs algorithmes stables peuvent diverger en présence de dynamiques non modélisées et/ou de perturbations de sortie [30].

Afin de préserver la stabilité et améliorer la robustesse plusieurs modifications d'algorithmes d'adaptation ont été proposées [28], [13].

## CHAPITRE II

### ETUDE DU MRAC

#### 2.1 Généralités

L'algorithme MRAC, développé par Landau, est un algorithme de contrôle adaptatif à modèle de référence basé sur la théorie de l'hyperstabilité [29]. Il est une généralisation d'une stratégie de commande linéaire avec modèle de référence [16], [17]. La loi de commande obtenue comporte une partie linéaire et une partie adaptative.

Solent le procédé et le MR donnés par les équations d'état (2.1) et (2.2) respectivement :

$$\dot{x}(t) = Ax(t) + Bu(t) + d(t) \quad (2-1)$$

$$\dot{x}_m(t) = A_m x_m(t) + B_m r(t) \quad (2-2)$$

où  $x$  et  $x_m$  sont les vecteurs d'état de dimension  $n$  et  $u$  et  $r$ , de dimension  $m$ , sont respectivement la commande du procédé et l'entrée de référence du système.

La loi de commande est :

$$\begin{aligned} u(t) &= u_1(t) + u_2(t) \\ &= -(K - \delta K(t))x(t) + (K_R + \delta K_R(t))r(t) \end{aligned} \quad (2-3-a)$$

$$\text{où } u_1(t) = -Kx(t) + K_R r(t) \quad (2-3-b)$$

$$u_2(t) = \delta K(t)x(t) + \delta K_R(t)r(t) \quad (2-3-c)$$

$u_1$  est la partie linéaire de la commande et

$u_2$  est la partie adaptative destinée à contrecarrer l'effet,

sur la trajectoire de l'état  $x$ , des incertitudes du processus et des perturbations éventuelles représentées par  $d(t)$  dans l'équation (2-1).

## 2.2 Détermination de la partie linéaire de la commande

Les gains  $K$  et  $K_R$  sont des matrices constantes. Ils peuvent être calculés en supposant que le MR et le procédé satisfont les conditions de poursuite parfaite d'un modèle. Ces conditions dites conditions d'Erzberger [17] doivent être vérifiées par les matrices  $A$ ,  $B$ ,  $A_m$  et  $B_m$  :

$$(I - BB^+)(A - A_m) = 0_n \quad (2-4-a)$$

$$(I_n - BB^+)B_m = 0_{n,m} \quad (2-4-b)$$

$$B^+ = (B^T B)^{-1} B^T \quad (2-4-c)$$

$B^+$  est la pseudo-inverse de  $B$ .

Les gains  $K$  et  $K_R$  sont alors donnés par [45] :

$$K = B^+(A - A_m) \quad (2-5-a)$$

$$K_R = B^+ B_m \quad (2-5-b)$$

## 2.3 Détermination de la partie adaptative de la commande

Pour ce faire, la méthode de synthèse basée sur la théorie de l'hyperstabilité est utilisée :

Le système initial représenté par les équations (2-1), (2-2), (2-3-a) est d'abord ramené, en exploitant les équations (2-4-a), (2-4-b), (2-5-a) et (2-5-b), à un système à CR équivalent donné par :

$$\dot{e} = A_m e - B w_e - d(t) \quad (2-6)$$

$$w_o(t) = \delta K_R(t)r(t) + \delta K(t)x(t) \quad (2-7)$$

$$e = x_m - x \quad (2-8)$$

$$y_o = C_o e \quad (2-9)$$

$e$  est l'erreur d'état

$C_o$  est un correcteur linéaire introduit pour permettre la vérification de la condition 1 énoncée ci-dessous.

Le système équivalent régit l'évolution de l'erreur d'état. Il est hyperstable si et seulement si :

1/ la matrice de transfert du bloc linéaire invariant dans le temps est réelle positive :

Pour cela, le lemme de Popov-Yakubovitch-Kalman [17] sur le caractère réel positif d'une matrice de transfert est utilisé.

Lemme de Popov-Yakubovitch-Kalman :

Soit le système multivariable, linéaire, invariant dans le temps, décrit par les équations

$$\left. \begin{aligned} \dot{x} &= Ax + Bu \\ v &= Cx + Ju \end{aligned} \right\} \quad (2-10)$$

où  $x$  est un vecteur d'état de dimension  $n$ ,  $u$  et  $v$  sont des vecteurs de dimension  $m$  représentant l'entrée et la sortie respectivement. Le système est supposé complètement commandable et observable.

Les matrices  $A, B, C, J$  définissent un système caractérisé par une matrice de transfert réelle positive si et seulement si le système d'équations suivant est vérifié :

$$\left. \begin{aligned} N^T N &= J + J^T \\ PA + A^T P &= -LL^T \\ B^T P + N^T L^T &= C \end{aligned} \right\} \quad (2-11)$$

où N et L sont des matrices arbitraires et P est une matrice symétrique définie positive.

L'application du lemme énoncé ci-dessus au système d'équations (2-6) et (2-9) permet de déterminer  $C_0$ .

$$C_0 = B^T P \quad (2-12)$$

où P est la solution définie positive de l'équation de Lyapunov :

$$PA_m + A_m^T P = -Q \quad Q > 0_n \quad (2-13)$$

2/ l'inégalité intégrale de Popov est vérifiée :

$$\int_{t_0}^{t_1} w_0^T(t) y_0(t) dt \geq -\gamma_0^2 \quad t_1 > t_0 \quad (2-14)$$

Landau [16] a obtenu des solutions générales de cette inégalité. Elle est en particulier satisfaite dans le cas d'une loi d'adaptation de type proportionnelle + intégrale (PI). C'est-à-dire pour  $\delta K$  et  $\delta K_R$  donnés par :

$$\delta K(t) = \int_0^t \alpha y_0(\tau) x^T(\tau) d\tau + \beta y_0(t) x^T(t) \quad (2-15)$$

$$\delta K_R(t) = \int_0^t \alpha y_0(\tau) r^T(\tau) d\tau + \beta y_0(t) r^T(t) \quad (2-16)$$

Dans le cas des systèmes à une seule entrée ( $m = 1$ ),  $\alpha$  et  $\beta$  sont des scalaires tels que :

$$\alpha > 0 \quad \text{et} \quad \beta \geq 0.$$

Pour les systèmes multi-entrées ( $m > 1$ ),  $\alpha$  et  $\beta$  sont des matrices carrées d'ordre  $m$  avec  $\alpha$  définie positive et  $\beta$  semi-définie positive.

#### 2.4 Conclusion

L'algorithme MRAC, décrit précédemment, nécessite pour sa mise en oeuvre :

1/ une identification du processus :

comme il peut être vérifié dans les équations (2-5-a) et (2-5-b), la connaissance des matrices A et B du procédé est nécessaire pour déterminer les gains K et  $K_R$ .

2/ la synthèse d'un contrôleur linéaire :

la commande du contrôleur linéaire est donnée par l'équation (2-3-b). Pour la calculer, il faut déterminer les gains linéaires K et  $K_R$ .



## CHAPITRE III

---

### ETUDE DU MCS

---

#### 3.1 Généralités

---

Dans une nouvelle approche, Stoten et Benchoubane [36], [37] ont proposé une extension de l'algorithme MRAC : l'algorithme MCS (Minimal Controller Synthesis) dans lequel les gains  $K$  et  $K_R$  donnés par les relations (2-5-a) et (2-5-b) sont pris égaux à zéro.

La mise en oeuvre du MCS ne nécessite

- ni la synthèse d'un contrôleur linéaire ( $u_1(t)=0$ ),
- ni l'identification du processus : il n'est pas nécessaire de connaître les valeurs des matrices  $A$  et  $B$  du processus. Toutefois, la structure de ce dernier (nombre de degrés de liberté, dimension de l'état) est supposée connue.

#### 3.2 L'algorithme MCS

---

Soient le procédé et le MR donnés par les équations

$$\dot{x}(t) = Ax(t) + Bu(t) + d(t) \quad (3-1)$$

$$\dot{x}_m(t) = A_m x_m(t) + B_m r(t) \quad (3-2)$$

Les matrices  $A$  et  $B$  sont inconnues.

La loi de commande de l'algorithme MCS est :

$$u(t) = \delta K(t)x(t) + \delta K_R(t)r(t) \quad (3-3)$$

avec

$$\delta K(t) = \int_0^t \alpha y_o(\tau) x^T(\tau) d\tau + \beta y_o(t) x^T(t) \quad (3-4-a)$$

$$\delta K_R(t) = \int_0^t \alpha y_o(\tau) r^T(\tau) d\tau + \beta y_o(t) r^T(t) \quad (3-4-b)$$

Pour des systèmes à une seule entrée ( $m = 1$ )  $\alpha$  et  $\beta$  sont des vecteurs-ligne d'ordre  $n$ .

Pour des systèmes multi-entrées ( $m > 1$ )  $\alpha$  et  $\beta$  sont des matrices à  $m$  lignes et  $n$  colonnes.

L'erreur d'état est

$$e = x_m - x \quad (3-5)$$

et

$$y_e = C_e e \quad (3-6)$$

où

$$C_e = P$$

$P$  est la solution de l'équation de Lyapunov :

$$P A_m + A_m^T P = -Q \quad Q > 0 \quad (3-7)$$

### 3.3 Hyperstabilité du MCS

L'hyperstabilité et la robustesse du système régissant l'évolution de l'erreur de l'algorithme MCS a été démontrée dans les cas suivants [37] :

$$\dot{x}(t) = A^*(t)x(t) + B^*(t)u(t)$$

où

$$A^*(t) = A + \delta A(t) \quad B^*(t) = B + \delta B(t)$$

$\delta A(t)$  et  $\delta B(t)$  représentent des variations paramétriques dues au procédé lui-même ou à des modifications de son environnement.

● processus mono-entrée et mono-sortie (SISO) représenté dans l'espace d'état par les matrices d'évolution et de commande ayant la forme canonique

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & \dots & 0 & 1 \\ -\alpha_1 & \dots & \dots & \dots & \dots & -\alpha_n \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ b_1 \end{bmatrix}$$

avec

$$\alpha = \begin{bmatrix} 0 & \dots & 0 & \alpha_n \end{bmatrix} \quad \text{et} \quad \beta = \begin{bmatrix} 0 & \dots & 0 & \beta_n \end{bmatrix}$$

$\alpha_n$  et  $\beta_n$  sont de même signe que  $b_1^*$  ( $b_1^*$  est le dernier élément et unique élément non nul de  $B^*(t)$ ).

• processus multi-entrées et multi-sorties (MIMO) décrit dans l'espace d'état par les matrices :

$$A = \begin{bmatrix} A_{11} & \dots & A_{1m} \\ \dots & \dots & \dots \\ A_{m1} & \dots & A_{mm} \end{bmatrix} \quad B = \begin{bmatrix} B_{11} & \dots & B_{1m} \\ \dots & \dots & \dots \\ B_{m1} & \dots & B_{mm} \end{bmatrix}$$

$m$  = nombre de degrés de liberté du processus.

Remarque : le nombre de commandes est supposé égal au nombre de sous-systèmes.

$$A_{ii} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & \dots & 0 & 1 \\ -a_{ii1} & -a_{ii2} & \dots & \dots & \dots & -a_{iin_i} \end{bmatrix}$$

$$A_{ij} = \begin{bmatrix} 0 & \dots & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & 0 \\ a_{ij1} & a_{ij2} & \dots & a_{ijn_j} \end{bmatrix}$$

$$B_{ii} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ b_{ii} \end{bmatrix}$$

$$B_{ij} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ -b_{ij} \end{bmatrix}$$

$n_i$  est la dimension de l'état associé au degré de liberté  $i$  de sorte que

$$\sum_{i=1}^m n_i = n$$

où  $n$  est l'ordre du processus

avec

$$\alpha_{m,n} = \text{diag} \left[ \left( 0 \dots 0 \alpha_1 \right), \left( 0 \dots 0 \alpha_2 \right), \dots, \left( 0 \dots 0 \alpha_m \right) \right]$$

$$\beta_{m,n} = \text{diag} \left[ \left( 0 \dots 0 \beta_1 \right), \left( 0 \dots 0 \beta_2 \right), \dots, \left( 0 \dots 0 \beta_m \right) \right]$$

$\alpha_i$  et  $\beta_i$  de mêmes signes que les  $b_{ii}^*$ .

### 3.4 Conclusion

Les algorithmes MRAC et MCS ont pour objectif commun la réalisation d'une commande multivariable robuste vis à vis :

- des variations des paramètres du processus
- des perturbations externes
- des non-linéarités du processus

et pour des spécifications sévères des performances du système en boucle fermée.

L'algorithme MCS présente l'avantage de ne pas comporter de partie linéaire dans l'expression de la commande. Sa synthèse est donc minimale et s'avère, par conséquent, moins coûteuse que celle de l'algorithme MRAC.

L'algorithme MCS devrait donc être plus avantageux dans

la pratique.

Afin de comparer les algorithmes MRAC et MCS et d'évaluer la performance de chacune des lois de commande, nous définissons le critère suivant :

$$J = \int_0^T 10 \sum_{i=1}^n (x_{mi}(t) - x_i(t))^2 dt + \int_0^T \sum_{i=1}^m (u_i(t) - u_{Ei})^2 dt \quad (3-8)$$

T est la durée de la simulation,

$x_{mi} - x_i = e_i$  est la  $i^{\text{ème}}$  composante de l'erreur d'état.

$u_{Ei}$  est la valeur que prend la composante  $u_i$  de la commande une fois l'adaptation réalisée. L'optimisation du critère se fera après l'obtention de  $u_{Ei}$ . Par la suite, nous noterons :

$$u_i - u_{Ei} = u_{Fi}$$

Le premier terme du critère représente l'erreur de poursuite ( $e = x_m - x$ ). Le deuxième terme permet d'éviter une trop grande augmentation de la commande  $u$ . Le coefficient 10 qui apparaît dans le premier terme du critère est un facteur de pondération choisi arbitrairement. Son rôle est d'accorder plus d'importance à l'erreur d'état qu'à la commande.

Le critère J sera utilisé dans les chapitres IV, V et VI suivants.

## CHAPITRE IV

## EXEMPLES DE DERIVATION DES LOIS DE COMMANDE MRAC

## 4.1 Cas d'un bras manipulateur

La figure 4-1 représente un bras manipulateur de classe quatre.

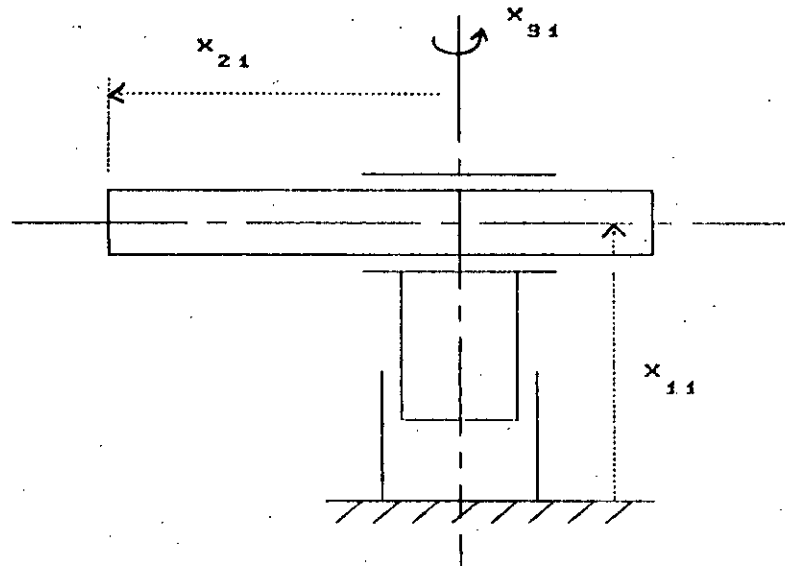


Fig 4-1 Schéma d'un bras manipulateur de classe 4

Il s'agit d'un processus d'ordre 6 à trois entrées dont les équations dynamiques présentées dans [35] conduisent à l'équation d'état suivante

$$\dot{x} = Ax + Bu + d(t) \quad (4-1)$$

où  $d(t)$  est un vecteur de perturbations qui tient compte de toutes les non-linéarités et des termes variant dans le temps du processus.

$$x = \left[ x_{11}, x_{12}, x_{21}, x_{22}, x_{31}, x_{32} \right]^T \quad (4-2)$$

$\dot{x}_{i2}$  est la dérivée de  $x_{i1}$ , pour  $i = 1, 2, 3$ .

$$A = \left[ a_{ij} \right] = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{pmatrix} \quad (4-3)$$

$$B = \left[ b_{ij} \right] = \begin{pmatrix} 0 & 0 & 0 \\ 3,333 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1,278 \end{pmatrix} \quad (4-4)$$

$$d(t) = \left[ 0 \quad d_1 \quad 0 \quad d_2 \quad 0 \quad d_3 \right]^T \quad (4-5)$$

$$d_1(t) = -9,81 \quad (4-6)$$

$$d_2(t) = (x_{21} - 0,75)x_{32}^2 \quad (4-7)$$

$$d_3(t) = - \left[ x_{21} (1,278 x_{21} - 1,917) x_{32} + (2,556 x_{21} - 1,917) x_{22} x_{32} \right] \quad (4-8)$$

Le modèle de référence est donné par :

$$\dot{x}_m(t) = A_m x_m(t) + B_m r(t) \quad (4-9)$$

$$x_m(t) = \left[ x_{m11} \quad x_{m12} \quad x_{m21} \quad x_{m22} \quad x_{m31} \quad x_{m32} \right] \quad (4-10)$$

$\dot{x}_{mi2}$  est la dérivée de  $x_{mi1}$ , pour  $i = 1, 2, 3$ .

$$A_m = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -16 & -8 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -16 & -8 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -16 & -8 \end{pmatrix} \quad (4-11)$$

$$B_m = \begin{pmatrix} 0 & 0 & 0 \\ 16 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 16 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 16 \end{pmatrix} \quad (4-12)$$

La loi de commande est

$$u(t) = - (K - \delta K(t)) x(t) + (K_R + \delta K_R(t)) r(t) \quad (4-13)$$

Elle fait intervenir les gains linéaires

$$K = B^+ (A - A_m) = \begin{pmatrix} 4,8000 & 2,1000 & 0 & 0 & 0 & 0 \\ 0 & 0 & 16,0000 & 6,000 & 0 & 0 \\ 0 & 0 & 0 & 0 & 12,5200 & 5,4775 \end{pmatrix} \quad (4-14)$$

$$K_R = B^+ B_m = \begin{pmatrix} 4,8000 & 0 & 0 \\ 0 & 16,0000 & 0 \\ 0 & 0 & 12,5200 \end{pmatrix} \quad (4-15)$$

où

$$B^+ = (B^T B)^{-1} B^T = \begin{pmatrix} 0 & 0,3000 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1,0000 & 0 & 0 \\ 0 & 0 & 0 & 0 & 12,5200 & 5,4775 \end{pmatrix} \quad (4-16)$$

et les gains adaptatifs



$$\delta K(t) = \int_0^t \alpha y_o(\tau) x^T(\tau) d\tau + \beta y_o(t) x^T(t) \quad (4-17)$$

$$\delta K_R(t) = \int_0^t \alpha y_o(\tau) r^T(\tau) d\tau + \beta y_o(t) r^T(t) \quad (4-18)$$

L'erreur d'état est

$$e(t) = x_m(t) - x(t) \quad (4-19)$$

et 
$$y_o(t) = C_o e(t) \quad (4-20)$$

où

$$C_o = B^T P = \begin{pmatrix} 0,9375 & 0,3255 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0,2813 & 0,0977 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0,3594 & 0,1248 \end{pmatrix} \quad (4-21)$$

avec P solution de l'équation de Lyapunov :

$$P A_m + A_m^T P = -Q, \quad Q > 0 \quad (4-22)$$

$$P = \text{diag} \left( P_1, P_2, P_3 \right) \quad (4-23)$$

$$Q = \text{diag} \left( Q_1, Q_2, Q_3 \right) \quad (4-24)$$

où

$$P_i = \begin{pmatrix} 3,813 & 0,2813 \\ 0,2813 & 9,77 \cdot 10^{-2} \end{pmatrix} \quad (4-25)$$

$$Q_i = \begin{pmatrix} 9 & 0 \\ 0 & 1 \end{pmatrix} \quad (4-26)$$

pour  $i = 1, 2, 3$ .

La condition initiale considérée est

$$x_o(0) = [0,3 \ 0 \ 0,4 \ 0 \ 0,5 \ 0]^T \quad (4-27)$$

Dans la loi de commande,  $\alpha$  et  $\beta$  sont des matrices

carrées de dimension 3.  $\alpha$  doit être définie positive et  $\beta$  semi-définie positive. On prendra :

$$\alpha = \begin{pmatrix} \alpha_1 & 0 & 0 \\ 0 & \alpha_2 & 0 \\ 0 & 0 & \alpha_3 \end{pmatrix} \quad \text{et} \quad \beta = \begin{pmatrix} \beta_1 & 0 & 0 \\ 0 & \beta_2 & 0 \\ 0 & 0 & \beta_3 \end{pmatrix}$$

avec  $\alpha_i > 0$  et  $\beta_i \geq 0$  pour  $i = 1, 2, 3$ .

Les paramètres  $\alpha_i$  et  $\beta_i$  initialement choisis à  $\alpha_i = 25$  et  $\beta_i = 15$ , pour  $i = 1, 2, 3$ , sont ensuite optimisés de manière à minimiser le critère  $J$  défini par l'équation (3-8). Cette optimisation utilise le logiciel de simulation PSI [43]. Il comporte un algorithme d'optimisation non linéaire de Hooke et Jeeves présenté en Annexe A. Les résultats sont rassemblés dans le tableau 4-1.

Tableau 4-1 Résultats obtenus pour le MRAC avant et après optimisation des paramètres.

	Avant optimisation ( $\alpha, \beta$ préalablement choisis)	Après optimisation ( $\alpha, \beta$ déterminés)
$\alpha_1$	25	18,08
$\alpha_2$	25	52,74
$\alpha_3$	25	0,214
$\beta_1$	15	3,46
$\beta_2$	15	9,61
$\beta_3$	15	16,20
J	16,76	15,26

La figure 4-2 montre que le MRAC conduit à une très bonne poursuite et que l'optimisation n'apporte qu'une très faible amélioration.

# Bras manipulateur de classe 4

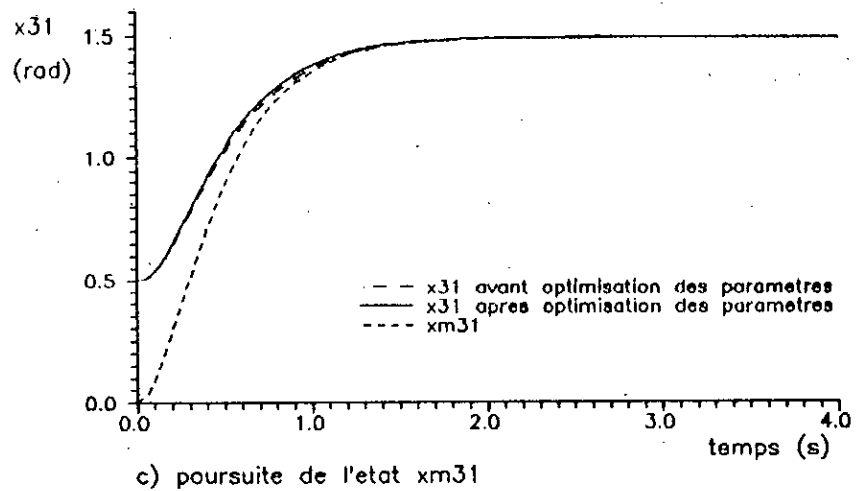
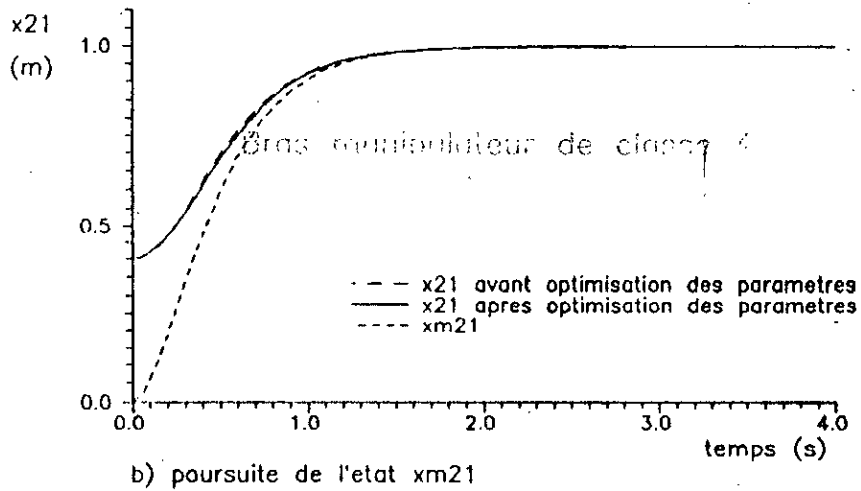
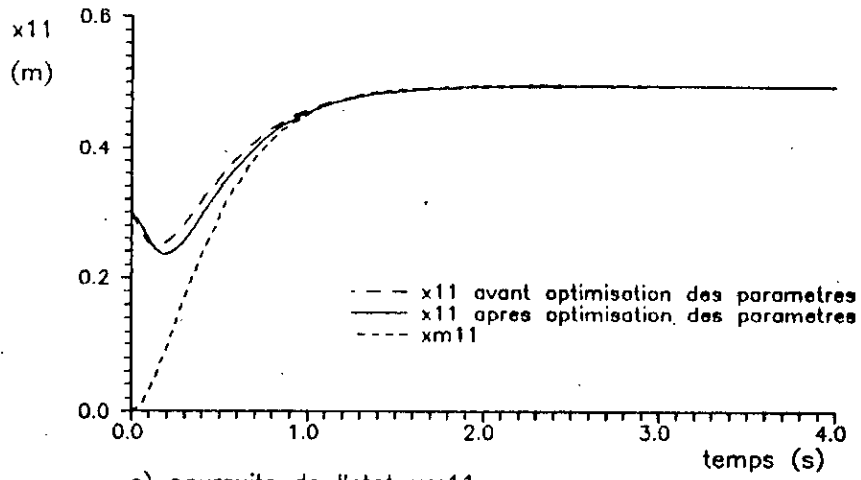


Fig 4-2 Etats du procede et du MR pour le MRAC

#### 4.1.1 Variation des conditions initiales

Nous considérons la loi de commande MRAC utilisant les paramètres  $\alpha_i, \beta_i$  ( $i = 1, 2, 3$ ) optimisés pour la condition initiale  $x_0(0)$  de l'équation (4-27). Puis, nous faisons subir des variations arbitraires

$$\begin{array}{lll} \text{de } 33\% & \text{à } & x_{11}(0) \\ \text{de } 25\% & \text{à } & x_{21}(0) \\ \text{de } 20\% & \text{à } & x_{31}(0) \end{array}$$

La condition initiale devient :

$$x_1(0) = [0,4 \quad 0 \quad 0,5 \quad 0 \quad 0,6 \quad 0]^T \quad (4-28)$$

Les figures 4-3 et 4-4 montrent que les courbes obtenues pour  $x_{ij}$  et  $u_{Fi}$  ( $i = 1, 2, 3, j = 1$ ) pour chacune des conditions initiales ( $x_0(0)$  et  $x_1(0)$ ) deviennent pratiquement confondues après environ 1s.

Avant la modification de la condition initiale, le critère avait la valeur donnée dans le tableau 4-1 :

$$J_{\text{opt}}|_{x_0(0)} = 15,26 \quad (4-29)$$

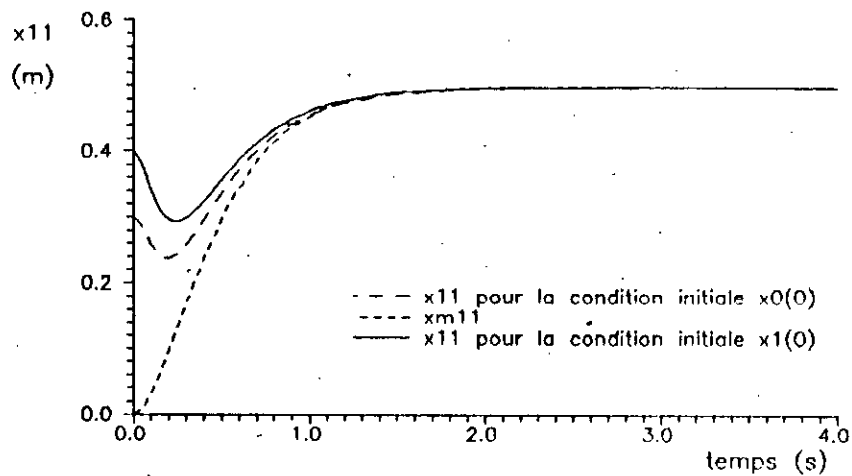
Après la variation de la condition initiale, il devient :

$$J|_{x_1(0)} = 17,87 = J_{\text{opt}}|_{x_0(0)} + \frac{17,10}{100} J_{\text{opt}}|_{x_0(0)}$$

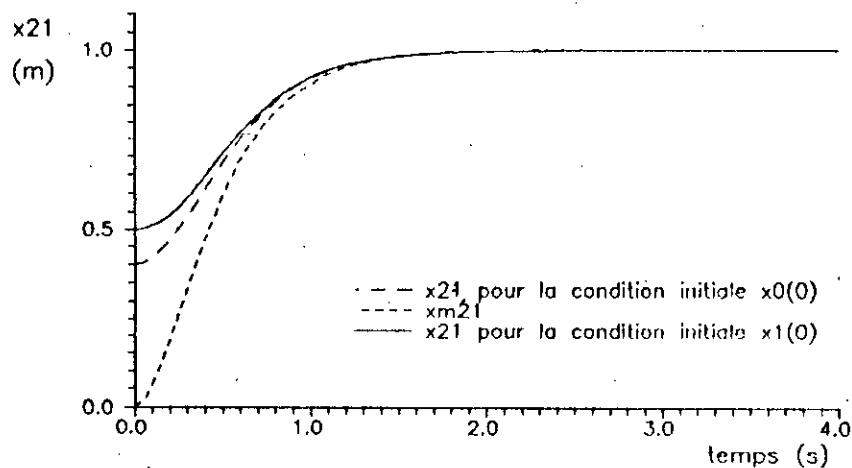
Autrement dit, des variations relatives de 20%, 25% et 33% des valeurs de  $x_{31}(0), x_{21}(0)$  et  $x_{11}(0)$ , respectivement, entraînent une variation relative du critère inférieure à 20%.

Après modification de la condition initiale, le critère obtenu est-il très éloigné de sa nouvelle valeur optimale ? Pour le savoir, nous optimisons les paramètres  $\alpha_i, \beta_i$  pour la condition initiale  $x_1(0)$ . Nous obtenons le critère

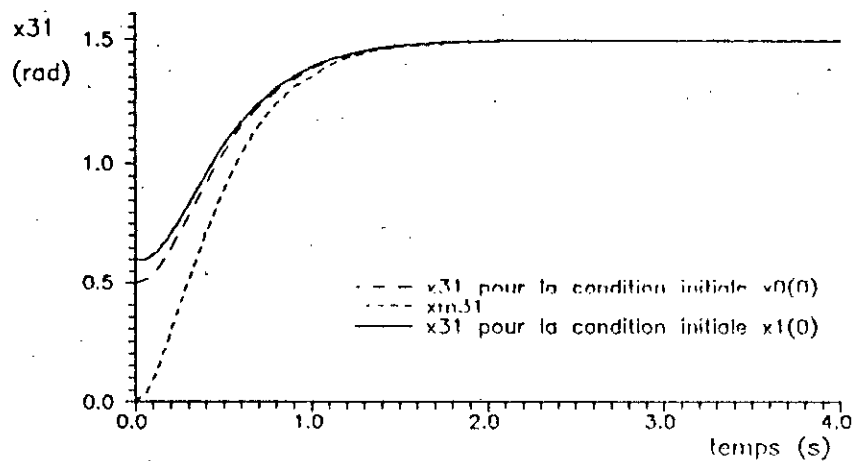
### Bras manipulateur de classe 4



a) poursuite de l'etat  $x_{m11}$



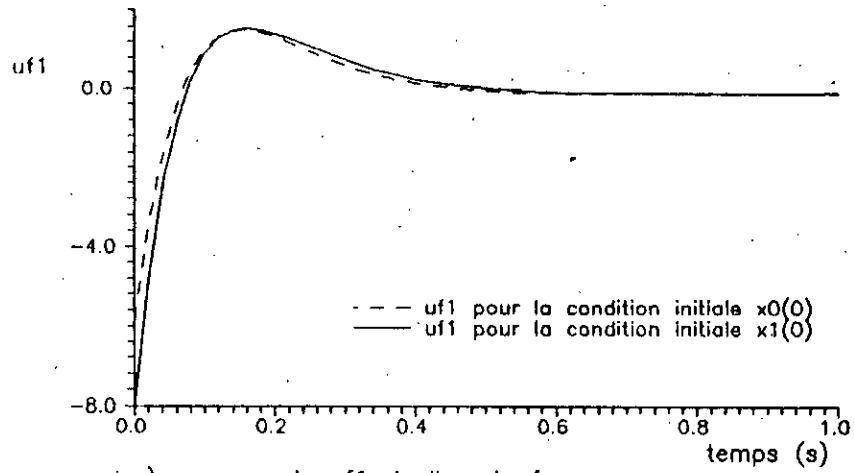
b) poursuite de l'etat  $x_{m21}$



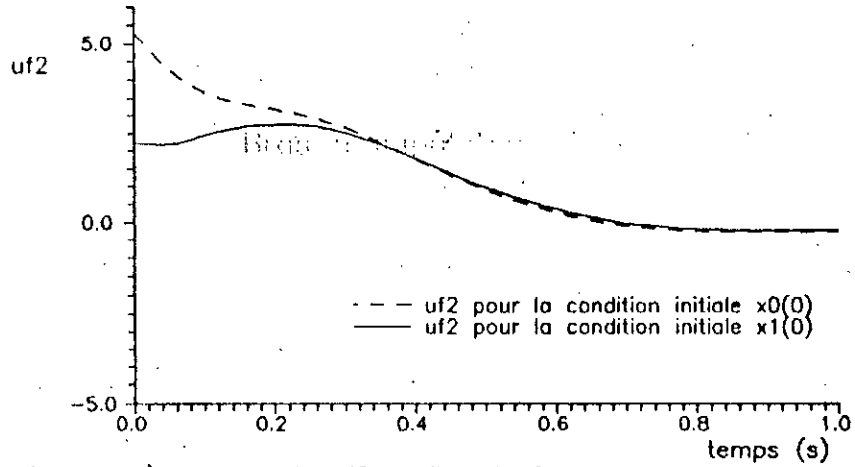
c) poursuite de l'etat  $x_{m31}$

Fig 4-3 Etats du procede et du MR pour le MRAC

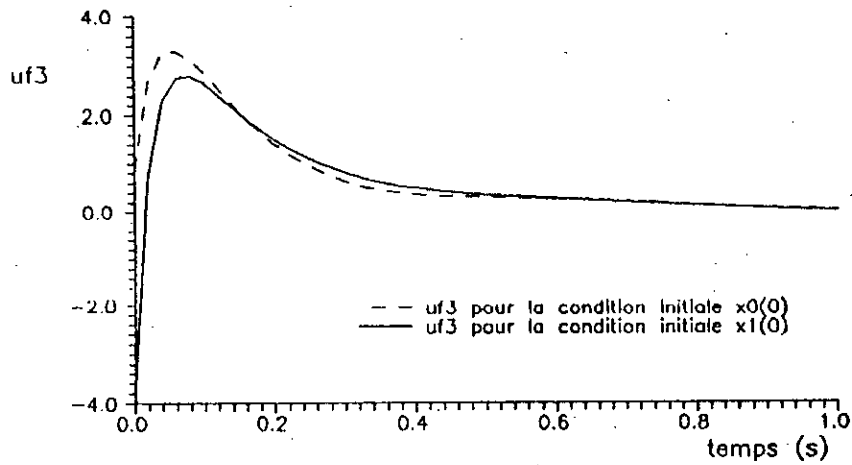
# Bras manipulateur de classe 4



a) composante  $uf_1$  de l'ecart  $uf$



b) composante  $uf_2$  de l'ecart  $uf$



c) composante  $uf_3$  de l'ecart  $uf$

Fig 4-4 Ecart  $uf$  pour le MRAC

$$J_{\text{opt}}|_{x_1(0)} = 17,48 = J|_{x_1(0)} - \frac{2,18}{100} J|_{x_1(0)}$$

L'optimisation n'apporte qu'une amélioration du critère d'environ 2%.

En résumé, des variations comprises entre 20% et 33% des valeurs initiales  $x_{i_1}(0)$  ( $i = 1, 2, 3$ ) n'entraînent qu'une augmentation inférieure à 20% du critère. De plus, la valeur de  $J$  alors obtenue n'est pas très éloignée de sa nouvelle valeur optimale puisque l'optimisation n'apporte qu'une réduction du critère voisine de 2%.

Par conséquent, si l'on peut considérer que la condition initiale  $x(0)$  est un vecteur aléatoire de moyenne connue et de variance faible, on pourra optimiser le critère  $J$  pour  $x(0) = x_{\text{moy}}$  et utiliser les paramètres optimaux  $\alpha_i, \beta_i$  ( $i = 1, 2, 3$ ) ainsi obtenus pour le calcul de la commande  $u$ , même lorsque  $x(0) \neq x_{\text{moy}}$ .

#### 4.1.2 Incertitudes paramétriques

Nous considérons à nouveau la loi de commande MRAC utilisant les paramètres  $\alpha_i$  et  $\beta_i$  optimisés pour la condition initiale  $x_0(0)$  et présentés dans le tableau 4-1. Mais, le procédé est caractérisé par des variations

de +20% de  $b_{21}$ ,  
de +20% de  $a_{44}$ ,  
de -20% de  $a_{56}$

par rapport à leurs valeurs dans les relations (4-3) et (4-4). Nous obtenons

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1,6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0,8 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{pmatrix} \quad (4-30)$$

$$B = \begin{pmatrix} 0 & 0 & 0 \\ 4 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1,278 \end{pmatrix} \quad (4-31)$$

Les gains  $K$  et  $K_R$  de la partie linéaire de la commande MRAC sont calculés au moyen des matrices  $A$  et  $B$  définies par les égalités (4-3) et (4-4). Cela revient à considérer que les résultats de l'identification paramétrique préalable nécessaire à la mise en oeuvre du MRAC ne sont pas parfaits.

Les figures 4-5 et 4-6 montrent les courbes obtenues pour  $x_{ij}$ ,  $x_{mij}$ ,  $u_{Fi}$  ( $i = 1, 2, 3$   $j = 1, 2$ ) en l'absence puis en présence d'incertitudes paramétriques. Les courbes représentant l'état du MR et celles représentant l'état du procédé sont pratiquement confondues après environ 3s.

L'introduction d'incertitudes sur les paramètres de  $A$  et  $B$  entraîne une augmentation du critère qui devient :

$$J_{ip} = 16,14 = J_{opt}|_{x_0(0)} + \frac{5,77}{100} J_{opt}|_{x_0(0)}$$

$J_{opt}|_{x_0(0)}$  est donné par l'égalité (4-29).

Une variation de 20% des paramètres  $b_{21}$ ,  $a_{44}$ ,  $a_{56}$  ne provoque qu'une variation d'environ 6% du critère.

En présence d'incertitudes paramétriques, le critère obtenu est-il très éloigné de sa nouvelle valeur optimale ? Afin de répondre à cette question, nous optimisons les paramètres  $\alpha_i$  et  $\beta_i$  en présence des incertitudes paramétriques. Nous obtenons la valeur du critère suivante :



# Bras manipulateur de classe 4

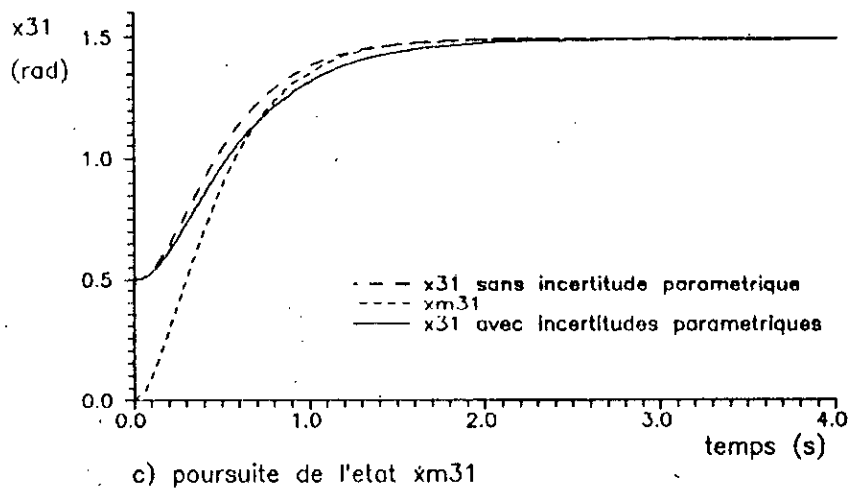
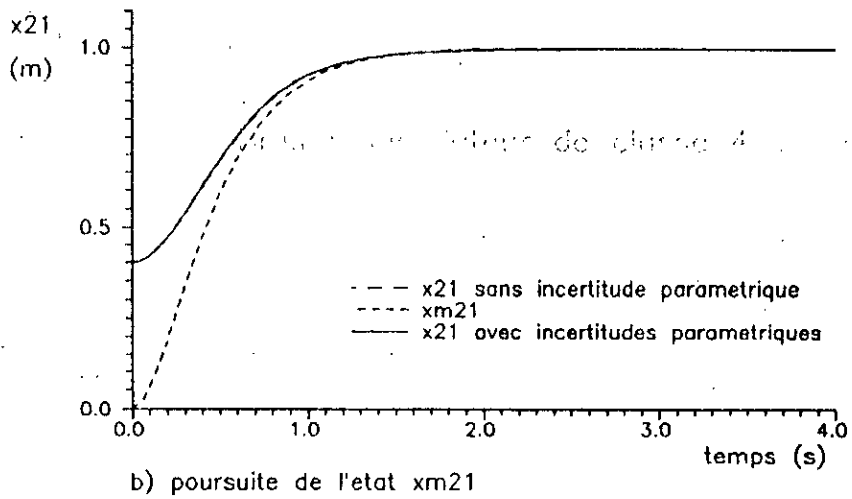
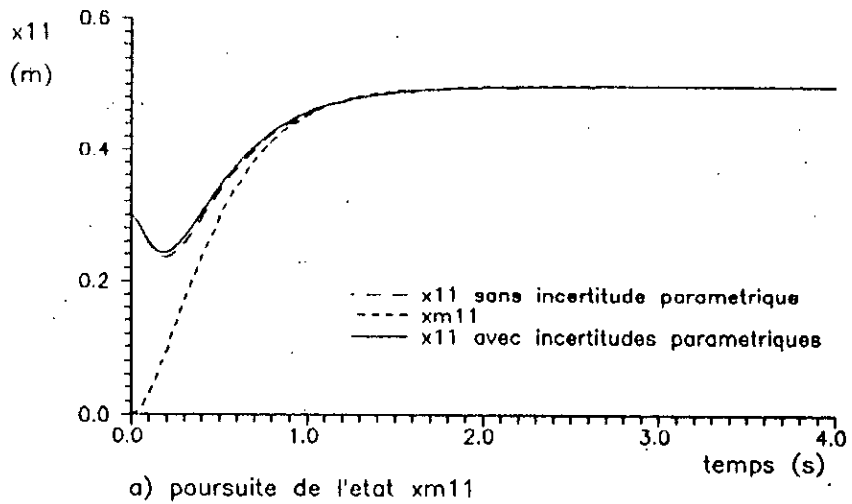
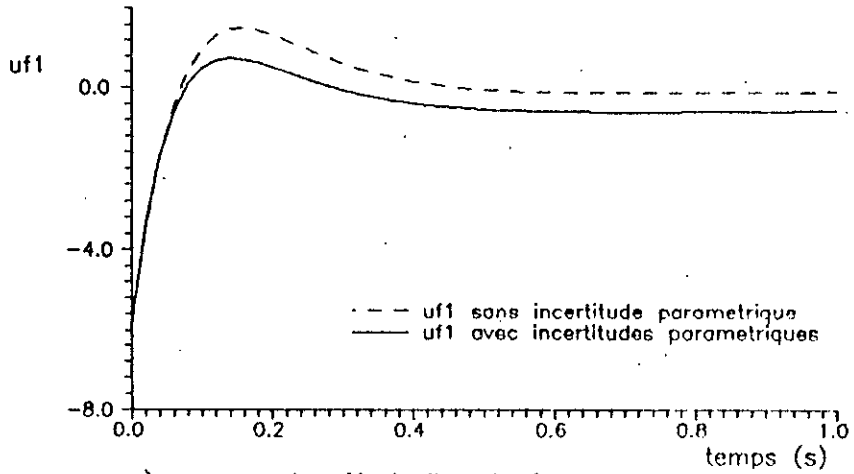
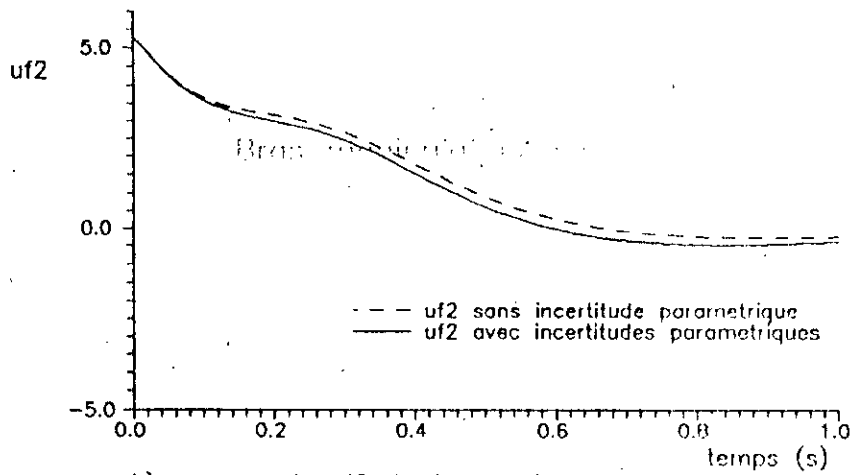


Fig 4-5 Etats du procede et du MR pour le MRAC

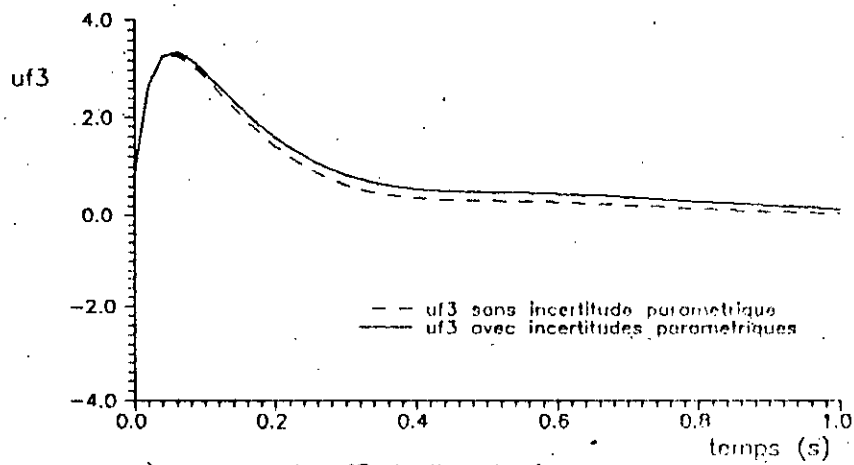
Bras manipulateur de classe 4



a) composante  $uf_1$  de l'ecart  $uf$



b) composante  $uf_2$  de l'ecart  $uf$



c) composante  $uf_3$  de l'ecart  $uf$

Fig 4-6 Ecart  $uf$  pour le MRAC

$$J_{opt}|_{ip} = 16,12 = J_{ip} - \frac{0,12}{100} J_{ip}$$

L'optimisation n'apporte qu'une amélioration du critère inférieure à 0,2% .

En résumé, une variation de 20% des paramètres  $b_{21}$ ,  $a_{44}$ ,  $a_{56}$  entraîne une augmentation inférieure à 6% du critère. De plus, la valeur de J ainsi obtenue est très peu différente de sa valeur optimale (1,001 fois plus élevée).

#### 4-2 Cas d'un servomécanisme

Nous étudions la commande d'un moteur électrique modélisé par une fonction de transfert d'ordre un donnée dans [36]. L'équation d'état de ce système est

$$\dot{x} = -5x + 3,833u \quad (4-32)$$

Le MR est caractérisé par

$$\dot{x}_m = -10x_m + 10r \quad (4-33)$$

La loi de commande est donnée par l'expression (4-13) soit

$$u(t) = - (K - \delta K(t)) x(t) + (K_R + \delta K_R(t)) r(t) \quad (4-34)$$

Les gains linéaires sont déterminés au moyen des relations (2-5-a) et (2-5-b), nous obtenons

$$K = 1,304$$

$$K_R = 2,60$$

En plus de la loi d'adaptation de type PI donnée par les équations (4-17) et (4-18), nous avons considéré une loi de type PID. Le terme dérivateur est alors implémenté au

moyen d'un bloc de fonction de transfert :

$$F(s) = \frac{s + 1}{as + 1} \quad 0 < a < 1 \quad (4-35)$$

et les gains adaptatifs sont de la forme :

$$\delta K(t) = \int_0^t \alpha y_e(\tau) x^T(\tau) d\tau + \beta y_e(t) x^T(t) + \gamma \mathcal{L}^{-1} \left[ (y_e x^T) \frac{s+1}{as+1} \right] \quad (4-36)$$

Le dernier terme de cette expression représente la sortie d'un bloc de fonction de transfert  $F(s)$  et d'entrée  $y_e(t)x^T(t)$ .

L'erreur d'état est

$$e = x_m - x$$

et

$$y_e = C_e e$$

Pour les systèmes d'ordre un, l'équation de Lyapunov devient

$$2a_m p = -q \quad (4-37)$$

et donc

$$p = -q / (2a_m)$$

Comme  $a_m$  est strictement négatif (le MR est stable) et  $q$  est positif arbitraire, toute valeur positive de  $p$  peut être considérée comme solution de l'équation de Lyapunov. Par conséquent, il suffira de prendre  $C_e$  positif.

La condition initiale considérée est

$$x_0(0) = 0,5$$

Les paramètres initialement choisis pour la loi de commande de type PI sont  $\alpha=4$ ,  $\beta=1$ ,  $C_e=0,2$ . Pour la loi PID, nous prenons, en outre,  $\gamma=1$ . Ces paramètres sont ensuite optimisés afin de rendre minimal le critère  $J$  défini par l'équation (3-8). Les résultats sont rassemblés dans le

tableau suivant :

Tableau 4-2 Résultats obtenus pour le MRAC avant et après optimisation des paramètres.

	Avant optimisation (paramètres choisis préalablement )		Après optimisation (paramètres déterminés)	
	$\alpha$	4		55,17
$\beta$	1		4,819	1,511
$C_0$	0,2		0,1813	0,255
$\gamma$		1		0,1656
J	0,2071	0,2648	0,1027	0,1015
loi	PI	PID	PI	PID

Partant des paramètres  $\alpha$  ,  $\beta$  ,  $C_0$  ,  $\gamma$  optimaux trouvés pour chacune des lois de commande, nous avons évalué la variation du critère, lorsque le système est soumis aux perturbations suivantes :

- variations de la condition initiale : nous avons fait varier  $x_0(0)$  de +20% ou -20% de sa valeur.

- incertitudes paramétriques : le processus est caractérisé par l'équation d'état

$$\dot{x} = -4x + 5u$$

et la condition initiale  $x_0(0) = 0,5$  , alors que les gains  $K$  et  $K_R$  sont calculés à l'aide de l'équation d'état

$$\dot{x} = -5x + 3,833u$$

Autrement dit, A et B ont subi respectivement une variation

de 20% et de 30% par rapport à leurs valeurs nominales  
 $A_{\text{nom}} = -5$  et  $B_{\text{nom}} = 3,833$ . La condition initiale est :  
 $x_0(0) = 0,5$ .

- présence d'une perturbation  $d(t)$  : le processus est représenté par l'équation d'état

$$\dot{x} = -5x + 3,833u + d(t)$$

où  $d(t)$  est une perturbation de distribution normale ayant une moyenne nulle ( $\mu = 0$ ) et une variance égale à 4 ( $\sigma^2 = 4$ ). La condition initiale est  $x_0(0) = 0,5$ .

Les résultats obtenus pour le critère J seront présentés dans le tableau 6-2 du chapitre VI de la partie A. Ils seront alors comparés avec ceux obtenus pour la loi de commande MCS.

## CHAPITRE V

## EXEMPLES DE DERIVATION DES LOIS DE COMMANDE MCS

## 5.1 Cas d'un bras manipulateur

Le bras manipulateur [35] est représenté en figure 4-1. Il est caractérisé par les équations (4-1) à (4-8). Les relations (4-9) à (4-12) définissent le modèle de référence.

La loi de commande du MCS est

$$u(t) = \delta K(t) x(t) + \delta K_R(t) r(t) \quad (5-1)$$

avec

$$\delta K(t) = \int_0^t \alpha y_o(\tau) x^T(\tau) d\tau + \beta y_o(t) x^T(t) \quad (5-2)$$

$$\delta K_R(t) = \int_0^t \alpha y_o(\tau) r^T(\tau) d\tau + \beta y_o(t) r^T(t) \quad (5-3)$$

L'erreur d'état est

$$e = x_m(t) - x(t)$$

et

$$y_o(t) = C_o e(t)$$

où

$$C_o(t) = P$$

P est solution de l'équation de Lyapunov :

$$PA_m + A_m^T P = -Q, \quad Q > 0$$

Les valeurs de Q, P sont définies par les relations (4-23) à (4-26).

La condition initiale considérée est

$$x_o(0) = [0,3 \ 0 \ 0,4 \ 0 \ 0,5 \ 0]^T \quad (5-4)$$

Dans la loi de commande,  $\alpha$  et  $\beta$  sont des matrices de

dimension 3 x 6 que l'on choisira, conformément à [37]. Elles auront la forme :

$$\alpha = \begin{pmatrix} 0 & \alpha_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \alpha_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \alpha_3 \end{pmatrix} \quad \beta = \begin{pmatrix} 0 & \beta_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \beta_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \beta_3 \end{pmatrix}$$

avec :  $\alpha_i > 0$  et  $\beta_i \geq 0$  ( $i = 1, 2, 3$ )

Les paramètres initialement choisis à  $\alpha_i = 25$  et  $\beta_i = 15$ , pour  $i = 1, 2, 3$ , sont ensuite optimisés de manière à minimiser le critère J défini par l'équation (3-8). Les résultats sont présentés dans le tableau 5-1.

Tableau 5-1 Résultats obtenus pour le MCS avant et après optimisation des paramètres.

	Avant optimisation ( $\alpha, \beta$ préalablement choisis.)	Après optimisation ( $\alpha, \beta$ déterminés.)
$\alpha_1$	25	92,05
$\alpha_2$	25	$3,59 \cdot 10^{-2}$
$\alpha_3$	25	$6,05 \cdot 10^{-2}$
$\beta_1$	15	19,49
$\beta_2$	15	4,11
$\beta_3$	15	10,38
J	29,22	23,94

La figure 5-1 montre que la poursuite est satisfaisante malgré une dégradation après l'optimisation des paramètres. Cette dégradation est due au fait que nous avons choisi de réduire aussi l'écart entre la commande et sa valeur en régime établi. Ainsi, comme le montrent les figures 5-2 et 5-3, la réduction de  $u_{F2}$  et  $u_{F3}$  s'est faite au détriment



de  $e_{21}$ .

### 5.1.1 Variation des conditions initiales

Nous considérons la loi de commande MCS utilisant les paramètres  $\alpha_i$ ,  $\beta_i$  ( $i = 1, 2, 3$ ) optimisés pour la condition initiale  $x_0(0)$  de l'équation (5-4). Puis, nous faisons subir des variations

$$\begin{aligned} & \text{de } 33\% \quad \text{à } x_{11}(0) \\ & \text{de } 25\% \quad \text{à } x_{21}(0) \\ & \text{de } 20\% \quad \text{à } x_{31}(0) \end{aligned}$$

La condition initiale devient :

$$x_1(0) = [0,4 \quad 0 \quad 0,5 \quad 0 \quad 0,6 \quad 0]^T \quad (5-5)$$

Les figures 5-4 et 5-5 montrent que les courbes obtenues pour  $x_{ij}$ ,  $u_{Fi}$  ( $i = 1, 2, 3$ ,  $j = 1, 2$ ) et pour chacune des conditions initiales deviennent pratiquement confondues après environ 1s.

Avant la modification de la condition initiale, le critère avait la valeur donnée dans le tableau 5-1 :

$$J_{\text{opt}}|_{x_0(0)} = 23,94 \quad (5-6)$$

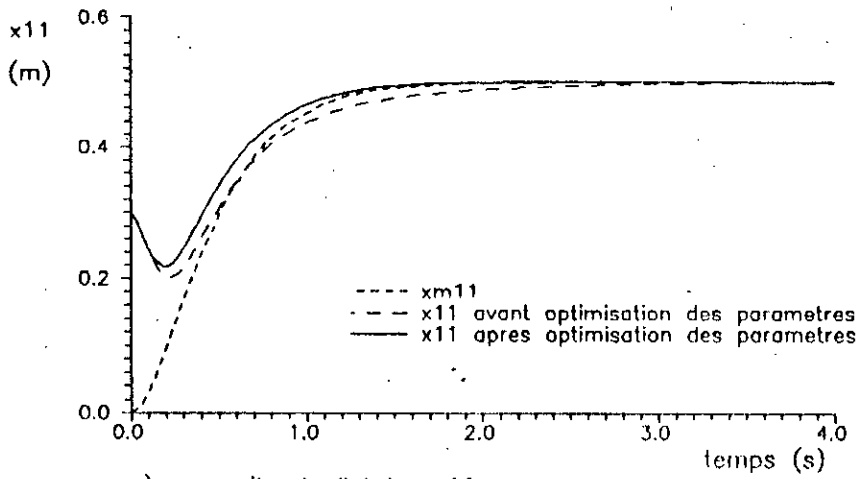
Après la variation de la condition initiale, il devient :

$$J|_{x_1(0)} = 27,38 = J_{\text{opt}}|_{x_0(0)} + \frac{14,37}{100} J_{\text{opt}}|_{x_0(0)}$$

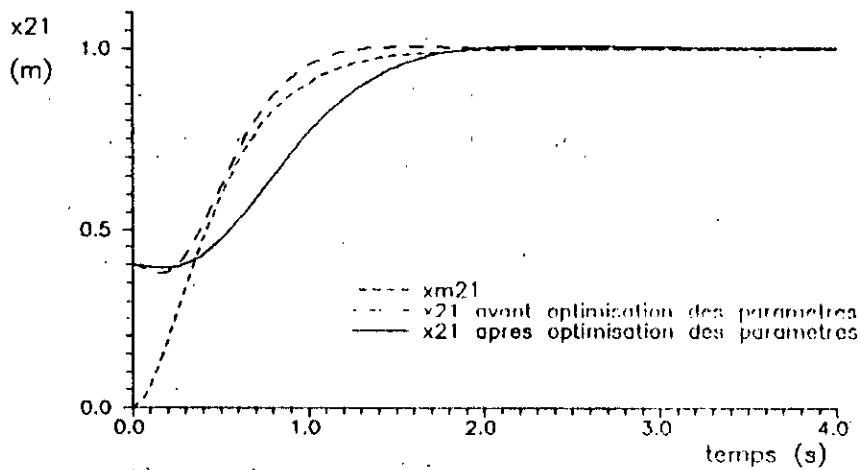
Autrement dit, des variations relatives de 20%, 25% et 33% des valeurs de  $x_{31}(0)$ ,  $x_{21}(0)$  et  $x_{11}(0)$ , respectivement entraînent une variation relative du critère inférieure à 15%

Après modification de la condition initiale, le critère obtenu est-il très éloigné de sa nouvelle valeur optimale ?

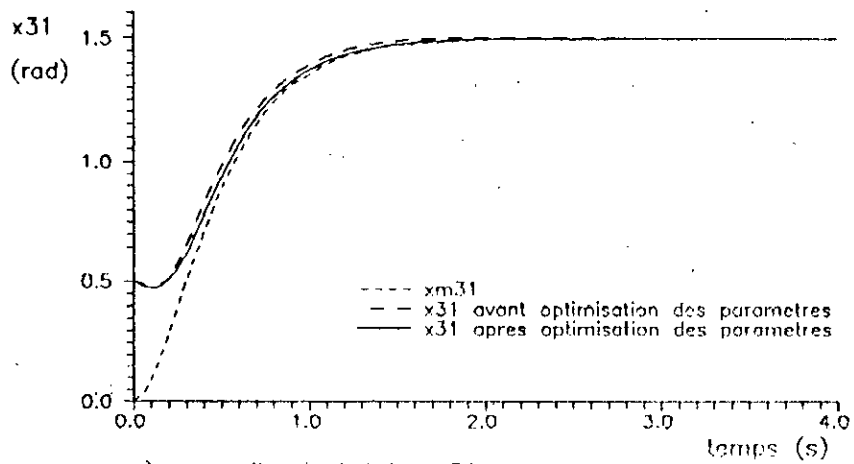
# Bras manipulateur de classe 4



a) poursuite de l'etat  $x_{m11}$



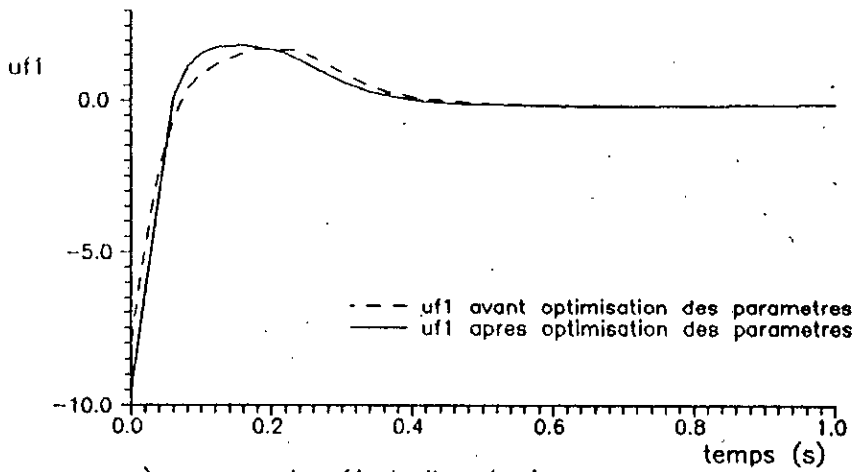
b) poursuite de l'etat  $x_{m21}$



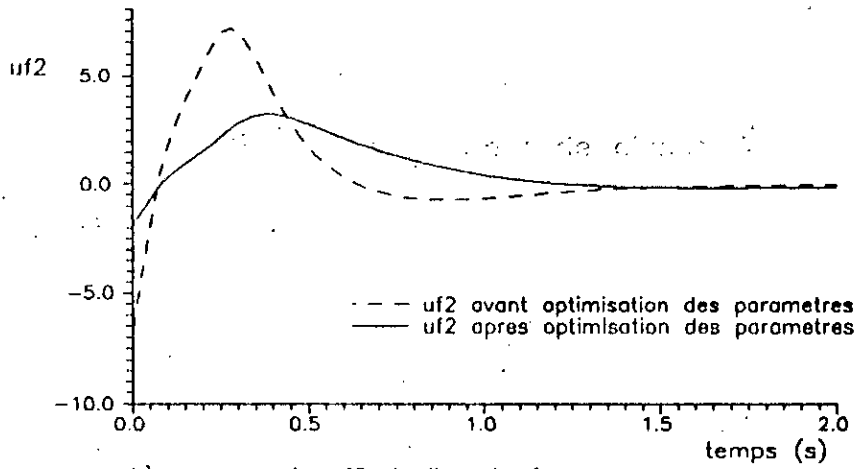
c) poursuite de l'etat  $x_{m31}$

Fig 5-1 Etats du procede et du MR pour le MCS

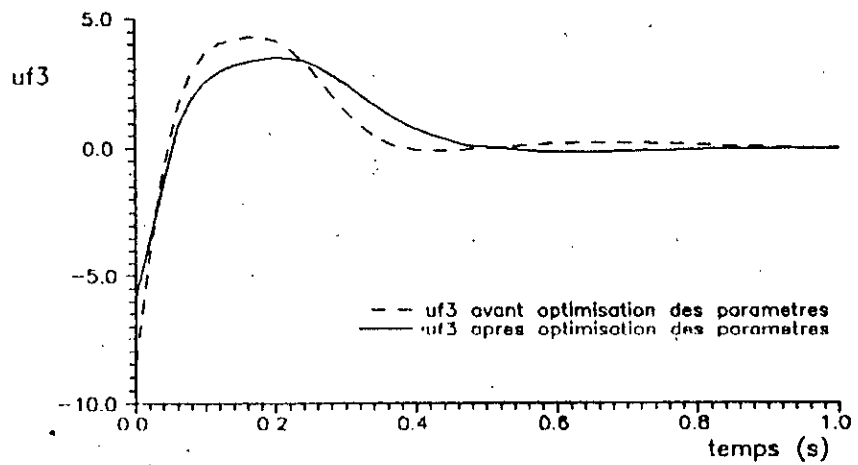
# Bras manipulateur de classe 4



a) composante uf1 de l'ecart uf



b) composante uf2 de l'ecart uf



c) composante uf3 de l'ecart uf

Fig 5-2 Ecart uf pour le MCS

### Bras manipulateur de classe 4

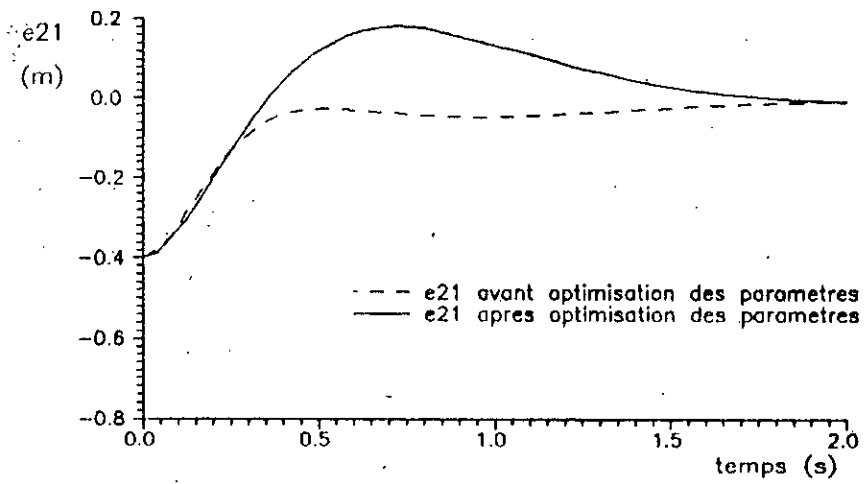


Fig 5-3 Composante  $e_{21}$  de l'erreur de poursuite du MCS

### Bras manipulateur de classe 4

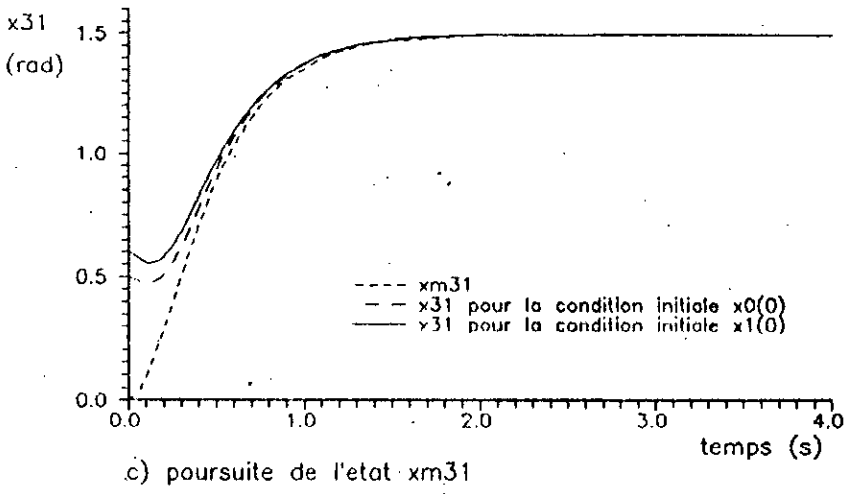
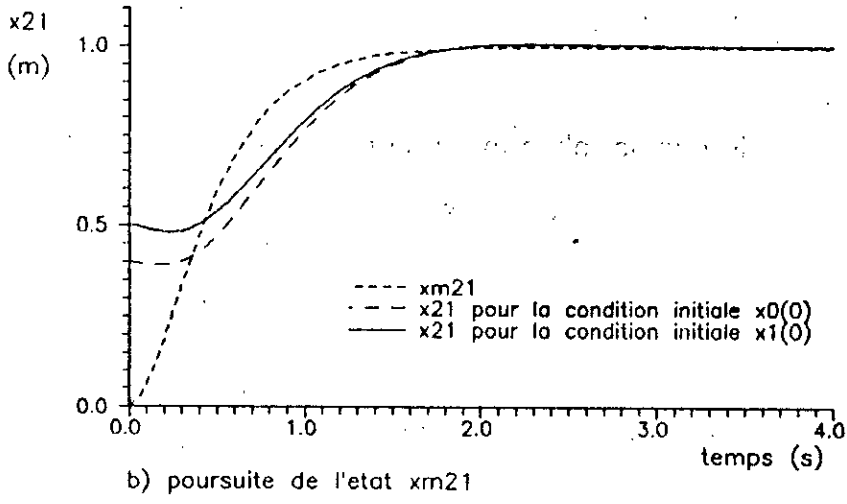
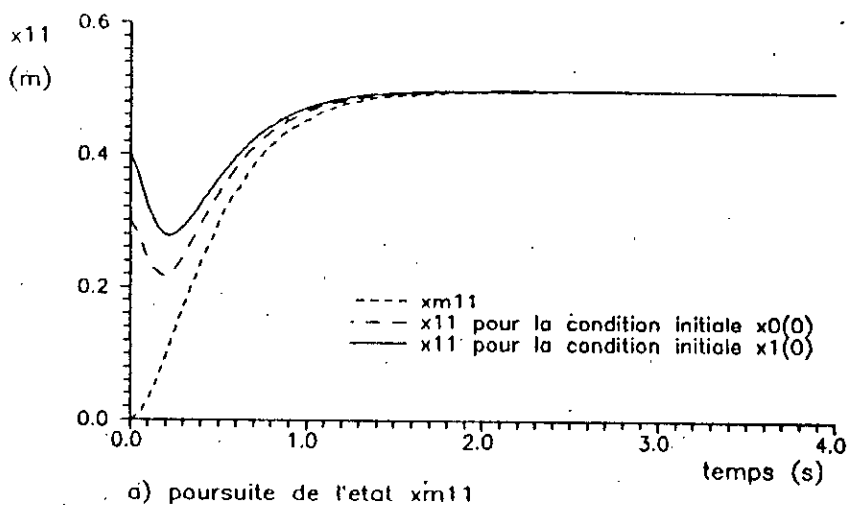


Fig 5-4 Etats du procede et du MR pour le MCS

# Bras manipulateur de classe 4

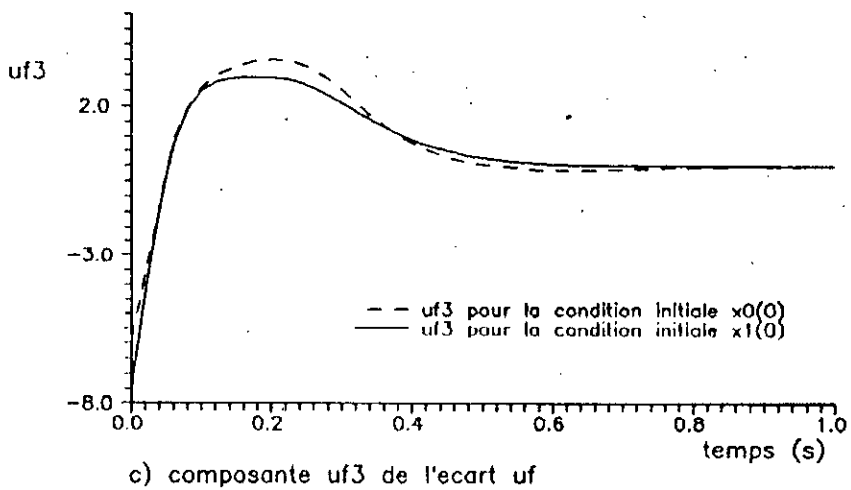
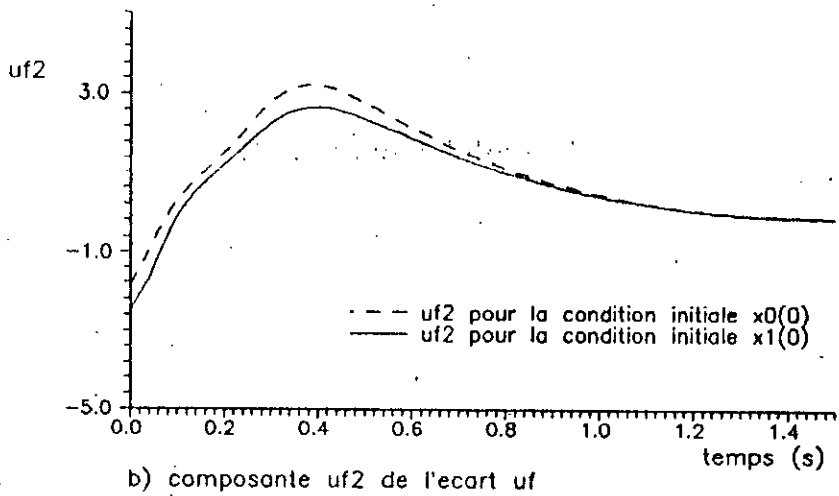
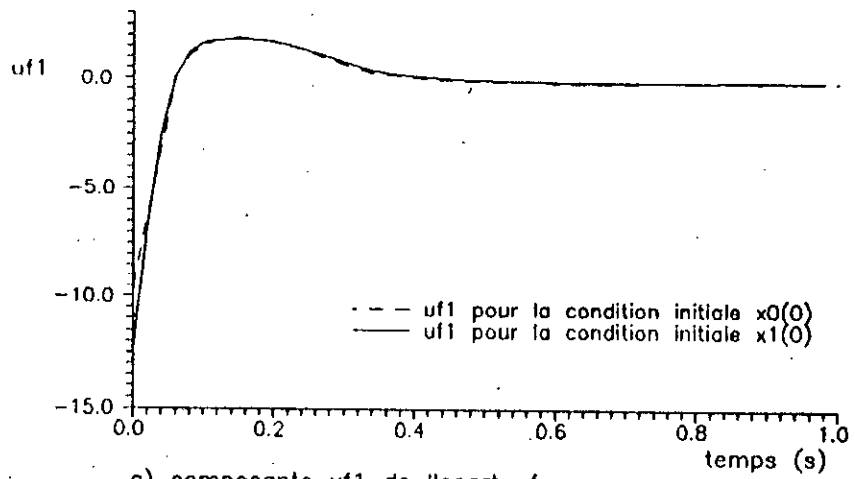


Fig 5-5 Ecart  $uf$  pour le MCS

Pour répondre à cette question, nous optimisons les paramètres  $\alpha_i$  et  $\beta_i$  pour la condition initiale  $x_1(0)$ . Nous obtenons le critère

$$J_{\text{opt}} |_{x_1(0)} = 27,22 = J |_{x_1(0)} - \frac{0,6}{100} J |_{x_1(0)}$$

L'optimisation n'apporte qu'une amélioration du critère de 0,6%.

En résumé, des variations comprises entre 20% et 33% des valeurs initiales  $x_{i1}(0)$  ( $i = 1, 2, 3$ ) n'entraînent qu'une augmentation inférieure à 15% du critère. En outre, le critère ainsi obtenu est très proche de sa valeur optimale puisque l'optimisation n'apporte qu'une réduction du critère voisine de 0,6%.

Par conséquent, comme pour le MRAC, si l'on peut considérer que la condition initiale  $x(0)$  est un vecteur aléatoire de moyenne connue et de variance faible, on pourra optimiser le critère  $J$  pour  $x(0) = X_{\text{moy}}$ .

### 5.1.2 Incertitudes paramétriques

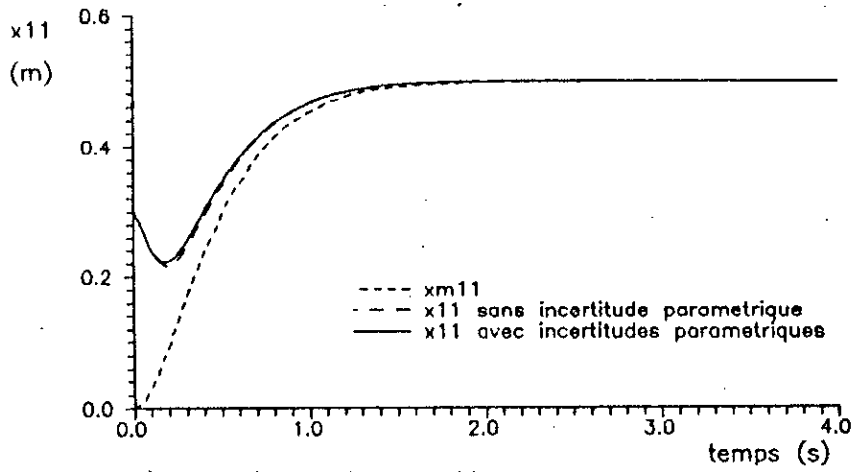
Nous considérons à nouveau la loi de commande MCS utilisant les paramètres  $\alpha_i$  et  $\beta_i$  optimisés pour la condition initiale  $x_0(0)$  et présentés dans le tableau 5-1. Mais, le procédé est caractérisé par des variations

de +20% de  $b_{21}$ ,  
 de +20% de  $a_{44}$ ,  
 de -20% de  $a_{56}$

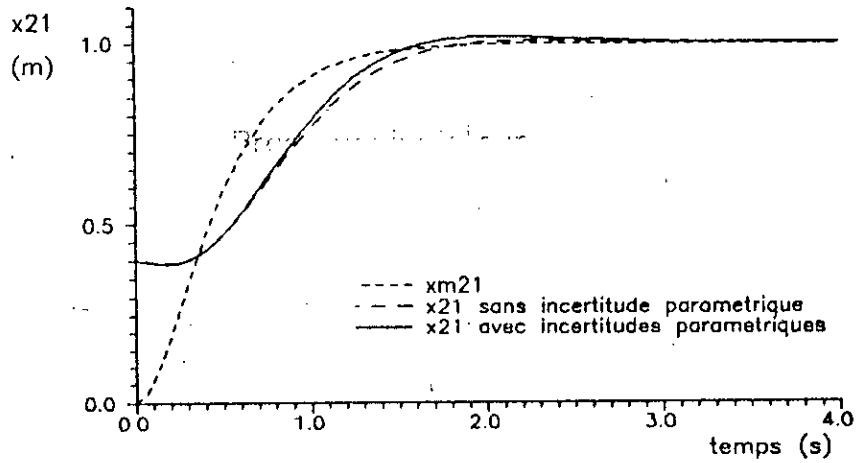
par rapport à leurs valeurs dans les relations (4-3) et (4-4). Les matrices d'évolution et de commande du procédé sont alors donnés par les égalités (4-30) et (4-31).

Les figures 5-6 et 5-7 montrent les courbes obtenues pour  $x_{ij}$ ,  $x_{mij}$ ,  $u_{Fi}$  ( $i = 1, 2, 3$ ,  $j = 1, 2$ ) en l'absence et en présence d'incertitudes paramétriques. Les courbes

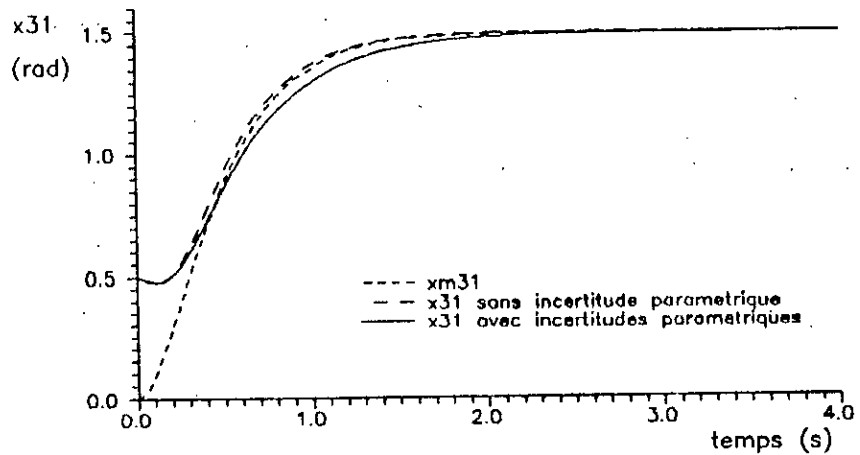
Bras manipulateur de classe 4



a) poursuite de l'etat  $x_{m11}$



b) poursuite de l'etat  $x_{m21}$

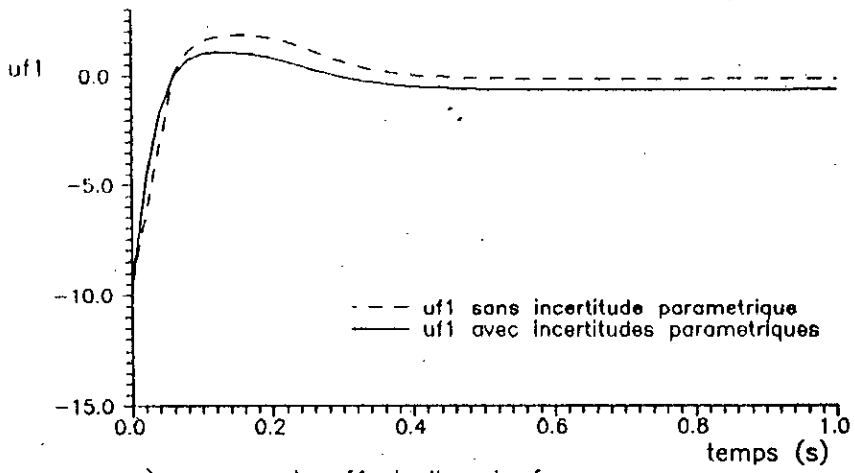


c) poursuite de l'etat  $x_{m31}$

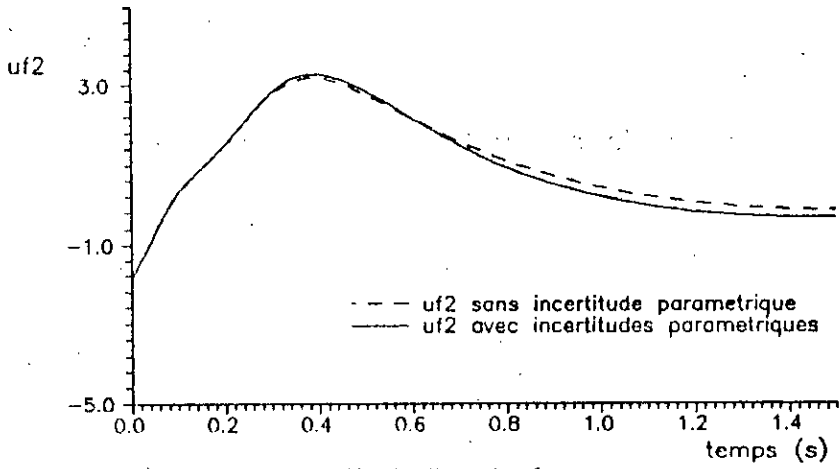
Fig 5-6 Etats du procede et du MR pour le MCS



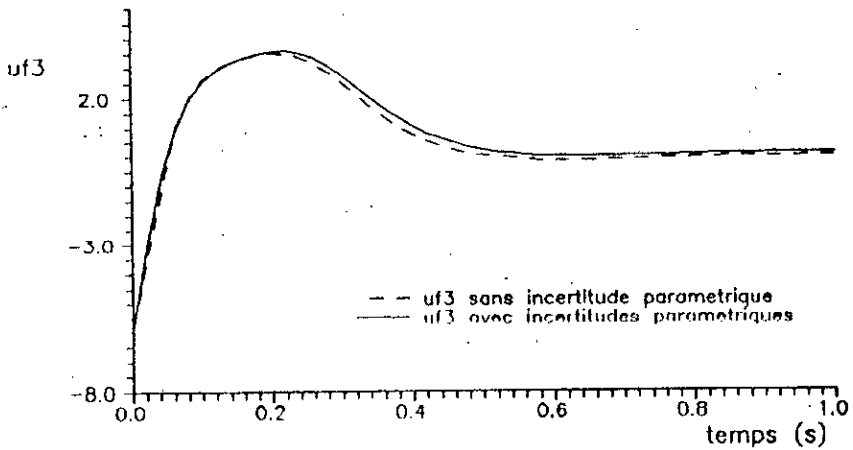
Bras manipulateur de classe 4



a) composante  $uf_1$  de l'ecart  $uf$



b) composante  $uf_2$  de l'ecart  $uf$



c) composante  $uf_3$  de l'ecart  $uf$

Fig 5-7. Ecart  $uf$  pour le MCS

représentant l'état du MR et celles représentant l'état du procédé deviennent rapidement confondues.

La modification des paramètres de A et B entraîne une augmentation du critère. Il devient

$$J_{ip} = 26,11 = J_{opt} |_{x_0(0)} + \frac{9,06}{100} J_{opt} |_{x_0(0)}$$

$J_{opt} |_{x_0(0)}$  est donné par l'égalité (5-6).

Une variation de 20% des paramètres  $b_{21}$ ,  $a_{44}$ ,  $a_{56}$  ne provoque qu'une variation d'environ 9% du critère.

En présence d'incertitudes paramétriques, le critère obtenu est-il très éloigné de sa nouvelle valeur optimale ? Afin de répondre à cette question, nous optimisons les paramètres  $\alpha_i$  et  $\beta_i$  en présence des incertitudes paramétriques. Nous obtenons la valeur du critère suivante :

$$J_{opt} |_{ip} = 26,08 = J_{ip} - \frac{0,11}{100} J_{ip}$$

L'optimisation n'apporte qu'une amélioration du critère inférieure à 0,2% .

En résumé, une variation de 20% des paramètres  $b_{21}$ ,  $a_{44}$ ,  $a_{56}$  entraîne une augmentation inférieure à 10% du critère. De plus, la valeur de J ainsi obtenue est très proche de sa valeur optimale (1,001 fois plus élevée).

## 5-2 Cas d'un servomécanisme

Le procédé étudié est un moteur électrique [36] représenté par l'équation d'état (4-32). Le modèle de référence est décrit par l'équation d'état (4-33).

La loi de commande MCS est donnée par l'équation (5-1). Comme au paragraphe 4.2, en plus de la loi d'adaptation PI

donnée par les équations (4-17) et (4-18), nous considérons la loi PID. Pour cette dernière, les gains adaptatifs sont de la forme (4-36), soit

$$\delta K(t) = \int_0^t \alpha y_o(\tau) x^T(\tau) d\tau + \beta y_o(t) x^T(t) + \gamma \mathcal{L}^{-1} \left[ (y_o x^T) \frac{s+1}{as+1} \right]$$

L'erreur d'état est

$$e = x_m - x$$

et

$$y_o = C_o' e$$

$C_o'$  est solution de l'équation de Lyapunov (4-37)

$$C_o' = p = -q/(2a_m)$$

Comme nous l'avons souligné au paragraphe 4.2, il suffit de prendre  $C_o'$  positif.

La condition initiale considérée est

$$x_o(0) = 0,5$$

Les paramètres initialement choisis pour la loi de commande de type PI sont  $\alpha=4$ ,  $\beta=1$ ,  $C_o'=0,2$ . Pour la loi PID, nous prenons, en outre,  $\gamma=1$ . Ces paramètres sont ensuite optimisés afin de rendre minimal le critère  $J$  défini par l'équation (3-8). Les résultats sont rassemblés dans le tableau 5-2.

Tableau 5-2 Résultats obtenus pour le MCS avant et après optimisation des paramètres.

	Avant optimisation (paramètres choisis préalablement )		Après optimisation (paramètres déterminés)	
$\alpha$	4		1,175	21,58
$\beta$	1		0,413	5,118
$C_0$	0,2		4,603	0,288
$\gamma$		1		0,144
J	5,246	2,420	0,712	0,707
loi	PI	PID	PI	PID

Partant des paramètres  $\alpha$ ,  $\beta$ ,  $C_0$ ,  $\gamma$  optimaux trouvés pour chacune des lois de commande, nous avons évalué la variation du critère, lorsque le système est soumis à des perturbations. Ces dernières ont été décrites au paragraphe 4.2.

Les résultats obtenus pour le critère J seront présentés dans le tableau 6-3 du chapitre VI de la partie A. Ils seront alors comparés aux résultats obtenus pour la loi de commande MRAC.

## CHAPITRE VI

## ANALYSE COMPARATIVE DU MRAC ET DU MCS

## 6.1 Cas d'un bras manipulateur

La figure 4-2 montre que le MRAC conduit à une très bonne poursuite. Par contre, pour le MCS, la poursuite est moins bonne comme on peut le voir en figure 5-1. En effet, pour le MRAC, on peut considérer que les courbes représentant l'état du processus et celles représentant l'état du MR sont confondues après une seconde. Pour le MCS par contre, il faut attendre environ trois secondes pour obtenir ce résultat.

Cette différence s'explique par le fait que l'algorithme MRAC dispose, dès l'instant initial, d'une connaissance a priori sur le système à commander. Si les paramètres du procédé ont été correctement identifiés, les gains  $K$  et  $K_R$  peuvent être calculés avec précision.

L'algorithme MCS ne dispose d'aucune connaissance préalable sur les paramètres du procédé à commander et pendant la phase d'adaptation, il recherche une commande adéquate sans autre information que l'erreur de poursuite.

Toutefois, après cette phase initiale d'adaptation, le MCS rattrape la performance du MRAC.

Il semble donc que l'algorithme MCS plus simple à mettre en oeuvre, devient très rapidement aussi performant que le MRAC.

En ce qui concerne le critère  $J$ , les résultats obtenus pour le MRAC et le MCS sont rassemblés dans le tableau 6-1 où

-  $J_0$  est le critère optimal quand la condition initiale est  $x_0(0)$ .

-  $\Delta J_0$  est la variation relative du critère  $J_0$  après introduction d'une perturbation dans le système (modification

de la condition initiale  $x_0(0)$  ou introduction d'incertitudes paramétriques).

- Le critère prend alors la valeur  $J_1$ .
- $\Delta J_1$  est la variation relative que doit subir  $J_1$  pour prendre la valeur  $J_2$ .
- $J_2$  est le critère optimal en présence de la perturbation considérée.

Tableau 6-1 Tableau comparatif des résultats du MRAC et du MCS.

CONDITIONS DE FONCTIONNEMENT	CRITERE	MRAC	MCS
$x_0(0)$	$J_0$	15,26	23,94
INCERTITUDES PARAMETRIQUES	$\Delta J_0$	+ 5,77%	+ 9,06%
	$J_1$	16,14	26,11
	$\Delta J_1$	- 0,12%	- 0,11%
	$J_2$	16,12	26,08
VARIATIONS DES CONDITIONS INITIALES $x_1(0)$	$\Delta J_0$	+ 17,10	+ 14,37%
	$J_1$	17,87	27,38
	$\Delta J_1$	- 2,18%	- 0,6%
	$J_2$	17,48	27,22

Dans les mêmes conditions de fonctionnement, la valeur du critère est beaucoup plus élevée pour l'algorithme MCS que pour l'algorithme MRAC. Par exemple, pour la condition initiale  $x_0(0) = [0,3 \ 0 \ 0,4 \ 0 \ 0,5 \ 0]^T$ ,  $J_{MRAC} = 15,26$  et  $J_{MCS} = 23,94$ .

La différence est essentiellement due à la phase initiale du processus d'adaptation. En effet, les résultats obtenus en poursuite pour chacune des deux lois de commande deviennent

rapidement à peu près identiques comme le montrent les figures 4-2, 4-3, 4-5 et 5-1, 5-4, 5-6.

Si on introduit des incertitudes paramétriques dans le processus, il en résulte une variation relative du critère supérieure dans le cas du MCS à celle obtenue pour le MRAC (cf. les valeurs respectives de  $\Delta J_0$ ).

Si au lieu des incertitudes paramétriques, le processus subit un changement de la condition initiale  $x_0(0)$ , on constate que la variation relative résultante du critère est inférieure dans le cas du MCS à celle obtenue pour le MRAC.

Dans les deux cas, le critère obtenu après modification des conditions de fonctionnement est relativement plus proche de sa valeur optimale (optimale dans les nouvelles conditions) pour l'algorithme MCS que pour l'algorithme MRAC.

Pour ce processus d'ordre 6, à 3 entrées, et en ce qui concerne le critère  $J_0$ , l'algorithme MCS apparaît moins sensible que le MRAC à la variation des conditions initiales; le critère augmente moins (en valeur relative). Tandis que l'algorithme MRAC est moins sensible aux incertitudes paramétriques.

## 6.2 Cas d'un servomécanisme

Les tableaux 6-2 et 6-3 rassemblent les résultats obtenus pour les algorithmes MRAC et MCS respectivement. Dans ces tableaux  $J_0$ ,  $\Delta J_0$ ,  $J_1$ ,  $\Delta J_1$ ,  $J_2$  ont la même signification que pour le tableau 6-1.

### 6.2.1 Lois PI et PID

Quel que soit l'algorithme considéré MRAC ou MCS, l'amélioration apportée au critère par l'utilisation de la loi de type PID au lieu de la loi PI est très faible. En effet, le rapport entre les critères obtenus pour chacune des lois est environ égal à 1 :  $\frac{J_{PID}}{J_{PI}} \simeq 1$ , comme on peut le voir dans les tableaux 6-2 et 6-3.

La loi PI, plus simple que la loi PID, donne des résultats comparables à ceux de cette dernière en ce qui concerne les valeurs du critère. Désormais, nous n'utiliserons plus la loi de type PID.

Tableau 6-2 Résultats obtenus pour l'algorithme MRAC.

Conditions de fonctionnement	critère	loi PI	loi PID	$J_{PID} / J_{PI}$
$x(0) = 0,5$	$J_0$	0,1027	0,1015	0,99
Incertitudes paramétriques	$\Delta J_0$	+ 48%	+ 45%	0,97
	$J_1$	0,1519	0,1477	
	$\Delta J_1$	- 24,6%	- 22,95%	
	$J_2$	0,1152	0,1138	
Perturbation	$\Delta J_0$	+ 75%	+ 78%	1,01
	$J_1$	0,1797	0,1810	
	$\Delta J_1$	- 4%	- 4,70%	
	$J_2$	0,1725	0,1725	
Variations des conditions initiales $x(0) = 0,4$	$\Delta J_0$	+ 8,67%	+ 8,18%	0,98
	$J_1$	0,1116	0,1098	
	$\Delta J_1$	- 1,97%	- 1,82%	
	$J_2$	0,1094	0,1078	
initiales $x(0) = 0,6$	$\Delta J_0$	+ 10,81%	+ 12,31%	1,00
	$J_1$	0,1138	0,1140	
	$\Delta J_1$	- 2,46%	- 2,63%	
	$J_2$	0,1110	0,1110	



Tableau 6-3 Résultats obtenus pour l'algorithme MCS.

Conditions de fonctionnement	critère	loi PI	loi PID	$J_{PID}/J_{PI}$
$x(0) = 0,5$	$J_0$	0,7121	0,7072	0,99
Incetitudes paramétriques	$\Delta J_0$	- 35%	- 35%	
	$J_1$	0,460	0,4589	1,00
	$\Delta J_1$	- 1,56%	- 1,68%	
	$J_2$	0,4528	0,4512	1,00
Perturbation	$\Delta J_0$	+ 15%	+ 15,79%	
	$J_1$	0,8198	0,8189	1
	$\Delta J_1$	- 0,78%	- 0,85%	
	$J_2$	0,8134	0,8119	1
Variation des conditions initiales	$\Delta J_0$	- 15,92%	- 16,29%	
	$J_1$	0,5987	0,5920	1
	$\Delta J_1$	- 1,08%	- 1,32%	
	$J_2$	0,5922	0,5842	0,99
conditions initiales	$\Delta J_0$	+ 19,60%	+ 20,05%	
	$J_1$	0,8517	0,8490	1
	$\Delta J_1$	- 0,81%	- 0,87%	
	$J_2$	0,8448	0,8416	1

### 6.2.2 Lois MRAC et MCS

La première observation que l'on peut faire, d'après les tableaux 6-2 et 6-3, est que dans les mêmes conditions de fonctionnement, le critère est beaucoup plus élevé pour l'algorithme MCS que pour l'algorithme MRAC.

Par exemple, lorsque la condition initiale est  $x(0) = 0,5$  et en l'absence d'incertitudes paramétriques, ou de toute autre perturbation :  $J_{MRAC} \approx 0,1$  et  $J_{MCS} \approx 0,7$ .

Si on introduit dans le processus une perturbation ou

des incertitudes paramétriques, il en résulte une variation relative du critère très supérieure dans le cas du MRAC à celle obtenue pour le MCS (cf.  $\Delta J_0$  dans les tableaux 6-2 et 6-3).

Si, au lieu d'introduire l'une des modifications des conditions de fonctionnement citées ci-dessus, on fait varier la condition initiale  $x(0)$ , on constate que la variation relative résultante du critère est supérieure dans le cas de l'algorithme MCS à celle obtenue pour le MRAC.

Dans tous les cas, le critère obtenu après modification des conditions de fonctionnement est beaucoup plus proche de sa valeur optimale (optimale dans ces nouvelles conditions), pour l'algorithme MCS que pour l'algorithme MRAC.

En ce qui concerne le critère  $J_0$ , l'algorithme MCS apparaît moins sensible que le MRAC vis-à-vis des incertitudes paramétriques et de la perturbation  $d(t)$ . Tandis que l'algorithme MRAC est moins sensible à la variation des conditions initiales.

Ces derniers résultats sont en contradiction avec ceux obtenus pour le procédé d'ordre 6. On ne peut pas, a priori, dire que la variation relative,  $\Delta J_0$ , du critère  $J_0$  provoquée par une modification donnée des conditions de fonctionnement sera plus importante ou plus faible pour le MRAC que pour le MCS. En fait, cela dépendra du procédé et de la modification des conditions de fonctionnement considérés.

### 6.3 Conclusion

L'algorithme MCS peut devenir (après une courte phase initiale d'adaptation) aussi performant, en ce qui concerne la poursuite, que l'algorithme MRAC. Il peut donc remplacer ce dernier de manière efficace lorsque les paramètres du procédé à commander sont inconnus.

Pour l'application de l'algorithme MCS, nous avons

adopté la structure PI pour la loi d'adaptation et nous avons choisi les paramètres  $\alpha$  et  $\beta$  qui assurent les meilleures performances du système adaptatif en fonction du critère J.

Les résultats précédents montrent que si l'on optimise les paramètres  $\alpha$  et  $\beta$  pour une situation donnée puis que l'on modifie les conditions de fonctionnement, le critère de performance se dégrade mais reste généralement proche de sa valeur optimale. Il apparaît donc intéressant d'optimiser les paramètres  $\alpha$  et  $\beta$ . Cependant, la mise en oeuvre de la procédure d'optimisation suppose la connaissance d'un modèle paramétrique du processus et des valeurs de ses paramètres. Ceci nous fait perdre un avantage important de l'algorithme MCS : à savoir, il n'est pas nécessaire de procéder à une identification paramétrique du procédé. Existe-t-il, alors, un moyen de choisir des valeurs adéquates de  $\alpha$  et de  $\beta$  sans connaître les paramètres du modèle du procédé à commander ? Le chapitre suivant tente de répondre à cette question.

## CHAPITRE VII

### ANALYSE DE LA CONVERGENCE DU MCS

#### 7.1 Généralités

Dans les chapitres précédents, les paramètres  $(\alpha, \beta)$  de la loi de commande étaient choisis par un algorithme d'optimisation chargé de minimiser un certain critère de performance. Nous n'avions alors aucun contrôle sur ce choix. Pour proposer des recommandations sur le choix de  $\alpha$  et  $\beta$ , nous commençons par présenter des résultats concernant l'effet de ces paramètres sur l'erreur de poursuite et sur la commande. Puis, nous développons une extension des travaux de Stoten et Benchoubane [39] à un système d'ordre 2.

#### 7.2 Influence des paramètres de convergence sur la commande

##### et sur l'erreur du MCS

Nous considérons le procédé et le MR décrits par les équations d'état ci-dessous :

$$\dot{x}_p = \begin{pmatrix} 0 & 1 \\ -4 & -7 \end{pmatrix} x_p + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u \quad (7-1)$$

$$\dot{x}_m = \begin{pmatrix} 0 & 1 \\ -10 & -10 \end{pmatrix} x_m + \begin{pmatrix} 0 \\ 2 \end{pmatrix} r \quad (7-2)$$

Dans l'équation de Lyapunov, la matrice  $Q$  est choisie égale à la matrice identité

$$Q = I_2$$

L'erreur de poursuite et la commande pour différentes

valeurs de  $\alpha$  et  $\beta$  sont présentées dans les figures 7-1 à 7-7.

### 7.2.1 Effet de $\alpha$ sur la commande

Une valeur élevée de  $\alpha$  provoque un pic important dans la commande (cf la courbe obtenue pour  $\alpha = 500$  en figure 7-1).

### 7.2.2 Effet de $\beta$ sur la commande

La figure 7-2 montre que plus  $\beta$  est grand, plus la commande augmente rapidement, au début de l'adaptation. D'autre part, une valeur élevée de  $\beta$  permet une réduction importante de l'erreur de poursuite, au début de l'adaptation (cf le paragraphe 7.2.4 et les figures 7-6 et 7-7). Ceci entraîne la diminution de la valeur maximale de la commande, observée dans la figure 7-2, lorsque  $\beta$  augmente. En effet, l'erreur de poursuite (qui intervient dans le calcul de la commande) étant devenue d'autant plus faible que  $\beta$  était grand, la valeur maximale de la commande diminue.

En figure 7-3, plus  $\beta$  est grand, plus la commande augmente rapidement, au début de l'adaptation, et plus l'amplitude du pic qui apparaît sur la commande est importante. Augmenter  $\beta$  revient à augmenter l'action proportionnelle de la commande. Plus  $\beta$  est grand, plus la commande est rapide et énergique.

### 7.2.3 Effet de $\alpha$ sur l'erreur de poursuite

Quand  $\alpha$  augmente,  $|e_i|_{\max}$  diminue (beaucoup pour  $e_1$  et très peu pour  $e_2$ ) et  $|e_i|$  tend plus vite vers zéro (Fig. 7-4 7-5) ( $|e_i|$ ,  $i = 1, 2$  désigne la valeur absolue de la 1<sup>ème</sup> composante de l'erreur de poursuite  $e = x_m - x$ ).

Une valeur élevée de  $\alpha$  accélère la convergence de l'erreur de poursuite vers zéro.

#### 7.2.4 Effet de $\beta$ sur l'erreur de poursuite

Quand  $\beta$  augmente,  $|e_i|_{\max}$  diminue au début de l'adaptation par la suite,  $|e_i|$  converge moins vite vers zéro (Fig. 7-6 et 7-7).

Une valeur élevée de  $\beta$  permet de réduire l'erreur de poursuite au début de l'adaptation.

#### 7.3 Etude de la convergence de l'erreur du MCS

Stoten et Benchoubane [39] ont obtenu des bornes de la vitesse de convergence de l'erreur de poursuite du MCS pour un processus SISO d'ordre 1. Nous nous proposons d'étendre ces résultats à un système d'ordre 2.

##### 7.3.1 Enveloppe de l'erreur du MCS

Le processus considéré est modélisé par :

$$\dot{x}(t) = A_p x(t) + B_p u(t) \quad (7-3)$$

où  $x(t)$  est le vecteur d'état du processus

$$x(t) = \begin{bmatrix} x_1 & x_2 \end{bmatrix}^T$$

$u(t)$  est la commande.

$$A_p = \begin{bmatrix} 0 & 1 \\ -a_{p1} & -a_{p2} \end{bmatrix}$$

et

$$B_p = \begin{bmatrix} 0 \\ b_{p1} \end{bmatrix}$$

$A_p$  et  $B_p$  sont inconnues mais un estimé du rapport  $b_{p1}/a_{p1}$  est disponible.

Le processus est supposé stable.

Le MR d'ordre 2 qui définit les trajectoires d'état

Effet des paramètres  $\alpha$  et  $\beta$  sur la commande du HCS

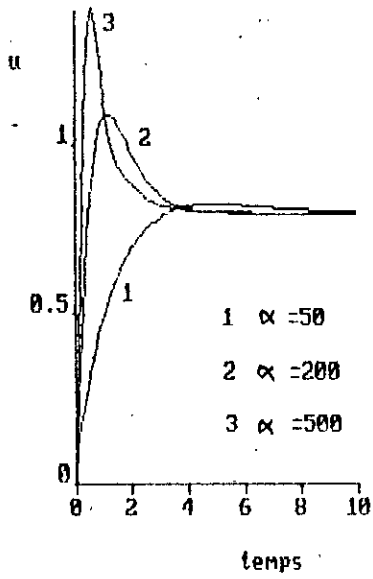


Fig 7-1 Effet de  $\alpha$  sur u

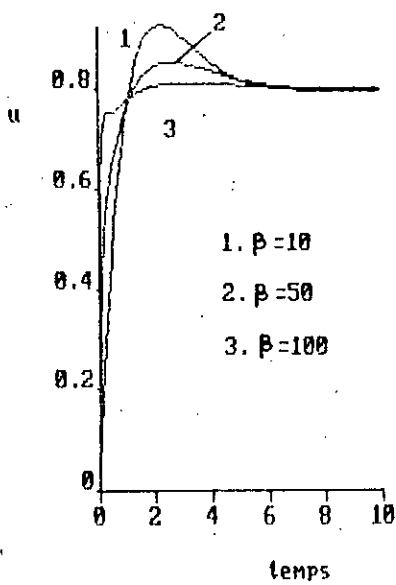


Fig 7-2 Effet de  $\beta$  sur u

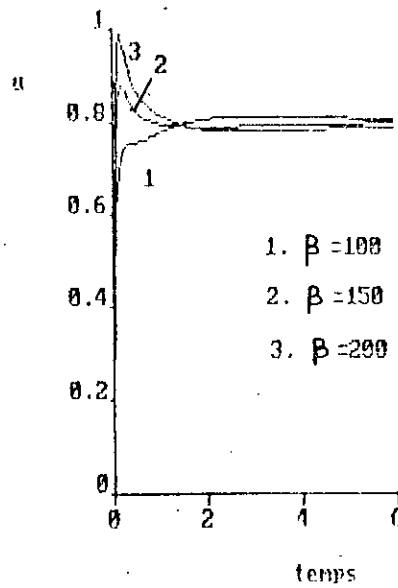


Fig 7-3 Effet de  $\beta$  sur u

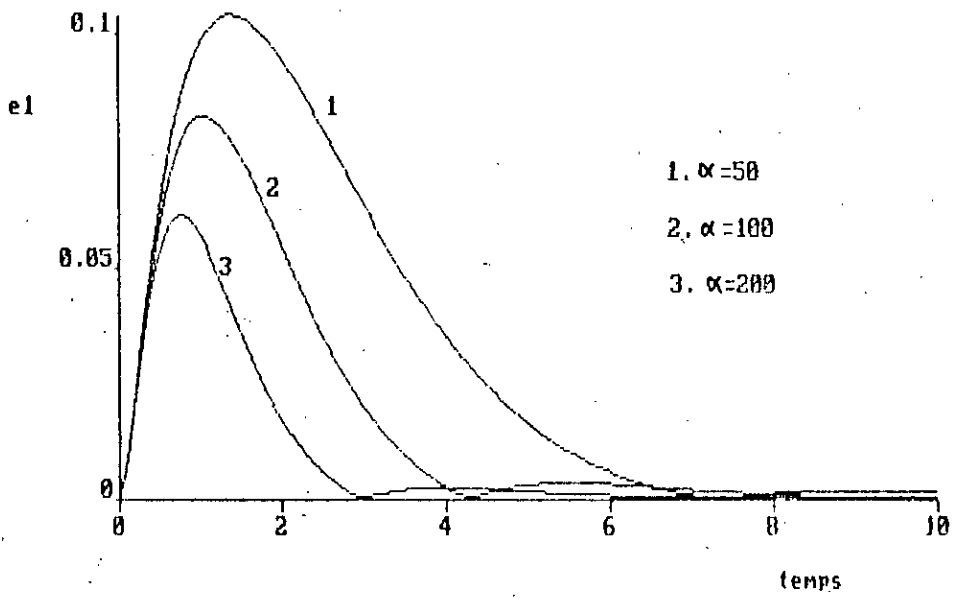


Fig 7-4 Influence de  $\alpha$  sur  $e_1$

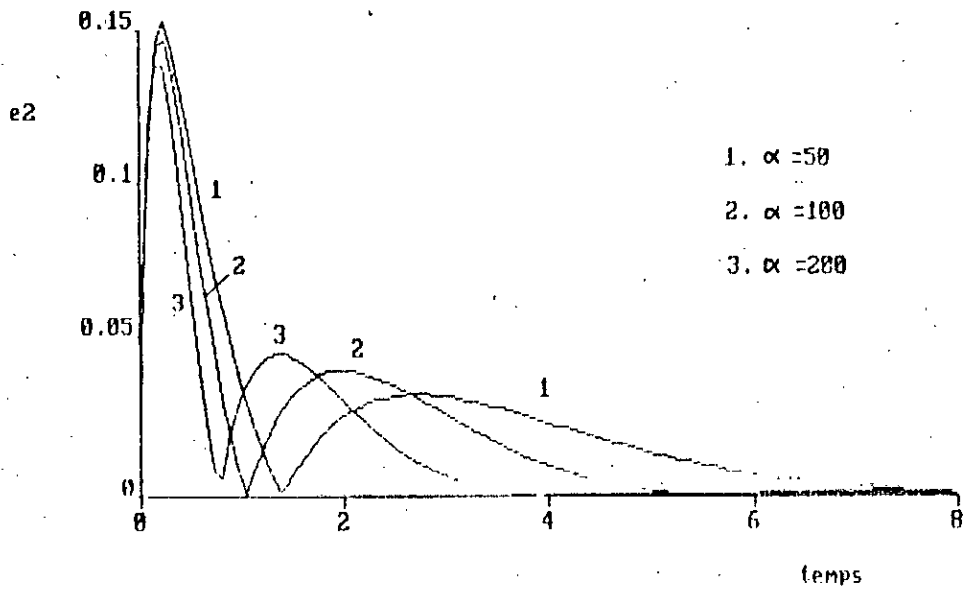


Fig 7-5 Influence de  $\alpha$  sur  $e_2$



# Influence des paramètres sur l'erreur de poursuite du MCS

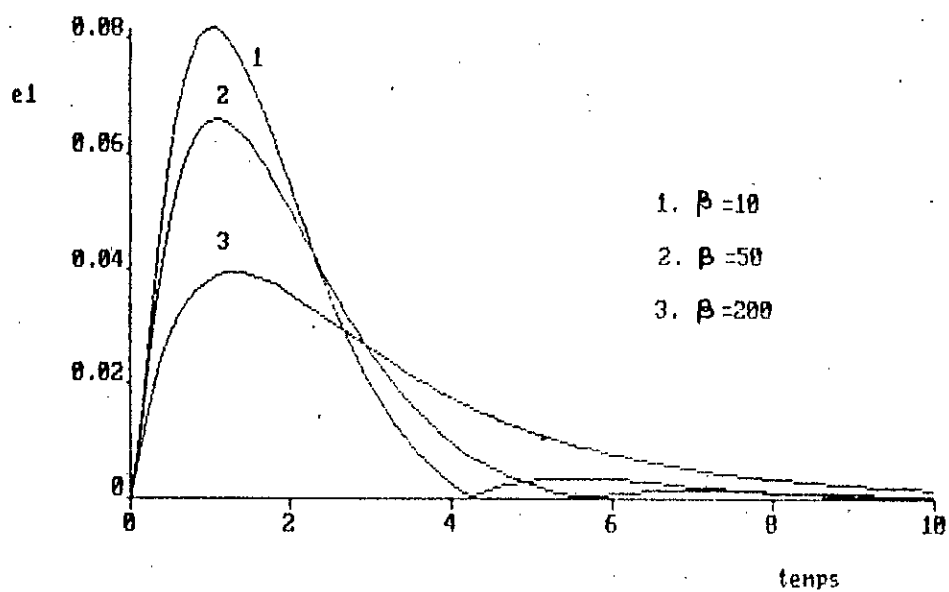


Fig 7-6 influence de  $\beta$  sur  $e_1$

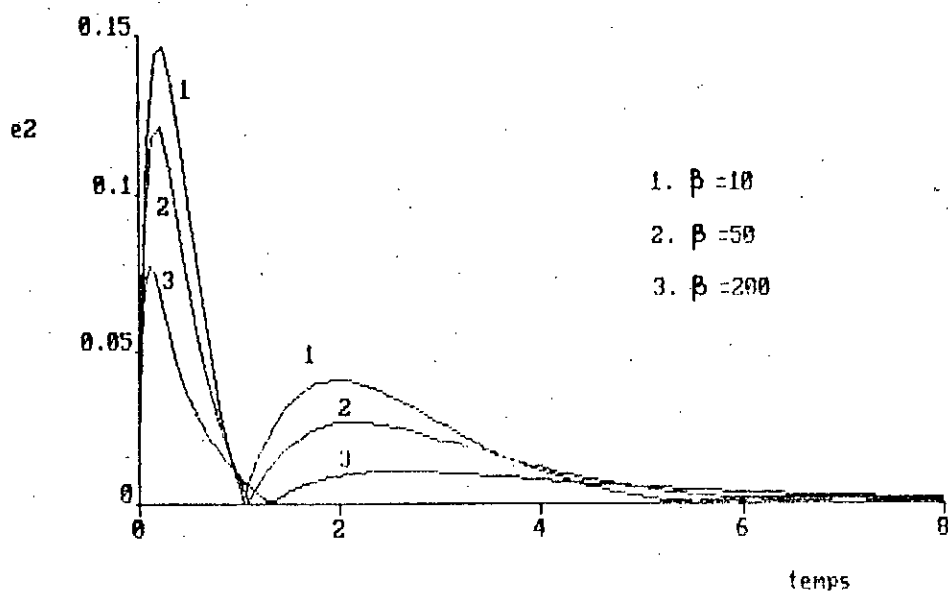


Fig 7-7 Influence de  $\beta$  sur  $e_2$

désirées, est décrit par :

$$\dot{x}_m(t) = A_m x_m(t) + B_m r(t) \quad (7-4)$$

où  $x_m$  représente la trajectoire de référence et  $r$  l'entrée de référence.

$$x_m = \begin{bmatrix} x_{m1} & x_{m2} \end{bmatrix}^T$$

$$A_m = \begin{bmatrix} 0 & 1 \\ -a_{m1} & -a_{m2} \end{bmatrix} \quad \text{et} \quad B_m = \begin{bmatrix} 0 \\ b_{m1} \end{bmatrix}$$

Nous considérons le cas où les pôles du MR,  $r_1$  et  $r_2$ , sont réels, négatifs, distincts :

$$r_1 = \frac{-a_{m2} + \sqrt{\Delta}}{2} \quad r_2 = \frac{-a_{m2} - \sqrt{\Delta}}{2}$$

$$\Delta = a_{m2}^2 - 4a_{m1} > 0$$

L'erreur de poursuite est définie par

$$e = x_m - x \quad (7-5)$$

La loi de commande du MCS est alors donnée par :

$$u(t) = \delta K_P(t)x(t) + \delta K_R(t)r(t) \quad (7-6)$$

où

$$\delta K_P(t) = \int_0^t [0 \quad \alpha] v x^T d\tau + [0 \quad \beta] v x^T \quad (7-7)$$

et

$$\delta K_R(t) = \int_0^t [0 \quad \alpha] v r d\tau + [0 \quad \beta] v r \quad (7-8)$$

$$v = De \quad (7-9)$$

[0  $\alpha$ ] : vecteur des paramètres de la matrice de gain intégral

[0  $\beta$ ] : vecteur des paramètres de la matrice de gain proportionnel

$\alpha$  et  $\beta$  sont constants et satisfont [37] :

$$\alpha b_{pi} > 0 \quad (7-10)$$

$$\beta b_{pi} > 0 \quad (7-11)$$

D satisfait l'équation de Lyapunov :

$$D\Lambda_m + \Lambda_m^T D = -Q \quad Q > 0 \quad (7-12)$$

En combinant les équations (7-3) à (7-8), on obtient le système à CR équivalent :

$$\dot{e} = \Lambda_m e - f \quad (7-13)$$

où

$$\begin{aligned} f &= \left[ B_p \delta K_p + A_p - A_m \right] x + \left[ B_p \delta K_R - B_m \right] r \\ &= \left[ B_p \left( \int_0^t [0 \ \alpha] v x^T d\tau + [0 \ \beta] v x^T \right) + A_p - A_m \right] x \\ &\quad + \left[ B_p \left( \int_0^t [0 \ \alpha] v r d\tau + [0 \ \beta] v r \right) - B_m \right] r \end{aligned} \quad (7-14)$$

soit, en développant

$$f_1 = 0$$

et

$$\begin{aligned}
 f_2 = & b_{p1} \alpha \left[ x_1 \int_0^t (d_{21} e_1 + d_{22} e_2) x_1 d\tau + x_2 \int_0^t (d_{21} e_1 + d_{22} e_2) x_2 d\tau \right. \\
 & \left. + r \int_0^t (d_{21} e_1 + d_{22} e_2) r d\tau \right] \\
 & + b_{p1} \beta \left[ (d_{21} e_1 + d_{22} e_2) x_1^2 + (d_{21} e_1 + d_{22} e_2) x_2^2 \right. \\
 & \left. + (d_{21} e_1 + d_{22} e_2) r^2 \right] \\
 & + (a_{m1} - a_{p1}) x_1 + (a_{m2} - a_{p2}) x_2 - b_{m1} r
 \end{aligned} \tag{7-15}$$

et l'inégalité intégrale de Popov :

$$\int_0^t v f d\tau \geq -\zeta^2 \quad \zeta^2 \text{ constante finie positive}$$

où

$$\zeta^2 = \frac{1}{2b_{p1} \alpha} \left[ (a_{m1} - a_{p1})^2 + (a_{m2} - a_{p2})^2 + b_{m1}^2 r^2 \right] \tag{7-16}$$

Puisque les équations (7-13), (7-9), (7-16) et (7-12) définissent un bloc hyperstable, l'erreur  $e$  est globalement asymptotiquement stable et par conséquent la fonction  $f$  tend toujours vers 0, en régime établi.

En supposant que les conditions initiales sont nulles, la solution de (7-13) est :

$$\begin{aligned}
 e(t) = \begin{pmatrix} e_1(t) \\ e_2(t) \end{pmatrix} = & \frac{e^{r_1 t}}{r_1 - r_2} \left( \int_0^t -f_2(\tau) e^{-r_1 \tau} d\tau \right) \begin{pmatrix} 1 \\ r_1 \end{pmatrix} \\
 & + \frac{e^{r_2 t}}{r_1 - r_2} \left( \int_0^t f_2(\tau) e^{-r_2 \tau} d\tau \right) \begin{pmatrix} 1 \\ r_2 \end{pmatrix}
 \end{aligned} \tag{7-17}$$

En postulant, pour tout  $t$

$$|f_2| \leq K e^{\gamma t} \tag{7-18}$$

( $K > 0$  et  $\gamma \leq 0$ ) on obtient, à l'aide de (7-17) :

$$|e_1(t)| \leq E_{s1}(t)$$

$$\begin{aligned} E_{s1}(t) &= \frac{Ke^{\gamma t}}{r_1 - r_2} \left[ \frac{1}{\gamma - r_1} + \frac{1}{\gamma - r_2} \right] \\ &= \frac{K}{r_1 - r_2} \left[ \frac{e^{r_1 t}}{\gamma - r_1} + \frac{e^{r_2 t}}{\gamma - r_2} \right] \end{aligned} \quad (7-19)$$

$$|e_2(t)| \leq E_{s2}(t)$$

$$\begin{aligned} E_{s2}(t) &= \frac{Ke^{\gamma t}}{r_1 - r_2} \left[ \frac{|r_1|}{\gamma - r_1} + \frac{|r_2|}{\gamma - r_2} \right] \\ &= \frac{K}{r_1 - r_2} \left[ \frac{|r_1| e^{r_1 t}}{\gamma - r_1} + \frac{|r_2| e^{r_2 t}}{\gamma - r_2} \right] \end{aligned} \quad (7-20)$$

Nous allons maintenant remplacer dans l'équation (7-14)  $e_1$  et  $e_2$  par leur enveloppe ( $E_{s1}$  et  $E_{s2}$ ) et dériver les expressions de  $K$  et  $\gamma$  (calculé numériquement) en fonction :

- a) de  $\alpha$  et  $\beta$
- b) de  $r$
- c) des paramètres du MR
- d) du rapport  $b_{p1}/a_{p1}$

Le signal de référence est supposé constant pendant le processus d'adaptation.

### 7.3.2 Dérivation du taux de convergence $\gamma$

En régime établi,  $t = t_s$ ,  $e = 0$ ,  $f = 0$  ( $f_2 = 0$ ) et

$$x = x_m = \begin{pmatrix} \frac{b_{m1}}{a_{m1}} r \\ 0 \end{pmatrix}$$

et en utilisant

$$x = x_m - e$$

dans l'équation (7-14), on obtient :

$$\begin{aligned} f_2(t_s) = 0 = & b_{p1} \alpha \frac{b_{m1}}{a_{m1}} \left( d_{21} \int_0^{t_s} e_1 x_{m1} d\tau + d_{22} \int_0^{t_s} e_2 x_{m1} d\tau \right. \\ & \left. - d_{21} \int_0^{t_s} e_1^2 d\tau - d_{22} \int_0^{t_s} e_1 e_2 d\tau \right) \\ & + b_{p1} \alpha r^2 \left( d_{21} \int_0^{t_s} e_1 d\tau + d_{22} \int_0^{t_s} e_2 d\tau \right) - \frac{a_{p1} b_{m1} r}{a_{m1}} \end{aligned} \quad (7-21)$$

où  $t_s$  est le temps d'établissement.

Puisque, d'après (7-4),

$$x_{m1}(t) = \frac{b_{m1} r}{r_1 - r_2} \left( \frac{e^{r_1 t}}{r_1} - \frac{e^{r_2 t}}{r_2} \right) + \frac{b_{m1} r}{r_1 r_2}$$

$$x_{m2}(t) = \frac{b_{m1} r}{r_1 - r_2} \left( e^{r_1 t} - e^{r_2 t} \right)$$

on peut remplacer  $x_{m1}(t)$  par son expression, dans (7-21). On a alors :

$$\begin{aligned}
& A_1 \int_0^t e_1 e^{r_1 \tau} d\tau + A_2 \int_0^t e_1 e^{r_2 \tau} d\tau + A_3 \int_0^t e_1 d\tau + \\
& A_4 \int_0^t e_2 e^{r_1 \tau} d\tau + A_5 \int_0^t e_2 e^{r_2 \tau} d\tau + A_6 \int_0^t e_2 d\tau + \\
& A_7 \int_0^t e_1^2 d\tau + A_8 \int_0^t e_1 e_2 d\tau + A_9 = 0 \quad (7-22)
\end{aligned}$$

où les coefficients  $A_i$ ,  $i = 1, \dots, 8$  (donnés en Annexe B) dépendent des paramètres du MR ( $a_{m1}$ ,  $b_{m1}$ ), des pôles du MR ( $r_1$ ,  $r_2$ ), des solutions de l'équation de Lyapunov ( $d_{21}$ ,  $d_{22}$ ), et de l'entrée de référence ( $r$ ). Le coefficient  $A_9$  dépend du paramètre  $\alpha$  et du rapport  $b_{p1}/a_{p1}$  (gain en basse fréquence du processus).

Dans l'équation (7-22), on note  $I_i$  l'intégrale précédée du coefficient  $A_i$  ( $i=1, \dots, 8$ ).

En appliquant la propriété générale des valeurs absolues à l'équation (7-22),

$$\sum_{i=1}^n g_i \leq \left| \sum_{i=1}^n g_i \right| \leq \sum_{i=1}^n |g_i|, \quad g_i \in \mathbb{R},$$

on obtient :

$$|A_9| \leq \sum_{i=1}^8 |A_i| |I_i| \quad (7-23)$$

On note

$$I_i = \int_0^t g(\tau) d\tau$$

et on utilise à nouveau la propriété des valeurs absolues :

$$|I_i| \leq \int_0^t |g(\tau)| d\tau \leq \int_0^\infty |g(\tau)| d\tau, \quad \forall g \in \mathbb{R}.$$

En remplaçant  $|e|$  par son enveloppe (7-19 et 7-20) on obtient une borne supérieure de chaque intégrale  $|I_i|$ . Ces bornes dépendent de  $K$ ,  $\gamma$  et des pôles du MR (celles sont données en

Annexe B). En utilisant ces bornes dans l'inégalité (7-23) et après simplification on obtient :

$$\frac{r_1 - r_2}{K} |A_0| \leq \sum_{i=1}^{18} \frac{B_i}{P_i(\gamma)} \quad (7-24)$$

Les  $B_i$  ( $i = 1$  à 18) sont exprimés en fonction des  $|A_i|$ , des pôles du MR et de  $K$  (ils sont donnés en Annexe B). Les  $P_i(\gamma)$  sont des polynômes en  $\gamma$  dont les coefficients dépendent des pôles du MR (les  $P_i(\gamma)$  sont donnés en Annexe B).

Après simplification de l'inégalité ci-dessus, on obtient :

$$P(\gamma) \geq 0 \quad (7-25)$$

où  $P(\gamma)$  est un polynôme de degré 7 en  $\gamma$  :

$$P(\gamma) = \gamma^7 + \sum_{i=0}^6 C_i \gamma^i$$

Les coefficients  $C_i$  sont donnés en Annexe B.

On cherche une borne supérieure de  $|e|$  et puisque  $\gamma$  est négatif, on s'intéresse aux solutions négatives de l'inégalité (7-25). Les racines de  $P(\gamma)$  sont obtenues numériquement (au moyen de la commande ROOTS du logiciel MATLAB). Dans les équations (7-19) et (7-20), on prendra  $\gamma = \gamma_0$  où  $\gamma_0$  est la plus grande racine négative de  $P(\gamma)$ .

### 7.3.3 Dérivation de l'amplitude $K$

Si le paramètre de la matrice de gain intégral  $\alpha$  est nul, le critère de Popov n'est plus satisfait pour le système défini par les équations (7-13), (7-16), (7-9) et (7-4). Donc, en régime établi, l'erreur ne tend plus vers zéro mais vers un vecteur constant non nul. On peut alors écrire, à partir des égalités (7-13) et (7-14) :



$$A_m e = f = (B_p [0 \ \beta] v x^T + A_p - A_m) x + (B_p [0 \ \beta] v r - B_m) r \quad (7-26)$$

ou encore :

$$e_2 = 0 \quad (7-27)$$

et

$$-a_{m1} e_1 = f_2 = b_{p1} \beta d_{21} e_1 (x_1^2 + x_2^2 + r^2) + (a_{m1} - a_{p1}) x_1 + (a_{m2} - a_{p2}) x_2 - b_{m1} r \quad (7-28)$$

Puisque  $e$  ne tend pas vers zéro,  $\gamma$  est nul et donc, à partir de l'inégalité (7-19)

$$|e_1| < \frac{-K(r_1 + r_2)}{r_1 r_2 (r_1 - r_2)} \quad (7-29)$$

En régime établi,

$$x_m = \begin{pmatrix} \frac{b_{m1}}{a_{m1}} r \\ 0 \end{pmatrix}$$

et l'on obtient, en utilisant l'égalité (7-28),

$$-a_{m1} e_1 = b_{p1} \beta d_{21} e_1 \left[ \left( \frac{b_{m1}}{a_{m1}} r - e_1 \right)^2 + r^2 \right] + (a_{m1} - a_{p1}) \left( \frac{b_{m1}}{a_{m1}} r - e_1 \right) - b_{m1} r$$

que l'on peut écrire sous la forme :

$$e_1^3 + a_2 e_1^2 + a_1 e_1 + a_0 = 0 \quad (7-30)$$

avec

$$a_2 = -\frac{2b_{m1}}{a_{m1}} r ; \quad a_1 = \frac{a_{p1}}{b_{p1} \beta d_{21}} + \left[ 1 + \left( \frac{b_{m1}}{a_{m1}} \right)^2 \right] r^2$$

$$a_0 = -\frac{a_{p1} b_{m1} r}{b_{p1} a_{m1} \beta d_{21}}$$

Pour résoudre cette équation, on applique la technique décrite dans [ 1]. On définit

$$q = \frac{1}{3} a_1 - \frac{1}{9} a_2^2$$

$$r_3 = \frac{1}{6} (a_1 a_2 - 3a_0) - \frac{1}{27} a_2^3$$

Une solution réelle existe pour l'équation (7-30) si

$$q^3 + r_3^2 > 0$$

Cette condition est en particulier vraie pour q positif. Ceci se produit lorsque :

$$a_{p1} > 0 \quad \text{et} \quad (b_{m1}/a_{m1})^2 < 3$$

En général, la solution réelle est donnée par

$$e_{1s} = s_1 + s_2 - \frac{a_2}{3} \quad (7-31)$$

où

$$s_1 = \left[ r_3 + (q^3 + r_3^2)^{1/2} \right]^{1/3}$$

$$s_2 = \left[ r_3 - (q^3 + r_3^2)^{1/2} \right]^{1/3}$$

La valeur correspondante de K est obtenue à partir de l'inégalité (7-29)

$$K > \frac{r_1 - r_2}{-(r_1 + r_2)} r_1 r_2 |e_{1s}| = K_{\text{min}} \quad (7-32)$$

### 7.3.4 Résultats et simulation

---

Nous appliquons les résultats précédents à l'exemple du paragraphe 7.2. Pour le processus, seule la connaissance du rapport  $b_{p1}/a_{p1}$  est, en fait, nécessaire.

Dans les figures 7-8 à 7-11, nous traçons l'erreur et son enveloppe pour différents ensembles de valeurs de  $\alpha$ ,  $\beta$  et  $r$ . Les courbes obtenues représentent bien l'enveloppe supérieure de l'erreur. Etudions, maintenant l'effet de  $\alpha$  et  $\beta$  sur cette enveloppe. Pour cela, nous traçons l'amplitude  $K$  de l'enveloppe de l'erreur et son taux de convergence  $\gamma$  en fonction de  $\alpha$  et  $\beta$ .

#### a) Effet de $\alpha$ sur le taux de convergence $\gamma$ (Fig. 7-12)

---

Lorsque  $\alpha$  augmente,  $\gamma$  ( $\gamma < 0$ ) diminue et par conséquent la convergence de l'enveloppe vers zéro devient plus rapide. En outre, la figure 7-12 montre qu'au-delà d'une certaine valeur de  $\alpha$  (voisine de  $\alpha = 150$ ),  $\gamma$  ne varie que très peu, pour une valeur donnée de  $\beta$ .

#### b) Effet de $\beta$ sur le taux de convergence $\gamma$ (Fig. 7-13)

---

Lorsque  $\beta$  augmente,  $\gamma$  croît et la convergence devient plus lente. La figure 7-13 montre aussi que plus  $\alpha$  est faible, plus l'augmentation de  $\gamma$ , provoquée par une augmentation donnée de  $\beta$ , est grande.

$\alpha$  et  $\beta$  ont donc des effets opposés sur la vitesse de convergence.

#### c) Effet de $\alpha$ sur l'amplitude $K$ (Fig. 7-14)

---

Le paramètre  $\alpha$  n'a aucune influence sur l'amplitude de l'enveloppe. Ceci est illustré en figure 7-14.

#### d) Effet de $\beta$ sur l'amplitude K (Fig. 7-15)

Comme on peut le voir sur la figure 7-15, l'amplitude de l'enveloppe diminue au fur et à mesure que  $\beta$  augmente.

A partir des observations précédentes est-il possible de proposer des recommandations pour le choix des valeurs de  $\alpha$  et  $\beta$  ?

#### 7.3.5 Recommandations sur le choix des paramètres

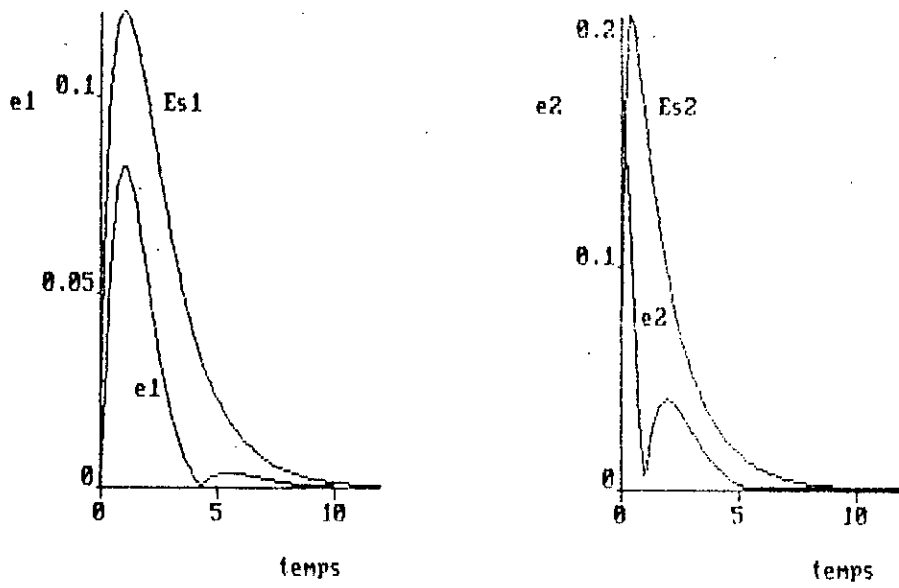
##### de convergence ( $\alpha$ et $\beta$ )

D'après la figure 7-12, il faut choisir  $\alpha$  assez grand pour permettre une convergence rapide de l'enveloppe. Cependant,  $\alpha$  ne devra pas être trop grand pour éviter de produire des pics importants dans la commande tels que ceux illustrés en figure 7-1. On pourra pour notre exemple prendre  $\alpha = 150$ .

Selon la figure 7-13,  $\beta$  devra être choisi assez faible pour que la convergence soit rapide et selon la figure 7-15,  $\beta$  devra être grand pour que l'amplitude de l'enveloppe de l'erreur soit faible.

Le choix de  $\beta$  devra donc résulter d'un compromis entre, rapidité de convergence et faiblesse de l'amplitude de l'erreur. On choisira pour exemple  $\beta = 50$ .

Le choix de  $\alpha$  et  $\beta$  n'est pas unique. On peut aussi prendre  $\alpha = 100$  et  $\beta = 50$  comme en figure 7-10.

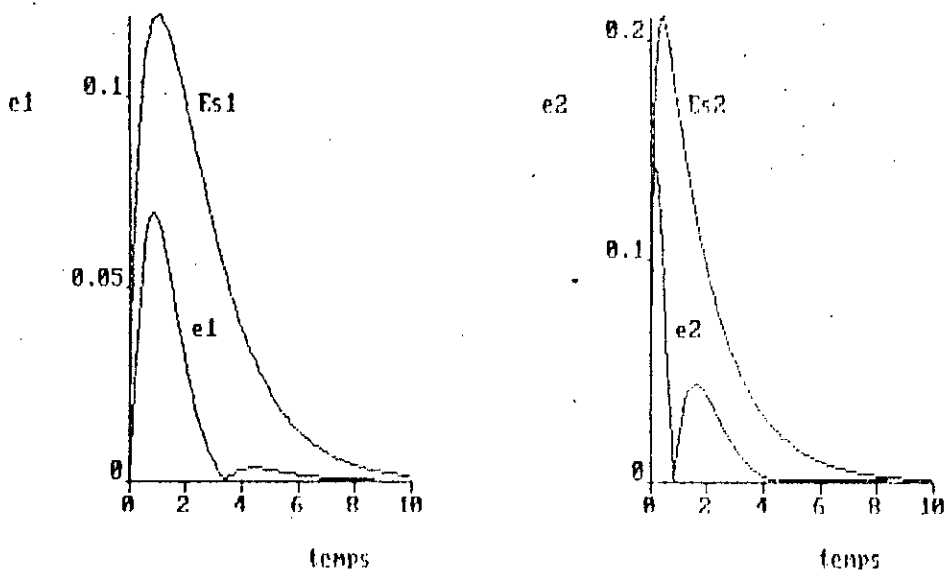


a) Enveloppe de  $e_1$

b) Enveloppe de  $e_2$

Fig 7-8 Enveloppe de l'erreur de poursuite du MCS

$$\alpha = 100 \quad \beta = 10 \quad r = 1$$

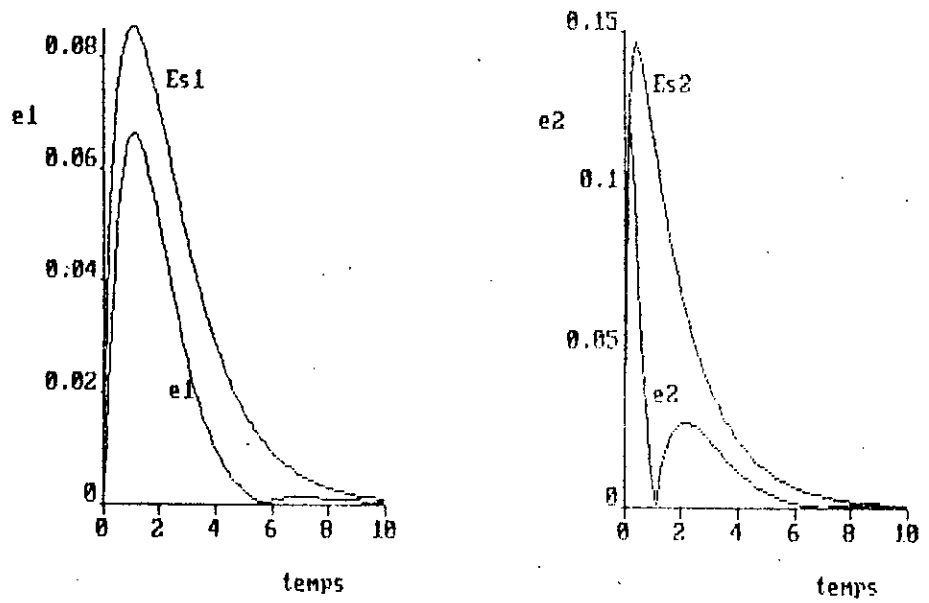


a) Enveloppe de  $e_1$

b) Enveloppe de  $e_2$

Fig 7-9 Enveloppe de l'erreur de poursuite du MCS

$$\alpha = 150 \quad \beta = 10 \quad r = 1$$

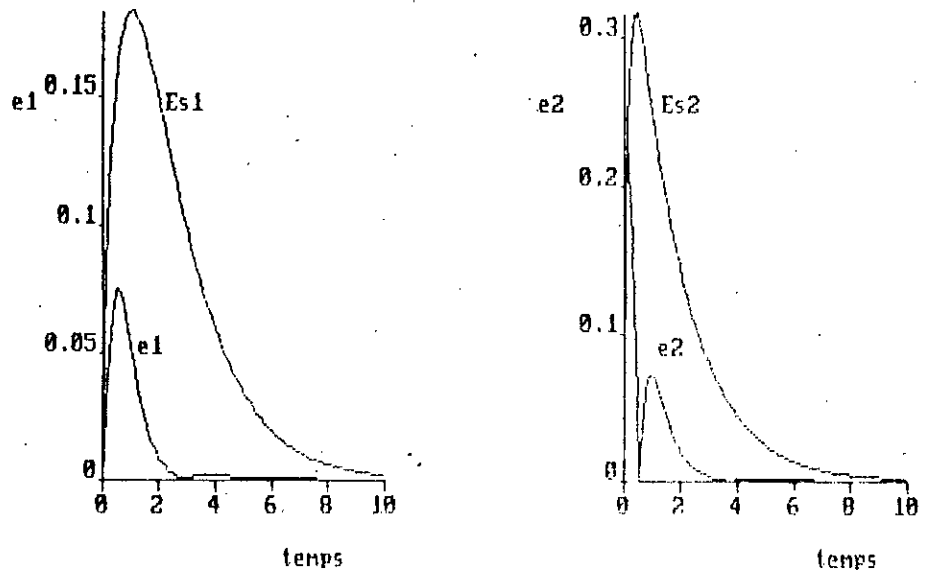


a) Enveloppe de  $e_1$

b) Enveloppe de  $e_2$

Fig 7-10 Enveloppe de l'erreur de poursuite du MCS

$$\alpha=100 \quad \beta=50 \quad r=1$$



a) Enveloppe de  $e_1$

b) Enveloppe de  $e_2$

Fig 7-11 Enveloppe de l'erreur de poursuite du MCS

$$\alpha=100 \quad \beta=10 \quad r=2$$

Effet des paramètres  $\alpha$  et  $\beta$  sur le taux de convergence du MCS

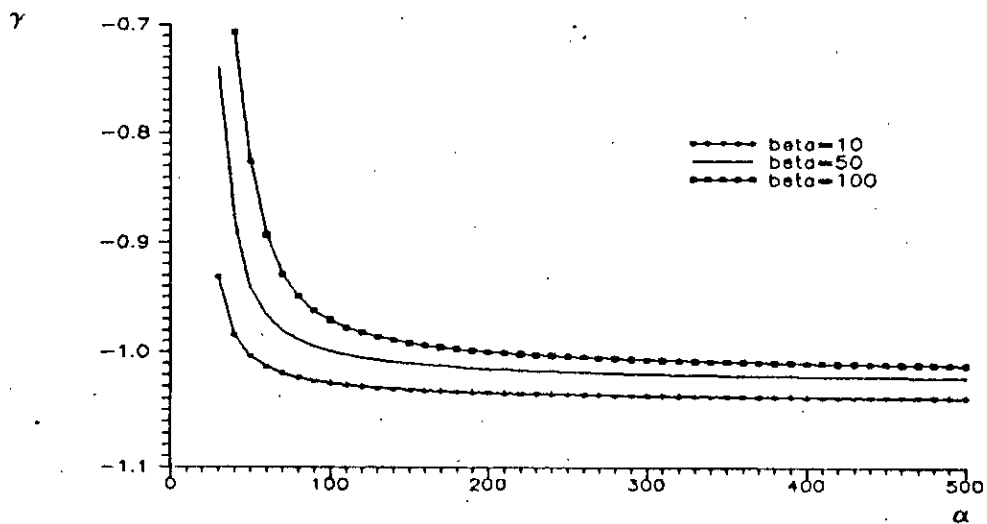


Fig 7-12 Effet de alpha sur le taux de convergence

Effet de beta sur le taux de convergence

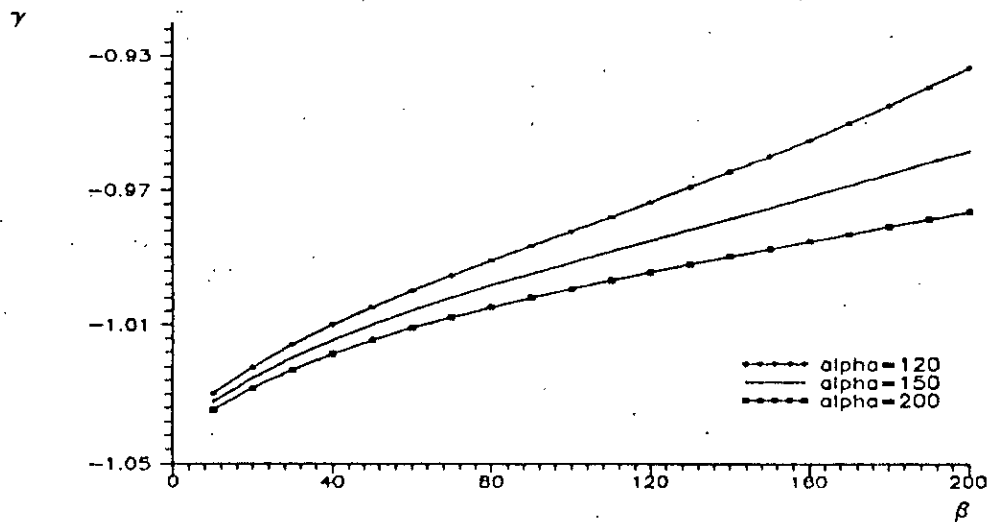


Fig 7-13 Effet de beta sur le taux de convergence

Effet des parametres  $\alpha$  et  $\beta$  sur l'amplitude de l'enveloppe

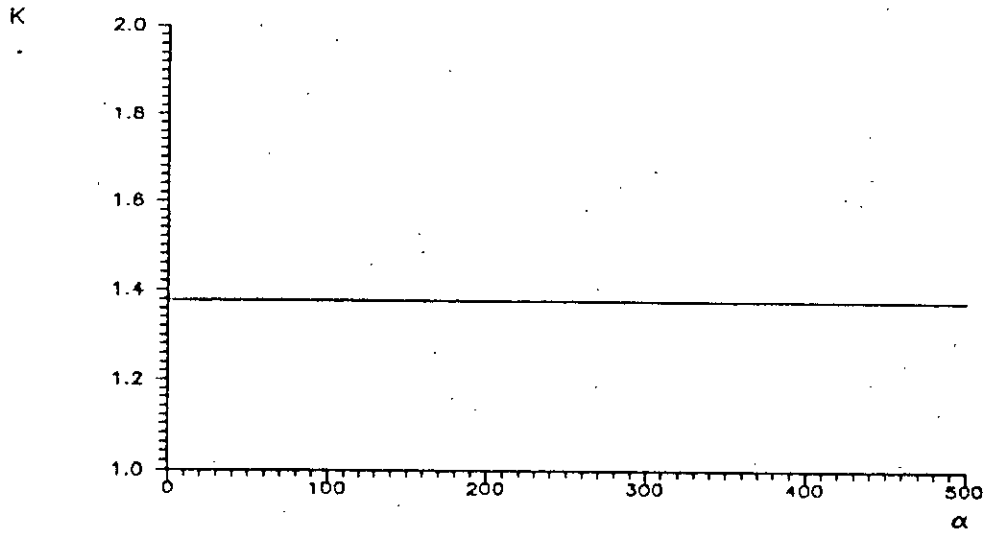


Fig 7-14 Effet de alpha sur K

Effet des parametres  $\alpha$  et  $\beta$  sur l'amplitude de l'enveloppe

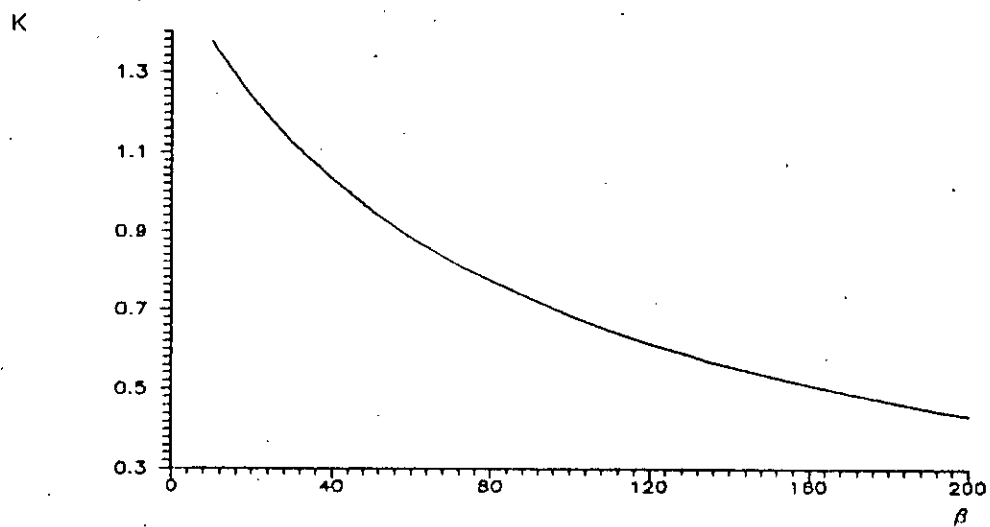


Fig 7-15 Effet de beta sur K



#### 7.4 Conclusion

Pour un processus et un MR d'ordre 2 connaissant :

- les paramètres du MR ( $a_{m1}$ ,  $b_{m1}$ ,  $a_{m2}$ )
- le gain en basse fréquence du processus ( $b_{p1}/a_{p1}$ ),
- les coefficients  $\alpha$  et  $\beta$  du contrôleur adaptatif et
- l'entrée de référence ( $r$ ),

nous avons pu construire une fonction, donnée par les relations (7-19) et (7-20), qui est l'enveloppe supérieure de l'erreur de poursuite (ou d'état  $e = x_m - x$ ) de l'algorithme MCS.

L'étude de l'influence des paramètres  $\alpha$  et  $\beta$  sur l'amplitude et sur le taux de convergence de l'enveloppe de l'erreur nous a permis de proposer des recommandations pour le choix de  $\alpha$  et  $\beta$ . Pour une application donnée, le concepteur pourra, connaissant le gain BF du processus à commander, tracer l'enveloppe de l'erreur pour différentes valeurs de  $\alpha$  et  $\beta$  et choisir celles qui satisfont les spécifications de performances désirées (amplitude de l'erreur tolérée et rapidité de convergence souhaitée).

Dans ce chapitre et dans les précédents, les processus considérés avaient des paramètres supposés inconnus mais constants. Le prochain chapitre s'intéresse à la CA des systèmes à paramètres variables dans le temps.

PARTIE. B

COMMANDE ADAPTATIVE

DE SYSTEMES

A PARAMETRES VARIABLES

CHAPITRE IGENERALITES

Une des motivations principales de la CA est de fournir des mécanismes capables de commander des systèmes à paramètres variables dans le temps.

Pour atteindre cet objectif, plusieurs schémas de CA ont été proposés par les chercheurs et souvent des contraintes sur le type de variations des paramètres ont été imposées.

De nombreux schémas utilisent la combinaison d'un algorithme d'identification paramétrique et d'une loi de CA [14], [42], [32], [46]. Ils exigent souvent la satisfaction de conditions sur certains signaux de la boucle adaptative, pour permettre une identification correcte des paramètres du procédé. Certains schémas [42] tentent de supprimer ces conditions.

Il existe aussi des schémas qui n'incorporent pas d'algorithme d'identification paramétrique [40][41]. Certains s'appuient sur des théories telles que l'hyperstabilité [4] ou la théorie des systèmes à structure variable [4], [6].

Dans cette partie de la thèse, nous étudions, les algorithmes MRAC, MCS et "MRAC modifié" [5] en présence de variations en amplitude et en fréquence des paramètres du procédé à commander. L'algorithme MRAC modifié (MRACM) combine un algorithme d'identification paramétrique et la loi de commande MRAC.

## CHAPITRE II

L'ALGORITHME MRAC MODIFIE

Dans le MRAC, les gains linéaires d'action et de réaction,  $K$  et  $K_R$ , sont calculés, de manière définitive, au moyen des paramètres préalablement estimés du procédé.

Si ces paramètres changent, pendant le fonctionnement du système adaptatif, les gains  $K$  et  $K_R$  ne reflètent plus le procédé à commander. Dans une étude précédente [5], on propose de tenir compte des variations des paramètres du procédé en les identifiant en ligne et en utilisant les estimés pour réajuster les gains  $K$  et  $K_R$ . L'algorithme obtenu dérive du MRAC. Il a été appelé algorithme MRAC modifié [5].

Le procédé à paramètres variables dans le temps est représenté par l'équation d'état suivante :

$$\dot{x} = A_p(t)x(t) + B_p(t)u(t) \quad (2-1)$$

où les paramètres  $A_p(t)$  et  $B_p(t)$  sont donnés par :

$$A_p(t) = A_p(0) + \Delta A_p(t) \quad (2-2)$$

$$B_p(t) = B_p(0)B(t) = B_p(0)(B + \Delta B(t)) \quad (2-3)$$

$B(t)$  est une matrice carrée inversible.

$A_p(0)$ ,  $B_p(0)$  et  $B$  sont des matrices constantes supposées connues, alors que les variations  $\Delta A_p(t)$  et  $\Delta B(t)$  sont inconnues. L'expression (2-3) a été utilisée dans [4] et [17]. L'équation d'état du MR est :

$$\dot{x}_m(t) = A_m x_m(t) + B_m r(t) \quad (2-4)$$

avec

$$B_m = B_p(0)$$

$x$  et  $x_m$  sont des vecteurs d'ordre  $n$ ,  
 $u$  et  $r$  sont d'ordre  $m$ .

### 2.1 Poursuite linéaire compensée

La partie linéaire de la commande de l'algorithme MRAC est remplacée par :

$$u_{L\alpha}(t) = -K'(t)x(t) + K'_R(t)r(t) \quad (2-5)$$

où les gains d'action et de réaction ne sont plus constants :

$$K'(t) = K^0 + \Delta K(t) \quad (2-6)$$

$$K'_R(t) = K_R^0 + \Delta K_R(t) \quad (2-7)$$

$\Delta K(t)$  et  $\Delta K_R(t)$  sont les réajustements dus à la variation des paramètres du procédé.

Les équations (2-1) à (2-7) permettent d'écrire :

$$\begin{aligned} \dot{e} = \dot{x}_m - \dot{x} = & A_m e \\ & + \left[ A_m - A_p(0) - \Delta A_p(t) + B_p(0)(B + \Delta B(t))(K^0 + \Delta K(t)) \right] x(t) \\ & + \left[ B_m - B_p(0)(B + \Delta B(t))(K_R^0 + \Delta K_R(t)) \right] r(t) \end{aligned}$$

Pour que

$$\lim_{t \rightarrow \infty} e(t) = 0 \quad \text{et} \quad \lim_{t \rightarrow \infty} \dot{e}(t) = 0$$

il faut que

$$A_m - A_p(0) - \Delta A_p(t) = -B_p(0)(B + \Delta B(t))(K^0 + \Delta K(t)) \quad (2-8)$$

et

$$B_m = B_p(0)(B + \Delta B(t))(K_R^0 + \Delta K_R(t)) \quad (2-9)$$

Les équations (2-8) et (2-9) sont vérifiées par les gains [ 5] :

$$K^0 = B_p^{-1} B_p^+(0) (A_p(0) - A_m) \quad (2-10)$$

$$K_R^0 = B_p^{-1} B_p^+(0) B_m \quad (2-11)$$

$$\Delta K(t) = (B + \Delta B(t))^{-1} B_p^+(0) (A_p(t) - B_p(0) \Delta B(t) K^0) \quad (2-12)$$

$$\Delta K_R(t) = -(B + \Delta B(t))^{-1} \Delta B(t) K_R^0 \quad (2-13)$$

Les conditions d'Erzberger sont :

$$(I - B_p(0) B_p^+(0)) (A_p - A_m) = 0$$

$$(I - B_p(0) B_p^+(0)) B_m = 0$$

$B_p^+(0)$  est la pseudo-inverse de la matrice  $B_p(0)$  :

$$B_p^+(0) = (B_p^T(0) B_p(0))^{-1} B_p^T(0)$$

Le calcul des gains  $\Delta K(t)$  et  $\Delta K_R(t)$  nécessite la connaissance des variations  $\Delta A_p(t)$  et  $\Delta B(t)$  des paramètres du procédé. Elles seront remplacées par leurs estimés.

## 2.2 Adaptation à MR

L'expression de la loi de commande adaptative est obtenue au moyen de la méthode de synthèse fondée sur la théorie de l'hyperstabilité. Pour cela, l'hyperstabilité du système à contre-réaction représentant l'évolution de l'erreur ( $e(t) = x_m(t) - x(t)$ ) est exigée [ 5].



propriété d'entrée bornée - sortie bornée des systèmes hyperstables conduisent à :

$$1) \quad D = B_p^T(O)P \quad (2-18)$$

P étant la solution de l'équation de Lyapunov :

$$PA_m + A_m^T P = -Q, \quad Q > 0$$

Ceci afin que la fonction de transfert du bloc linéaire invariant du système à contre-réaction équivalent soit réelle positive.

$$2) \quad \delta K(t) = (B + \Delta B(t))^{-1} \left[ \int_0^t \alpha v x^T dt + \beta v x^T \right] \quad (2-19)$$

$$\delta K_R(t) = (B + \Delta B(t))^{-1} \left[ \int_0^t \alpha v r^T dt + \beta v r^T \right] \quad (2-20)$$

$\alpha$  : matrice carrée d'ordre m définie positive.

$\beta$  : matrice carrée d'ordre m semi-définie positive.

Ceci après résolution de l'inégalité intégrale de Popov et sélection d'une loi de CA de type PI.

Afin d'assurer l'hyperstabilité du système adaptatif, les variations ( $\Delta A_p(t)$  et  $\Delta B(t)$ ) des paramètres du procédé doivent être bornés :

$$\|\Delta A_p(t)\| \leq a \quad (2-21)$$

$$\|\Delta B(t)\| \leq b \quad (2-22)$$

où a et b sont des constantes finies positives.

De plus, la contrainte suivante doit être satisfaite :



$$B_m = B_p(O) \quad (2-23)$$

En résumé, l'algorithme MRACM est caractérisé par la loi de commande :

$$u(t) = -(K^O + \Delta K(t) - \delta K(t))x(t) + (K_R^O + \Delta K_R(t) + \delta K_R(t))r(t) \quad (2-24)$$

où les gains sont donnés par les équations (2-10) à (2-13), (2-19) et (2-20). L'algorithme MRACM exige, en outre, la satisfaction des contraintes exprimées par les relations (2-21) à (2-23).

D'autre part, le calcul des gains de la commande requiert la connaissance des variations des paramètres du procédé. Ces variations seront remplacées par leurs estimés obtenus grâce à un algorithme d'identification paramétrique récursif. Ce dernier sera présenté dans le chapitre suivant.

## CHAPITRE III

## IDENTIFICATION PARAMETRIQUE

## 3.1 Généralités

L'objectif d'un algorithme d'identification paramétrique est de déterminer ou d'estimer les paramètres du modèle d'un procédé en utilisant des mesures effectuées sur ce procédé.

Plusieurs algorithmes d'identification paramétrique récursifs possédant une formulation adaptée à leur mise en oeuvre sur micro-ordinateur ont été développés [19], [22].

Le principe de l'estimation des paramètres d'un modèle échantillonné est représenté dans la figure suivante.

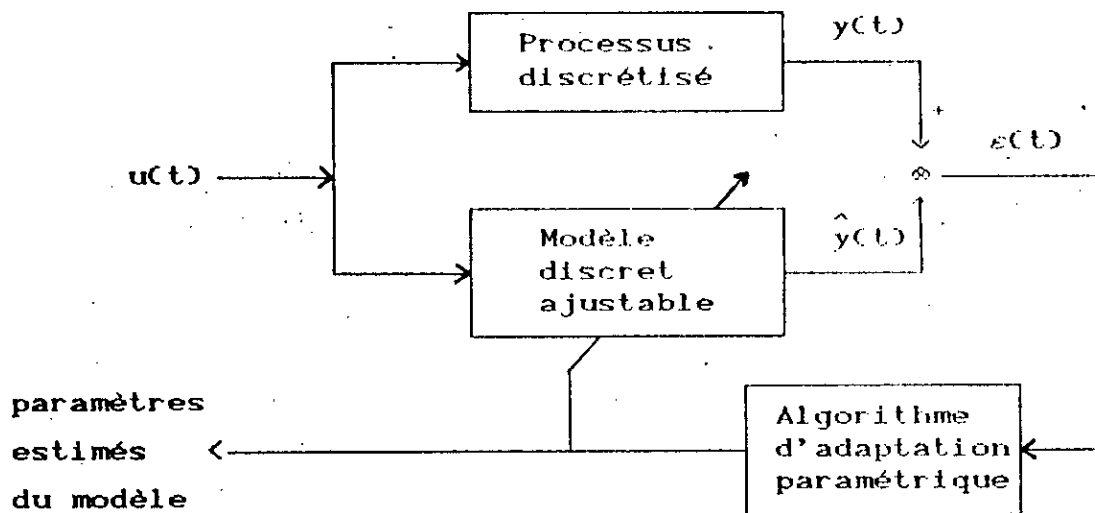


Fig. 3-1 Principe de l'estimation adaptative des paramètres

Un modèle échantillonné à paramètres ajustables est implanté sur le calculateur. L'erreur entre la sortie du

procédé,  $y(t)$ , et la sortie prédite par le modèle ajustable,  $\hat{y}(t)$ , est appelée erreur de prédiction. Elle est utilisée par un algorithme d'adaptation paramétrique qui modifie, à chaque instant d'échantillonnage, les paramètres estimés afin de minimiser l'erreur de prédiction.

L'algorithme d'adaptation paramétrique possède une structure récursive : les nouvelles valeurs des paramètres estimés sont égales aux anciennes valeurs plus un terme de correction qui dépend des mesures les plus récentes.

$$\begin{pmatrix} \text{NOUVEL} \\ \text{ESTIME} \\ \text{DES} \\ \text{PARAMETRES} \end{pmatrix} = \begin{pmatrix} \text{ESTIME} \\ \text{PRECEDENT} \\ \text{DES} \\ \text{PARAMETRES} \end{pmatrix} + \begin{pmatrix} \text{GAIN} \\ \text{D'ADAPTATION} \end{pmatrix} \times \begin{pmatrix} \text{FONCTION} \\ \text{DES} \\ \text{MESURES} \end{pmatrix} \times \begin{pmatrix} \text{FONCTION} \\ \text{DE L'ERREUR} \\ \text{DE PREDICTION} \end{pmatrix}$$

Parmi les algorithmes d'adaptation paramétrique nous nous intéressons à celui des moindres carrés récursifs.

Soit le procédé SISO échantillonné représenté par l'équation aux différences

$$y(t+1) = -\sum_{i=1}^n a_i y(t+1-i) + \sum_{i=0}^m b_i u(t-i) = \theta^T \phi(t) \quad (3-1)$$

où

$$\theta^T = [a_1 \dots a_n, b_0 \dots b_m]$$

est le vecteur des paramètres du procédé, et

$$\phi^T(t) = [-y(t), -y(t-1), \dots, -y(t+1-n), u(t), \dots, u(t-m)]$$

est le vecteur des mesures.

Le modèle de prédiction ajustable est décrit par :

$$\begin{aligned}\hat{y}^0(t+1) &= -\sum_{i=1}^n \hat{a}_i(t) y(t+1-i) + \sum_{i=0}^m \hat{b}_i(t) u(t-i) \\ &= \hat{y}(t+1 / \hat{\theta}(t)) = \hat{\theta}(t)^T \phi(t)\end{aligned}$$

où

$$\hat{\theta}(t) = \left[ \hat{a}_1(t) \dots \hat{a}_n(t), \hat{b}_0(t) \dots \hat{b}_m(t) \right]$$

$\hat{\theta}(t)$  est le vecteur des paramètres estimés à l'instant  $t$ .

$\hat{y}^0(t+1)$  est la prédiction "a priori" de la sortie du procédé.

La prédiction "a posteriori" est donnée par :

$$\hat{y}(t+1) = \hat{y}(t+1 / \hat{\theta}(t+1)) = \hat{\theta}(t+1)^T \phi(t)$$

On définit l'erreur de prédiction "a priori" :

$$\varepsilon^0(t+1) = y(t+1) - \hat{y}^0(t+1)$$

et l'erreur de prédiction "a posteriori" :

$$\varepsilon(t+1) = y(t+1) - \hat{y}(t+1)$$

L'objectif de l'algorithme d'adaptation paramétrique des moindres carrés est d'estimer un vecteur de paramètres  $\hat{\theta}(t)$ , à l'instant  $t$ , pour qu'il minimise la somme des carrés des écarts entre le procédé et le modèle de prédiction sur un horizon de  $t$  mesures. Le critère à minimiser est donc :

$$J(t) = \sum_{i=1}^t \left[ y(i) - \hat{\theta}(t)^T \phi(i-1) \right]^2$$

Le paramètre  $\hat{\theta}(t)$  s'obtient en annulant

$$\frac{\partial J(t)}{\partial \hat{\theta}(t)}$$

La formulation mathématique de l'algorithme des moindres carrés récurrents (MCR) est donnée ci-dessous [19] :

$$\hat{\theta}(t+1) = \hat{\theta}(t) + F(t) \phi(t) \varepsilon(t+1) \quad (3-2)$$

Dans (3-2),  $\varepsilon(t+1)$  est l'erreur de prédiction "a posteriori". Elle peut être calculée au moyen de l'expression (3-3) qui donne la relation entre l'erreur de prédiction "a posteriori" et l'erreur de prédiction "a priori" [19].

$$\varepsilon(t+1) = \frac{y(t+1) - \hat{\theta}(t)^T \phi(t)}{1 + \phi(t)^T F(t) \phi(t)} = \frac{\varepsilon^0(t+1)}{1 + \phi(t)^T F(t) \phi(t)} \quad (3-3)$$

$\varepsilon^0(t+1)$  est l'erreur de prédiction "a priori".

$$F(t+1)^{-1} = F(t)^{-1} + \phi(t)\phi(t)^T \quad (3-4)$$

$$F(t+1) = F(t) - \frac{F(t)\phi(t)\phi(t)^T F(t)}{1 + \phi(t)^T F(t)\phi(t)} \quad (3-5)$$

F est la matrice de covariance.

Dans la pratique, l'algorithme MCR possède une convergence initiale rapide mais après quelques itérations, le gain d'adaptation devient très faible. En fait, cet algorithme possède un gain décroissant et donne de moins en moins de poids aux nouvelles mesures. Il ne conviendra donc pas pour l'estimation des paramètres variables dans le temps.

### 3.2 Algorithme MCR à trace constante (SISO)

Plusieurs modifications ont été proposées pour rendre l'algorithme MCR applicable aux systèmes variants dans le temps [11], [19]. Une méthode présentée par Landau [19] consiste à généraliser la formule de l'inverse du gain d'adaptation en introduisant deux séquences de pondération,  $\lambda_1(t)$  et  $\lambda_2(t)$ , choisies de façon à empêcher le gain de tendre vers zéro.

$$F(t+1)^{-1} = \lambda_1(t)F(t)^{-1} + \lambda_2(t)\phi(t)\phi(t)^T \quad (3-6)$$

$$0 < \lambda_1(t) \leq 1 \quad \text{et} \quad 0 < \lambda_2(t) < 2, \quad F(0) > 0$$

$\lambda_1(t)$  et  $\lambda_2(t)$  ont des effets opposés :

$\lambda_1(t) < 1$  tend à augmenter le gain d'adaptation,

$\lambda_2(t) > 0$  tend à diminuer le gain d'adaptation.

A chaque choix des séquences  $\lambda_1(t)$  et  $\lambda_2(t)$  correspond un profil de variation du gain d'adaptation. Un profil convenable pour l'identification des systèmes à paramètres variables dans le temps est obtenu en choisissant  $\lambda_1(t)$  et  $\lambda_2(t)$  automatiquement de façon à assurer une trace constante de la matrice de gain :

$$\text{tr } F(t+1) = \text{tr } F(t) = \text{tr } F(0) = nGI$$

$n$  est le nombre de paramètres et  $GI$  est le gain initial.

La matrice  $F(0)$  a la forme :

$$F(0) = \begin{pmatrix} GI & 0 \\ 0 & GI \end{pmatrix}$$

En utilisant le lemme d'inversion de matrice <sup>1)</sup>, l'équation (3-6) devient:

$$F(t+1) = \frac{1}{\lambda_1(t)} \left[ F(t) - \frac{F(t)\phi(t)\phi^T(t)F(t)}{\frac{\lambda_1(t)}{\lambda_2(t)} + \phi(t)^T F(t)\phi(t)} \right] \quad (3-7-a)$$

Les valeurs de  $\lambda_1(t)$  et  $\lambda_2(t)$  sont déterminées à partir de l'équation :

$$\text{tr } F(t+1) = \frac{1}{\lambda_1(t)} \text{tr} \left[ F(t) - \frac{F(t)\phi(t)\phi(t)^T F(t)}{\alpha(t) + \phi(t)^T F(t)\phi(t)} \right] = \text{tr } F(t) \quad (3-7-b)$$

1) Lemme d'inversion de matrice :

Soient  $F$  une matrice régulière de dimension  $n \times n$  et  $\phi$  un vecteur de dimension  $n$  alors :

$$(F^{-1} + \phi\phi^T)^{-1} = F - \frac{F\phi\phi^T F}{1 + \phi^T F\phi}$$

en fixant le rapport :

$$\alpha(t) = \frac{\lambda_1(t)}{\lambda_2(t)}$$

Formulation de l'algorithme des MCR à trace constante:

$$\hat{\theta}(t+1) = \hat{\theta}(t) + F(t)\phi(t)\varepsilon(t+1) \quad (3-8)$$

$$F(t+1) = \frac{1}{\lambda_1(t)} \left[ F(t) - \frac{F(t)\phi(t)\phi(t)^T F(t)}{\alpha(t) + \phi(t)^T F(t)\phi(t)} \right] \quad (3-9)$$

$$\varepsilon(t+1) = \frac{y(t+1) - \hat{\theta}(t)^T \phi(t)}{\alpha(t) + \phi(t)^T F(t)\phi(t)} \quad (3-10)$$

$\alpha(t) = \lambda_1(t)/\lambda_2(t)$  fixé (valeurs typiques : 0,9 ; 0,95).

$\lambda_1(t)$  est calculé à partir de l'équation (3-7-b).

### 3.3 Algorithme MCR à trace constante (MIMO)

#### 3.3.1 Dérivation de l'algorithme

L'équation aux différences (3-1) peut être généralisée pour des systèmes MIMO [22]. On obtient

$$\begin{aligned} y(t+1) + A_1 y(t) + \dots + A_{n_a} y(t+1-n_a) \\ = B_1 u(t) + \dots + B_{n_b} u(t+1-n_b) \end{aligned} \quad (3-11)$$

$y(t)$ , vecteur de dimension  $n$ , est la sortie du procédé.

$u(t)$ , vecteur de dimension  $m$ , est l'entrée du procédé.

$A_i$  : matrices de dimension  $n \times n$ , pour  $i = 1, \dots, n_a$

$B_i$  : matrices de dimension  $n \times m$ , pour  $i = 1, \dots, n_b$

L'équation (3-11) peut être écrite sous la forme

$$y(t+1) = \theta^T \phi(t) \quad (3-12)$$

où

$$\theta^T = \left[ -A_1 \dots -A_{n_a}, B_1 \dots B_{n_b} \right]$$

$$\phi(t)^T = \left[ -y(t), \dots -y(t+1-n_a), u(t), \dots u(t+1-n_b) \right]$$

$\theta$  est une matrice de dimension  $l \times n$

$\phi$  est un vecteur de dimension  $l \times 1$

$$l = n_a \cdot n + n_b \cdot m$$

L'objectif de l'algorithme d'identification paramétrique est de trouver un estimé  $\hat{\theta}(t)$  qui minimise le critère :

$$J(t) = \sum_{i=1}^t f(t, i) \left[ y(i) - \hat{\theta}(t)^T \phi(i-1) \right]^T \left[ y(i) - \hat{\theta}(t)^T \phi(i-1) \right]$$

(3-13)

où  $f(t, i)$  est une fonction scalaire à valeurs positives.

Pour cela, on cherche  $\hat{\theta}(t)$  qui annule 
$$\frac{\partial J(t)}{\partial \hat{\theta}(t)}$$

$$\frac{\partial J(t)}{\partial \hat{\theta}(t)} = 0 \Rightarrow$$

$$\sum_{i=1}^t f(t, i) \phi(i-1) \phi(i-1)^T \hat{\theta}(t) = \sum_{i=1}^t f(t, i) \phi(i-1) y^T(i) \quad (3-14)$$

On en déduit



$$\hat{\theta}(t) = F(t) \left( \sum_{i=1}^t f(t, i) \phi(i-1) y^T(i) \right) \quad (3-15)$$

et

$$F(t) = \left[ \sum_{i=1}^t f(t, i) \phi(i-1) \phi(i-1)^T \right]^{-1} \quad (3-16)$$

Pour obtenir une expression récursive de  $\hat{\theta}(t)$ , calculons d'abord  $F(t+1)$  :

$$F(t+1)^{-1} = \sum_{i=1}^{t+1} f(t+1, i) \phi(i-1) \phi(i-1)^T \quad (3-17)$$

posons

$$f(t+1, i) = \lambda_1(t) \cdot f(t, i) \quad \text{pour } i < t+1$$

et

$$f(t+1, t+1) = \lambda_2(t)$$

avec

$$0 < \lambda_1(t) < 1 \quad \text{et} \quad 0 < \lambda_2(t) < 2.$$

l'équation (3-17) devient alors:

$$F(t+1)^{-1} = \lambda_1(t) F(t)^{-1} + \lambda_2(t) \phi(t) \phi^T(t) \quad (3-18)$$

et l'application du lemme d'inversion de matrice donne:

$$F(t+1) = \frac{1}{\lambda_1(t)} \left[ F(t) - \frac{F(t) \phi(t) \phi^T(t) F(t)}{\alpha(t) + \phi^T(t) F(t) \phi(t)} \right] \quad (3-19)$$

$$\alpha(t) = \frac{\lambda_1(t)}{\lambda_2(t)}$$

Nous pouvons maintenant calculer  $\hat{\theta}(t+1)$ , en utilisant les expressions de  $F(t+1)$  et de  $f(t+1, i)$

$$\hat{\theta}(t+1) = F(t+1) \left[ \sum_{i=1}^{t+1} f(t+1, i) \phi(i-1) y^T(i) \right]$$

$$\hat{\theta}(t+1) = \frac{1}{\lambda_1(t)} \left[ F(t) - \frac{F(t)\phi(t)\phi(t)^T F(t)}{\alpha(t) + \phi(t)^T F(t)\phi(t)} \right] \times$$

$$\left[ \lambda_1(t) \sum_{i=1}^l f(t, i) \phi(i-1) y^T(i) + \lambda_2(t) \phi(t) y^T(t+1) \right] \quad (3-20)$$

et après simplification nous obtenons :

$$\hat{\theta}(t+1) = \hat{\theta}(t) + F(t)\phi(t) \frac{y^T(t+1) - \phi(t)^T \hat{\theta}(t)}{\alpha(t) + \phi(t)^T F(t)\phi(t)} \quad (3-21)$$

Finalement, l'algorithme des moindres carrés récursifs à trace constante est donné, dans le cas des systèmes MIMO, par :

$$\hat{\theta}(t+1) = \hat{\theta}(t) + F(t)\phi(t)\varepsilon(t+1) \quad (3-22)$$

$$F(t+1) = \frac{1}{\lambda_1(t)} \left[ F(t) - \frac{F(t)\phi(t)\phi(t)^T F(t)}{\alpha(t) + \phi(t)^T F(t)\phi(t)} \right] \quad (3-23)$$

$$\varepsilon(t+1) = \frac{y^T(t+1) - \phi^T(t)\hat{\theta}(t)}{\alpha(t) + \phi(t)^T F(t)\phi(t)} \quad (3-24)$$

$$\text{tr } F(t+1) = \frac{1}{\lambda_1(t)} \text{tr} \left[ F(t) - \frac{F(t)\phi(t)\phi(t)^T F(t)}{\alpha(t) + \phi(t)^T F(t)\phi(t)} \right] \quad (3-25)$$

$$\alpha(t) = \frac{\lambda_1(t)}{\lambda_2(t)} \quad (3-26)$$

### 3.3.2 Stabilité numérique

La propagation des erreurs dues aux arrondis fait que la matrice de covariance,  $F$ , peut perdre son caractère symétrique définie positive. Plusieurs algorithmes ont été proposés pour permettre une meilleure stabilité numérique [24]. Dans nos simulations nous avons utilisé l'algorithme de Carlson [24].

Soit l'équation ci-dessous donnant la matrice de covariance :

$$F(t+1) = F(t) - \frac{F(t) \phi(t) \phi^T(t) F(t)}{\phi(t)^T F(t) \phi(t) + \alpha(t)}$$

on peut écrire :

$$F(t) = S(t) S^T(t) \quad (3-27)$$

où  $S(t)$  est une racine carrée<sup>1)</sup> de  $F(t)$ . On a alors :

$$S(t+1) S^T(t+1) = S(t) \left[ I - a(t) a^T(t) / d(t) \right] S^T(t) \quad (3-28)$$

où

$$\begin{aligned} a(t) &= S^T(t) \phi(t) \\ d(t) &= a^T(t) a(t) + \alpha(t) \end{aligned}$$

$a(t)$  est un vecteur d'ordre  $l$  et  $d(t)$  est un scalaire.

Supposant que  $S(t)$  est triangulaire supérieure, l'algorithme cherche

$$\left[ I - a(t) a^T(t) / d(t) \right]^{1/2} \quad (3-29)$$

1) Soit  $P(n, n)$  une matrice carrée symétrique définie-positive. Toute matrice carrée,  $S$ , d'ordre  $n$ , telle que  $P = SS^T$ , est une racine carrée de  $P$ .

telle que  $S(t+1)$  soit aussi triangulaire supérieure. L'algorithme ne calcule pas cette racine explicitement et ne requiert pas le calcul du produit matriciel ci-dessous pour générer  $S(t+1)$  :

$$S(t+1) = S(t) [I - a(t)a^T(t)/d(t)]^{1/2} \quad (3-30)$$

L'algorithme est initialisé en posant :

$$\left. \begin{aligned} d_0 &= \alpha(t) \\ e_0 &= 0 \\ a &= S^T(t) \phi(t) \end{aligned} \right\} \quad (3-31)$$

$d_0$  est un scalaire,  $e_0$  et  $a$  sont des vecteurs d'ordre  $l$ . puis on procède aux itérations suivantes pour  $k = 1, 2, \dots, l$

$$d_k = d_{k-1} + a_k^2 \quad (3-32)$$

$$b_k = (d_{k-1} / d_k)^{1/2} \quad (3-33)$$

$$c_k = a_k / (d_{k-1} d_k)^{1/2} \quad (3-34)$$

$$e_k = e_{k-1} + S_k(t) a_k \quad (3-35)$$

$$S_k(t+1) = S_k(t) b_k - e_{k-1} c_k \quad (3-36)$$

$a_k$  est la  $k^{\text{ème}}$  composante du vecteur  $a$ .

$S_k(t)$  désigne la  $k^{\text{ème}}$  colonne de  $S(t)$ .

$S_k(t)$  et  $e_k$  ont des zéros au-dessous de leur  $k^{\text{ème}}$  élément.

Après  $l$  itérations, on obtient :

$$S(t+1) = [S_1(t+1) S_2(t+1) \dots S_l(t+1)] \quad (3-37)$$

Pour l'algorithme à trace constante, nous tenons compte

du coefficient  $\lambda_1(t)$  qui intervient lors du calcul de la matrice de covariance, de la façon suivante:

l'équation (3-36) est remplacée par :

$$S_k(t+1) = \frac{1}{\sqrt{\lambda_1(t)}} [S_k(t)b_k - e_{k-1}c_k] \quad (3-38)$$

$$\sqrt{\lambda_1} = \text{tr } S(t+1) / \text{tr } S(0) \quad (3-39)$$

$$\text{tr } S(t+1) = \text{tr}[S_1(t)b_1 - e_{10}c_1, S_2(t)b_2 - e_{12}c_2, \dots, S_l(t)b_l - e_{l-1}c_l]$$

L'estimé des paramètres est donné par :

$$\hat{\theta}(t+1) = \hat{\theta}(t) + e_l (ly^T(t+1) - \phi^T(t) \hat{\theta}(t)) / d_l \quad (3-40)$$

L'intérêt de la factorisation (3-27) est que l'on peut mettre à jour la racine carrée S au lieu de la matrice F elle-même. Par conséquent, le produit  $F = SS^T$  sera toujours défini positif.

### 3.4 Convergence paramétrique

La convergence vers zéro de l'erreur de prédiction  $\varepsilon(t)$  n'implique pas toujours la convergence des paramètres estimés vers ceux du processus. En fait, pour permettre une identification correcte des paramètres du modèle du processus, il faut que l'entrée  $u(t)$  du procédé à identifier soit "riche" en fréquence [19][22]. Par exemple, pour permettre l'identification de  $2n$  (ou de  $2n+1$ ) paramètres, l'entrée  $u(t)$  doit être la somme d'au moins  $n$  (ou  $n+1$ ) sinusoides de fréquences distinctes [19].

Nous nous intéressons à l'identification des paramètres d'un système en boucle fermée. Dans ce cas, la condition sur

l'entrée ne suffit pas pour permettre une estimation correcte des paramètres [22]. Il faut que le mécanisme de la contre-réaction ne soit pas trop simple. Les contrôleurs non linéaires ou variant dans le temps ou d'ordre élevé devraient, en général, permettre une identification correcte des paramètres.

Pour l'identification en boucle fermée, il est conseillé [22] d'appliquer les méthodes d'identification à erreur de prédiction, directement, sans précautions particulières pour tenir compte de la contre-réaction.

L'algorithme des MCR est un cas particulier d'algorithme à erreur de prédiction. Nous utiliserons donc l'algorithme des MCR à trace constante pour estimer les paramètres des procédés.

## CHAPITRE IV

DISCRETISATION DE LA COMMANDE

L'algorithme d'identification paramétrique donne des estimés des paramètres du procédé, à chaque pas d'échantillonnage. Ces estimés sont utilisés pour le calcul de la commande qui est appliquée au système. Nous utilisons dans nos simulations une version discrétisée des algorithmes MRACM, MRAC et MCS.

Pour le passage des équations du système analogique aux équations du système discrétisé, nous utilisons l'approximation ZOHDE (zéro order hold discrete equivalent) [36]. Cette méthode donne des résultats précis si la période d'échantillonnage est choisie convenablement. En contrôle adaptatif, la période d'échantillonnage doit être environ égale à un cinquième de la constante de temps la plus rapide [11].

L'approximation ZOHDE fait correspondre à la variable  $s$  de la transformée de Laplace, l'expression

$$\frac{1 - z^{-1}}{z^{-1} T_e}$$

dans le domaine de la transformée en  $z$ .

$T_e$  est la période d'échantillonnage.

Ainsi en considérant l'expression (2-19) du gain  $\delta K(t)$  de l'algorithme MRACM en continu

$$\delta K(t) = (B + \Delta B(t))^{-1} \left[ \int_0^t \alpha y x^T dt + \beta v x^T \right]$$

on peut écrire:

$$\frac{d}{dt} ((B+\Delta B(t))\delta K(t)) = \frac{d}{dt} \left( \int_0^t \alpha v x^T dt + \beta v x^T \right) \quad (4-1)$$

et en utilisant la transformation ci-dessus :

$$\frac{(B+\Delta B(k))\delta K(k) - (B+\Delta B(k-1))\delta K(k-1)}{T_e} =$$

$$\alpha v(k-1) x^T(k-1) + \beta \frac{v(k)x^T(k) - v(k-1)x^T(k-1)}{T_e} \quad (4-2)$$

soit

$$\delta K(k) = (B+\Delta B(k))^{-1} \left[ (B+\Delta B(k-1))\delta K(k-1) + \beta v(k)x^T(k) - \alpha v(k-1)x^T(k-1) \right] \quad (4-3)$$

$$\sigma = \beta - T_e \alpha \quad (4-4)$$

La version discrétisée de l'algorithme MRACM est donnée ci-dessous:

$$x_m(k+1) = (I_n + T_e A_m) x_m(k) + T_e B_m r(k) \quad (4-5)$$

$$u(k) = -K(k) x(k) + K_R(k) r(k) \quad (4-6)$$

$$K(k) = K^0 + \Delta K(k) - \delta K(k) \quad (4-7)$$

$$K_R(k) = K_R^0 + \Delta K_R(k) + \delta K_R(k) \quad (4-8)$$

où

$$\Delta K(k) = (B+\Delta B(k))^{-1} B_p^+(0) (\Delta \hat{A}_p(k) - B_p(0) \Delta \hat{B}(k)) K_p^0 \quad (4-9)$$

$$\Delta K_R(k) = - (B+\Delta B(k))^{-1} \Delta \hat{B}(k) K_R^0 \quad (4-10)$$



$$\delta K(k) = (B + \Delta \hat{B}(k))^{-1} \left[ (B + \Delta \hat{B}(k-1)) \delta K(k-1) + \beta v(k) x^T(k) - \sigma v(k-1) x^T(k-1) \right] \quad (4-11)$$

$$\delta K_R(k) = (B + \Delta \hat{B}(k))^{-1} \left[ (B + \Delta \hat{B}(k-1)) \delta K_R(k-1) + \beta v(k) r^T(k) - \sigma v(k-1) r^T(k-1) \right] \quad (4-12)$$

$$\sigma = \beta - T_0 \alpha$$

## CHAPITRE V

### TESTS ET SIMULATION DES ALGORITHMES

#### MRACM, MRAC ET MCS

Dans d'autres études, plusieurs types de variations des paramètres ont été considérés (variations lentes de grande amplitude, variations rapides mais de faible amplitude, paramètres tendant exponentiellement vers une constante, paramètres présentant des discontinuités) [27], [40].

Pour tester les algorithmes MRAC, MCS et MRACM, nous considérons deux types de variations des paramètres :

- variations sinusoïdales de différentes amplitudes et fréquences,
- variations de type rectangulaire (les paramètres sont constants par morceaux avec des transitions instantanées d'un palier à l'autre).

#### 5.1 Application à un système SISO

Le procédé considéré est donné par l'équation d'état:

$$\dot{x}(t) = a(t) x(t) + b_0 b(t) u(t) \quad (5-1)$$

Le MR est décrit par :

$$\dot{x}_m(t) = -10 x_m(t) + 10 r(t) \quad (5-2)$$

La référence,  $r$ , choisie est un signal carré.

$x$  et  $x_m$  sont des vecteurs d'état de dimension 1.

Pour  $Q = 1$ , la solution de l'équation de Lyapunov est  $P=0,05$ .

Nous appliquons les contrôleurs MRAC, MCS et MRACM à ce système.

### 5.1.1 Variations rectangulaires des paramètres

Dans l'équation d'état (5-1) du procédé, nous prenons :

$$a(t) = a + \Delta a(t) \quad \text{où} \quad a = -5$$

$\Delta a(t)$  est un signal carré d'amplitude  $\pm 0,05a$ .

$$b(t) = b_0 (b + \Delta b(t)) \quad \text{où} \quad b_0 = 10, \quad b = 0,3833$$

$\Delta b(t)$  est un signal carré d'amplitude  $\pm 0,05b$ .

Les valeurs de  $\Delta a(t)$  et  $\Delta b(t)$  sont choisies arbitrairement.

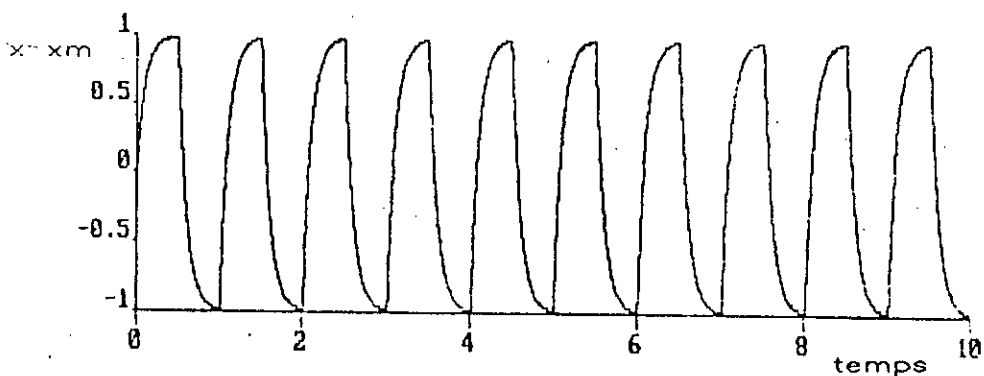
Les résultats de simulation sont présentés dans les figures 5-1 à 5-5.

Les figures 5-1-a, 5-3-a et 5-4-a montrent que la poursuite de l'état du modèle est satisfaisante pour les trois algorithmes. L'erreur de poursuite la plus faible est obtenue pour le MRACM. Elle est représentée en figure 5-1-b. En ce qui concerne le MCS, l'erreur de poursuite (fig. 5-4-b) est plus importante que pour les autres algorithmes au début de l'adaptation (fig. 5-1-b et 5-3-b).

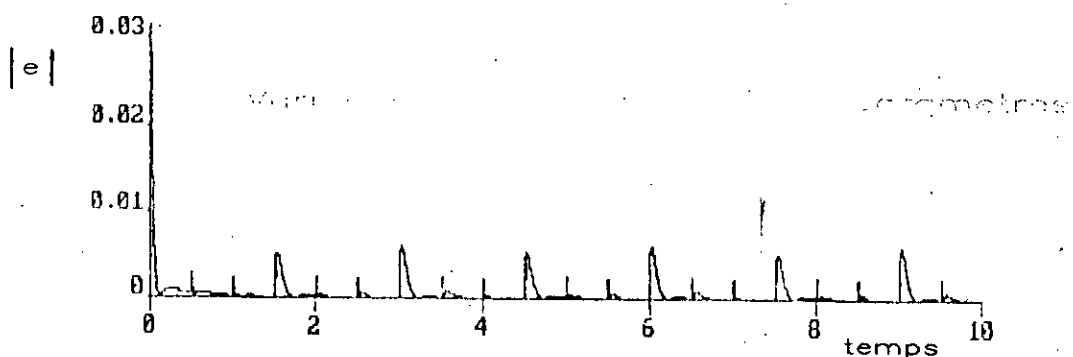
Par la suite (fig. 5-5-a, 5-5-b et 5-1-b) elle devient inférieure à celle obtenue pour le MRAC mais reste supérieure à celle du MRACM.

La valeur maximale de la commande est la même pour le MRAC (fig. 5-3-c) et le MCS (fig. 5-4-c). Elle est inférieure à celle du MRACM pour laquelle on distingue des pics plus nets (fig. 5-1-c).

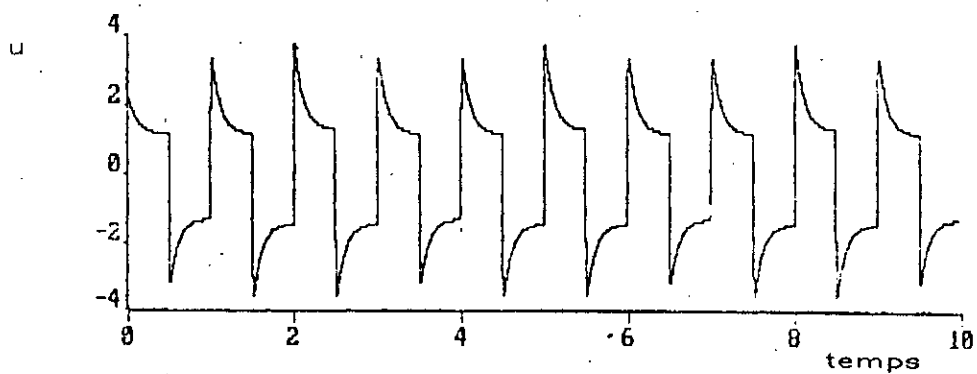
L'algorithme MRACM permet une bonne estimation des paramètres du procédé entre deux transitions successives de ces paramètres (fig. 5-2-a et 5-2-b). Cependant, l'erreur d'identification augmente brusquement à chaque transition des paramètres. Ceci est dû au fait qu'il faut quelques périodes d'échantillonnage pour que l'identificateur obtienne des



a) Etats du procede et du MR  
(2 courbes confondues)

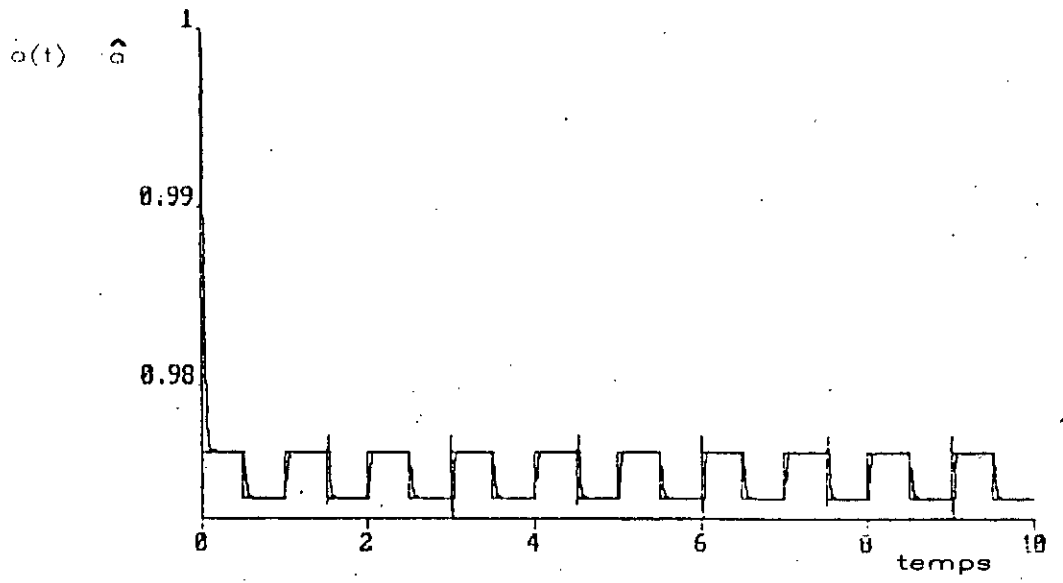


b) Erreur de poursuite  $e=xm-x$

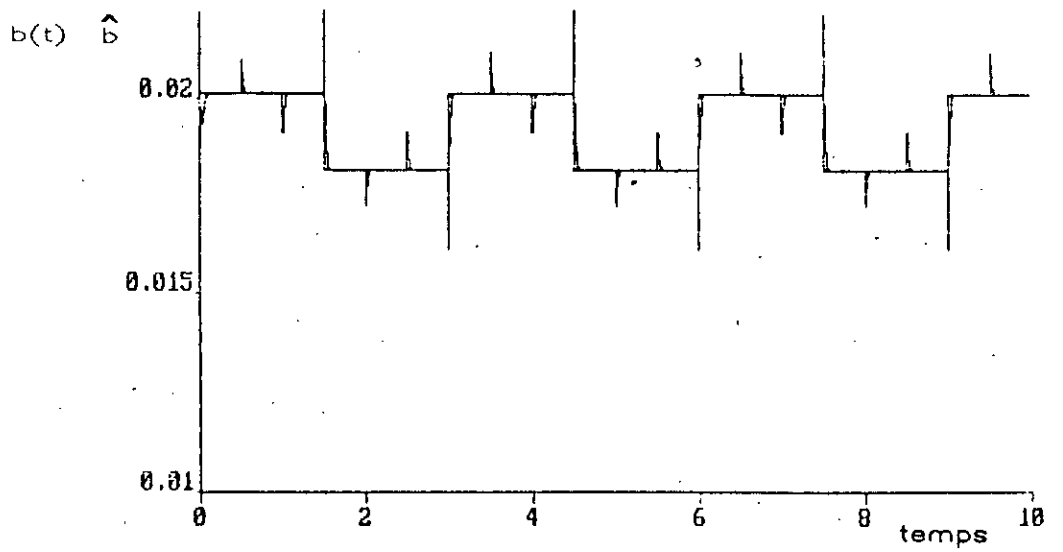


c) Commande

Fig 5-1 Algorithme MRACM



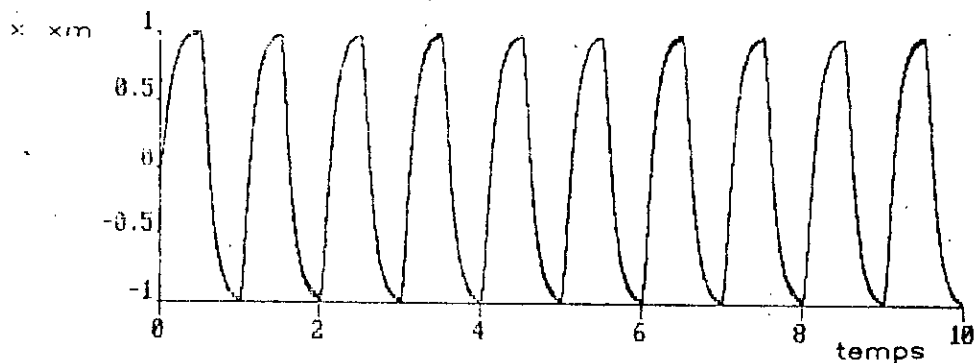
a) Estime du parametre  $a(t)$



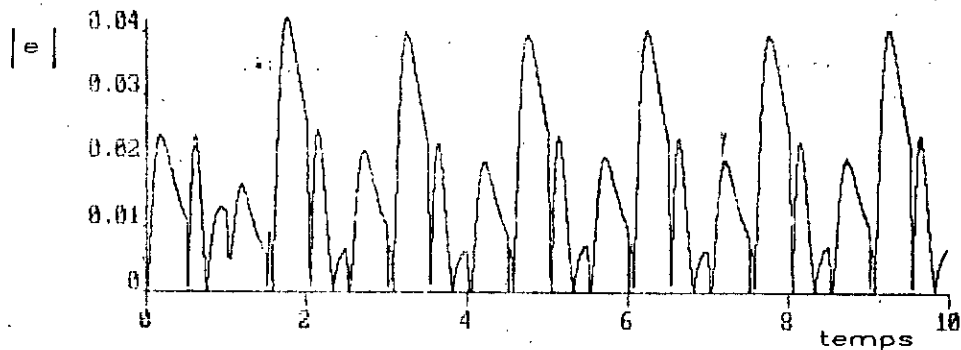
b) Estime du parametre  $b(t)$

Fig. 5-2 Algorithme MRACM

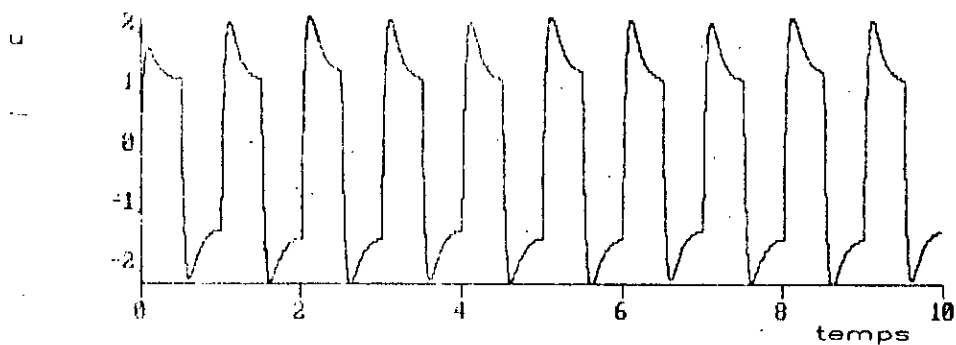
variations rectangulaires des parametres



a) Etats du procede et du MR



b) Erreur de poursuite  $e=xm-x$



c) Commande

Fig 5-3 Algorithme MRAC

Variations rectangulaires des parametres

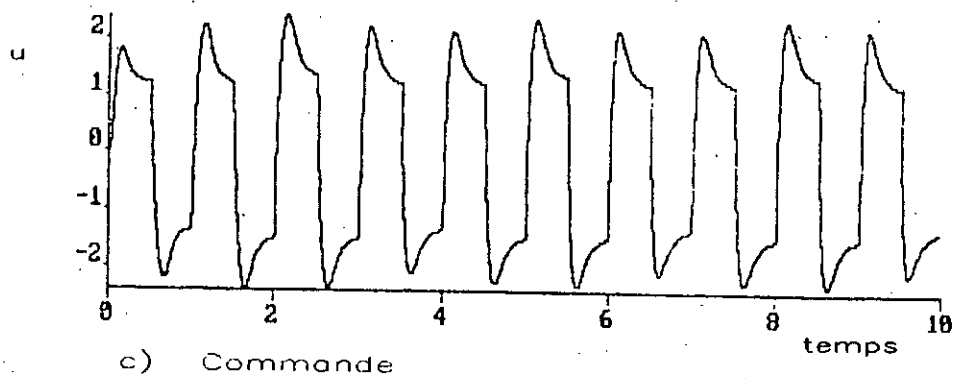
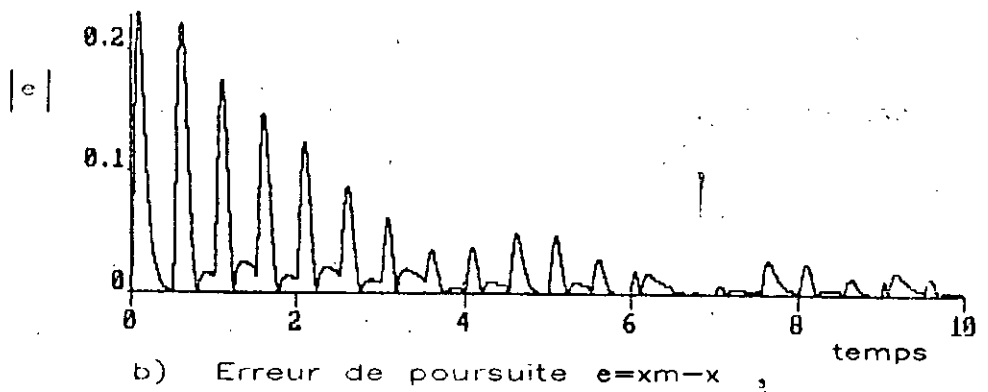
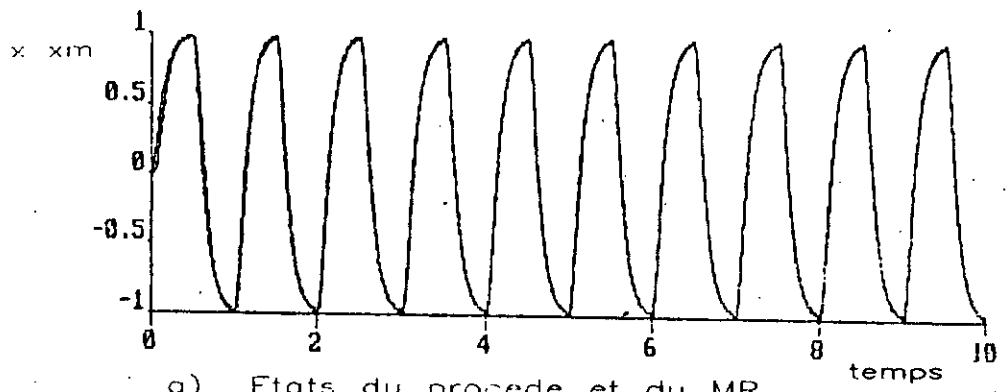
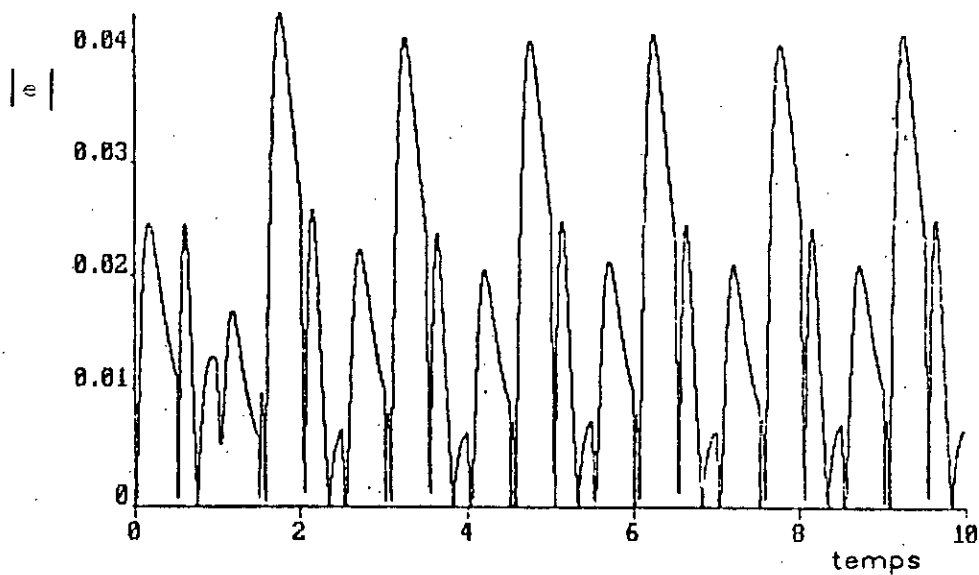
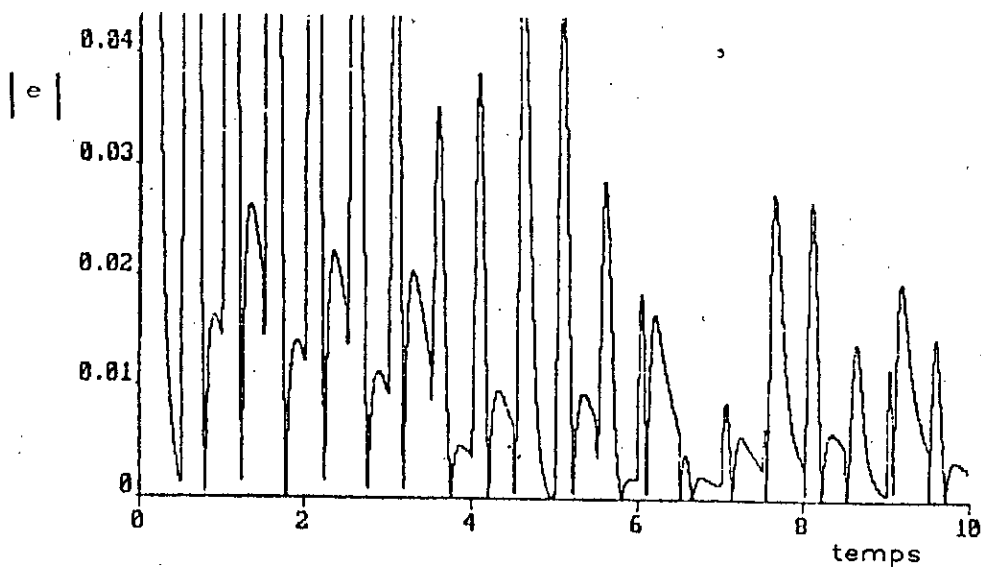


Fig 5-4 Algorithme MCS

Variations rectangulaires des parametres



a) Erreur de poursuite du MRAC



b) Erreur de poursuite du MCS

Fig 5-5 Erreurs de poursuite du MRAC et du MCS



estimés corrects des nouveaux paramètres.

### 5.1.2 Variations sinusoïdales des paramètres

Dans l'équation d'état (5-1) du procédé, nous prenons :

$$a(t) = a + \Delta a(t) , \quad a = -5$$
$$\Delta a(t) = \text{ampa} \sin(2\pi t/T_1) , \quad \text{ampa} = 2$$
$$b(t) = b_0(b + \Delta b(t)) , \quad b_0 = 10 , \quad b = 0,3833$$
$$\Delta b(t) = \text{ampb} \sin(2\pi t/T_2) , \quad \text{ampb} = 0,01$$

$T_1 = T_2 = 700T_e$  de façon à avoir des variations lentes.  
 $\Delta a(t)$  et  $\Delta b(t)$  sont choisies de façon arbitraire.

Les erreurs de poursuite obtenues par simulation, sont présentées en figure 5-6.

Comme précédemment, le MRACM possède l'erreur de poursuite la plus faible. Le MCS possède l'erreur de poursuite initiale la plus élevée. Par la suite cette erreur devient inférieure à celle du MRAC tout en restant supérieure à celle du MRACM.

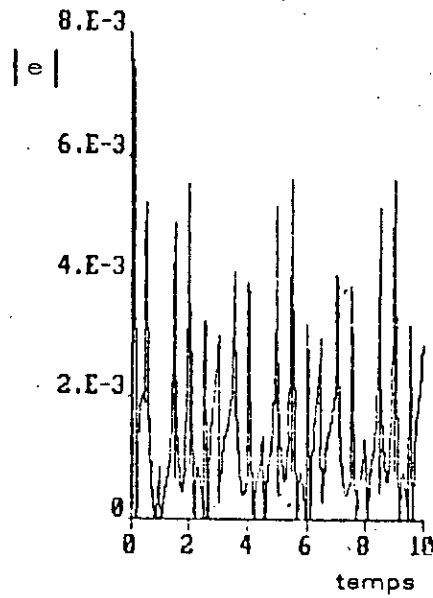
Pour cet exemple, nous avons fait augmenter l'amplitude de  $\Delta a(t)$  et  $\Delta b(t)$ . Les erreurs de poursuite des trois algorithmes ont alors augmenté mais les tendances décrites ci-dessus sont toujours observées.

La commande du MRAC comprend une partie linéaire,  $u_{\text{Lin}}$ , calculée à partir des paramètres  $a$  et  $b$  et une partie adaptative chargée de corriger les erreurs de poursuite dues notamment aux variations  $\Delta a(t)$  et  $\Delta b(t)$  des paramètres par rapport à leurs valeurs  $a$  et  $b$  nominales.

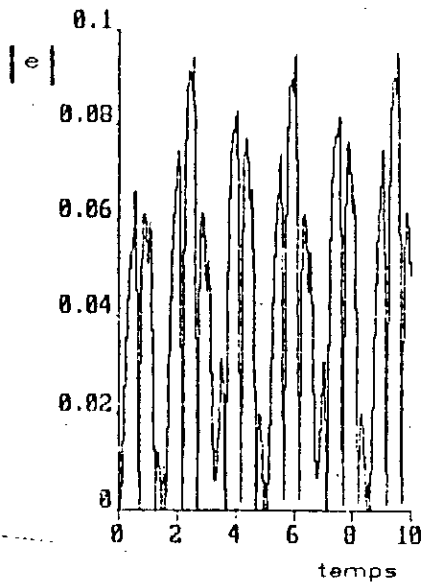
La commande du MCS ne comprend que la partie adaptative de la "commande du MRAC"

Cette différence explique le fait que l'erreur de poursuite est plus faible, au début de l'adaptation, pour le MRAC que pour le MCS. Le MCS ne dispose en effet d'aucune connaissance a priori sur les paramètres du procédé à commander et doit pendant une phase transitoire chercher la

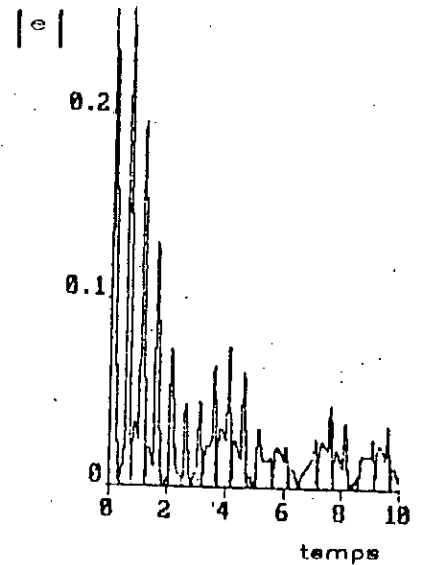
Variations sinusoidales des parametres



a) Erreur de poursuite du MRACM



b) Erreur de poursuite du MRAC



c) Erreur de poursuite du MCS

Fig 5-6 Erreurs de poursuite

commande adéquate qui permet de réduire l'erreur de poursuite.

Le MRAC a dès le départ une connaissance assez grande (si  $\Delta a(t)$  et  $\Delta b(t)$  sont faibles) du système à commander et  $u_{Lin}$  permet dès le début de l'adaptation de réduire considérablement l'erreur de poursuite.

Par contre, par la suite  $\Delta a(t)$  et  $\Delta b(t)$  éloignent  $a(t)$  et  $b(t)$  des valeurs nominales utilisées pour le calcul de  $u_{Lin}$  et la partie adaptative de la commande du MRAC doit alors corriger les "erreurs" de poursuite dues à  $u_{Lin}$  et donc l'erreur de poursuite du MCS devient inférieure à celle du MRAC.

Dans le MRAC modifié, les valeurs des paramètres sont estimées en temps réel et utilisées pour ajuster la partie linéaire de la commande. C'est pourquoi le MRAC modifié conduit à l'erreur de poursuite la plus faible.

## 5.2 Application à un système MIMO

Le processus que nous considérons est donné par l'équation d'état [36] :

$$\dot{x} = Ax(t) + Bu(t) \quad (5-3)$$

$$x(t) = [x_1 \quad x_2]^T \quad u(t) = [u_1 \quad u_2]^T$$

$$A = \begin{bmatrix} -0,02265 & 0,01735 \\ 0,01735 & -0,02265 \end{bmatrix} \quad (5-4)$$

$$B = 0,0167 \quad (5-5)$$

Le MR est représenté dans l'espace d'état par

$$\dot{x}_m = A_m x_m + B_m r \quad (5-6)$$

$$\begin{aligned} A_m &= -0,04 I_2 \\ B_m &= 0,04 I_2 \end{aligned}$$

La période d'échantillonnage est  $T_e = 1s$ .

Dans l'équation de Lyapunov, Q est la matrice identité.

### 5.2.1 Variations sinusoidales des paramètres

Dans l'équation (5-3), A est remplacée par  $A+\Delta A(t)$  où A est donnée par l'équation (5-4), et B est remplacée par  $B_0(B+\Delta B(t))$  où  $B_0 = 0,04 I_2$  et  $B = 0,4175 I_2$ .

Les éléments de  $\Delta A(t)$  et  $\Delta B(t)$  varient de manière sinusoidale :

$$\Delta A(t) = \begin{pmatrix} \Delta a_{11} & 0 \\ 0 & \Delta a_{22} \end{pmatrix} \quad \Delta B(t) = \begin{pmatrix} \Delta b_{11} & 0 \\ 0 & \Delta b_{22} \end{pmatrix}$$

$$\Delta a_{11}(t) = a_1 \sin(2\pi t/kT) \quad (5-7)$$

$$\Delta a_{22}(t) = a_2 \sin(2\pi t/kT) \quad (5-8)$$

$$\Delta b_{11}(t) = b_1 \sin(2\pi t/kT) \quad (5-9)$$

$$\Delta b_{22}(t) = b_2 \sin(2\pi t/kT) \quad (5-10)$$

#### a) Variations lentes des paramètres

Dans les équations (5-7) à (5-10), nous choisissons arbitrairement

$$a_1 \simeq 0,04 a_{11}$$

$$a_2 \simeq 0,07 a_{22}$$

$$b_1 \simeq 0,07 b_{11}$$

$$b_2 \simeq 0,1 b_{22}$$

$$k = 200.$$

#### a-1) Référence constante

Considérons dans un premier temps que l'entrée de référence est constante :

$$r = [r_1 \quad r_2]^T = [3 \quad 2,6]^T$$

Dans le cas de l'algorithme MRACM, pour permettre une meilleure convergence des estimés des paramètres vers leurs valeurs réelles, nous ajoutons au signal de commande un signal aléatoire uniformément distribué et de variance 0,01.

Les résultats de simulations sont représentés en figures 5-7 à 5-9.

Pour les trois algorithmes, la poursuite du modèle est très satisfaisante. Ceci est illustré en figure 5-7-a pour le MRACM et en figure 5-7-b pour le MCS. Les courbes obtenues pour le MRAC et le MCS étant pratiquement identiques, nous n'avons représenté que celles correspondant au MCS.

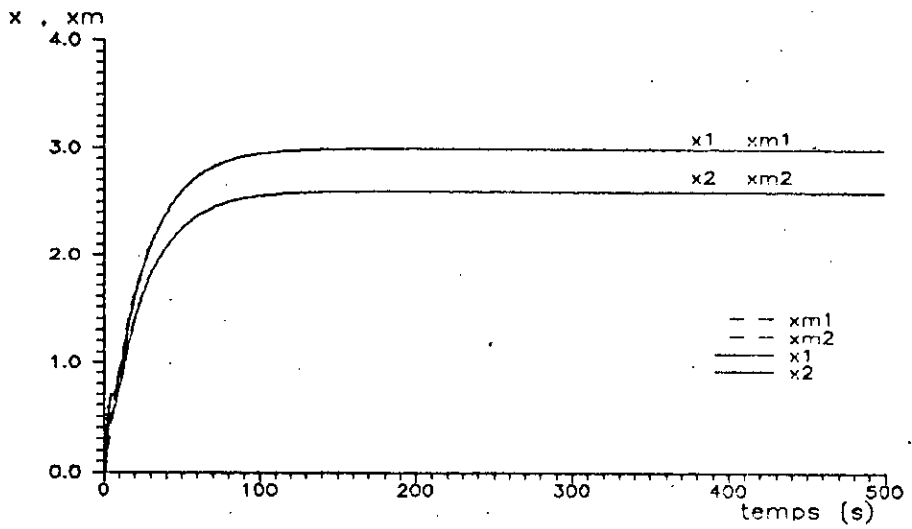
Au début de l'adaptation, l'erreur de poursuite la plus importante est obtenue pour le MRACM (Fig. 5-8-a). Celle du MCS représentée en figure 5-8-b et celle du MRAC sont à peu près identiques. Par la suite, l'erreur de poursuite du MRACM reste supérieure à celle du MRAC, elle-même supérieure à celle du MCS, comme on peut le voir dans les figures 5-8-c et 5-8-d. Cependant, pour les trois algorithmes, l'erreur de poursuite est inférieure à 5% de la valeur, en régime établi, de l'état.

Le terme aléatoire ajouté à la commande ne permet pas une estimation correcte des paramètres du procédé (fig 5-9-a et 5-9-b), tout en entraînant une erreur de poursuite pour le MRACM supérieure (en moyenne et en amplitude) à celles obtenues pour le MRAC et le MCS.

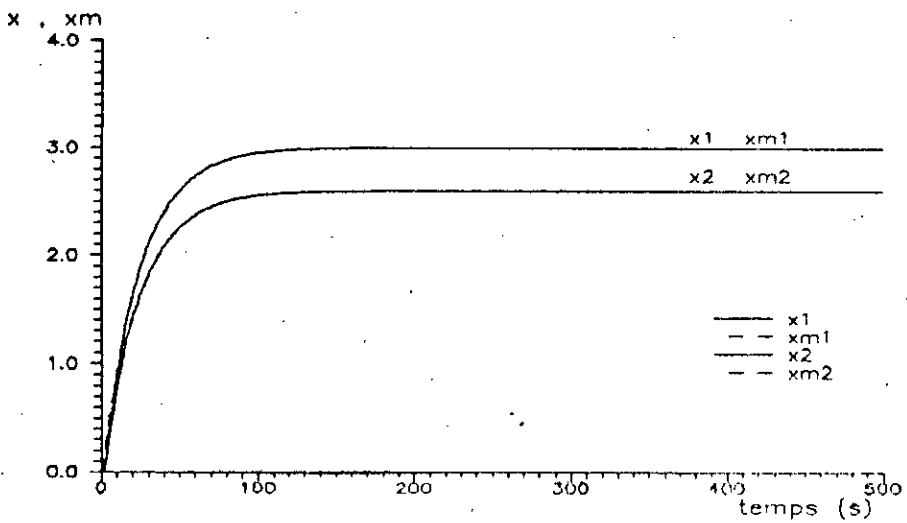
#### a-2 Référence carrée

Afin de permettre une meilleure estimation des paramètres du procédé et tirer ainsi un meilleur parti de l'algorithme MRACM, nous considérons un signal de référence carré.

Variations sinusoidales des parametres  
 Reference constante



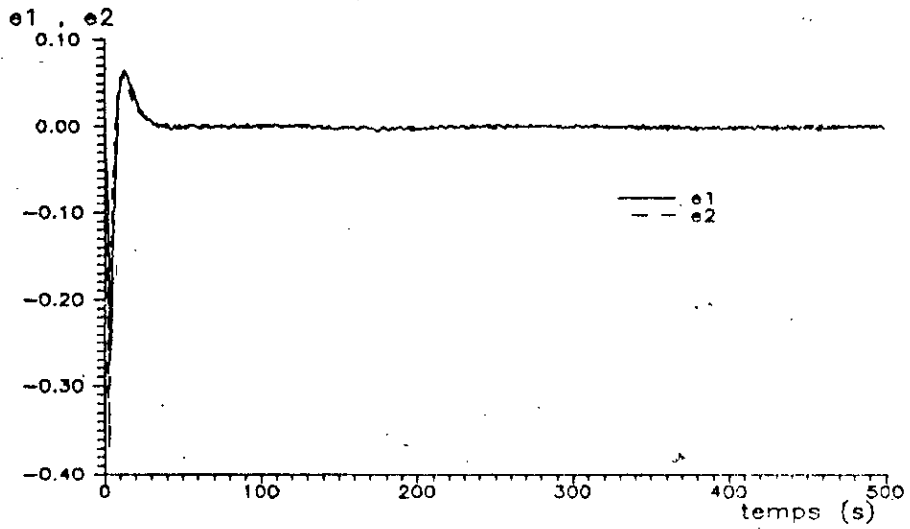
a) M R A C modifie



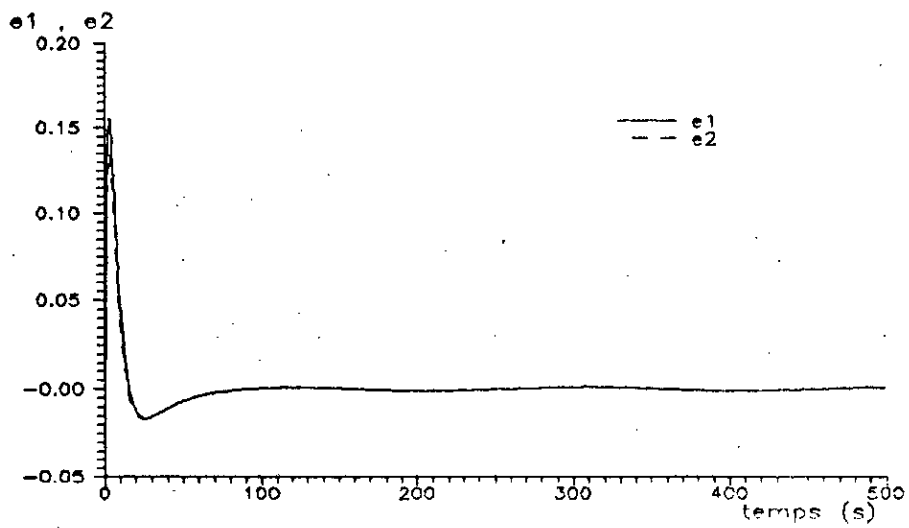
b) M C S

Fig 5-7 Poursuite des etats

Variations sinusoidales des paramètres  
Reference constante



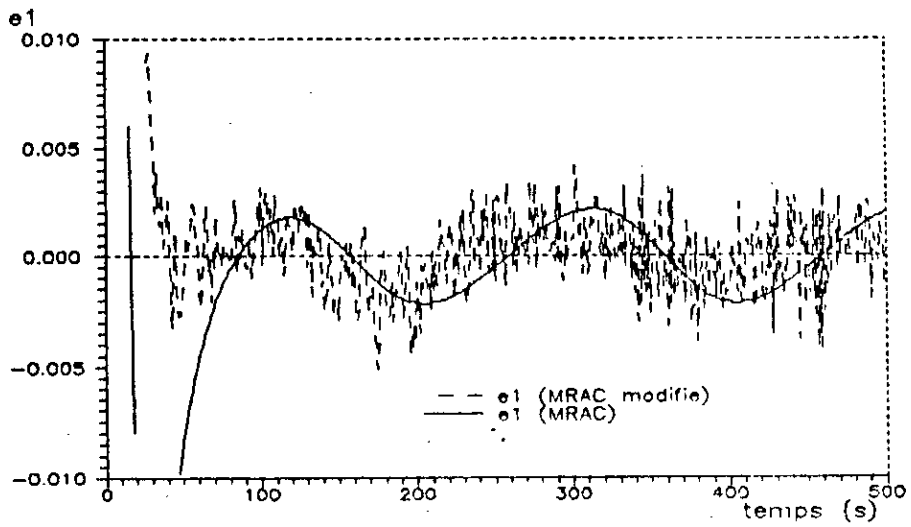
a) M R A C modifie



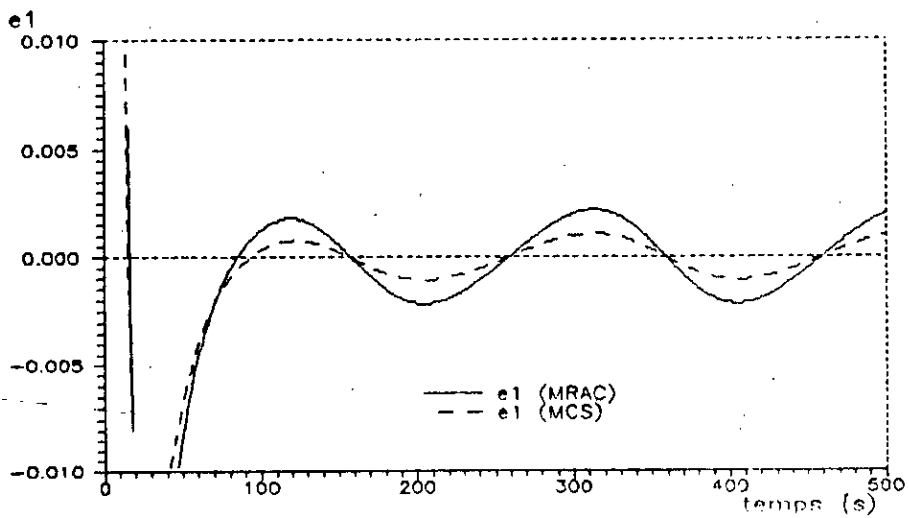
b) M C S

Fig 5-8 Erreur de poursuite

Variations sinusoidales des paramètres  
Reference constante



c) M R A C et M R A C modifié

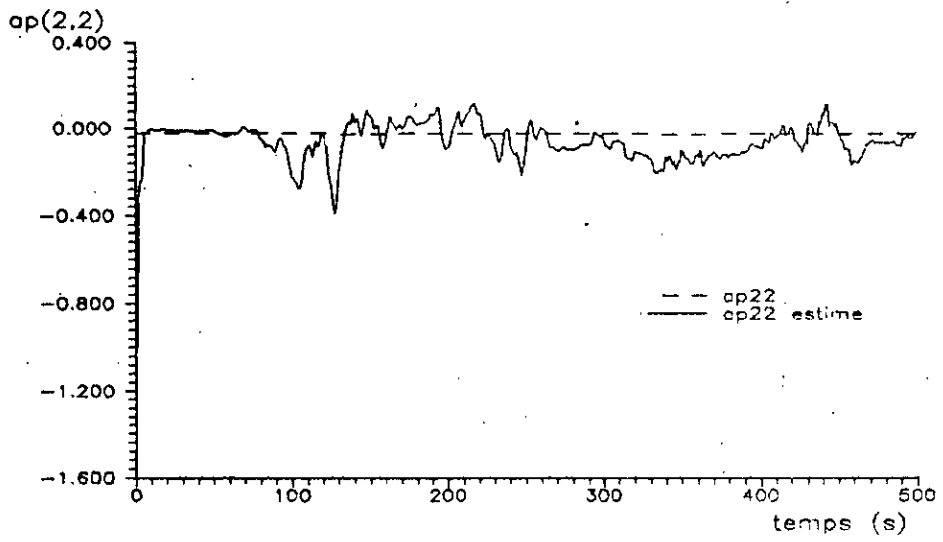


d) M R A C et M C S

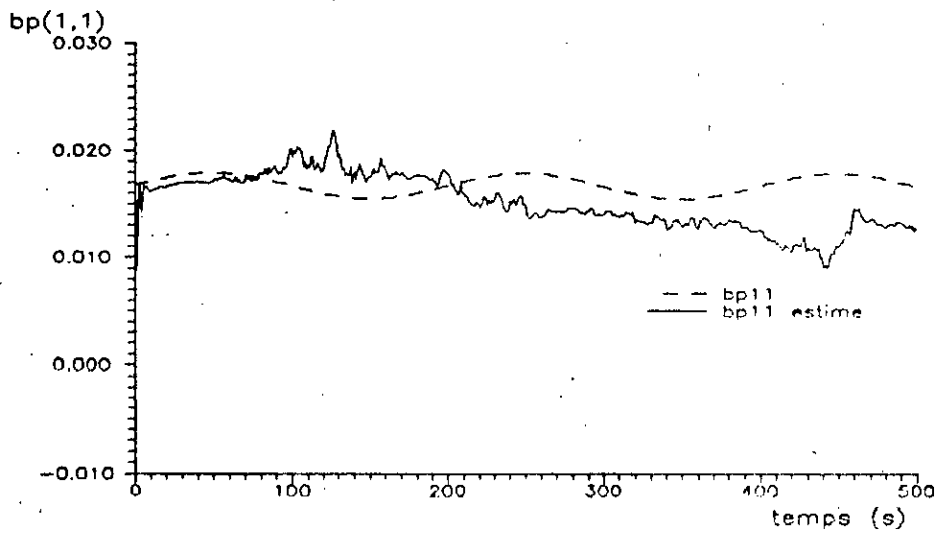
Fig 5-8 Erreur de poursuite



Variations sinusoidales des parametres  
Reference constante



a) M R A C modifie



b) M R A C modifie

Fig 5-9 Parametres estimes

Les figures 5-10 à 5-13 rassemblent les résultats de simulation.

La poursuite est satisfaisante pour les trois algorithmes. Le résultat obtenu pour le MRACM est illustré en figure 5-10. La figure 5-11 montre qu'au début de l'adaptation, l'erreur de poursuite la plus élevée est celle du MRACM. Celles du MRAC et du MCS sont comparables. Par la suite, l'erreur de poursuite la plus faible est obtenue pour le MRACM et la plus élevée pour le MRAC.

La figure 5-12-a montre des pics importants dans la commande du MRACM, au début de l'adaptation. Par la suite on constate la présence de légers pics qui coïncident avec ceux observés dans les estimés des paramètres en figure 5-13.

Les commandes du MRAC et du MCS ne présentent pas de telles transitions et sont à peu près identiques. Celle du MCS est représentée en figure 5-12-b.

L'utilisation de la référence de type carré permet une meilleure poursuite des paramètres mais avec des erreurs d'estimation importantes à chaque transition du signal de référence. Ceci peut être observé en comparant les figures 5-9 et 5-13. En fait, une transition du signal de référence correspond à une modification rapide de l'environnement du système à identifier. Pour que l'identificateur obtienne des estimés corrects, il faut attendre quelques périodes d'échantillonnage.

Nous conservons cette référence pour tous les tests suivants de ce système.

#### b) Variations en amplitude des paramètres de la matrice A

---

Nous considérons de nouvelles valeurs dans les équations (5-7) et (5-8) :

$$a_1 \approx 0,5 a_{11} \qquad a_2 \approx 0,55 a_{22}$$

Variations sinusoidales des parametres  
Reference carree

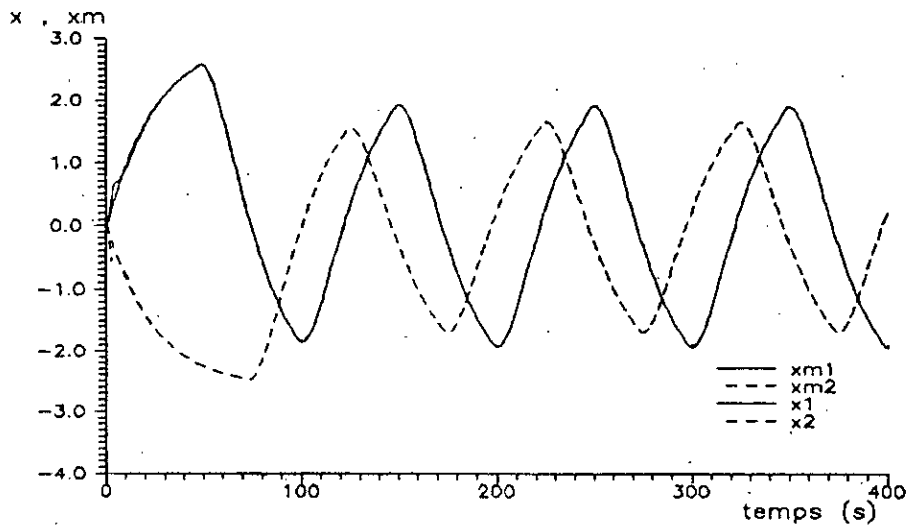
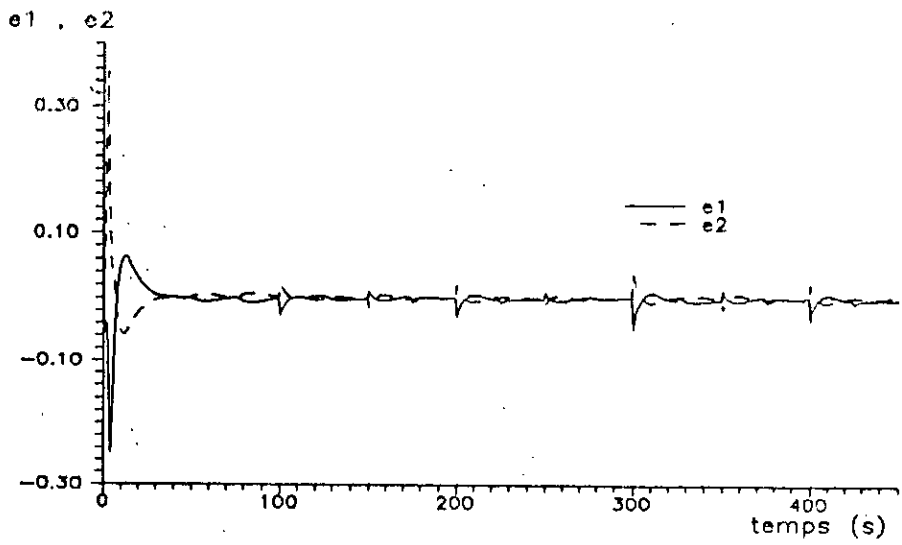
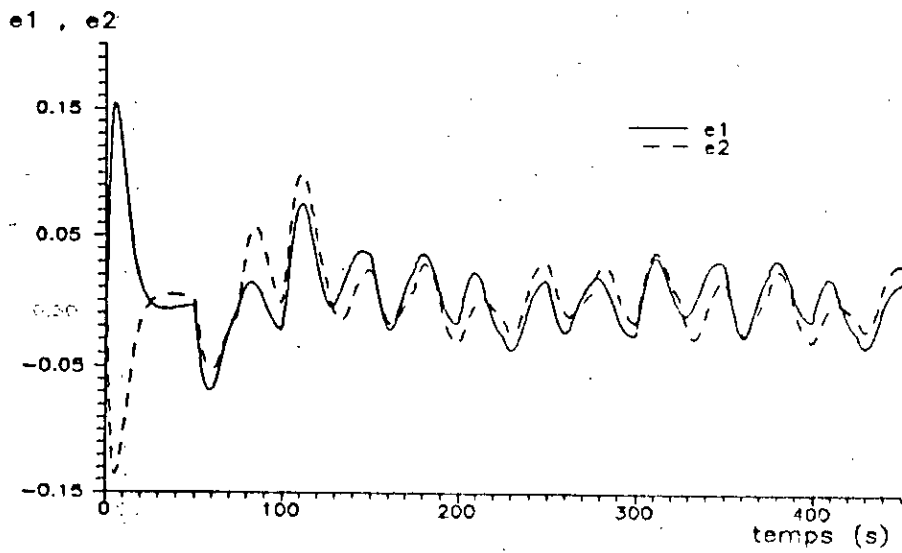


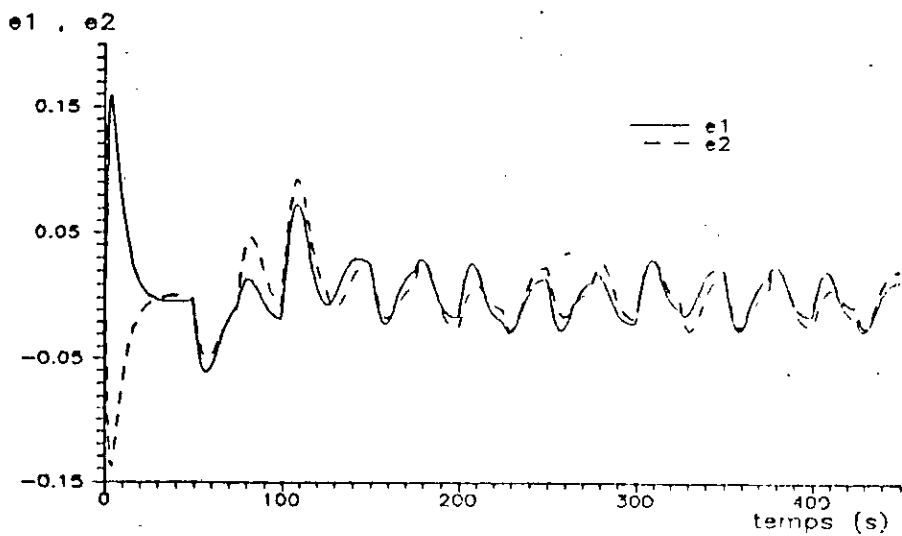
Fig 5-10 Poursuite des etats pour le M R A C modifie



a) M R A C modifié



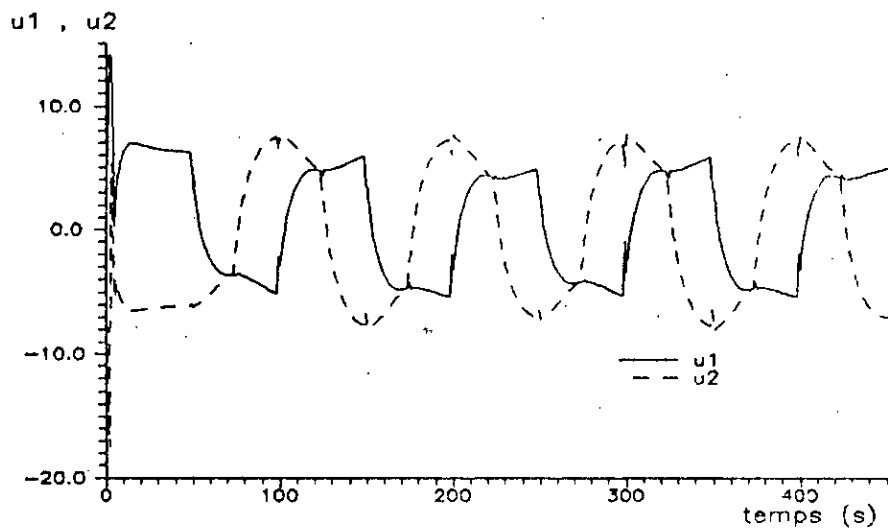
b) M R A C



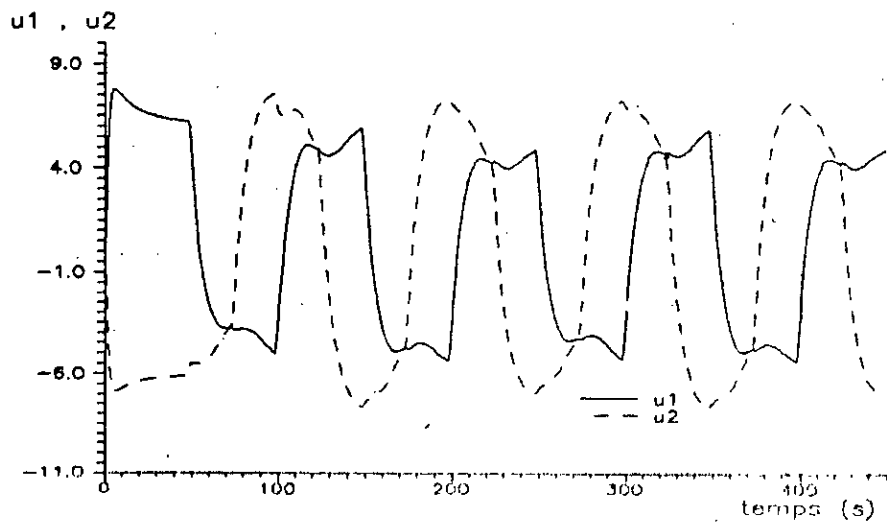
c) M C S

Fig 5-11 Erreur de poursuite

Variations sinusoidales des parametres  
Reference carree



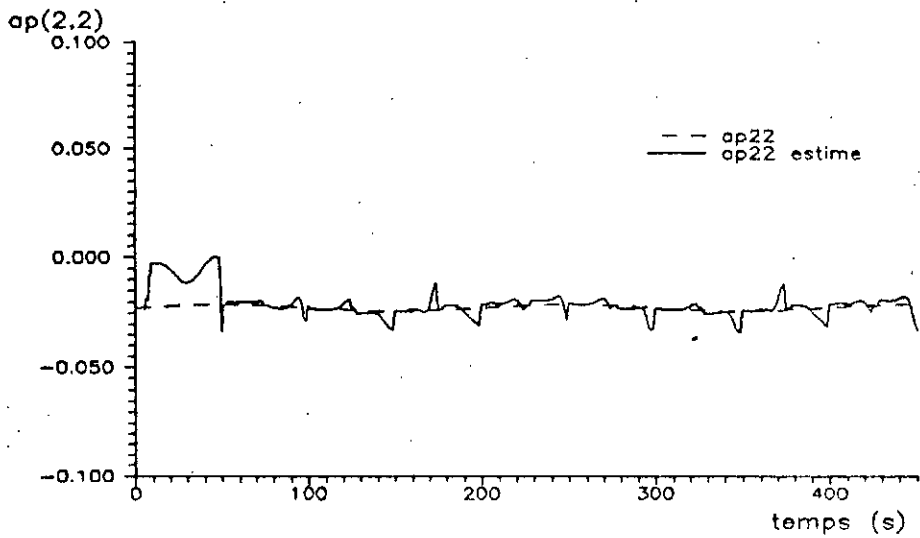
a) M R A C modifie



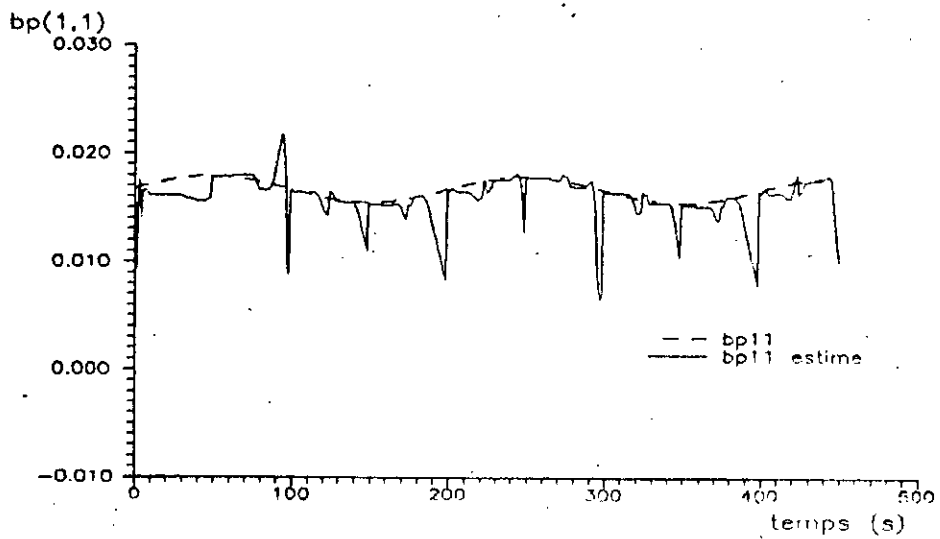
b) M C S

Fig 5-12 Commande

Variations sinusoidales des parametres  
Reference carree



a) M R A C modifie



b) M R A C modifie

Fig 5-13 Parametres estimees

$\Delta B(t)$  est inchangé par rapport au cas étudié au paragraphe précédent.

Les résultats de simulation sont présentés dans les figures 5-14 à 5-17.

La poursuite est assez bonne pour le MRACM, mais l'on décèle en figure 5-14, pour  $t \geq 270$  s, de légers écarts entre  $x$  et  $x_m$ .

L'amplitude de l'erreur de poursuite augmente quand les paramètres de A augmentent en amplitude. Ceci peut être observé en comparant les figures 5-11 et 5-15.

Les erreurs d'amplitude maximale sont obtenues pour le MRACM. Le MRAC et le MCS ont la même amplitude maximale de l'erreur de poursuite, au début de l'adaptation.

Par la suite l'amplitude de l'erreur de poursuite du MCS devient inférieure à celle du MRAC. La figure 5-15 illustre ces résultats.

L'erreur maximale est obtenue pour le MRACM mais, en moyenne cette erreur est inférieure à celles des algorithmes MRAC et MCS :

$$s_i = \sum_{k=1}^{450} e_i^2, \quad i = 1, 2$$

	MRACM	MRAC	MCS
$s_1$	0,36	0,70	0,40
$s_2$	0,34	0,80	0,44

Les figures 5-12-a et 5-16-a montrent que l'augmentation de l'amplitude des variations de A entraîne une augmentation de l'amplitude des pics qui apparaissent dans la commande du MRACM. Ces pics peuvent être indésirables pour les organes de commande. La commande du MCS représentée en figure 5-16-b et celle du MRAC sont à peu près identiques. Elles ne présentent

Variations en amplitude des parametres de A

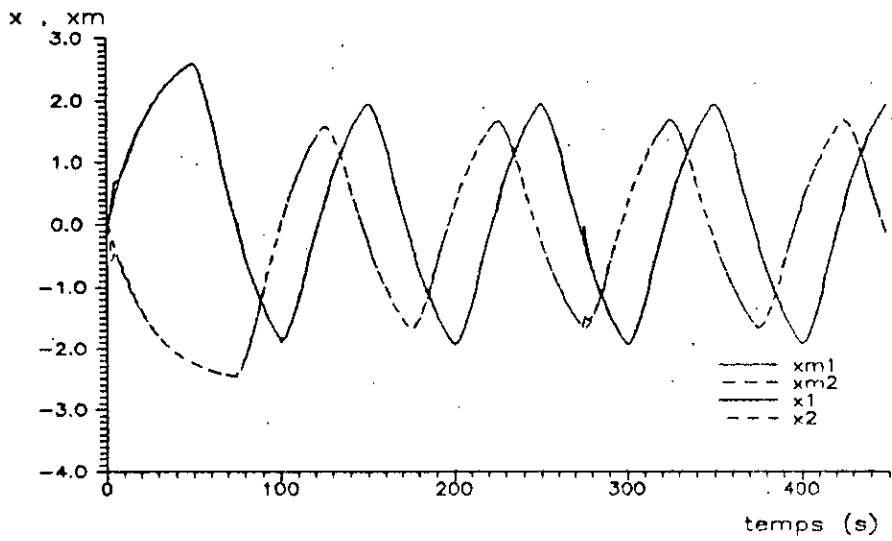
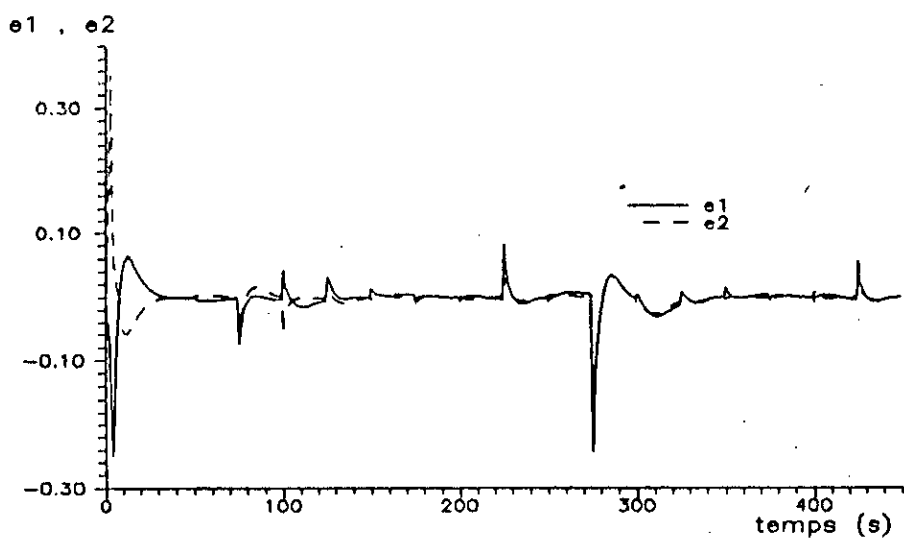
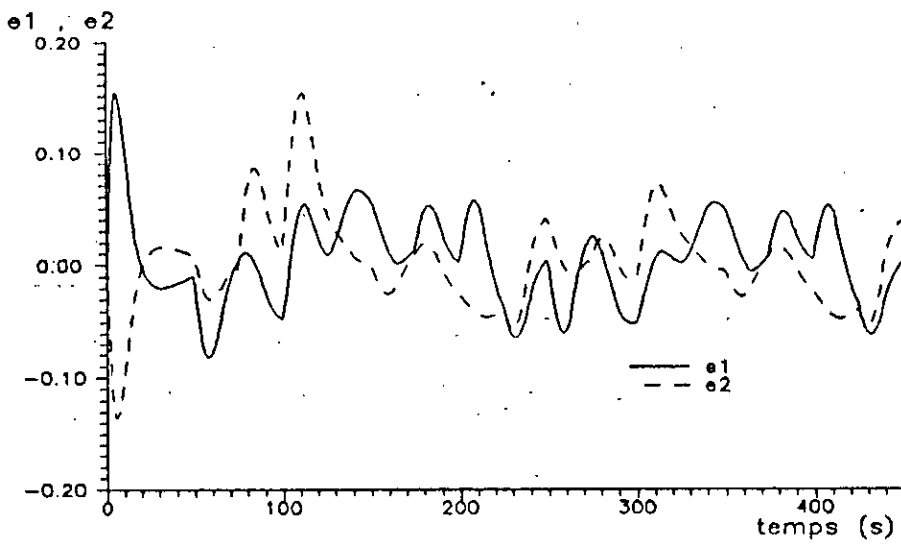


Fig 5-14 Poursuite des etats pour le MRAC modifie

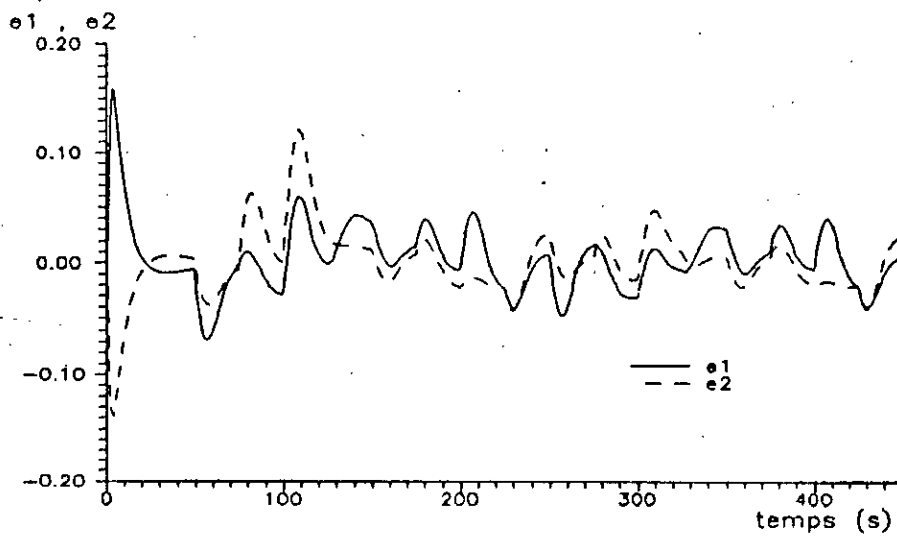




a) M R A C modifie



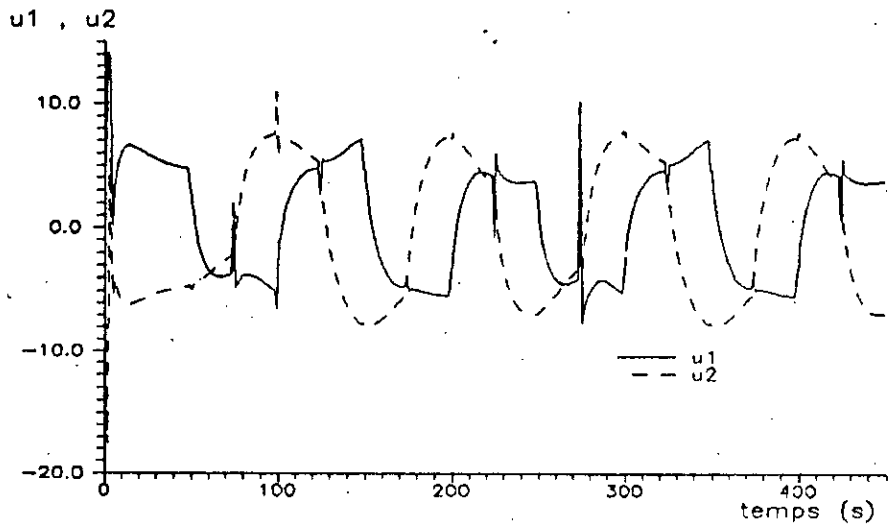
b) M R A C



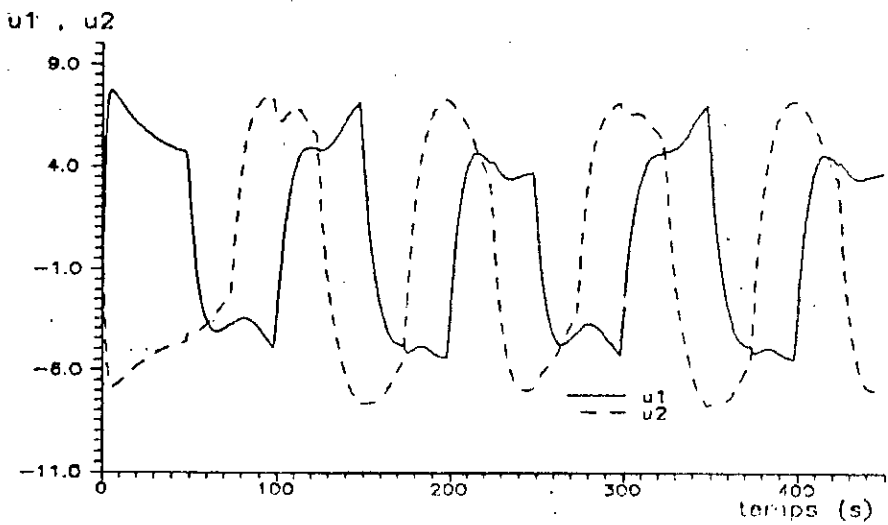
c) M C S

Fig 5-15 Erreur de poursuite

Variations en amplitude des paramètres de A



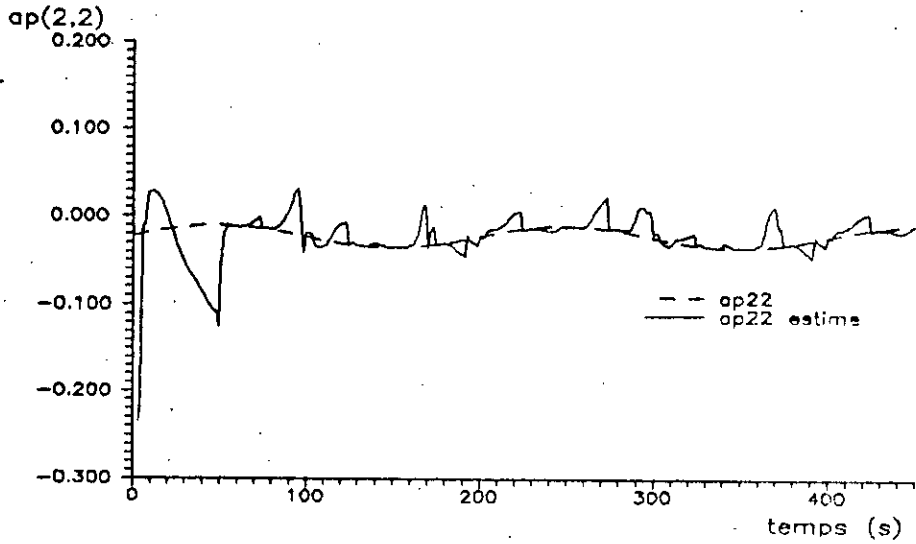
a) M R A C modifie



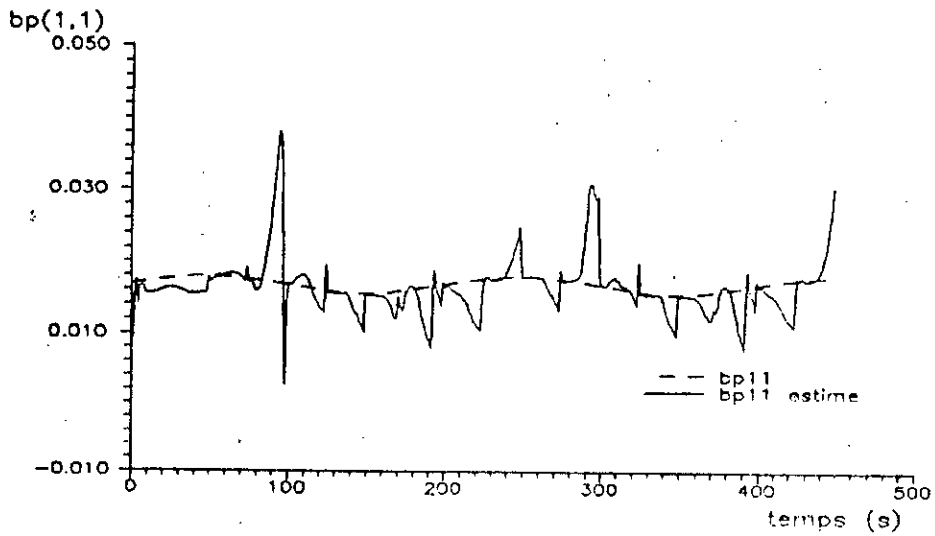
b) M C S

Fig 5-16 Commande

Variations en amplitude des paramètres de A



a) M R A C modifié



b) M R A C modifié

Fig 5-17 Paramètres estimés

pas de pics et varie très peu pour la modification considérée de l'amplitude des paramètres du procédé.

Quand l'amplitude des variations des paramètres de  $\Delta A(t)$  augmente, l'amplitude des pics dans l'erreur d'estimation augmente aussi, comme on peut le voir en comparant les figures 5-13-b et 5-17-b.

### c) Variations en amplitude des paramètres de la matrice B

---

Nous prenons maintenant les valeurs :

$$b_1 \approx 0,2 \quad b_{11} \quad b_2 \approx 0,25 \quad b_{22}$$

La matrice  $\Delta A(t)$  est identique à celle considérée au paragraphe a).

Les figures 5-18 à 5-21 rassemblent les résultats de simulation.

Comme au paragraphe précédent, on constate des écarts entre  $x$  et  $x_m$  pour le MRACM, mais la poursuite reste bonne. Ceci est illustré en figure 5-18.

Quand l'amplitude des paramètres de B augmente, on peut remarquer, en comparant les figures 5-11 et 5-19, que l'amplitude des erreurs de poursuite augmente pour les trois algorithmes.

L'erreur de poursuite du MRACM est généralement inférieure à celles du MRAC et du MCS avec toutefois des pics très importants que l'on ne retrouve pas dans les deux autres algorithmes. L'erreur de poursuite du MCS est inférieure à celle du MRAC. Ces résultats sont illustrés en figure 5-19.

On retrouve, comme dans le cas du paragraphe précédent, des pics importants dans la commande. Ceci peut être observé en figure 5-20-a, pour  $t \approx 424$  s. A cet instant, les éléments de la matrice  $(B + \Delta B)^{-1}$  ont des valeurs élevées. Comme cette matrice multiplie les gains  $\Delta K$ ,  $\delta K$  et  $\Delta K_R$ ,  $\delta K_R$ , ceci donne une commande trop importante.

Dans le cas du MRACM, il faut prendre en compte la

Variations en amplitude des parametres de B

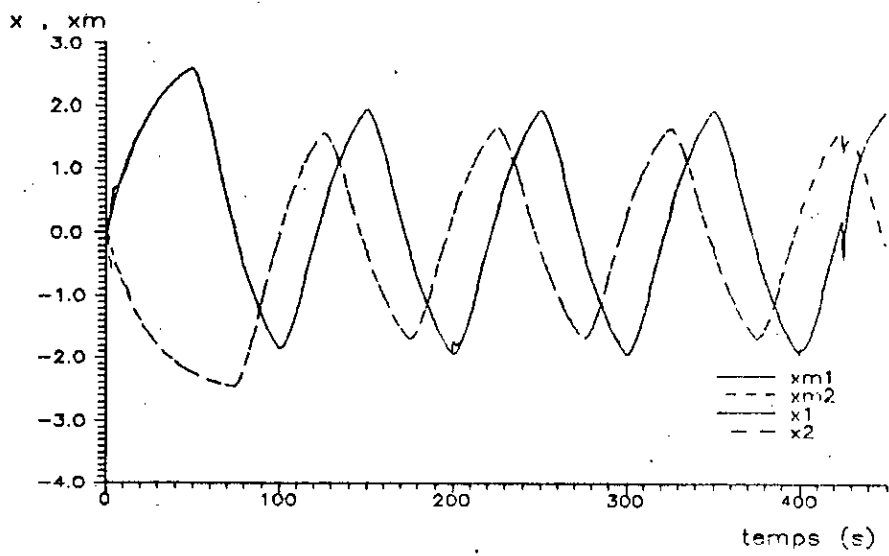
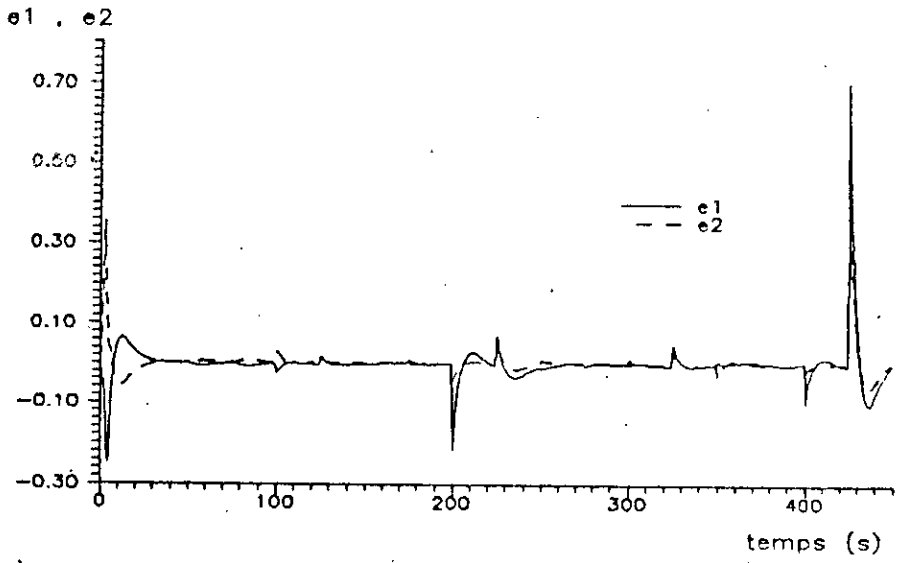
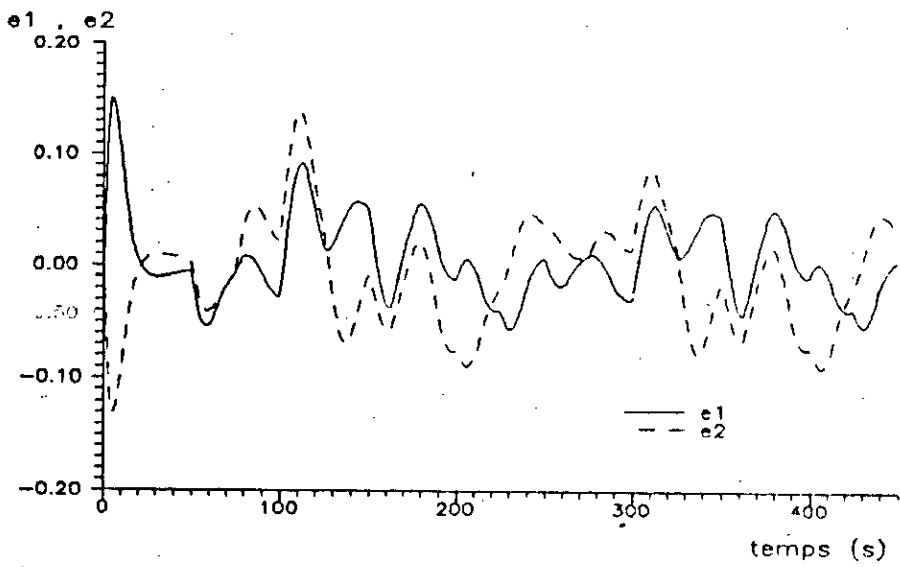


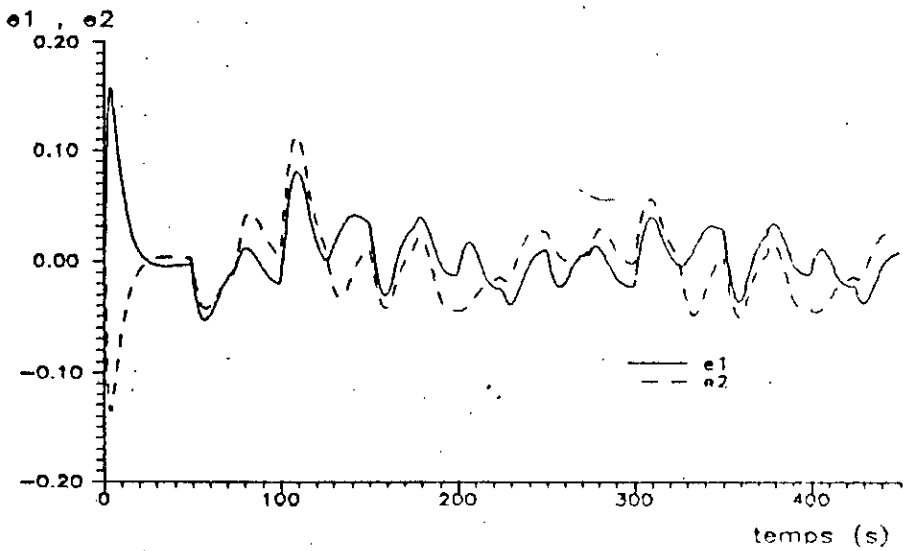
Fig 5-18 Poursuite des etats pour le MRAC modifie



a) M R A C modifie



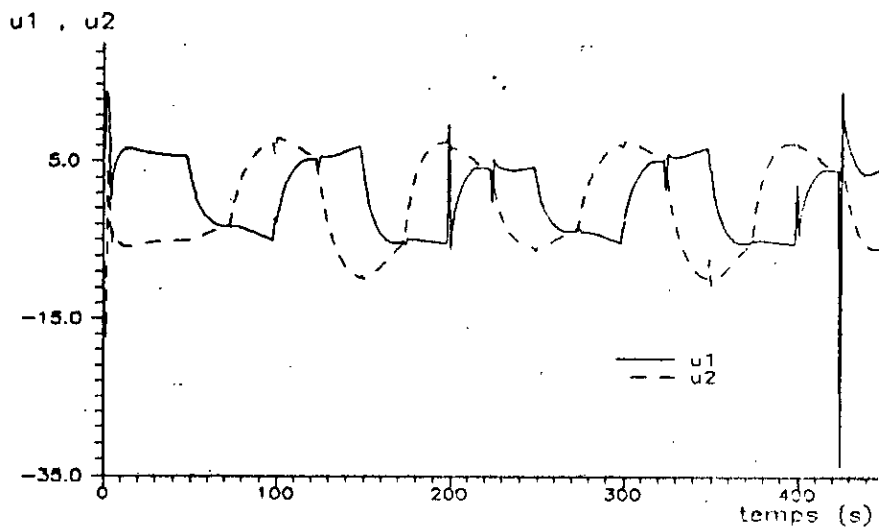
b) M R A C



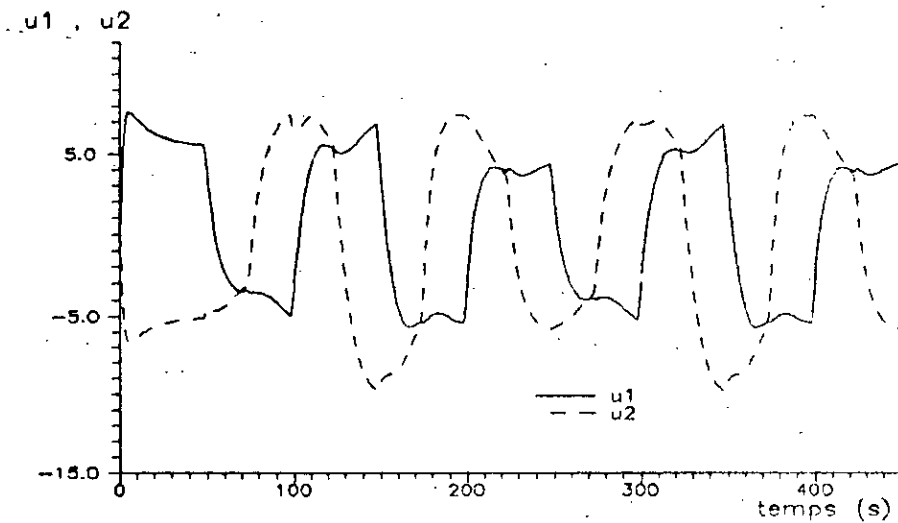
c) M C S

Fig 5-19 Erreur de poursuite

Variations en amplitude des parametres de B



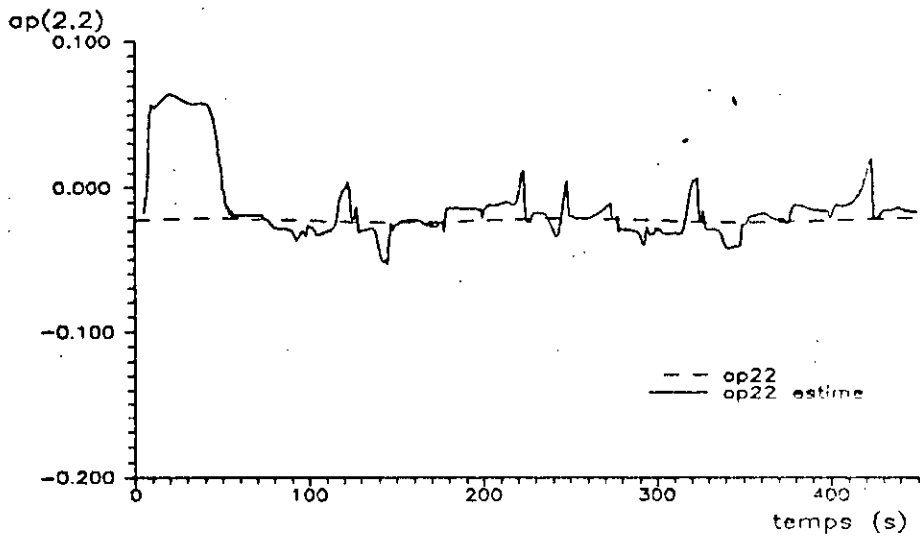
a) M R A C modifie



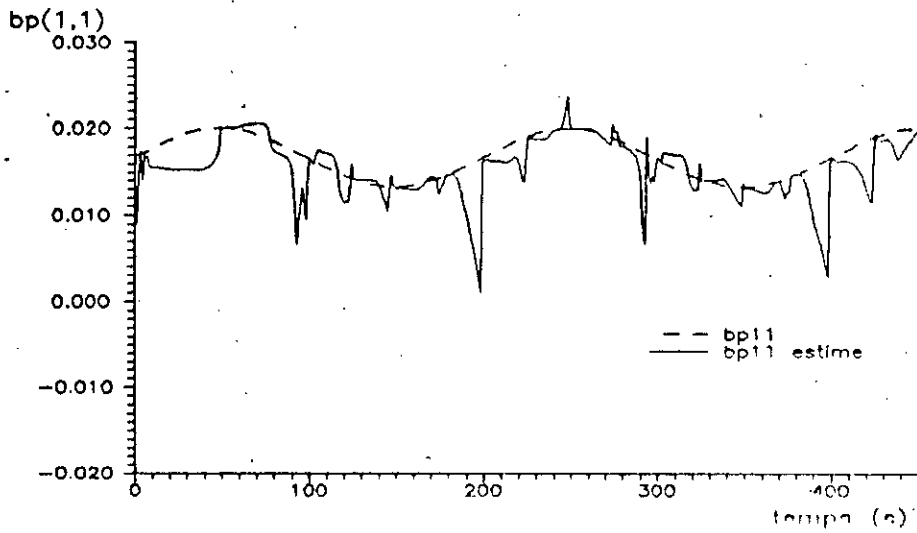
b) M C S

Fig 5-20 Commande

Variations en amplitude des parametres de B



a) M R A C modifie



b) M R A C modifie

Fig 5-21 Parametres estimates



contrainte suivante : la matrice  $B + \Delta B(t)$  doit être inversible.

Quand l'amplitude des éléments de la matrice  $\Delta B(t)$  augmente, l'erreur d'estimation des paramètres augmente aussi. On peut l'observer en comparant les figures 5-13-a et 5-21-a.

#### d) Variations rapides des paramètres

Dans ce paragraphe, nous nous intéressons aux variations fréquentielles des éléments de  $A(t)$  et  $B(t)$ . Dans les équations (5-7) à (5-10), les amplitudes des paramètres sont identiques à celles considérées au paragraphe a). Par contre, les variations de ces paramètres sont plus rapides. En effet nous prenons  $k = 30$ .

Les résultats de simulation sont présentés dans les figures 5-22 à 5-25.

Dans le cas du MRACM, il apparaît un écart entre  $x$  et  $x_m$  que l'on ne retrouve pas dans le cas du MCS. Ceci est illustré par la figure 5-22.

L'amplitude des pics de l'erreur de poursuite du MRACM, représentée en figure 5-23-a, augmente par rapport au cas du paragraphe a) où les variations des paramètres sont lentes.

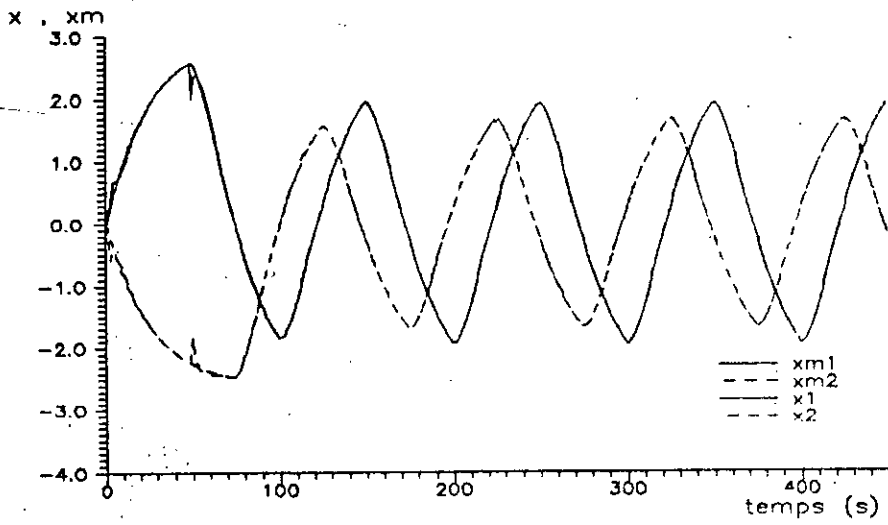
Pour le MRAC et le MCS, l'erreur de poursuite reste du même ordre de grandeur qu'au paragraphe a). Ces résultats peuvent être observés en comparant les figures 5-23 et 5-11.

La figure 5-24-a montre des pics très importants dans la commande du MRACM au début de l'adaptation.

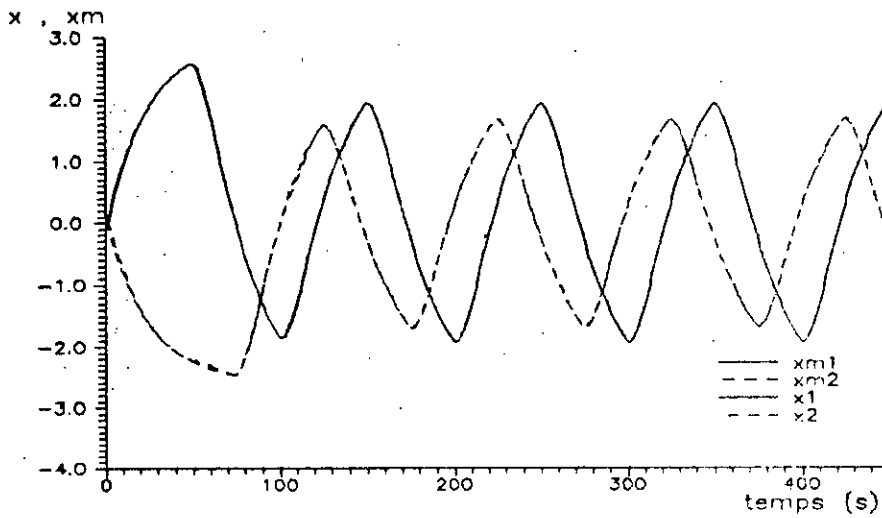
La commande du MCS est représentée en figure 5-24-b.

Pendant les deux premières périodes ( $t \in [0, 60s]$ ) les estimés des paramètres sont parfois très éloignés des vraies valeurs de ces derniers, comme le montre la figure 5-25. Ces mauvais résultats d'estimation expliquent les pics observés dans la commande du MRACM, en figure 5-24-a, à  $t \approx 50$  s. Cette commande inadéquate appliquée au procédé est à l'origine du

Variations sinusoidales rapides des parametres

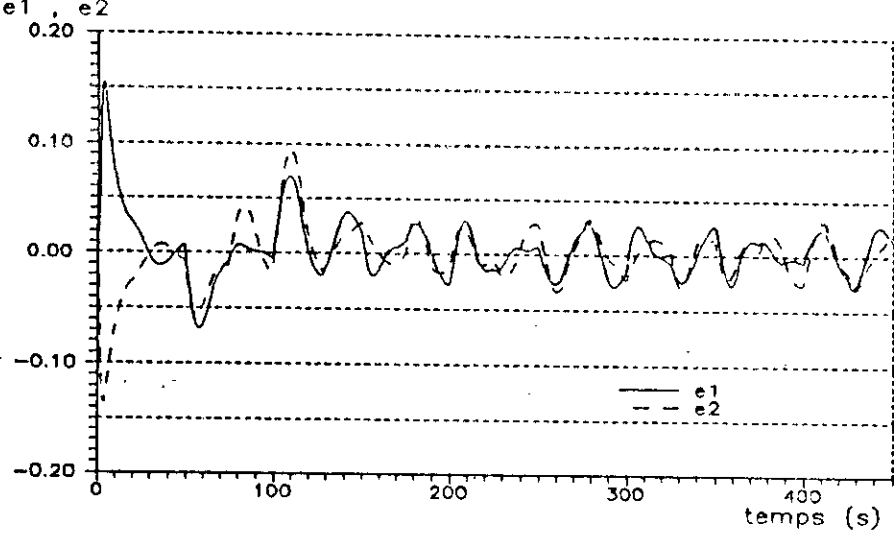


a) M R A C modifie

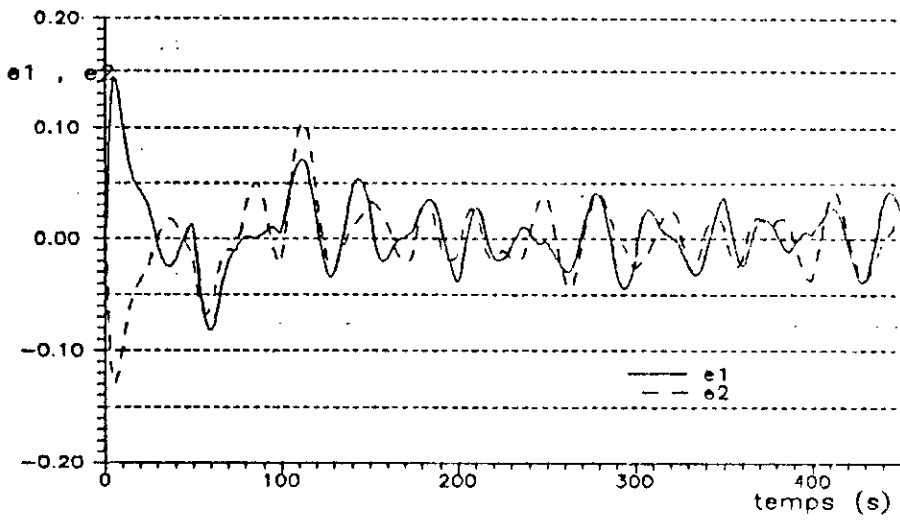


b) M C S

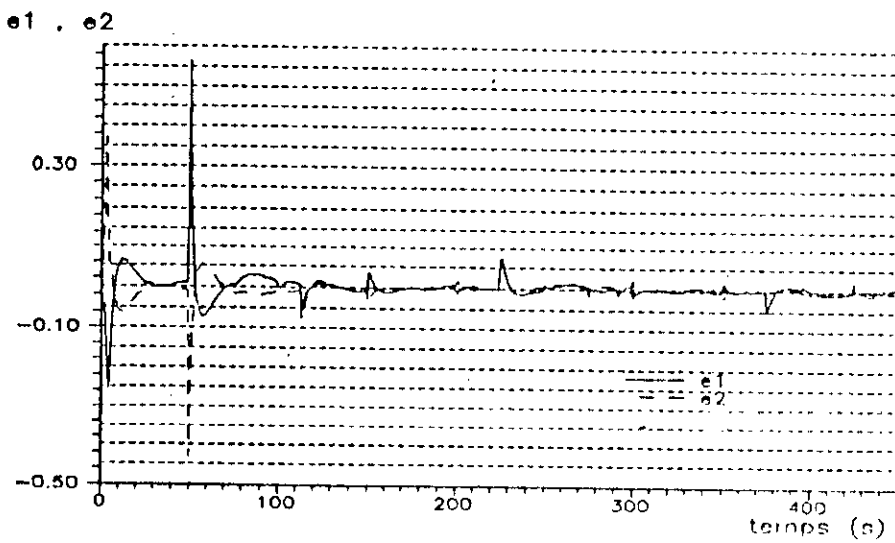
Fig 5-22 Poursuite des etats



c) M C S



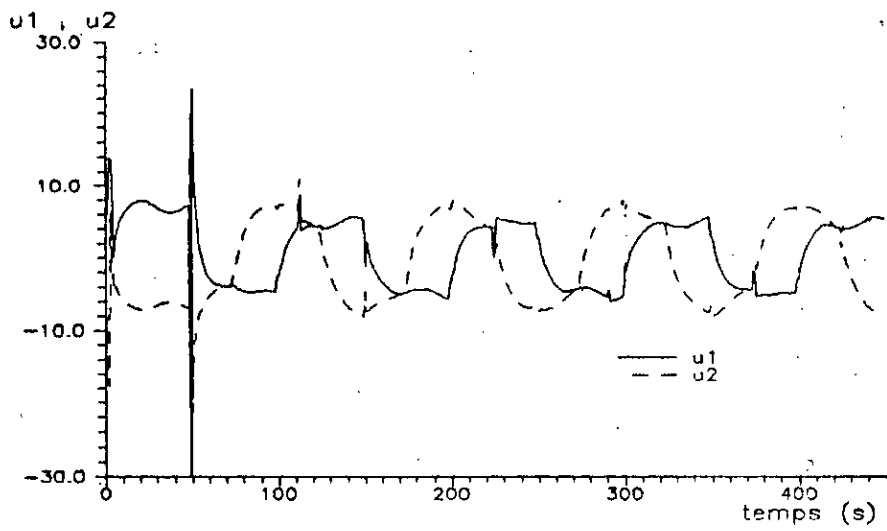
b) M R A C



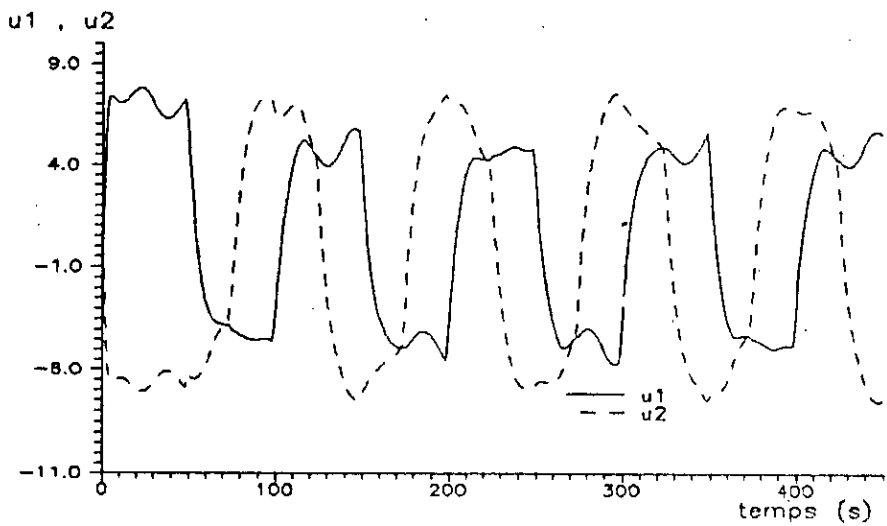
a) M R A C modifie

Fig 5-23 Erreur de poursuite

Variations sinusoidales rapides des parametres



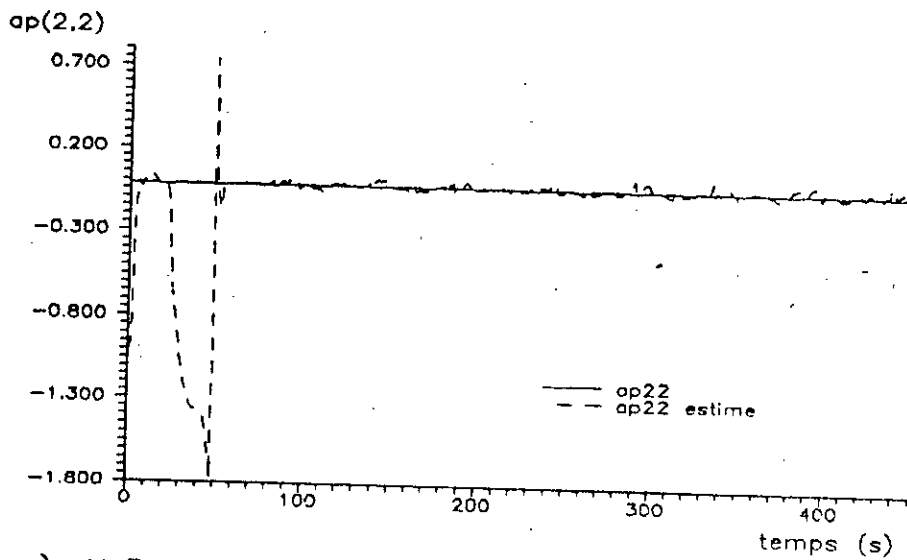
a) M R A C modifie



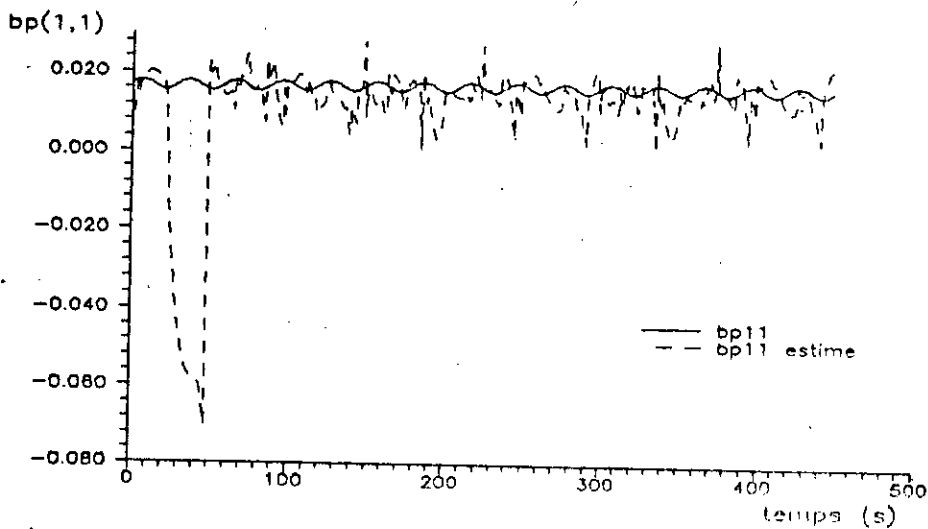
b) M C S

Fig 5-24 Commande

# Variations sinusoidales rapides des parametres



a) M R A C modifie



b) M R A C modifie

Fig 5-25 Parametres estimates

pic important observé dans l'erreur de poursuite, en figure 5-23-a, à  $t \approx 50$  s.

Tous ces pics n'existent pas dans le cas du MRAC et du MCS. Ces deux algorithmes n'utilisent pas d'estimés en temps réel des paramètres du procédé à commander.

### 5.2.2 Variations rectangulaires des paramètres

Dans l'équation d'état (5-3) du procédé, la matrice d'évolution A et la matrice de commande B sont respectivement remplacées par

$$A + \Delta A(t) \qquad B_0 (B + \Delta B(t))$$

où A est donné par l'équation (5-4) et

$$B_0 = 0,04I_2 \qquad B = 0,4175I_2$$

$$\Delta A(t) = \begin{bmatrix} \Delta a_{11} & 0 \\ 0 & \Delta a_{22} \end{bmatrix} \qquad \Delta B(t) = \begin{bmatrix} \Delta b_{11} & 0 \\ 0 & \Delta b_{22} \end{bmatrix}$$

$\Delta a_{11}$ ,  $\Delta a_{22}$ ,  $\Delta b_{11}$ ,  $\Delta b_{22}$  sont des signaux constants par morceaux, d'amplitudes assez faibles (10% du paramètre nominal) mais présentant des transitions brusques.

Les résultats de simulation sont rassemblés dans les figures 5-26 à 5-29.

La poursuite est satisfaisante pour les trois algorithmes. La figure 5-26 montre le résultat obtenu pour le MRACM.

L'erreur de poursuite du MRACM initialement très importante devient inférieure à celle du MCS.

L'erreur de poursuite du MCS comparable à celle du MRAC, au début de l'adaptation, devient par la suite inférieure à cette dernière. Ces résultats sont présentés en figure 5-27.

Comme le montre la figure 5-28, la commande du MRACM présente des valeurs initiales très élevées. Par la suite, elle est à peu près identique à celles du MRAC et du MCS mais, elle peut présenter de légers pics (comme à  $t \approx 150$  s).

### Variations rectangulaires des parametres

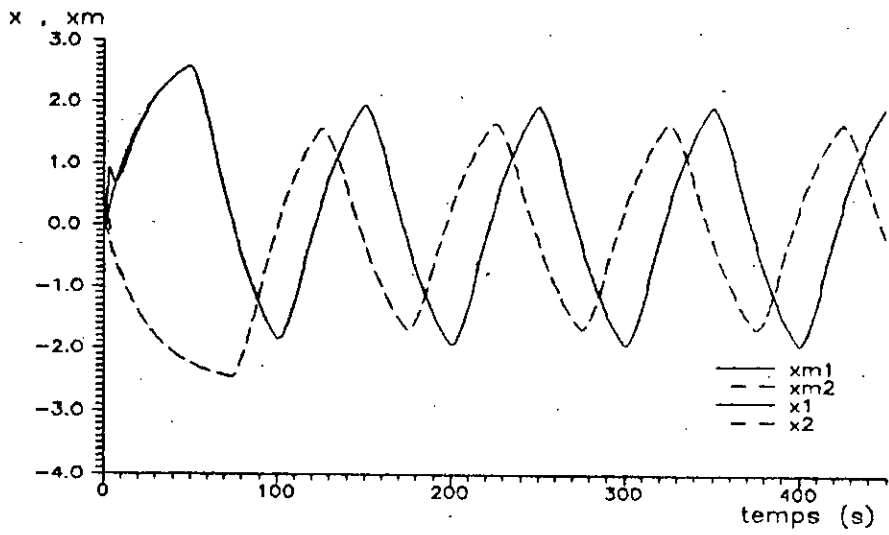
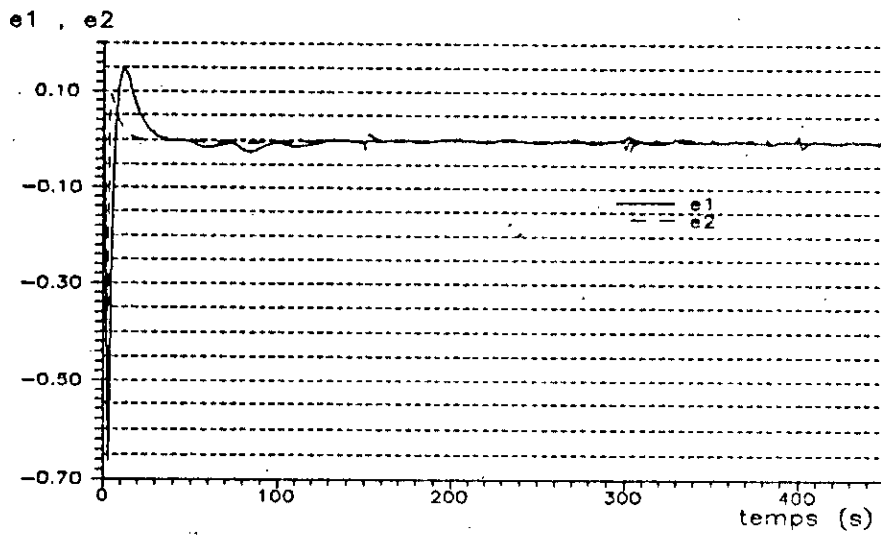
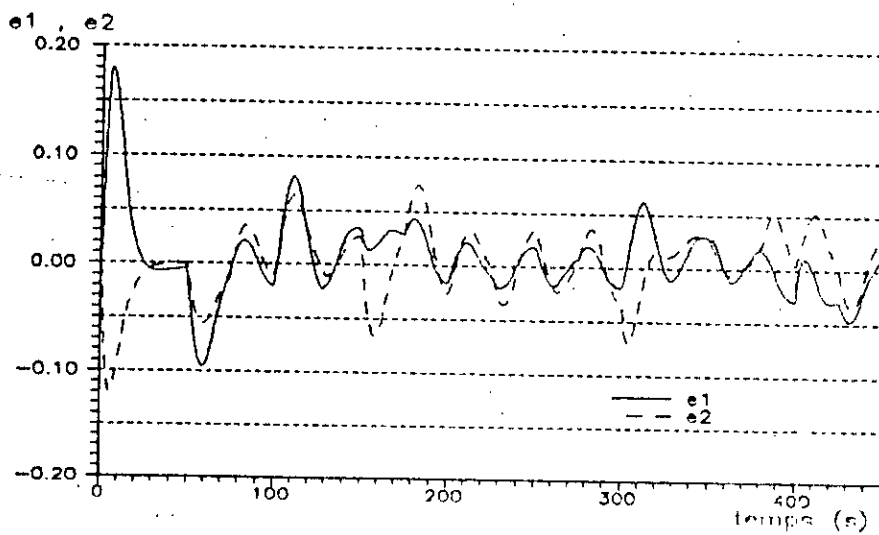


Fig 5-26 Poursuite des etats pour le M R A C modifie

# Variations rectangulaires des parametres



a) M R A C modifie

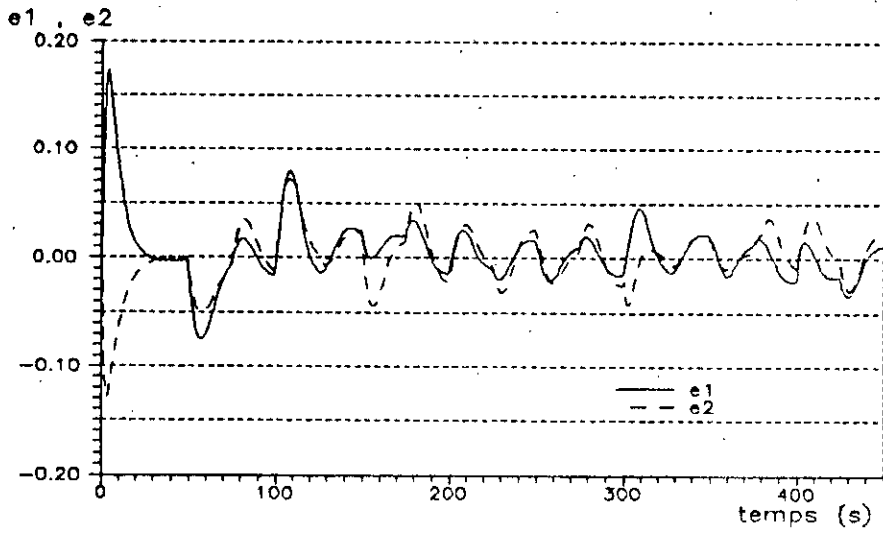


b) M R A C

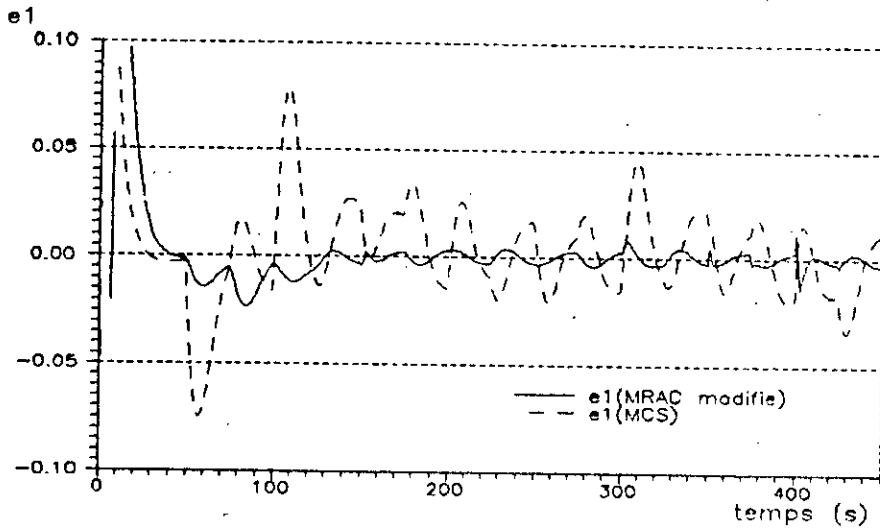
Fig 5-27 Erreur de poursuite



Variations rectangulaires des parametres



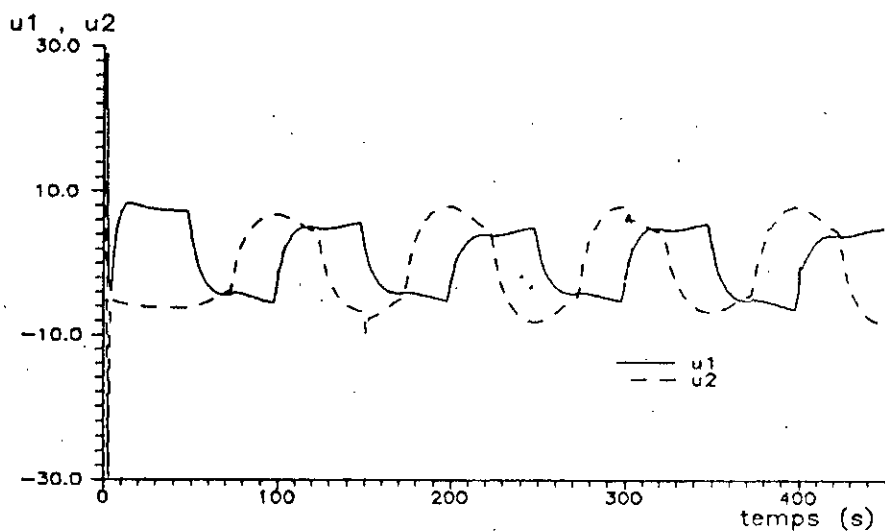
c) M C S



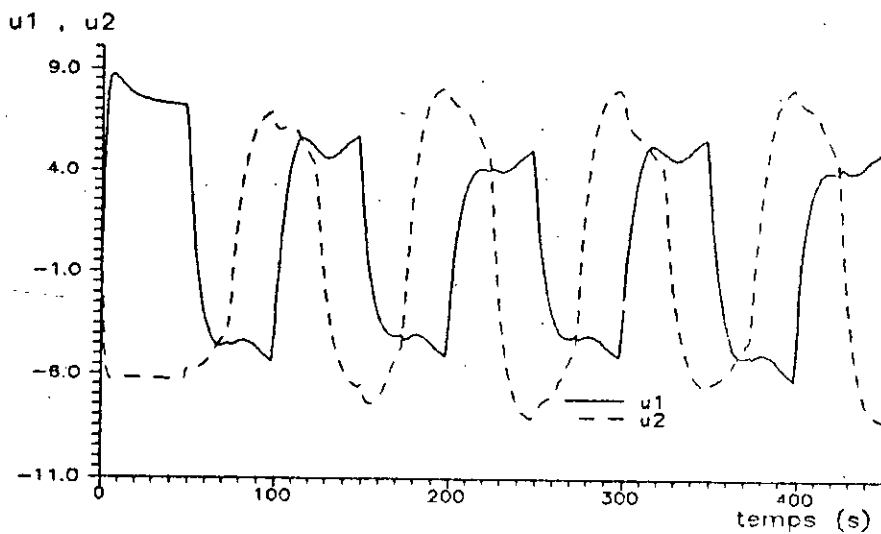
d) M R A C modifie et M C S

Fig 5-27 Erreur de poursuite

Variations rectangulaires des parametres



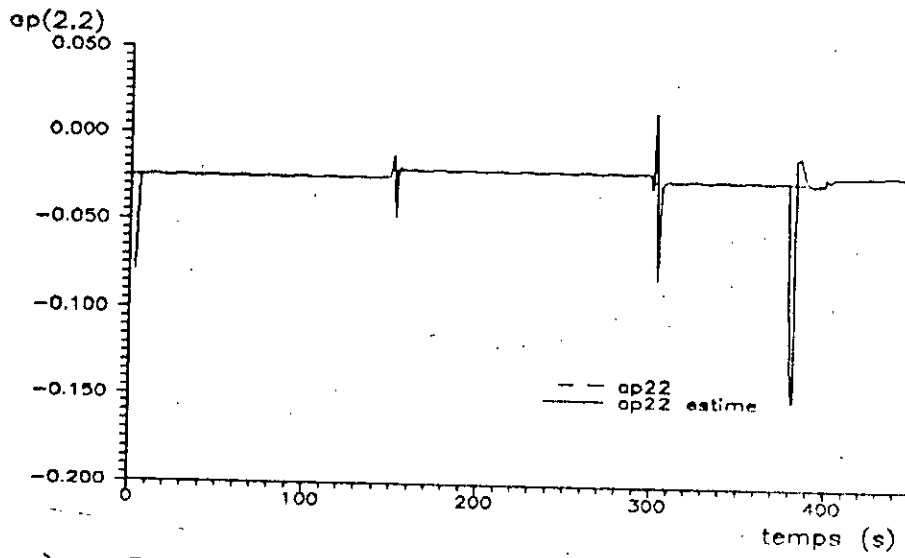
a) M R A C modifie



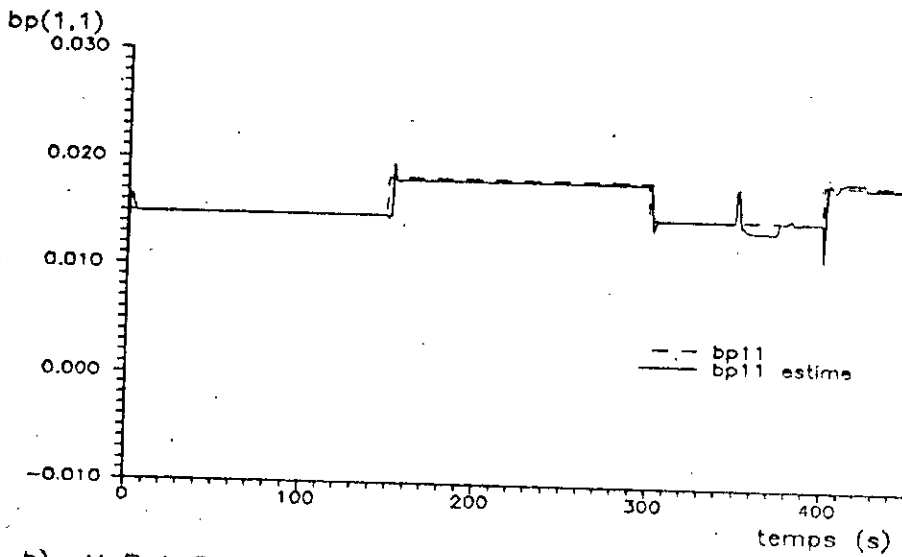
b) M C S

Fig 5-28 Commande

# Variations rectangulaires des parametres



a) M R A C modifie



b) M R A C modifie

Fig 5-29 Parametres estimes

La commande du MCS est montrée en figure 5-28-b.

Les valeurs des paramètres sont, dans l'ensemble, correctement estimées mais des pics très importants apparaissent sur les estimés quand des transitions des paramètres se produisent. Ceci peut être observé en figure 5-29.

CHAPITRE VICONCLUSION

Les tests du chapitre précédent ont montré que pour les trois algorithmes, la poursuite du modèle est satisfaisante malgré les variations des paramètres du procédé.

La loi de commande MRACM permet, en outre, une plus grande réduction de l'erreur de poursuite mais au prix de l'apparition de pics (qui peuvent être indésirables : détérioration des organes de commande, consommation d'énergie) sur la commande. L'algorithme MRACM apparaît plus sensible (présence des pics) aux transitions rapides des paramètres et du signal de référence que les algorithmes MRAC et MCS. Même si, entre deux transitions, la performance en poursuite, de l'algorithme MRACM redevient supérieure à celles des deux autres algorithmes. Les pics observés peuvent être attribués au fait que l'identificateur ne fournit pas instantanément des estimés corrects des paramètres. En fait, il lui faut quelques périodes d'échantillonnage pour les obtenir. Les algorithmes MRAC et MCS qui n'utilisent pas d'estimés en ligne des paramètres dans l'expression de leur commande ne présentent pas de tels pics.

Afin de tirer le meilleur parti de l'algorithme MRACM, nous avons voulu améliorer les résultats de l'identification en ligne. En effet, si l'identification ne donne pas des estimés assez proches des vraies valeurs des paramètres, il est beaucoup plus simple d'avoir directement recours à l'algorithme MRAC. Afin de permettre une meilleure identification nous avons donc utilisé une référence de type carré. L'intérêt de l'algorithme MRACM est alors aussi de fournir une connaissance du processus. Les informations

obtenues par l'identification pourraient éventuellement être utilisées pour surveiller l'évolution des paramètres du procédé.

D'autre part, s'il est possible d'interpréter les paramètres identifiés en termes d'usure ou de rendement, ceci pourrait être exploité dans le cadre d'une maintenance préventive. Il en résulterait une augmentation de la fiabilité du système qui pourrait encore justifier la complexité du MRACM par rapport au MRAC et au MCS.

Remarquons que notre choix de référence peut paraître restrictif (pas de référence constante). En fait, la plupart des algorithmes d'identification paramétrique nécessitent la satisfaction d'une condition de richesse sur l'entrée pour réaliser la convergence paramétrique. Signalons cependant l'existence de méthodes d'identification paramétrique qui permettent, selon leurs auteurs, de réduire considérablement la contrainte imposée à l'entrée de référence [23], [42].

Nous pouvons dire :

- L'algorithme MRACM ne devrait être utilisé que lorsqu'une erreur de poursuite particulièrement faible est indispensable ou lorsqu'il est important de connaître les paramètres du procédé commandé. Sa mise en oeuvre nécessite la vérification de contraintes. Par exemple, l'estimé de la matrice  $B(t)$  devra être inversible. Des précautions devront être prises pour satisfaire ces contraintes.

- L'algorithme MRAC demande pour sa mise en oeuvre une assez bonne connaissance des valeurs initiales des paramètres du procédé. Il est moins complexe que l'algorithme précédent mais moins performant en ce qui concerne la réduction de l'erreur de poursuite pour des procédés à paramètres variables.

En outre, il n'apporte aucune information sur le procédé

commandé. Il devrait être utilisé pour des procédés dont les paramètres sont très lentement variables.

- L'algorithme MCS n'apporte aucune information sur les paramètres du procédé mais possède, dans tous les cas, le meilleur rapport performance/simplicité de mise en oeuvre. L'erreur de poursuite peut devenir inférieure à celle de l'algorithme MRAC pour des systèmes à paramètres variables. L'algorithme MCS est le plus simple des trois. Il ne requiert pas d'identification en ligne (en temps réel). Une connaissance préalable des valeurs des paramètres du procédé n'est pas nécessaire.

PARTIE C

COMMANDE ADAPTATIVE

DE SYSTEMES

DANS UN ENVIRONNEMENT

STOCHASTIQUE



CHAPITRE IGENERALITES

Nous abordons, ici, la commande de procédés dans un environnement stochastique. Pour ces systèmes, le vecteur d'état est perturbé par des bruits aléatoires.

L'idée de base de la CA des systèmes stochastiques est de combiner une procédure d'estimation paramétrique en ligne et une technique de synthèse d'un système de commande pour produire une loi de commande adaptative.

Si l'on considère les paramètres du système comme des états constants inconnus, on peut poser le problème de commande comme un problème de commande optimale stochastique non linéaire [3], [25]. La commande est synthétisée en minimisant une fonction de coût qui est une fonction scalaire des états et des signaux de commande. Ce problème est, en général, extrêmement difficile à résoudre. Si l'on suppose qu'une solution existe, on peut obtenir, en faisant appel à la programmation dynamique, une équation fonctionnelle, dite équation de Bellman, pour le coût optimal. Malheureusement, cette dernière ne peut être résolue numériquement que dans des cas très simples.

Les calculs numériques nécessaires limitent l'applicabilité du contrôleur optimal. Cependant, sa structure duale est très importante. En effet, le contrôleur optimal prend en compte l'incertitude sur les paramètres estimés et tente de la réduire. Sa commande a donc un rôle dual de régulation et d'apprentissage. Elle permet l'obtention d'information sur les paramètres dynamiques du système.

Le contrôleur optimal est difficilement exploitable voire irréalisable dans la pratique. Il s'est donc avéré

nécessaire de développer des stratégies de commande sous-optimales réalisables [ 3], [10].

Une approche de la CA stochastique consiste à ignorer les incertitudes sur les paramètres estimés et à construire une loi de commande qui utilise les estimés comme s'ils étaient les vraies valeurs des paramètres du système. Cette démarche caractérise les contrôleurs dits à équivalence certaine [ 7], [12].

L'étude du comportement asymptotique des schémas de CA était initialement faite au moyen de simulations intensives. Par la suite, les développements de certains résultats théoriques ont permis d'analyser ces algorithmes.

Ainsi, un schéma de CA à MR peut être étudié par la méthode de l'ODE ("Ordinary Differential Equation") [ 9][20]. C'est une approche générale pour l'analyse du comportement asymptotique d'algorithmes récursifs stochastiques. Une équation différentielle déterministe est associée à l'algorithme. Le comportement asymptotique de ce dernier est déduit de l'analyse de la stabilité de l'équation différentielle déterministe.

L'utilisation de la méthode suppose que les différentes variables intervenant dans le schéma sont bornées.

D'autres approches permettent également l'étude de la convergence des algorithmes de CA [11], [34], [26].

Pour cette partie de notre travail, nous avons, dans un premier temps, utilisé l'algorithme MCS dans un environnement stochastique. Mais, les résultats des simulations nous ont conduit à dériver un nouvel algorithme de commande. Nous avons développé cette approche pour des systèmes SISO. Le cas des systèmes MIMO sera envisagé dans des études ultérieures.

## CHAPITRE II.

TESTS ET SIMULATION DE L'ALGORITHME MCS

Le processus considéré est un système SISO d'ordre 1 représenté dans l'espace d'état par le système d'équations

$$\begin{cases} \dot{x}(t) = ax(t) + bu(t) + d(t) & (2-1) \\ y(t) = x(t) + v(t) & (2-2) \end{cases}$$

$d$  et  $v$  sont des bruits blancs centrés.

$$b = 3,833$$

Le paramètre  $a$  est modélisé par [24] :

$$\dot{a} = -\frac{1}{T}a + \frac{1}{T}w \quad (2-3)$$

où  $w$  est un bruit blanc centré et  $T$  est grand ( $T = 10^4$  s dans les simulations). Le paramètre  $a$  varie assez lentement.

Le MR est caractérisé par les équations :

$$\begin{cases} \dot{x}_m = -10x_m + 3,833r & (2-4) \\ y_m = x_m & (2-5) \end{cases}$$

Puisque l'état  $x$  n'est pas directement mesurable, la commande est calculée à l'aide de  $y$  de la façon suivante :

$$u(t) = \delta K_P(t)y + \delta K_R(t)r \quad (2-6)$$

où

$$\delta K_P(t) = \alpha \int_0^t C_e(y_m - y) y dt + \beta C_e(y_m - y) y \quad (2-7)$$

$$\delta K_R(t) = \alpha \int_0^t C_e(y_m - y) r dt + \beta C_e(y_m - y) r \quad (2-8)$$

L'erreur utilisée pour le calcul des gains adaptatifs est donc :

$$e = y_m - y = x_m - x - v \quad (2-9)$$

où  $v$  est un bruit blanc centré.

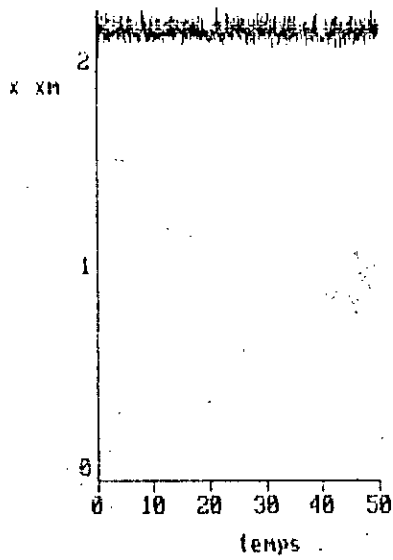
La figure 2-1 montre que lorsque le bruit de sortie,  $v$ , est de variance faible, la poursuite de l'état  $x_m$  est satisfaisante au début de l'adaptation. Cependant, plus la variance du bruit  $v$  augmente, plus le système devient rapidement instable : l'amplitude des variations de  $x$  autour de  $x_m$  ne cesse de croître. De même, l'amplitude des variations de  $u$  augmente et le paramètre  $\tilde{a}$  du système en boucle fermée s'écarte de plus en plus de  $a_m$ . Ceci est illustré par la figure 2-2.

Ces résultats ont été obtenus pour des bruits gaussiens. Comme le montre la figure 2-3, l'instabilité se produit également lorsque les bruits sont uniformément distribués.

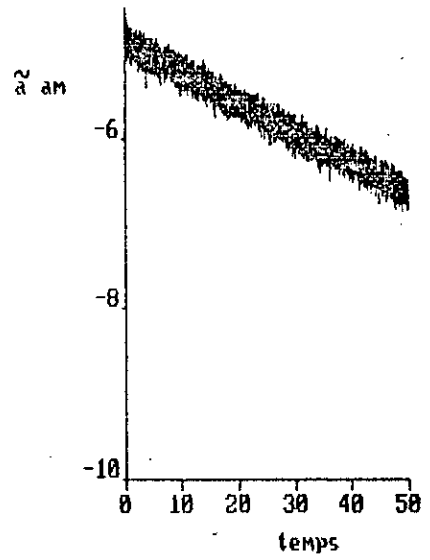
L'algorithme MCS est basé sur la théorie de l'hyperstabilité développée par Popov [29] dans un cadre déterministe. Cette dernière ne convient pas pour l'analyse des systèmes stochastiques.

En réalité, en présence d'un bruit de sortie, l'algorithme MCS peut devenir instable même dans le cas déterministe. Rohrs et ses collaborateurs [30] ont en effet montré que, pour des procédés possédant des dynamiques non modélisées, de nombreux algorithmes de CA peuvent devenir instables. Selon eux, l'instabilité est due à des opérateurs inhérents au mécanisme d'adaptation et dont le gain peut devenir infini.

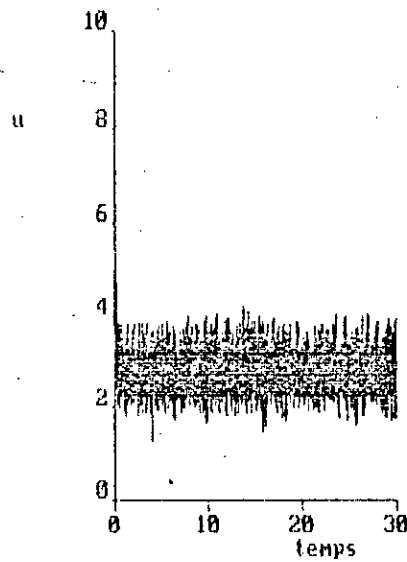
L'instabilité de l'algorithme MCS, pour des systèmes dans un environnement stochastique, nous a conduit à développer une nouvelle approche présentée dans le chapitre suivant.



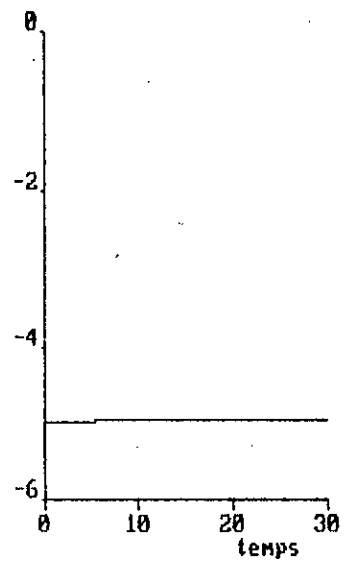
a) Poursuite de l'état  $x_m$   
(  $x_m = \text{constante}$  )



b) Poursuite du paramètre  $a_m$   
(  $a_m = -10$  )



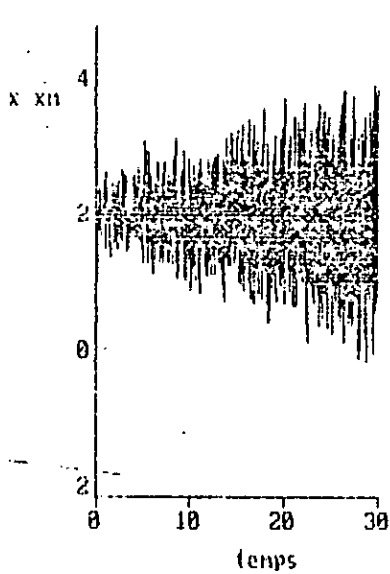
c) Commande



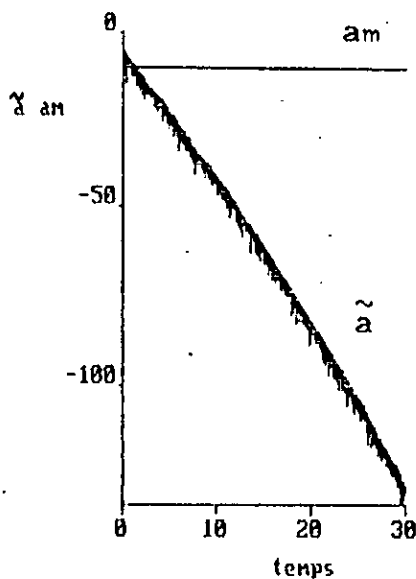
d) Paramètre a

Fig 2-1 Algorithme MCS : bruits gaussiens

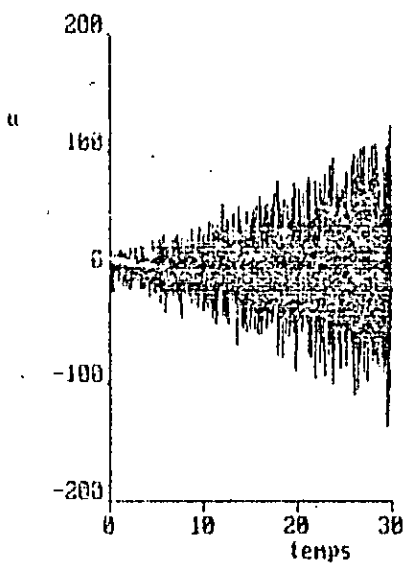
$$\text{Var}(d) = 1 \quad \text{Var}(v) = 0.01 \quad \text{Var}(w) = 0.25$$



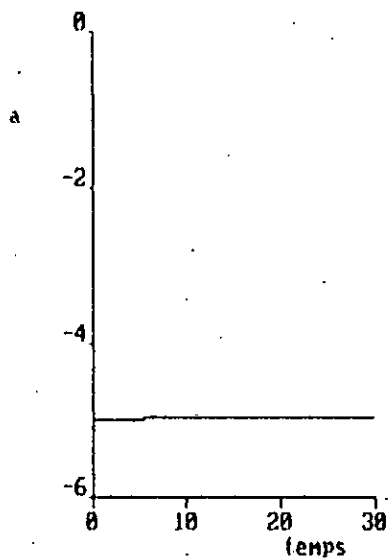
a) Poursuite de l'état  $x_m$   
( $x_m = \text{constante}$ )



b) Poursuite du paramètre  $a_m$



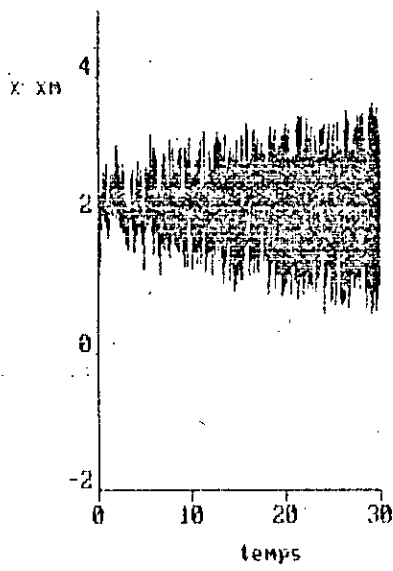
c) Commande



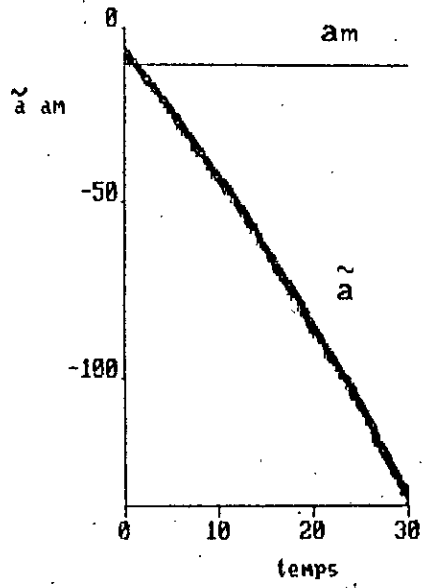
d) Paramètre a

Fig 2-2 Algorithme MCS : bruits gaussiens

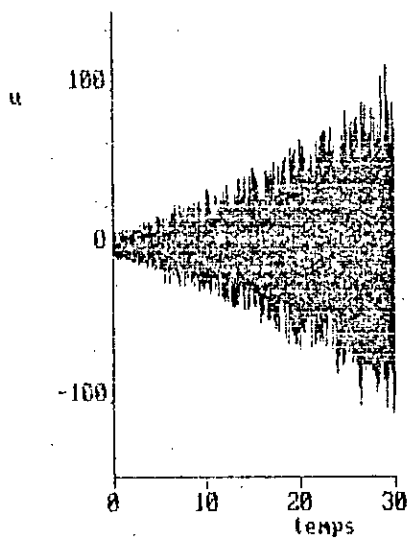
$$\text{Var}(d) = 1 \quad \text{Var}(v) = 1 \quad \text{Var}(w) = 0.25$$



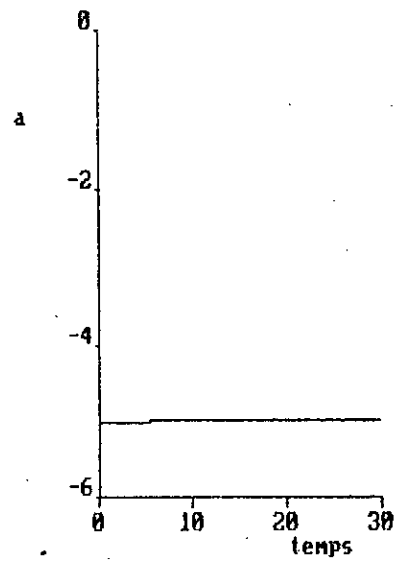
a) Poursuite de l'état  $x_m$   
( $x_m = \text{constante}$ )



b) Poursuite du paramètre  $a_m$



c) Commande



d) Paramètre a

Fig 2-3 Algorithme MCS : bruits uniformément distribués

$$\text{Var}(d) = 1 \quad \text{Var}(v) = 1 \quad \text{Var}(w) = 0.25$$



## CHAPITRE III

DERIVATION DE L'ALGORITHME STOCHASTIQUE

Cet algorithme fait appel au filtrage de Kalman étendu [24] pour estimer les états et les paramètres du procédé à commander. Puis, une loi de commande à contre-réaction est calculée en utilisant les estimés comme s'ils étaient les vraies valeurs des états et des paramètres du système (principe de l'équivalence certaine).

Nous développons tout d'abord l'algorithme stochastique pour un procédé SISO d'ordre 1. Puis nous étendons cette approche au cas de systèmes d'ordre  $n$ .

3.1 Le filtre de Kalman étendu

Le filtre de Kalman étendu [24] est un filtre approximatif pour des systèmes non linéaires (cf. Annexe F). son utilisation pour l'estimation conjointe des paramètres et états d'un système linéaire à paramètres inconnus est largement répandue. En fait, il est naturel d'inclure les paramètres inconnus dans le vecteur d'état et d'exploiter ensuite les résultats du filtrage de Kalman linéaire [24] (cf. Annexe F). L'algorithme obtenu n'est pas très complexe. Cependant, des précautions doivent être prises avant de l'utiliser. Ainsi, une étude approfondie en simulation s'avère nécessaire car les estimés peuvent être biaisés ou même diverger.

Ljung [21] a étudié les causes du biais et de la divergence. Ces problèmes proviennent d'une combinaison de facteurs dont :

- une spécification incorrecte des covariances des bruits,

• le fait que le gain de Kalman dépende des estimés des paramètres.

Plusieurs modifications de l'algorithme ont été proposées pour faire face à ces problèmes [21]. Goodwin [11] a suggéré de combiner un algorithme d'estimation paramétrique (à erreur de prédiction) pour estimer les paramètres du système et un filtre de Kalman standard pour estimer les états du modèle du procédé. Dans le modèle, les paramètres inconnus sont remplacés par leurs estimés. CHUI et CHEN [ 8] ont proposé d'utiliser en parallèle un filtre de Kalman standard et un filtre de Kalman étendu pour l'identification, en temps réel, des paramètres inconnus de systèmes stochastiques linéaires et variant dans le temps.

### 3.2 Cas de systèmes SISO d'ordre 1

Le procédé à commander est décrit dans l'espace d'état par le système d'équations

$$\left. \begin{aligned} \dot{x}(t) &= ax(t) + bu(t) + d(t) \\ y(t) &= x(t) + v(t) \end{aligned} \right\} \quad (3-1)$$

où  $d$ ,  $v$  sont des bruits blancs centrés tels que :

$$E(d(t)d(\tau)) = \bar{d}\delta(t-\tau) \quad (3-2)$$

$$E(v(t)v(\tau)) = \bar{v}\delta(t-\tau) \quad (3-3)$$

$b$  est supposé connu,

$a = a(t)$  est un paramètre aléatoire que nous décrirons plus loin,

$x$  est l'état du système et  $y$  est l'état mesuré (bruité).

Le MR est représenté par :

$$\left. \begin{aligned} \dot{x}_m(t) &= a_m x_m(t) + b_m r(t) \\ y_m(t) &= x_m(t) \end{aligned} \right\} \quad (3-4)$$

avec

$$b_m = b$$

Supposons qu'il soit possible de calculer la commande :

$$u = -K(t)x + r \quad (3-5)$$

Le système en boucle fermée est alors :

$$\dot{x} = \tilde{a}x + br + d \quad (3-6)$$

$$\tilde{a} = \tilde{a}(a, K) = a - bK(t) \quad (3-7)$$

L'objectif du mécanisme d'adaptation est d'assurer

- la meilleure approximation du paramètre  $a_m$  du MR par le paramètre  $\tilde{a}$  du système en boucle fermée :

$$\begin{aligned} \tilde{a}(K^*, a) &= a_m \Rightarrow \\ K^* &= \frac{1}{b} (a - a_m) \end{aligned} \quad (3-8)$$

ou ce qui revient au même,

- la meilleure approximation du paramètre inconnu,  $a(t)$ , du procédé par la quantité  $bK(t) + a_m$ .

Le paramètre  $a$  étant inconnu et l'état  $x$  non mesurable directement, on prendra :

$$u(t) = -K(t)\hat{x} + r \quad (3-9)$$

$$K(t) = \frac{1}{b} (\hat{a} - a_m) \quad (3-10)$$

où  $\hat{a}$  et  $\hat{x}$  sont estimés, à partir des observations  $y$ , de  $x_m$  et des caractéristiques probabilistes de  $a(t)$ ,  $d$  et  $v$ , par un filtre de Kalman [24].

Il s'agit donc, pour permettre la synthèse du mécanisme

d'adaptation, d'estimer le vecteur d'état élargi  $\begin{bmatrix} x(t) \\ a(t) \end{bmatrix}$ .

C'est un problème non linéaire car le produit  $ax$  intervient dans le système d'équations (3-1) qui donne  $x$ . Pour le résoudre, il nous faudra linéariser par rapport aux estimés  $\hat{x}$  et  $\hat{a}$  des variables d'état (en supposant que les erreurs d'estimation sont faibles). Le filtre utilisé est donc un filtre de Kalman étendu (cf. Annexe F) [24].

Le paramètre  $a(t)$  est modélisé par :

$$\dot{a}(t) = -\frac{1}{T} a(t) + \frac{1}{T} w(t) \quad (3-11)$$

où  $w$  est un bruit blanc centré tel que

$$E\{w(t)w(\tau)\} = \bar{w}\delta(t-\tau) \quad (3-12)$$

$a(t)$  est un processus markovien formé, à partir du bruit blanc  $w$  à l'aide d'un filtre donné par la fonction de transfert :  $\frac{1}{Ts + 1}$  (Si  $T$  tend vers l'infini, on obtient le cas particulier où  $a$  est constant).

Nous avons donc :

$$\dot{x} = ax - bK\hat{x} + br + d \quad (3-13)$$

$$\dot{a} = -\frac{1}{T} a + \frac{1}{T} w \quad (3-14)$$

$$\Delta y = x - x_m + v = y - y_m \quad (3-15)$$

Soit  $z = \begin{bmatrix} x \\ a \end{bmatrix}$ , en tenant compte de l'égalité (3-10), nous pouvons écrire :

$$\left. \begin{aligned} \dot{z} &= f(z) + \eta \\ \Delta y &= h(z) + v \end{aligned} \right\} \quad (3-16)$$

où

$$f(z) = \begin{bmatrix} ax - \hat{a}\hat{x} + a_m\hat{x} + br \\ -\frac{1}{T} a \end{bmatrix} \quad \eta = G\zeta$$

$$G = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & \frac{1}{T} \end{pmatrix} \quad \zeta = \begin{pmatrix} d \\ v \\ w \end{pmatrix}$$

$$h(z) = x - x_m$$

Pour

$$\hat{z} = \begin{pmatrix} \hat{x} \\ \hat{a} \end{pmatrix}$$

nous avons

$$f(\hat{z}) = \begin{pmatrix} a_m \hat{x} + br \\ -\frac{1}{T} \hat{a} \end{pmatrix} \quad (3-17)$$

$$h(\hat{z}) = \hat{x} - x_m \quad (3-18)$$

et

$$A(\hat{z}) = \frac{\partial f(z)}{\partial z} \Big|_{z = \hat{z}} = \begin{pmatrix} \hat{a} & \hat{x} \\ 0 & -\frac{1}{T} \end{pmatrix} \quad (3-19)$$

$$C(\hat{z}) = \frac{\partial h(z)}{\partial z} \Big|_{z = \hat{z}} = \begin{pmatrix} 1 & 0 \end{pmatrix} \quad (3-20)$$

A partir des égalités (3-2), (3-3) et (3-12), nous pouvons écrire :

$$E \langle \zeta(t) \zeta^T(\tau) \rangle = \bar{Q} \delta(t-\tau) \quad (3-21)$$

où

$$\bar{Q} = \begin{pmatrix} \bar{d} & 0 & 0 \\ 0 & \bar{v} & 0 \\ 0 & 0 & \bar{w} \end{pmatrix}$$

$$E \left\langle \begin{pmatrix} \zeta(t) \\ v(t) \end{pmatrix} \begin{pmatrix} \zeta(\tau) \\ v(\tau) \end{pmatrix}^T \right\rangle = \begin{pmatrix} \bar{Q} & \bar{S} \\ \bar{S}^T & R \end{pmatrix} \delta(t-\tau) \quad (3-22)$$

où  $\bar{S} = [0 \quad \bar{v} \quad 0]^T$  et  $R = \bar{v}$

$$E \langle \eta(t) \eta^T(\tau) \rangle = Q \delta(t - \tau) \quad (3-23)$$

où

$$Q = G \bar{Q} G^T = \begin{pmatrix} \bar{d} & 0 \\ 0 & \frac{\bar{w}}{T^2} \end{pmatrix}$$

$$E \left\langle \begin{pmatrix} \eta(t) \\ v(t) \end{pmatrix} \begin{pmatrix} \eta(\tau) \\ v(\tau) \end{pmatrix}^T \right\rangle = \begin{pmatrix} Q & S \\ S^T & R \end{pmatrix} \delta(t - \tau) \quad (3-24)$$

où

$$S = G \bar{S} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Le filtre de Kalman étendu qui joue le rôle de mécanisme d'adaptation, a la forme :

$$\dot{\hat{z}} = f(\hat{z}) + L (\Delta y - h(\hat{z})) \quad (3-25)$$

soit :

$$\dot{\hat{x}} = a_m \hat{x} + br + l_1 (\Delta y - (\hat{x} - x_m)) \quad (3-26)$$

$$\dot{\hat{a}} = -\frac{1}{T} \hat{a} + l_2 (\Delta y - (\hat{x} - x_m)) \quad (3-27)$$

où

$$L = \begin{pmatrix} l_1 \\ l_2 \end{pmatrix} = PC^T R^{-1} \quad (3-28)$$

et  $P = P(t)$  est solution de l'équation différentielle de

Riccati :

$$\dot{P} = AP + PA^T - PC^T R^{-1} CP + Q \quad (3-29)$$

### 3.3 Cas de systèmes SISO d'ordre n

Le procédé à commander est décrit par le système d'équations

$$\left. \begin{aligned} \dot{x}(t) &= A_p(t)x(t) + B_p u(t) + d(t) \\ y(t) &= C_p x(t) + v(t) \end{aligned} \right\} \quad (3-30)$$

$x$  est un vecteur d'état de dimension  $n$ .

$d$  est un vecteur d'ordre  $n$ ,  $v$  est un scalaire,

$d$  et  $v$  sont des bruits blancs centrés tels que :

$$E\{d(t)d^T(\tau)\} = \bar{D} \delta(t-\tau) \quad (3-31)$$

$$E\{v(t)v(\tau)\} = \bar{v} \delta(t-\tau) \quad (3-32)$$

$$A_p = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & 0 & 1 \\ a_{p_1} & a_{p_2} & \dots & a_{p_n} & \dots \end{bmatrix}$$

Les paramètres  $a_{p_i}$  ( $i=1, \dots, n$ ) sont aléatoires.

$$\begin{aligned} B_p &= [0 \dots 0 \ b]^T \\ C_p &= [1 \ 0 \dots 0] \end{aligned}$$

Le MR est donné par:

$$\left. \begin{aligned} \dot{x}_m &= A_m x_m + B_m r \\ y_m &= C_m x_m \end{aligned} \right\} \quad (3-33)$$

$x_m$  est un vecteur d'état de dimension  $n$ .

$$A_m = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & 0 & 1 \\ a_{m_1} & \dots & \dots & \dots & a_{m_n} \end{pmatrix} \quad \begin{aligned} B_m &= [0 \dots 0 \ b_m] \\ b_m &= b_p \\ C_m &= [1 \ 0 \dots 0] \end{aligned}$$

Comme au paragraphe précédent, nous supposons, dans un premier temps, qu'il est possible de calculer la commande

$$u = -K(t)x + r \quad (3-34)$$

où  $K(t) = [K_1(t) \dots K_n(t)]$

Le système en boucle fermée devient :

$$\left. \begin{aligned} \dot{x} &= \tilde{A}x + B_p r + d \\ \tilde{A} &= \tilde{A}(A_p, K) = A_p - B_p K \end{aligned} \right\} \quad (3-35)$$

et le mécanisme d'adaptation a pour objectif d'assurer la meilleure approximation de la matrice  $A_m$  du MR par la matrice  $\tilde{A}$  du système en boucle fermée :

$$\begin{aligned} \tilde{A}(A_p, K^*) &= A_m \quad \rightarrow \quad B_p K^* = A_p - A_m \\ \rightarrow K_i^* &= \frac{1}{b_p} (a_{p_i} - a_{m_i}) \quad i = 1, \dots, n \end{aligned}$$

Les paramètres  $a_{p_i}$  étant inconnus et l'état  $x$  non mesurable directement, nous prendrons :

$$u(t) = -\hat{K}x + r \quad (3-36)$$

et  $K_i(t) = \frac{1}{b_p} (\hat{a}_{p_i} - a_{m_i}) \quad i = 1, \dots, n \quad (3-37)$

Les paramètres  $\hat{a}_{p_i}$  et  $\hat{x}$  sont estimés au moyen d'un filtre de Kalman étendu.

En utilisant les équations (3-30) et (3-36) nous avons



$$\dot{x} = A_p x - B_p K \hat{x} + B_p r + d \quad (3-38)$$

Nous définissons

$$\Delta y = C_p x - C_m x_m + v = y - y_m \quad (3-39)$$

et

$$\dot{a} = -\Gamma^{-1} a + \Gamma^{-1} w \quad (3-40)$$

où

$$a = \begin{pmatrix} a_{p_1} & \dots & a_{p_n} \end{pmatrix}^T$$

$$\Gamma^{-1} = \begin{pmatrix} \frac{1}{\Gamma_1} & & 0 \\ & \dots & \\ 0 & & \frac{1}{\Gamma_n} \end{pmatrix}$$

w, vecteur de dimension n, est un bruit blanc tel que :

$$E\{w(t)w^T(\tau)\} = \bar{w} \delta(t-\tau) \quad (3-41)$$

Soit le vecteur d'état élargi  $z = \begin{pmatrix} x \\ a \end{pmatrix}$ , en tenant compte de

$$B_p K = \hat{A}_p - A_m$$

nous pouvons écrire :

$$\left. \begin{aligned} \dot{z} &= f(z) + \eta \\ \Delta y &= h(z) + v \end{aligned} \right\} \quad (3-42)$$

où

$$f(z) = \begin{pmatrix} A_p x - \hat{A}_p \hat{x} + A_m \hat{x} + B_p r \\ -\Gamma^{-1} a \end{pmatrix} \quad \eta = G\zeta$$

$$G = \begin{pmatrix} I & 0 & 0 \\ 0 & 0 & \Gamma^{-1} \end{pmatrix} \quad \zeta = \begin{pmatrix} d \\ v \\ w \end{pmatrix}$$

$$h(z) = C_p x - C_m x_m$$

Pour

$$z = \hat{z} = \begin{pmatrix} \hat{x} \\ \hat{a} \end{pmatrix}$$

nous avons

$$f(\hat{z}) = \begin{pmatrix} A_m \hat{x} + B_p r \\ -T^{-1} \hat{a} \end{pmatrix} \quad (3-43)$$

$$h(\hat{z}) = C_p \hat{x} - C_m x_m \quad (3-44)$$

$$A(\hat{z}) = \frac{\partial f(\hat{z})}{\partial z} \Big|_{z = \hat{z}} = \begin{pmatrix} \hat{A}_p & \begin{matrix} 0_{n-1, n} \\ \hat{x}^T \end{matrix} \\ 0 & -T^{-1} \end{pmatrix} \quad (3-45)$$

$$C(\hat{z}) = \frac{\delta h(\hat{z})}{\delta z} \Big|_{z = \hat{z}} = \begin{pmatrix} 1 & 0_{1 \times (2n-1)} \end{pmatrix} \quad (3-46)$$

A partir des égalités (3-31), (3-32) et (3-41), nous pouvons écrire :

$$E \langle \zeta(t) \zeta(\tau)^T \rangle = \begin{pmatrix} \bar{D} & 0 & 0 \\ 0 & \bar{v} & 0 \\ 0 & 0 & \bar{W} \end{pmatrix} \delta(t-\tau) = \bar{Q} \delta(t-\tau) \quad (3-47)$$

$$E \left\langle \begin{pmatrix} \zeta(t) \\ v(t) \end{pmatrix} \begin{pmatrix} \zeta(\tau) \\ v(\tau) \end{pmatrix}^T \right\rangle = \begin{pmatrix} \bar{Q} & \bar{S} \\ \bar{S}^T & R \end{pmatrix} \delta(t-\tau) \quad (3-48)$$

$$\text{où } \bar{S} = \begin{pmatrix} 0 \\ \bar{v} \\ 0 \end{pmatrix} \quad R = \bar{v}$$

$$E \langle \eta(t) \eta(\tau)^T \rangle = Q \delta(t-\tau) \quad (3-49)$$

où

$$Q = G \bar{Q} G^T = \begin{pmatrix} \bar{D} & 0 \\ 0 & T^{-1} \bar{W} T^{-1} \end{pmatrix}$$

$$E \left\langle \begin{pmatrix} \eta(t) \\ v(t) \end{pmatrix} \begin{pmatrix} \eta(\tau) \\ v(\tau) \end{pmatrix}^T \right\rangle = \begin{pmatrix} Q & S \\ S^T & R \end{pmatrix} \delta(t-\tau) \quad (3-50)$$

Le filtre qui joue le rôle de mécanisme d'adaptation a la forme :

$$\dot{\hat{z}} = f(\hat{z}) + L (\Delta y - h(\hat{z})) \quad (3-51)$$

soit

$$\dot{\hat{x}} = A_m \hat{x} + B_p r + L_1 (\Delta y - (C_p \hat{x} - C_m x_m)) \quad (3-52)$$

$$\dot{\hat{a}} = -T^{-1} \hat{a} + L_2 (\Delta y - h(\hat{z})) \quad (3-53)$$

$L_1$  et  $L_2$  sont des vecteurs d'ordre  $n$ .

$$L = \begin{pmatrix} L_1 \\ L_2 \end{pmatrix} = PC^T R^{-1} \quad (3-54)$$

$L$  est le gain du filtre de Kalman et  $P$  est la matrice de covariance de l'erreur d'estimation.  $P$  est solution de l'équation différentielle de Riccati :

$$\dot{P} = AP + PA^T - PC^T R^{-1} CP + Q \quad (3-55)$$

Les équations (3-54) et (3-51) montrent que :

• si  $P(t)$  est grand, alors la différence entre la mesure  $y$  et l'estimé  $C_p \hat{x}$  est fortement pondérée (Dans l'équation (3-51), la quantité  $\Delta y - h(\hat{z})$  est en fait égale à  $y - C_p \hat{x}$ ). Autrement dit

si l'estimé  $\hat{z}$  est très incertain, le filtre accorde une grande importance à l'information apportée par les mesures.

• si  $R(t)$  est faible, c'est-à-dire si les mesures sont très précises, alors l'information fournie par les mesures est fortement pondérée.

Dans l'équation (3-55),

• les deux premiers termes dépendent des estimés par l'intermédiaire de la matrice  $A$  donnée dans l'équation (3-45).

• le troisième terme reflète l'effet de diminution de la covariance dû à l'incorporation de l'information apportée par les mesures.

• le quatrième terme représente l'effet d'augmentation de la covariance dû aux bruits  $d(t)$  et  $w(t)$ .

Dans l'algorithme de commande proposé, le filtre de Kalman étendu joue le rôle de mécanisme d'adaptation. Il permet d'estimer le vecteur d'état élargi  $z = [x^T \ a^T]^T$ , dans un environnement stochastique. A partir de l'estimé,  $\hat{a}$ , du vecteur de paramètres  $a$ , on forme le gain  $K(t)$  qui intervient dans l'expression (3-36) de la commande.

CHAPITRE IV

TESTS ET SIMULATION

DE L'ALGORITHME STOCHASTIQUE

4.1 Application à un système SISO d'ordre 1

Le processus considéré est représenté dans l'espace d'état par le système d'équations :

$$\left. \begin{aligned} \dot{x}(t) &= ax(t) + bu(t) + d(t) \\ y &= x + v \end{aligned} \right\} \quad (4-1)$$

$d$  et  $v$  sont des bruits blancs centrés.

$$b = 3,833$$

Le paramètre  $a$  est modélisé par :

$$\dot{a} = -\frac{1}{T}a + \frac{1}{T}w \quad (4-2)$$

où  $w$  est un bruit blanc centré et  $T$  est grand ( $T = 10^4$  s dans les simulations). Le paramètre  $a$  varie assez lentement.

Le MR est caractérisé par les équations :

$$\left. \begin{aligned} \dot{x}_m &= -10x_m + 3,833r \\ y_m &= x_m \end{aligned} \right\} \quad (4-3)$$

La commande est

$$u_p = -K\hat{x} + r \quad (4-4)$$

avec 
$$K = \frac{1}{3,833}(\hat{a}_p + 10) \quad (4-5)$$

$\hat{a}_p$  et  $\hat{x}$  sont estimés au moyen du filtre de Kalman étendu donné dans le chapitre précédent.

#### 4.1.1 Bruits gaussiens

Les bruits  $d$ ,  $v$ ,  $w$  sont blancs gaussiens centrés. Les résultats de simulation sont présentés dans les figures 4-1 et 4-2. On obtient une bonne poursuite de l'état du MR et une bonne estimation du paramètre  $a$ , pour diverses valeurs des variances des bruits. La commande est pratiquement constante.

#### 4.1.2 Bruits uniformément distribués

Lorsque les bruits  $d$ ,  $v$ ,  $w$  sont uniformément distribués, la figure 4-3, montre que l'on obtient également, une bonne poursuite, une bonne estimation du paramètre et une commande qui varie très peu.

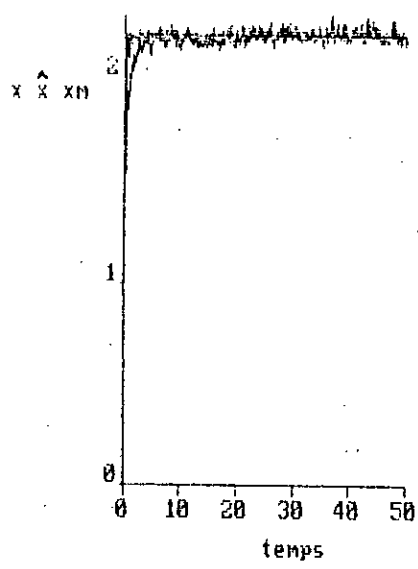
#### 4.1.3 Bruits gaussiens corrélés

Pour ce test, nous avons supposé que les bruits intervenant dans la représentation d'état du procédé sont corrélés :

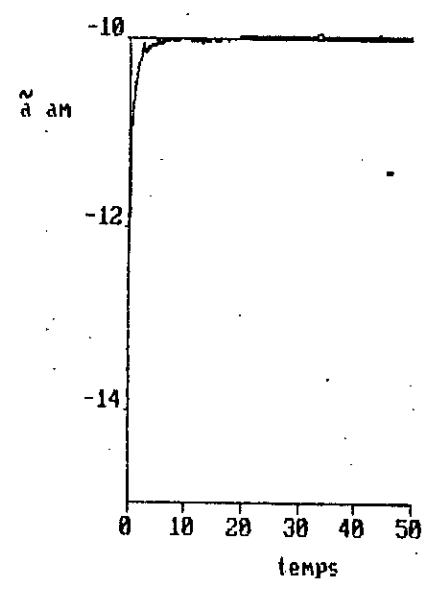
$$\left. \begin{aligned} \dot{x} &= (a - bk)x + br + d_c \\ y &= x + v \\ d_c &= d + \alpha v \end{aligned} \right\} \quad (4-6)$$

$\alpha$  est une constante.

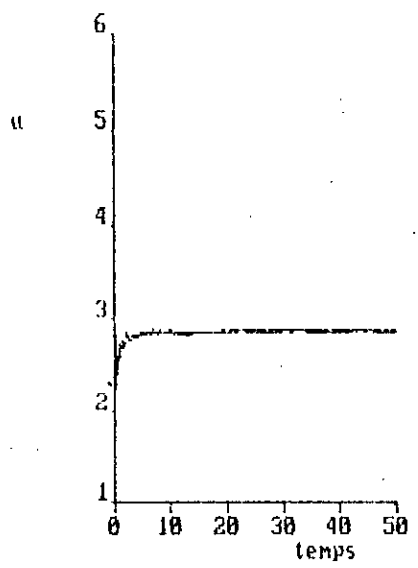
Nous avons donc :



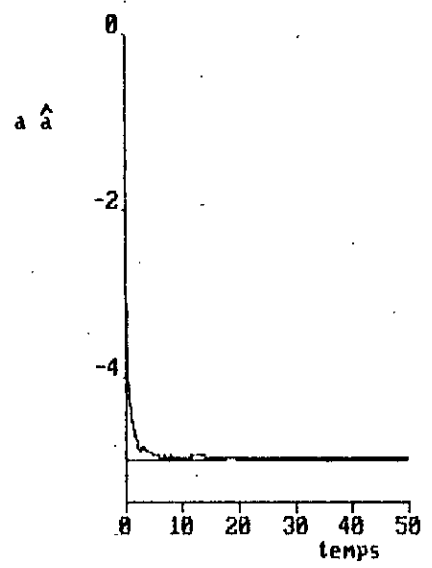
a) Poursuite de l'état  $x(t)$



b) Poursuite du paramètre  $a(t)$



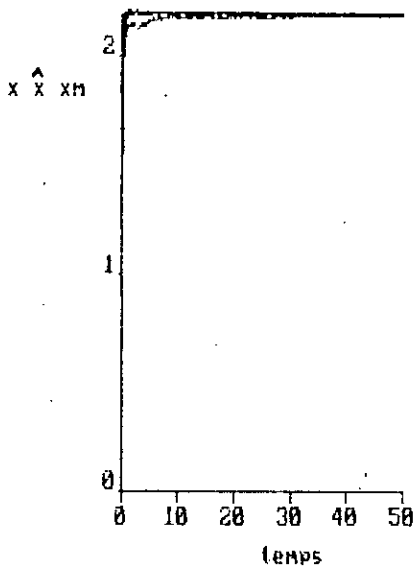
c) Commande



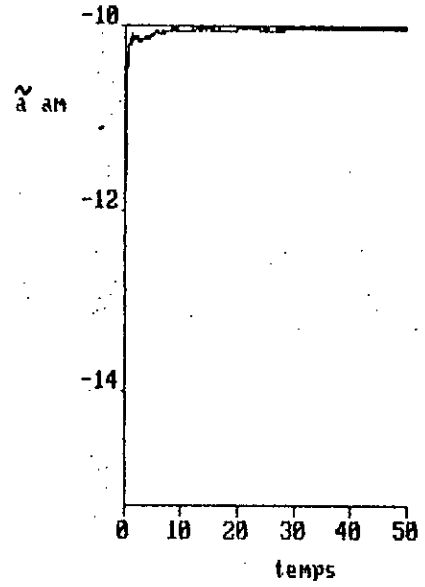
d) Estime du paramètre  $a$

Fig 4-1 Algorithme stochastique avec bruits gaussiens

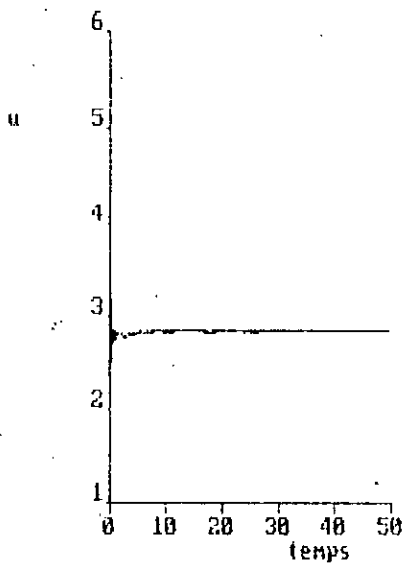
$$\text{Var}(d) = 1 \quad \text{Var}(v) = 1 \quad \text{Var}(w) = 0.25$$



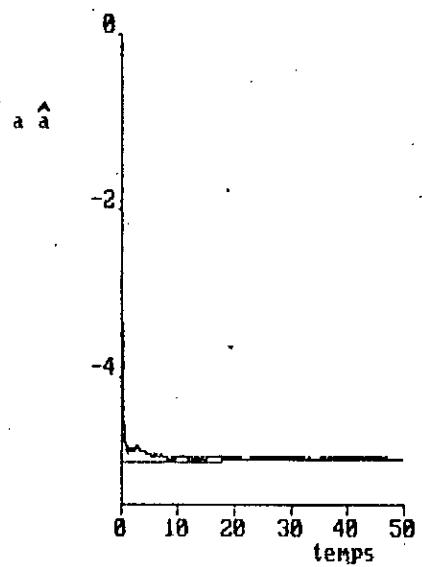
a) Poursuite de l'état  $x_m$



b) Poursuite du paramètre  $a_m$



c) Commande

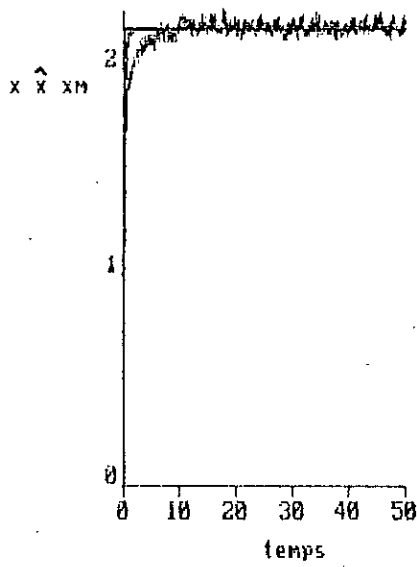


d) Estime du paramètre  $a$

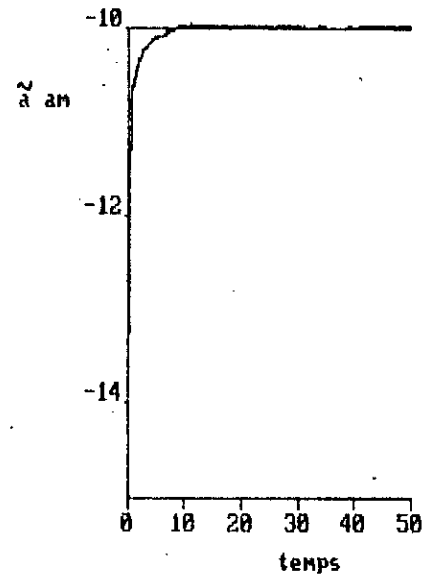
Fig 4-2 Algorithme stochastique avec bruits gaussiens

$$\text{Var}(d) = 0.01 \quad \text{Var}(v) = 0.01 \quad \text{Var}(w) = 0.25$$

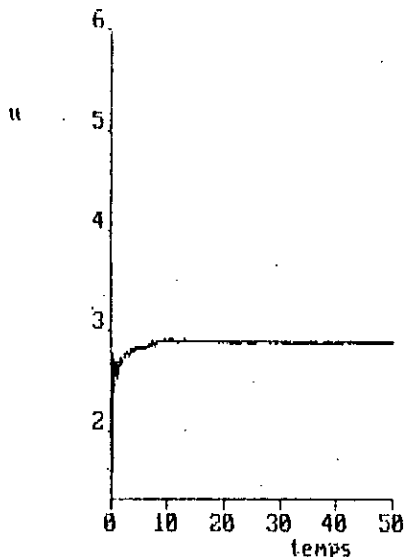




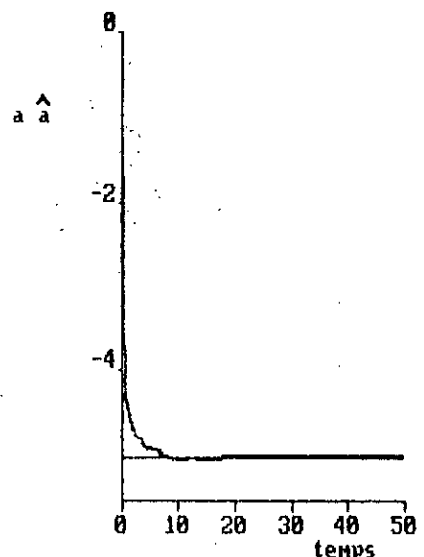
a) Poursuite de l'état  $x_m$



b) Poursuite du paramètre  $a_m$



c) Commande



d) Estimé du paramètre  $a$

Fig 4-3 Algorithme stochastique avec bruits

uniformement distribués

$$\text{Var}(d) = 1 \quad \text{Var}(v) = 1 \quad \text{Var}(w) = 0.25$$

$$\begin{aligned}\dot{z} &= f(z) + \eta \\ \eta &= G \zeta\end{aligned}\tag{4-7}$$

$$\zeta = \begin{bmatrix} d \\ v \\ w \end{bmatrix} \quad G = \begin{bmatrix} 1 & \alpha & 0 \\ 0 & 0 & \frac{1}{T} \end{bmatrix}$$

et en procédant comme au paragraphe 3.2 du chapitre précédent, nous obtenons :

$$E(\eta(t)\eta(\tau)^T) = Q\delta(t-\tau) \quad Q = \begin{bmatrix} \bar{d} + \alpha^2 \bar{v} & 0 \\ 0 & \frac{\bar{w}}{T^2} \end{bmatrix}$$

$$E \left( \begin{bmatrix} \eta(t) \\ v(t) \end{bmatrix} \begin{bmatrix} \eta(\tau) \\ v(\tau) \end{bmatrix}^T \right) = \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \delta(t-\tau)$$

$$S = \begin{bmatrix} \alpha \bar{v} \\ 0 \end{bmatrix} \quad r = \bar{v}$$

Dans ce cas, le filtre est donné par :

$$\dot{\hat{z}} = f(\hat{z}) + L(\Delta y - h(\hat{z}))\tag{4-8}$$

$$L = (PC^T + S)R^{-1}\tag{4-9}$$

P est solution de l'équation de Riccati :

$$P = A^*P + PA^{*T} - PC^TR^{-1}CP + Q^*\tag{4-10}$$

$$A^* = A - SR^{-1}C\tag{4-11}$$

$$Q^* = Q - SR^{-1}S^T\tag{4-12}$$

A et C sont donnés par les équations (3-19) et (3-20) du

chapitre précédent.

Comme on peut le voir en figure 4-4, on obtient une bonne poursuite des états du MR et une bonne estimation du paramètre  $a$ .

#### 4.1.4 Paramètre variable

Nous modélisons le paramètre  $a$  de la façon suivante

$$\dot{a}(t) = -\frac{1}{T} a(t) + w(t)$$

où  $T$  est grand ( $T \approx 10^4$  s) et  $w(t)$  est un bruit blanc gaussien centré.

La figure 4-5 montre que malgré les variations du paramètre  $a$ , la poursuite de l'état du MR et l'estimation du paramètre  $a(t)$  sont correctes. La commande a des variations assez importantes.

Remarques :

##### REMARQUE 1

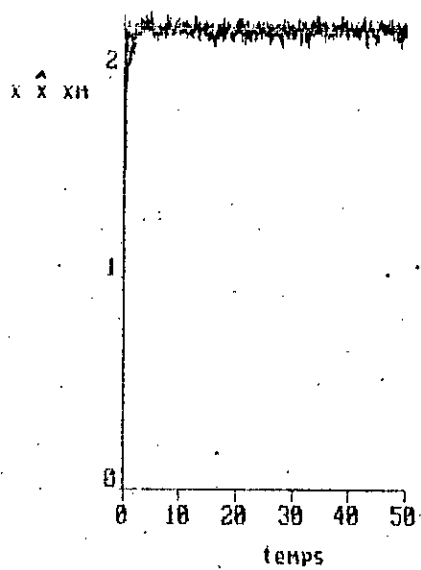
Nous aurions pu utiliser la loi de commande :

$$u(t) = -K(t)y(t) + r(t) \quad (4-13)$$

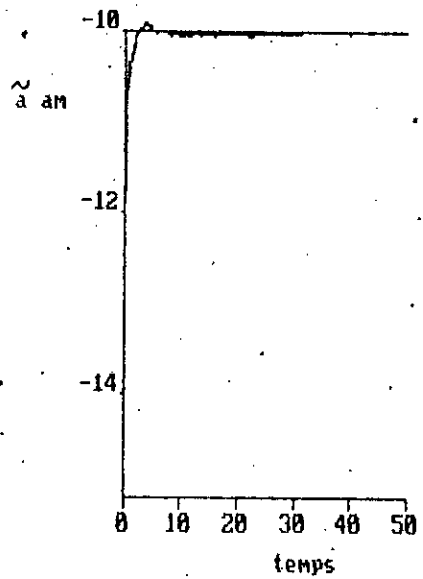
Cependant, cette loi introduit, par l'intermédiaire de  $y$ , le bruit  $v$  sur la commande. Ceci peut entraîner des variations importantes de la commande comme on peut le voir en figure 4-6. La loi

$$u = -K(t)\hat{x} + r$$

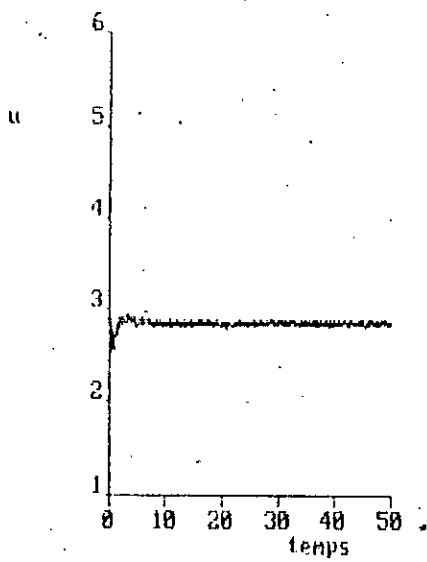
permet d'éviter cela.



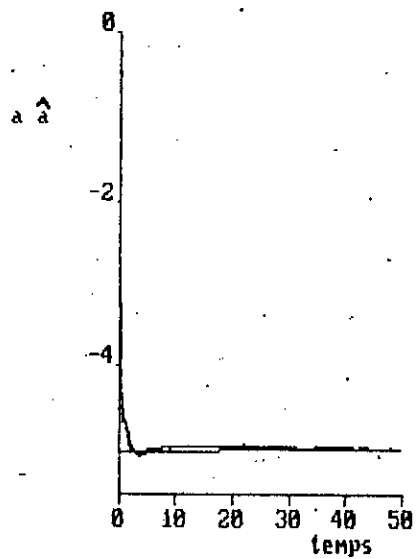
a) Poursuite de l'état  $x_m$



b) Poursuite du paramètre  $a_m$



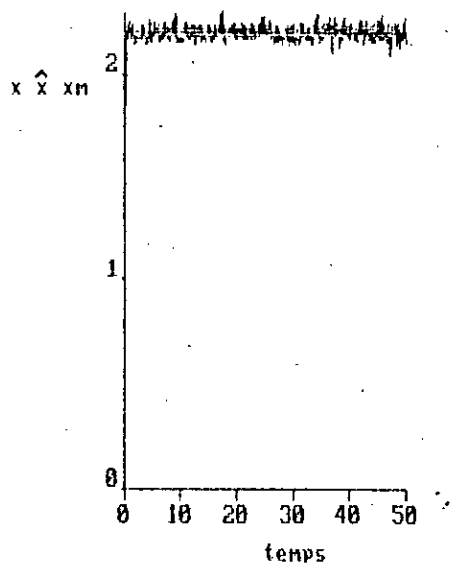
c) Commande



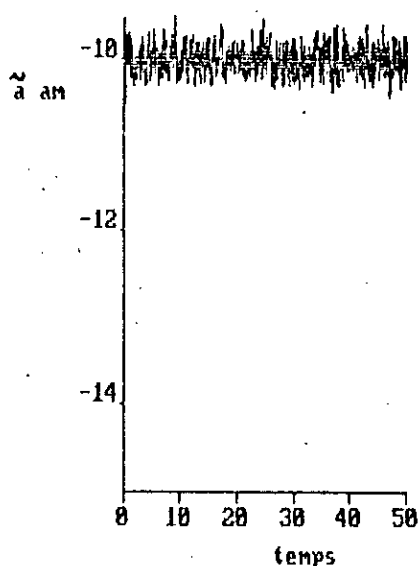
d) Estimé du paramètre  $a$

Fig 4-4 Algorithme stochastique avec bruits corrélés

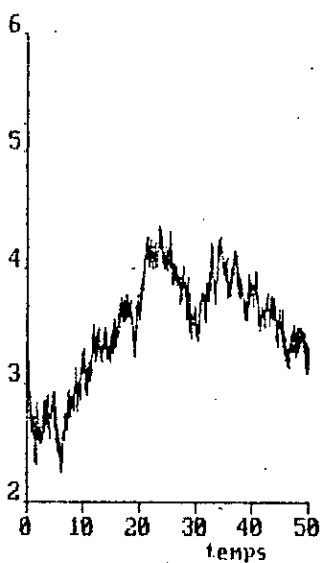
$$d_c = d + v \quad \text{Var}(d) = 1 \quad \text{Var}(v) = 0.5$$



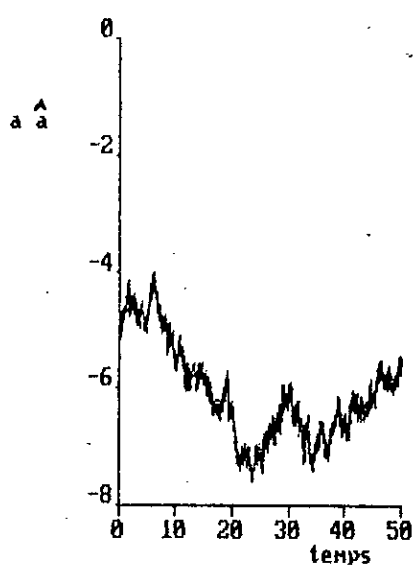
a) Poursuite de l'état  $x_n$



b) Poursuite du paramètre  $a_n$



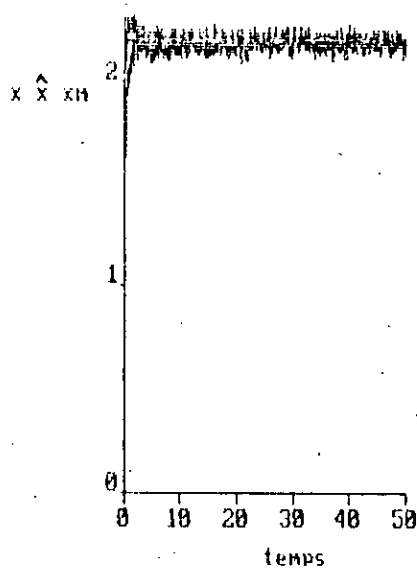
c) Commande



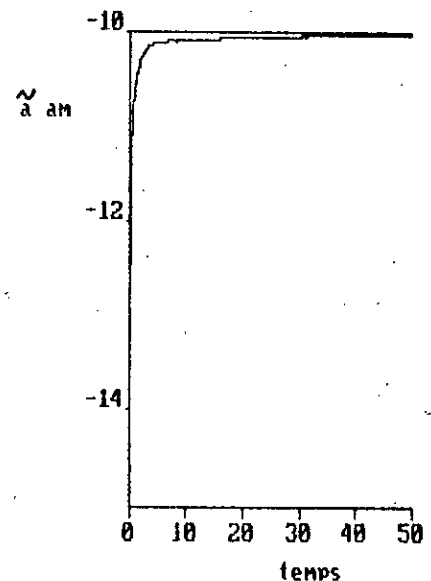
d) Estimé du paramètre  $a$

Fig 4-5 Algorithme stochastique : paramètre  $a$  variable

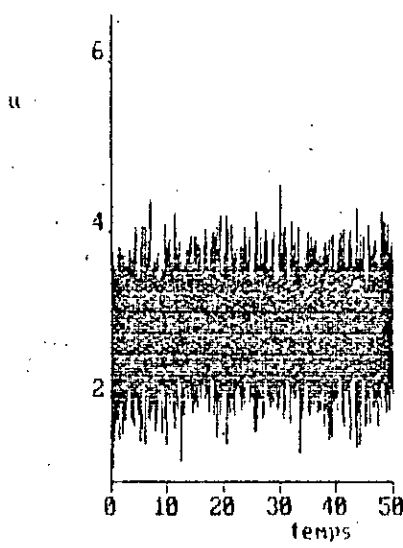
$$\text{Var}(d) = 0.01 \quad \text{Var}(v) = 0.01 \quad \text{Var}(w) = 25$$



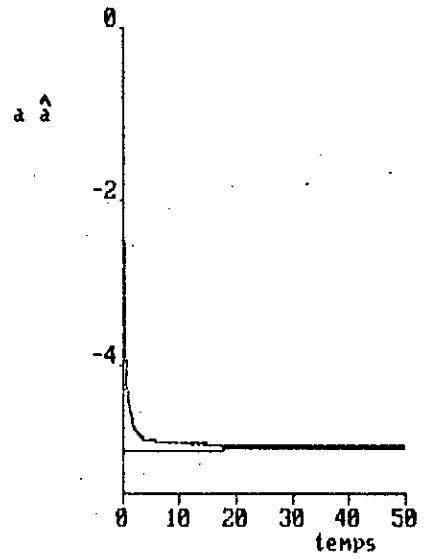
a) Poursuite de l'état  $x_n$



b) Poursuite du paramètre  $a_n$



c) Commande



d) Estimé du paramètre  $a$

Fig 4-6 Algorithme stochastique :

loi de commande  $u = -K(t)y + r$

$\text{Var}(d) = 1 \quad \text{Var}(v) = 1 \quad \text{Var}(u) = 0.25$

## REMARQUE 2

Si le paramètre  $a(t)$  est modélisé par

$$\dot{a}(t) = -\frac{1}{T} a(t) + \frac{1}{T} w(t)$$

où  $w(t)$  a une moyenne non nulle (et  $T$  est faible), alors un biais apparaît dans les estimés de l'état  $x$  et du paramètre  $a$ . En ajoutant un terme  $K_e(\hat{x} - x_m)$  à la commande  $u(t)$  :

$$u(t) = -K(t)\hat{x}(t) + r + K_e(\hat{x} - x_m) \quad (4-14)$$

on peut corriger le biais entre l'état du MR et l'état du processus. Ceci est illustré en figure 4-7. Cependant, la valeur  $K_e$  (constante) est obtenue par des essais en simulation.

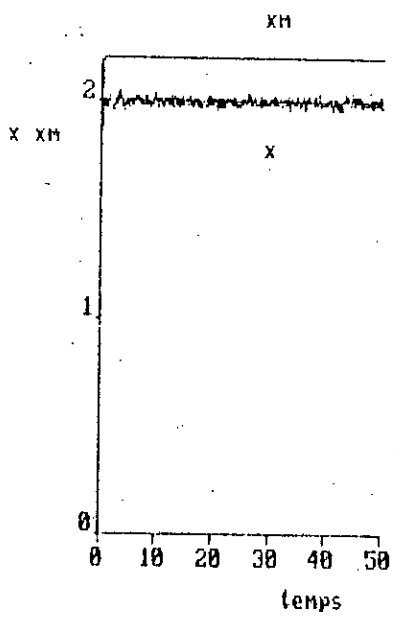
Les résultats de simulation présentés montrent que l'algorithme proposé possède un très bon comportement pour un processus d'ordre 1. Nous passons maintenant à un processus d'ordre plus élevé.

## 4-2 Application à un système SISO d'ordre 4

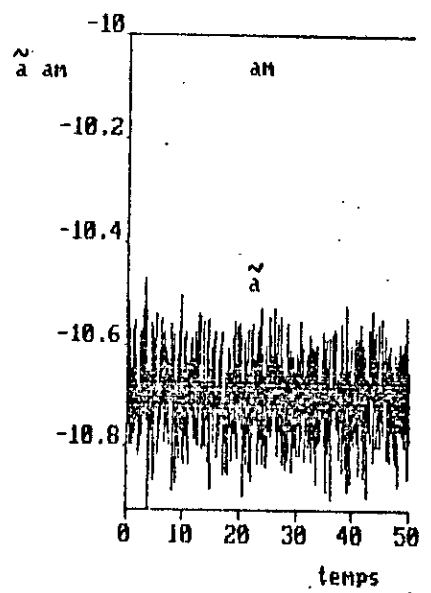
Le processus et le MR considérés sont décrits par des équations de la forme (3-30), (3-40) et (3-33). Les pôles du processus sont à  $-\frac{1}{2}$  ;  $-1$  ;  $-2$  ; et  $-3$ . Ceux du MR sont à  $-4$ .

### 4.2.1 Bruits Gaussiens

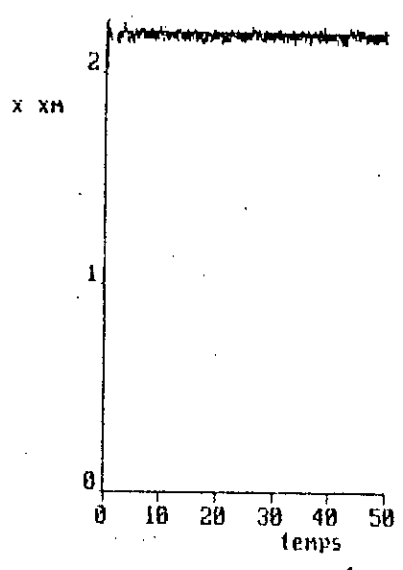
Comme le montrent les figures 4-8 et 4-13, la poursuite des états du MR est satisfaisante et les états du procédé sont correctement estimés. De même, le paramètre  $a_{p1}$  est bien



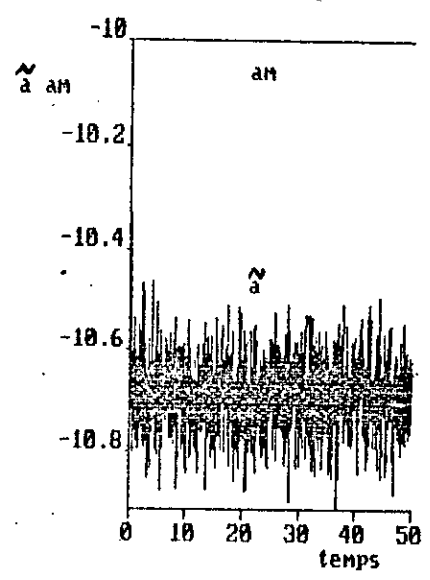
a) Poursuite de l'état  $x_M$   
(  $K_e = 0$  )



b) Poursuite du paramètre  $a_M$   
(  $K_e = 0$  )



a) Poursuite de l'état  $x_M$   
(  $K_e \neq 0$  )



b) Poursuite du paramètre  $a_M$   
(  $K_e \neq 0$  )

Fig 4.7 Algorithme stochastique : bruit  $w$  non centre

$$E(u) = -5 \quad \text{Var}(u) = 0.25 \quad \text{Var}(d) = 0.01 \quad \text{Var}(v) = 0.01$$



estimé. Par contre, un biais apparaît entre les paramètres  $a_{pi}$  ( $i > 1$ ) et leurs estimés. Ceci est illustré dans la figure 4-10 où l'on peut voir les paramètres du système en boucle fermée et ceux du MR. Dans les figures 4-11 et 4-12, on observe que les termes diagonaux de la solution de l'équation de Riccati deviennent très rapidement constants.

Remarquons que pour le MR choisi, les  $x_{mi}$  ( $i > 1$ ) tendent rapidement vers zéro. Par exemple,  $x_2$  et  $\hat{x}_2$  oscillent faiblement autour de zéro comme le montre la figure 4-8. Nous avons donc

$$u \simeq -K_1 \hat{x}_1 + r.$$

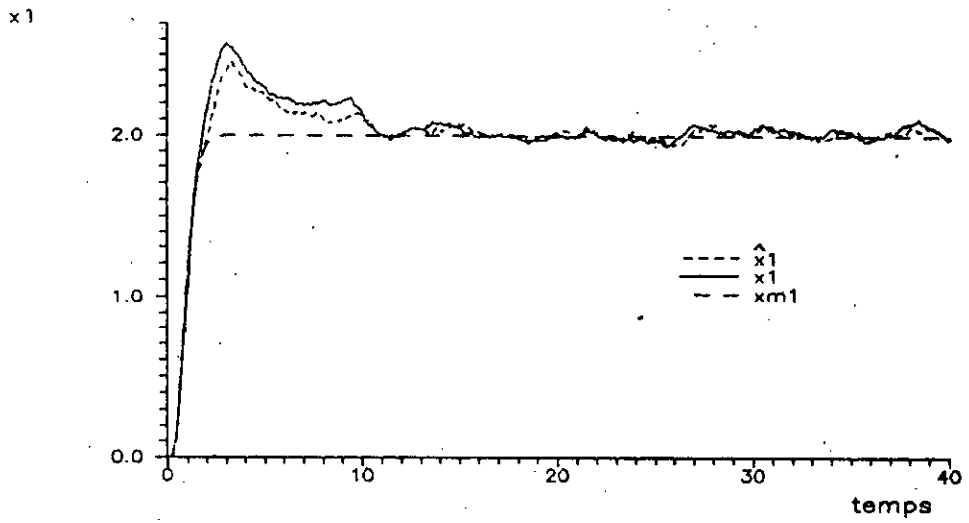
Autrement dit, il est important que  $\hat{a}_{p1}$  tende vers  $a_{p1}$ , mais, le biais entre  $\hat{a}_{pi}$  et  $a_{pi}$  ( $i > 1$ ) ne perturbe que très peu la performance car les  $x_i$  ( $i > 1$ ) sont presque nuls.

#### 4.2.2 Bruits uniformément distribués

Dans ce cas, nous obtenons également une bonne poursuite des états du MR et des estimés corrects des états du procédé. Ceci est illustré dans la figure 4-14. Le paramètre  $a_{p1}$  est correctement estimé mais un biais apparaît entre les paramètres  $a_{pi}$  ( $i > 1$ ) et leurs estimés. La figure 4-15 montre les paramètres du système en boucle fermée et ceux du MR.

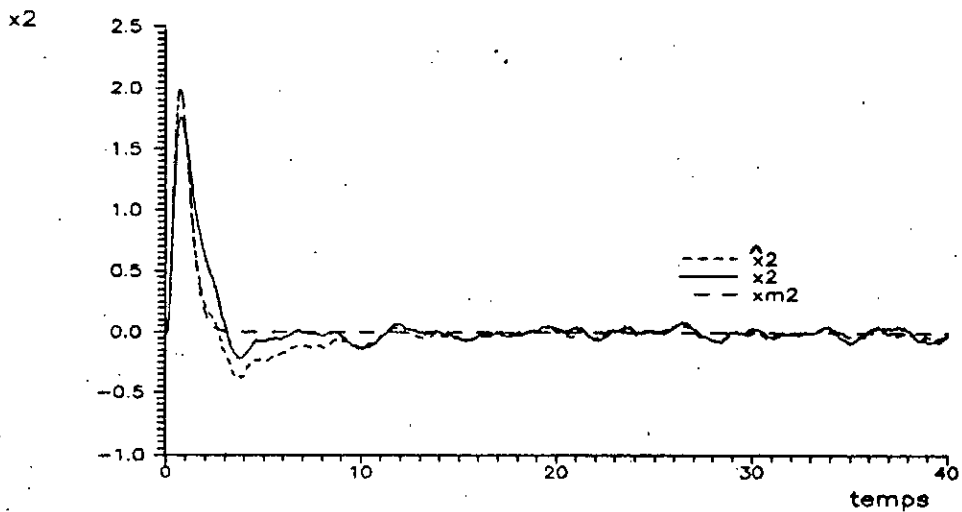
Algorithme stochastique

Bruits gaussiens



a) Poursuite de l'état  $x_{m1}$

Bruits



b) Poursuite de l'état  $x_{m2}$

Fig 4-8 Poursuite des états

$$\text{Var}(d)=0.01 \quad \text{Var}(v)=0.01$$

Algorithme stochastique

Bruits gaussiens

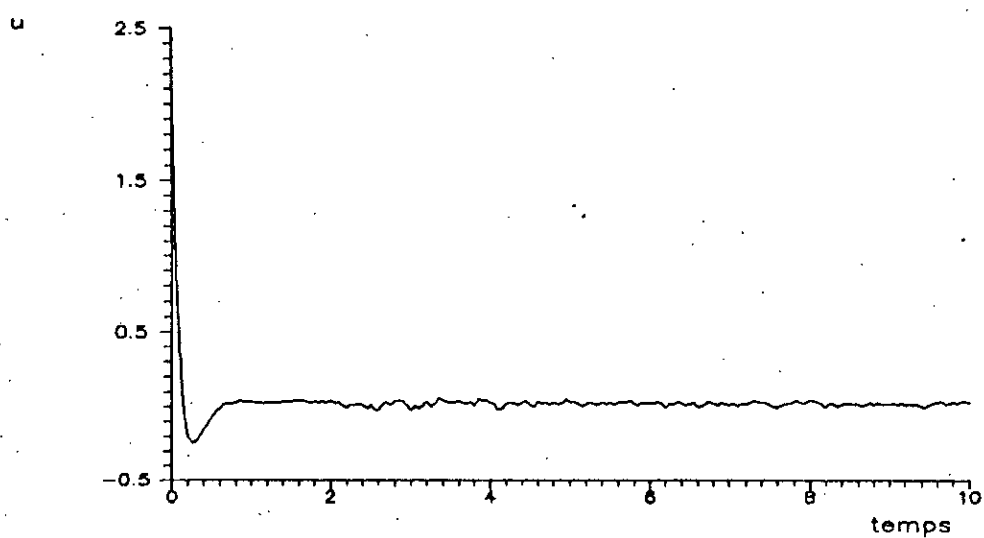
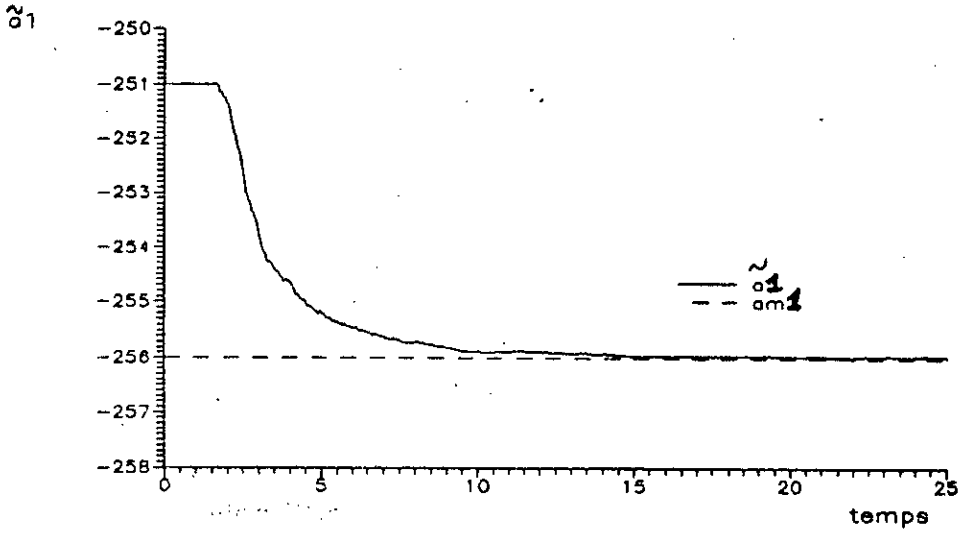


Fig 4-9 Commande

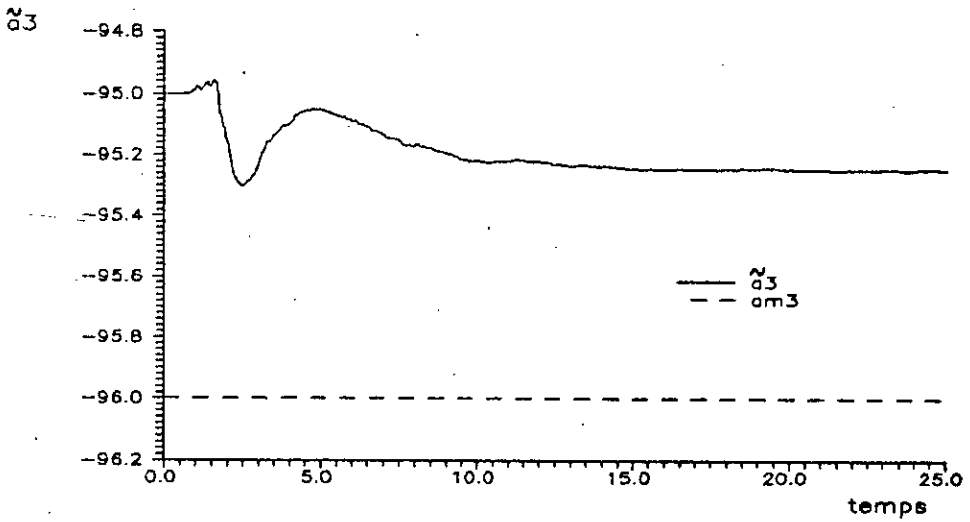
$$\text{Var}(d)=0.01 \quad \text{Var}(v)=0.01$$

Algorithme stochastique

Bruits gaussiens



a) Poursuite du paramètre  $a_{m1}$

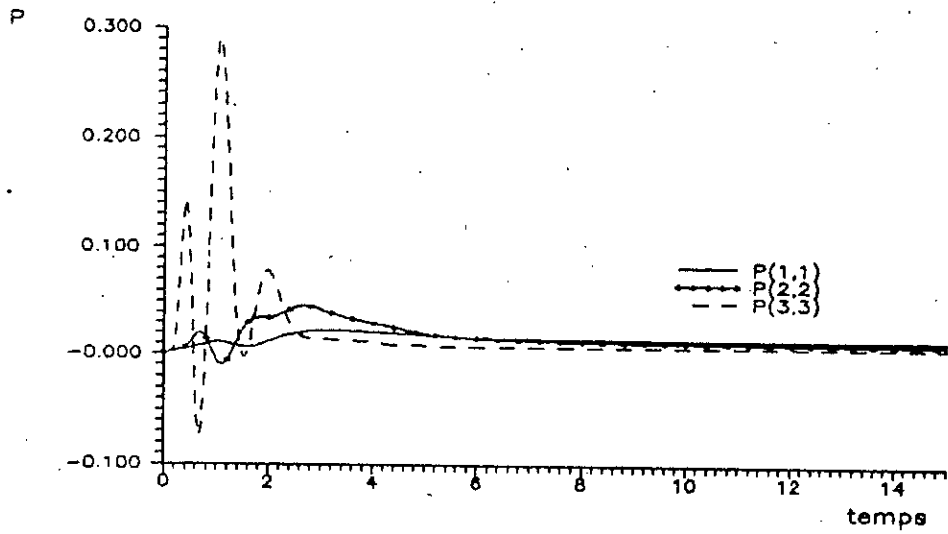


b) Poursuite du paramètre  $a_{m3}$

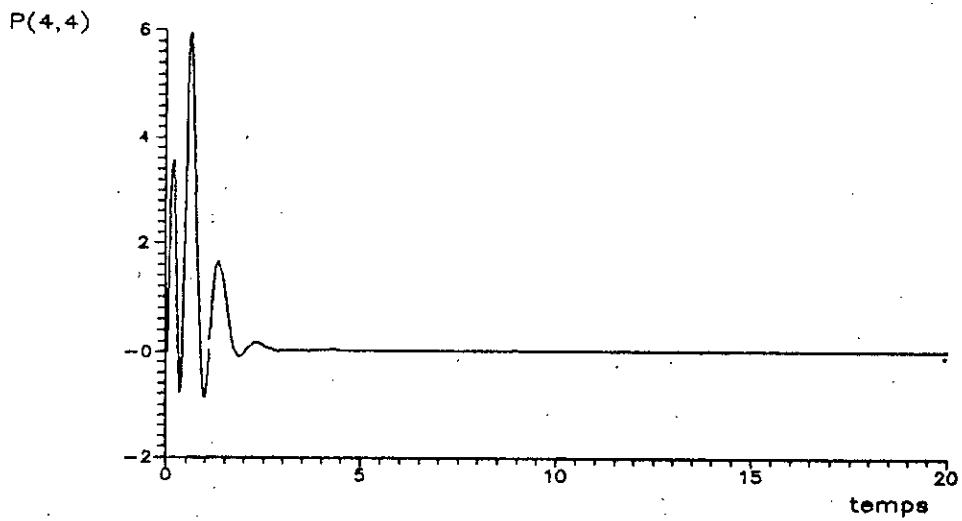
Fig 4-10 Poursuite des paramètres du MR  
 $\text{Var}(d)=0.01$   $\text{Var}(v)=0.01$

Algorithme stochastique

Bruits gaussiens



a)



b)

Fig 4-11. Solutions de l'equation de Riccati

$$\text{Var}(d)=0.01 \quad \text{Var}(v)=0.01$$

Algorithme stochastique

Bruits gaussiens

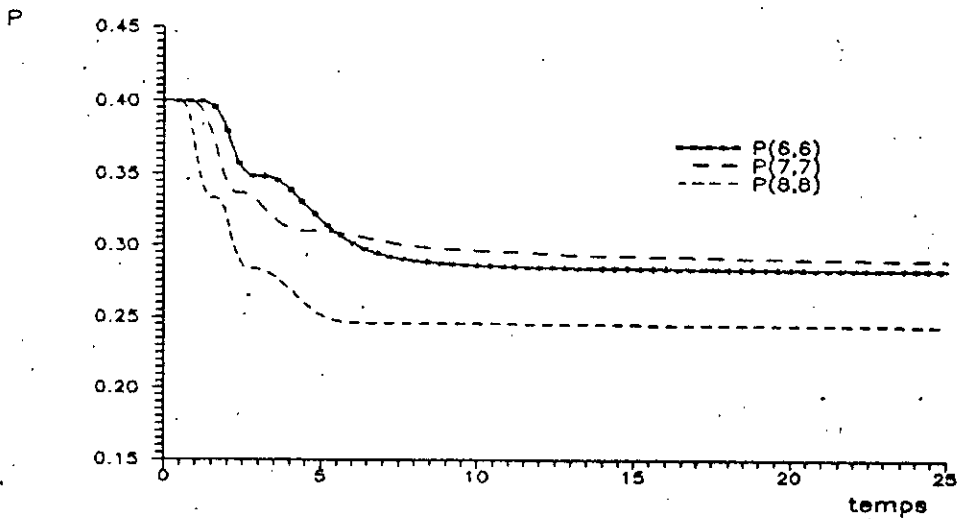
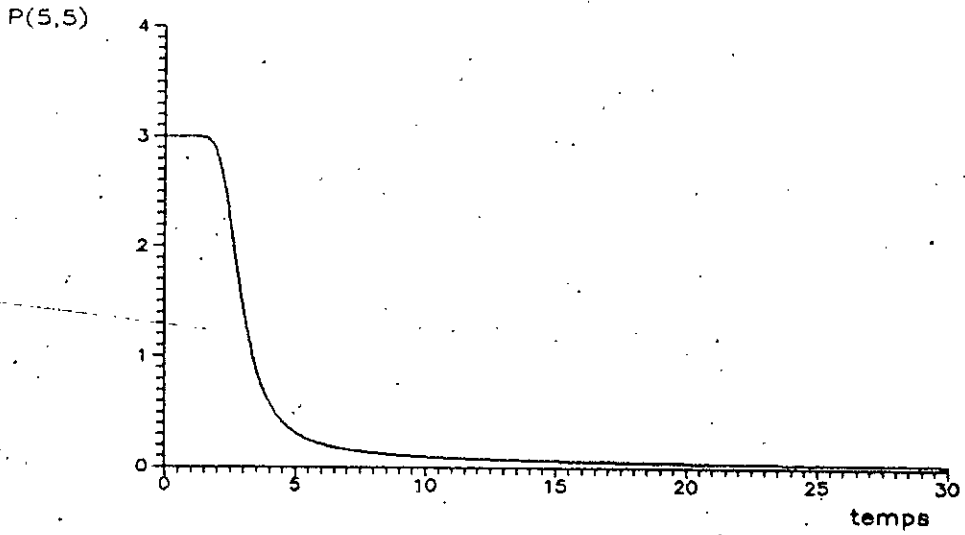


Fig 4-12 Solutions de l'equation de Riccati

$$\text{Var}(d)=0.01 \quad \text{Var}(v)=0.01$$

Algorithme stochastique

Bruits gaussiens

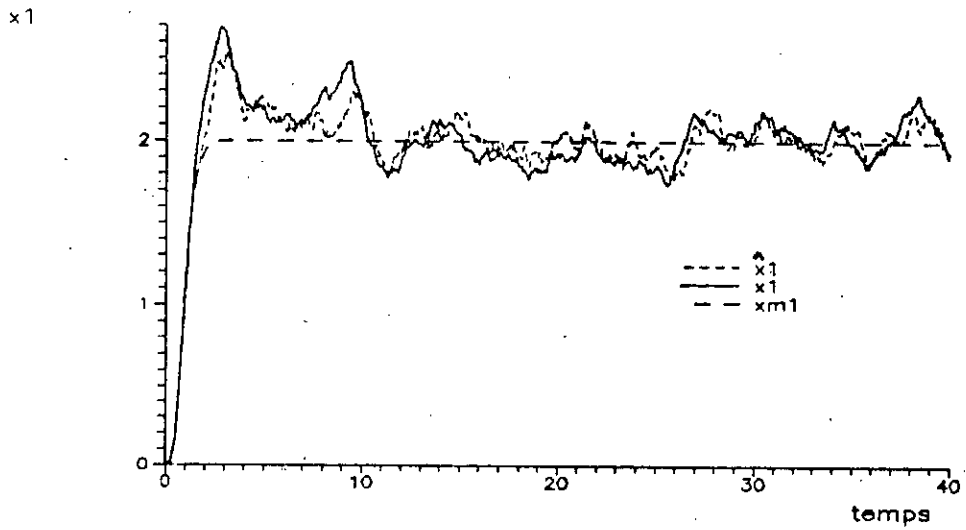
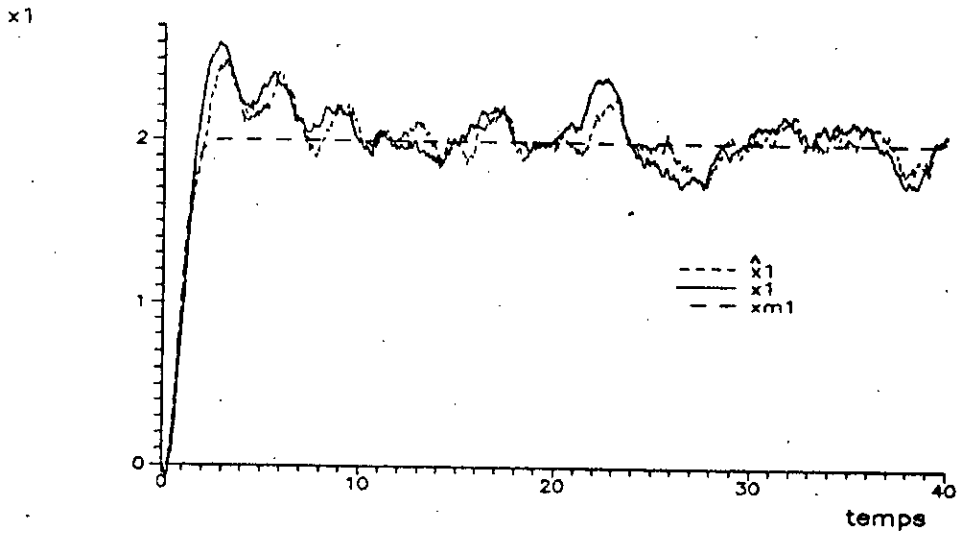


Fig 4-13 Poursuite de l'état  $x_{m1}$

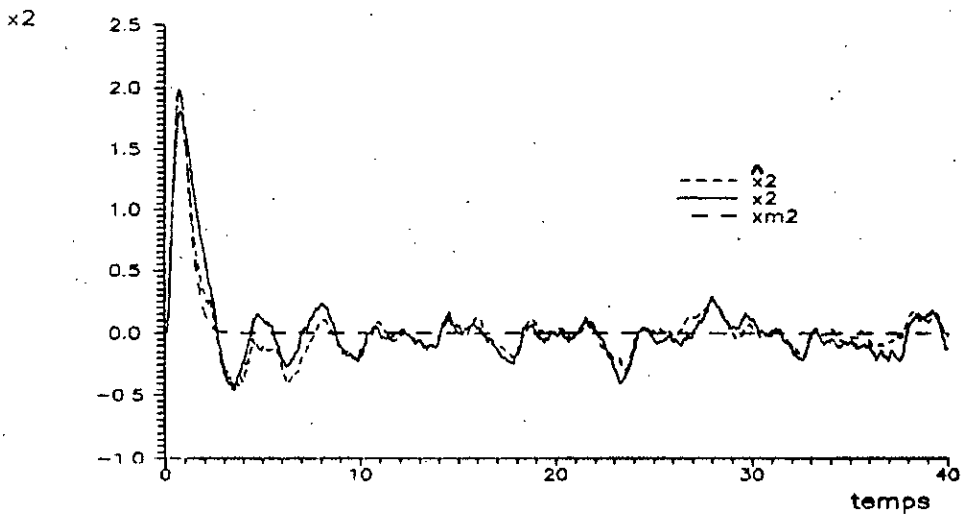
$$\text{Var}(d)=0.1 \quad \text{Var}(v)=0.1$$

Algorithme stochastique

Bruits uniformement distribués



a) Poursuite de l'état  $x_1$



b) Poursuite de l'état  $x_{m2}$

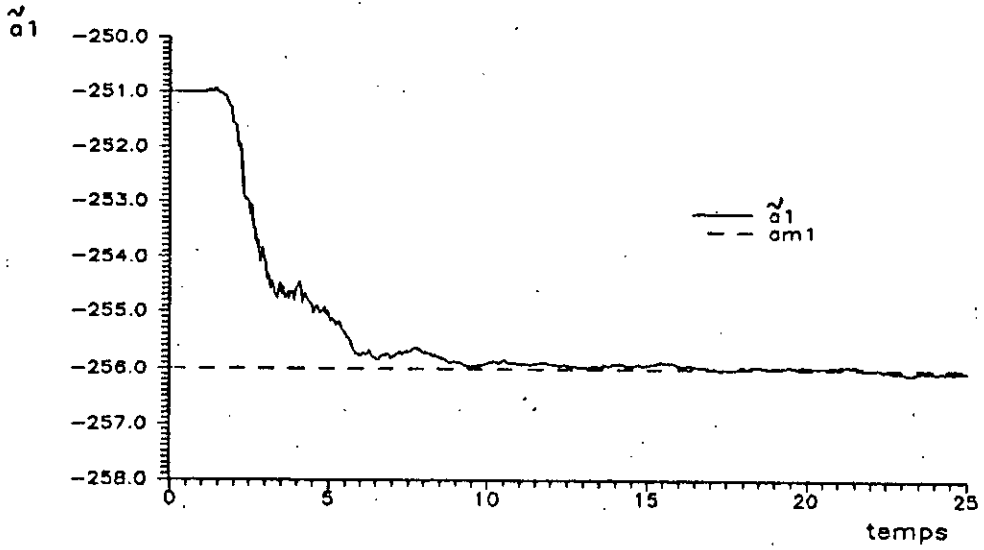
Fig 4-14 Poursuite des états

$$\text{Var}(d)=0.1 \quad \text{Var}(v)=0.1$$

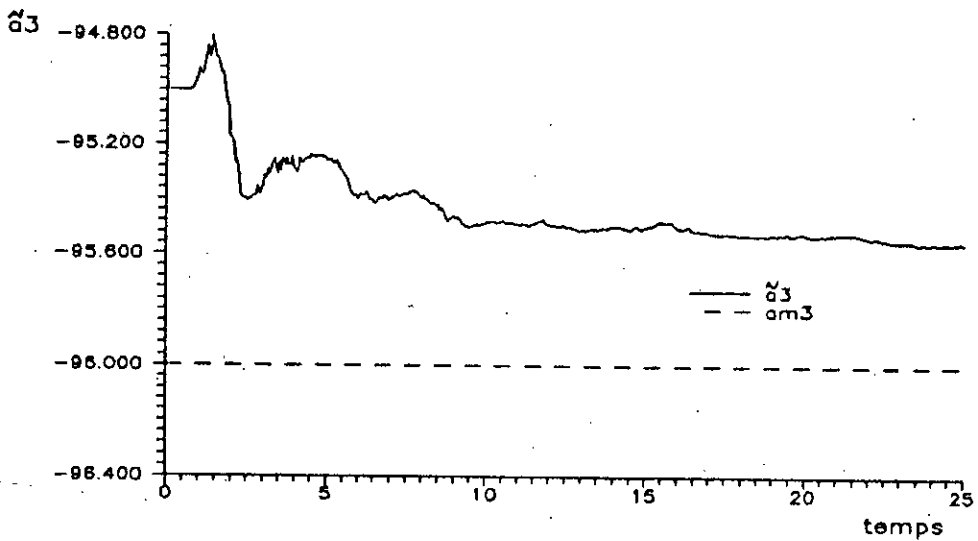


Algorithme stochastique

Bruits uniformement distribués



a) poursuite du paramètre  $a_{m1}$



b) Poursuite du paramètre  $a_{m3}$

Fig 4-15 Poursuite des paramètres du MR

$$\text{Var}(d)=0.1 \quad \text{Var}(v)=0.1$$

CHAPITRE VCONCLUSION

Les résultats de simulation du chapitre précédent montrent que l'algorithme de commande proposé conduit à des résultats très satisfaisants pour un procédé d'ordre un. Dans le cas d'un système de dimension quatre, les résultats obtenus pour la poursuite des états par ceux du MR restent satisfaisants. Cependant, les estimés des paramètres du procédé peuvent être biaisés.

Nous avons utilisé, dans nos simulations, un filtre de Kalman étendu pour estimer un vecteur d'état élargi composé des états du procédé et de ses paramètres inconnus. Or, comme nous l'avons signalé dans le chapitre III de la partie C, le filtre de Kalman étendu est un filtre approximatif pour des systèmes non linéaires. Il peut donner lieu à des estimés biaisés ou même diverger. Des modifications de l'algorithme, introduites dans le chapitre III de la partie C, ont été proposées pour faire face à ces problèmes. Dans nos simulations, nous avons utilisé un filtre de Kalman étendu non modifié. L'utilisation des versions modifiées de ce filtre permettent éventuellement une meilleure connaissance des paramètres du procédé commandé. Toutefois, celle-ci est obtenue au prix d'une complexité accrue de l'algorithme de commande.

CONCLUSIONS

ET

PERSPECTIVES

## CONCLUSIONS ET PERSPECTIVES

---

Nous avons analysé et simulé trois algorithmes de CA : MRAC, MCS et MRACM.

Dans la première partie, nous avons tout d'abord repris l'étude d'algorithmes existants : le MRAC développé par Landau [16], [17] et le MCS proposé par Stoten et Benchoubane [36], [37]. Nous avons ensuite simulé et comparé les lois de commande MRAC et MCS pour des systèmes à paramètres constants et nous avons pris comme nouvelle application un bras manipulateur de classe 4. Les deux algorithmes ont donné de bonnes performances : l'erreur de poursuite transitoire du MCS est supérieure à celle du MRAC mais ces erreurs se rejoignent rapidement.

Nous avons ensuite traité le taux de convergence du MCS. Une contribution nouvelle concernant la convergence du MCS a été apportée en étendant les résultats de Stoten et Benchoubane [39] à un système du deuxième ordre. Une enveloppe supérieure de l'erreur de poursuite du MCS a été construite. L'analyse de l'influence des paramètres  $\alpha$  et  $\beta$  sur l'amplitude et sur le taux de convergence de l'enveloppe nous a permis de proposer des recommandations pour le choix de ces paramètres. Pour une application donnée, le concepteur pourra tracer l'enveloppe de l'erreur pour différentes valeurs de  $\alpha$  et  $\beta$  et choisir celles qui répondent au mieux aux spécifications désirées.

A partir des analyses précédentes, nous avons essayé de généraliser les algorithmes MRAC et MCS à une autre classe de systèmes : les systèmes à paramètres variables dans le temps. Dans ce contexte, notre étude présente une contribution nouvelle qui consiste à introduire une identification dans l'algorithme MRAC afin de tenir compte des variations des paramètres du procédé. L'algorithme obtenu a été appelé MRAC modifié (MRACM). Comme applications des systèmes SISO et MIMO

ont été simulés et testés. Nous avons ensuite comparé les lois de commande MRAC, MCS et MRACM. Pour cela, nous avons considéré des paramètres sinusoïdaux de différentes amplitudes et de différentes fréquences ainsi que des paramètres donnés par des fonctions rectangulaires. L'algorithme MRACM présente une erreur de poursuite généralement inférieure à celles du MRAC et du MCS. Celle-ci peut toutefois présenter des pics importants en cas de variations trop rapides ou d'amplitude trop grande des paramètres du procédé.

D'autre part, une meilleure identification des paramètres nécessite une entrée de référence suffisamment riche en fréquences. Comme nous l'avons signalé dans le chapitre VI de la partie B, cette condition devrait pouvoir être réduite [42], [23].

L'erreur de poursuite du MCS peut quant à elle devenir inférieure à celle du MRAC.

L'avantage du MCS est que, contrairement au MRAC, sa mise en oeuvre ne nécessite qu'une connaissance minimale du système à commander.

Nous avons entamé la troisième partie de notre étude, avec l'idée de généraliser l'application du MCS à un environnement stochastique. Cependant, l'instabilité mise en évidence lors de l'application du MCS dans un tel environnement, nous a conduit à adopter une nouvelle approche. Nous avons alors développé un algorithme stochastique basé sur le filtrage de Kalman étendu. Cet algorithme conduit à une poursuite satisfaisante des états du MR mais les estimés des paramètres du procédé peuvent être biaisés. Ce biais devrait pouvoir être réduit en utilisant des versions modifiées du filtre de Kalman étendu. L'algorithme obtenu serait plus complexe mais plus performant. D'autre part, l'algorithme stochastique développé

pour des systèmes SISO concerne une représentation d'état sous forme canonique. Il reste à généraliser cet algorithme pour des systèmes représentés sous une forme non canonique et pour des systèmes multivariables.

Enfin, il serait intéressant de considérer des applications effectives de ces algorithmes.

REFERENCES

BIBLIOGRAPHIQUES

## REFERENCES BIBLIOGRAPHIQUES

- [ 1 ] M. ABRAMOWITZ and I. A. STEGUN  
"Handbook of Mathematical Functions"  
Dover Pubs. New-York 1968.
- [ 2 ] K. J. ASTROM  
"Theory and Applications of Adaptive Control - A Survey"  
Automatica, Vol. 19, n° 5, pp. 471-486, 1983.
- [ 3 ] K. J. ASTROM and B. WITTENMARK  
"Adaptive Control"  
Addison - Wesley 1989.
- [ 4 ] A. BALESTRINO, G. DE MARIA, A. S. I. ZINOBER  
"Nonlinear Adaptive Model-Following Control"  
Automatica, Vol. 20, n° 5, 1984.
- [ 5 ] M. BOURI et M. RACHEDI  
"Analyse et synthèse de contrôleurs adaptatifs à modèle  
de référence :  
- cas déterministe  
- cas de structures à paramètres variables"  
Projet de fin d'études - Département Electronique  
ENP, juin 1992.
- [ 6 ] H. BUHLER  
"Réglage par mode de glissement"  
Presses Polytechniques Romandes  
Coll. Complément au Traité d'Electricité, 1986.



- [ 7] P. E. CAINES and S. LAFORTUNE  
 "Adaptive Control with Recursive Identification for Stochastic Linear Systems"  
 IEEE Trans. on Automatic Control, Vol. AC-29, n° 4,  
 pp. 312-320, April 1984.
- [ 8] C. K. CHUI, G. CHEN and H. C. CHUI  
 "Modified Extended Kalman Filtering and a Real-Time Parallel Algorithm for System Parameter Identification"  
 IEEE Trans. on Automatic Control, Vol. AC-35, n° 1,  
 pp. 100-104, January 1990.
- [ 9] L. DUGARD  
 "Etude des systèmes adaptatifs avec modèle de référence dans un environnement stochastique"  
 Thèse de Docteur-Ingénieur, INP Grenoble 1980.
- [10] Y. M. EL-FATTAH et C. FOULARD  
 "Commande adaptative sous-optimale de systèmes dynamiques stochastiques récurrents linéaires avec des paramètres variables inconnus"  
 Revue Française d'Automatique, Informatique et Recherche Opérationnelle - RAIRO, Novembre 1975, J-3, p.5 à 31.
- [11] G. C. GOODWIN and K. S. SIN  
 "Adaptive Filtering Prediction and Control"  
 Prentice-Hall, Information and System Sciences Series, 1984.
- [12] G. C. GOODWIN and D. Q. MAYNE  
 "Continuous-Time Stochastic Model Reference Adaptive Control"  
 IEEE Trans. on Automatic Control, Vol. AC-36, n° 11,  
 pp. 1254-1263, November 1991.

- [13] P. A. IOANNOU and P. V. KOKOTOVIC  
"Adaptive Systems with Reduced Models"  
Lecture Notes in Control and Information Sciences,  
1983.
- [14] H. KURZ, R. ISERMANN and R. SCHUMANN  
"Experimental Comparison and Application of Various  
Parameter - Adaptive Control Algorithms"  
Automatica, Vol. 16, pp. 117-133, 1980.
- [15] I. A. LANDAU  
"Les systèmes adaptatifs avec modèle"  
Automatisme, tome XVI, n° 5, p. 272 à 291, mai 1971.
- [16] I. D. LANDAU  
"Sur une méthode de synthèse de systèmes adaptatifs avec  
modèle utilisés pour la commande et l'identification  
d'une classe de procédés physiques"  
Thèse de Docteur Es Sciences physiques  
Université Scientifique et Médicale de Grenoble,  
juin 1973.
- [17] I. D. LANDAU  
"Adaptive Control : The Model Reference Approach"  
Marcel Dekker, 1979.
- [18] I. D. LANDAU et L. DUGARD  
"Commande Adaptative : Aspects pratiques et Théoriques"  
Ed. Masson, 1986.
- [19] I. D. LANDAU  
"Identification et Commande des Systèmes"  
Traité des Nouvelles Technologies : Série Automatique  
Paris : Hermes, 1988.

- [20] L. LJUNG  
"Analysis of Recursive Stochastic Algorithms"  
IEEE Trans. on Automatic Control,  
Vol. AC-22, N° 4, pp. 551-575, August 1977.
- [21] L. LJUNG  
"Asymptotic Behavior of the Extended Kalman Filter  
as a Parameter Estimator for Linear Systems"  
IEEE Trans. on Automatic Control,  
Vol. AC-24, n° 1, pp. 36-50, February 1979.
- [22] L. LJUNG  
"System Identification : Theory for the User"  
Prentice-Hall 1987.
- [23] P. LOHNBERG, A. STIENSTRA and H. J. A. F. TULLEKEN  
"Time-Varying Parameter Estimation Combining Directional  
and Uniform Forgetting"  
Preprints of the 11<sup>th</sup> IFAC World Congress  
" Automatic in the Service of Mankind "  
Tallin, Estonia USSR August 13-17, 1990  
Vol. 4, pp. 232-237,  
Edited by V. Utkin, U. Jaksoo.
- [24] P. S. MAYBECK  
"Stochastic Models, Estimation, and Control"  
Academic Press 1982.
- [25] D. MEHDI  
"Contribution à l'analyse des systèmes stochastiques  
dont les paramètres sont aléatoires"  
Thèse de Doctorat en Automatique,  
Université de Nancy I, février 1986.

- [26] H. MORIMOTO  
 "Adaptive LQG Regulator via the Separation Principle"  
 IEEE Trans. on Automatic Control, Vol. AC-35, n° 1,  
 pp. 85-88, January 1990.
- [27] K. S. NARENDRA and A. M. ANNASWAMY  
 "Stable Adaptive Systems"  
 Prentice Hall,  
 Information and System Sciences Series, 1989.
- [28] R. ORTEGA and Y. TANG  
 "Robustness of Adaptive Controllers - A Survey"  
 Automatica, Vol. 25, n°5, pp. 651-677, 1989.
- [29] V. M. POPOV  
 "L'Hyperstabilité des Systèmes Automatiques"  
 Dunod 1973.
- [30] C. E. ROHRS, L. VALAVANI, M. ATHANS, and G. STEIN  
 "Robustness of Continuous Time Adaptive Control  
 Algorithms in the Presence of Unmodelled Dynamics"  
 IEEE Trans. on Automatic Control, Vol. AC-30, N° 9,  
 pp. 881-889, September 1985.
- [31] S. SASTRY and M. BODSON  
 "Adaptive Control  
 Stability, Convergence, and Robustness"  
 Prentice - Hall 1989.
- [32] E. SINNER  
 "Régulateur adaptatif à variables d'état pour processus  
 industriels"  
 Revue Française d'Automatique, Informatique et de  
 Recherche Opérationnelle J-1, p. 103 à 123, février 1975.

- [33] K. SOBEL, H. KAUFMAN, L. MABIUS  
"Implicit Adaptive Control for a Class of MIMO Systems"  
IEEE Trans. on Aerospace and Electronic Systems,  
Vol. AES-18, n°5, pp. 576-590, September 1982.
- [34] V. SOLO  
"Stochastic Adaptive Control and Martingale Limit  
Theory"  
IEEE Trans. on Automatic Control, Vol. AC-35, N°1,  
pp. 67-71, January 1990.
- [35] D. P. STOTEN  
"Generalized Manipulator Dynamics, with Regard to Model  
Reference Adaptive Control"  
Int. J. of Control, 1989, Vol. 50, n° 6, pp. 2249-2268.
- [36] D. P. STOTEN and H. BENCHOUBANE  
"Empirical Studies of an MRAC Algorithm with Minimal  
Controller Synthesis"  
Int. J. of Control 1990, Vol. 51, n° 4, pp. 823-849.
- [37] D. P. STOTEN and H. BENCHOUBANE  
"Robustness of a Minimal Controller Synthesis  
Algorithm"  
Int. J. of Control, 1990, Vol. 51, n° 4, pp. 851-861.
- [38] D. P. STOTEN and S. P. HODGSON  
"Comparative Implementation Studies of the Minimal  
Control Synthesis Algorithm on a Class 1 Manipulator"  
Journal of systems and Control Engineering  
Proc. Instn Mech. Engrs, Vol. 205, pp. 23-33, 1991.

- [39] D. P. STOTEN and H. BENCHOUBANE  
 "Convergence Rates of an Adaptive Control Algorithm with Application to the Speed Control of a DC Machine"  
 IEEE IECON'90, Vol. I, pp. 390-395, 1990.
- [40] K. TSAKALIS and P. IOANNOU  
 "Adaptive Control of Linear Time-Varying Plants a New Controller Structure"  
 American Control Conference, pp. 583-588, 1987.
- [41] K. TSAKALIS and P. IOANNOU  
 "Adaptive Control of Linear Time-Varying Plants"  
 Automatica, Vol.23, n° 4, pp. 459-468, 1987.
- [42] M. UCHIDA, T. HIROSAKI, Y. TOYOTA, H. NAKAMURA  
 "Computing-Predictor-Network-Based MRACS for Non-Linear Time-Varying Processes with Unknown Parameters Subject to Unpredictable Disturbances"  
 First IFAC Symposium on Design Methods of Control Systems Preprints, Vol. 1, pp. 436-441, Zurich, 1991.
- [43] P. P. J. VAN DEN BOSCH  
 "PSI version 6.1"  
 Delft University of Technology, The Netherlands, 1984.
- [44] M. VIDYASAGAR  
 "Non Linear Systems Analysis"  
 Prentice-Hall, Electrical Engineering series, 1978.
- [45] C. A. WINSOR and R. J. ROY  
 "The Application of Specific Optimal Control to the Design of Desentized Model Following Control Systems"  
 IEEE Trans. on Automatic Control, Vol. AC-15, n°3, pp. 326-333, June 1970.

- [46] X. XIANYA and R. J. EVANS  
"Adaptive Control of Discrete-Time Time-Varying Systems  
with Unknown Deterministic Disturbances"  
IEE Proceedings, Vol. 131, Pt. D, n° 3, pp. 81-84,  
May 1984.
- [47] G. ZWINGELSTEIN et A. POUJOL  
"Estimation de paramètres de réacteurs nucléaires par  
filtrage de Kalman étendu"  
Revue Française d'Automatique, Informatique et  
Recherche Opérationnelle - RAIRO , p. 31 à 46, avril 1976.

ANNEXES

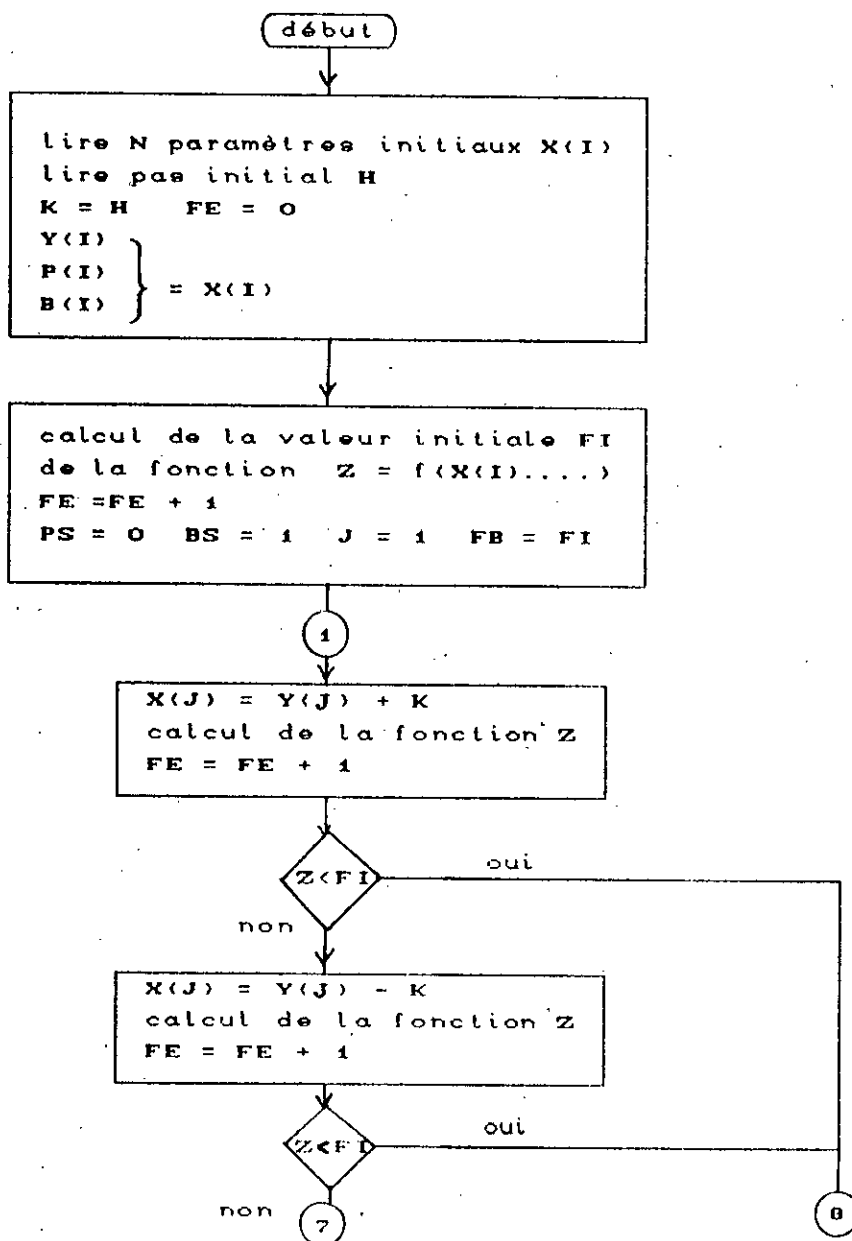


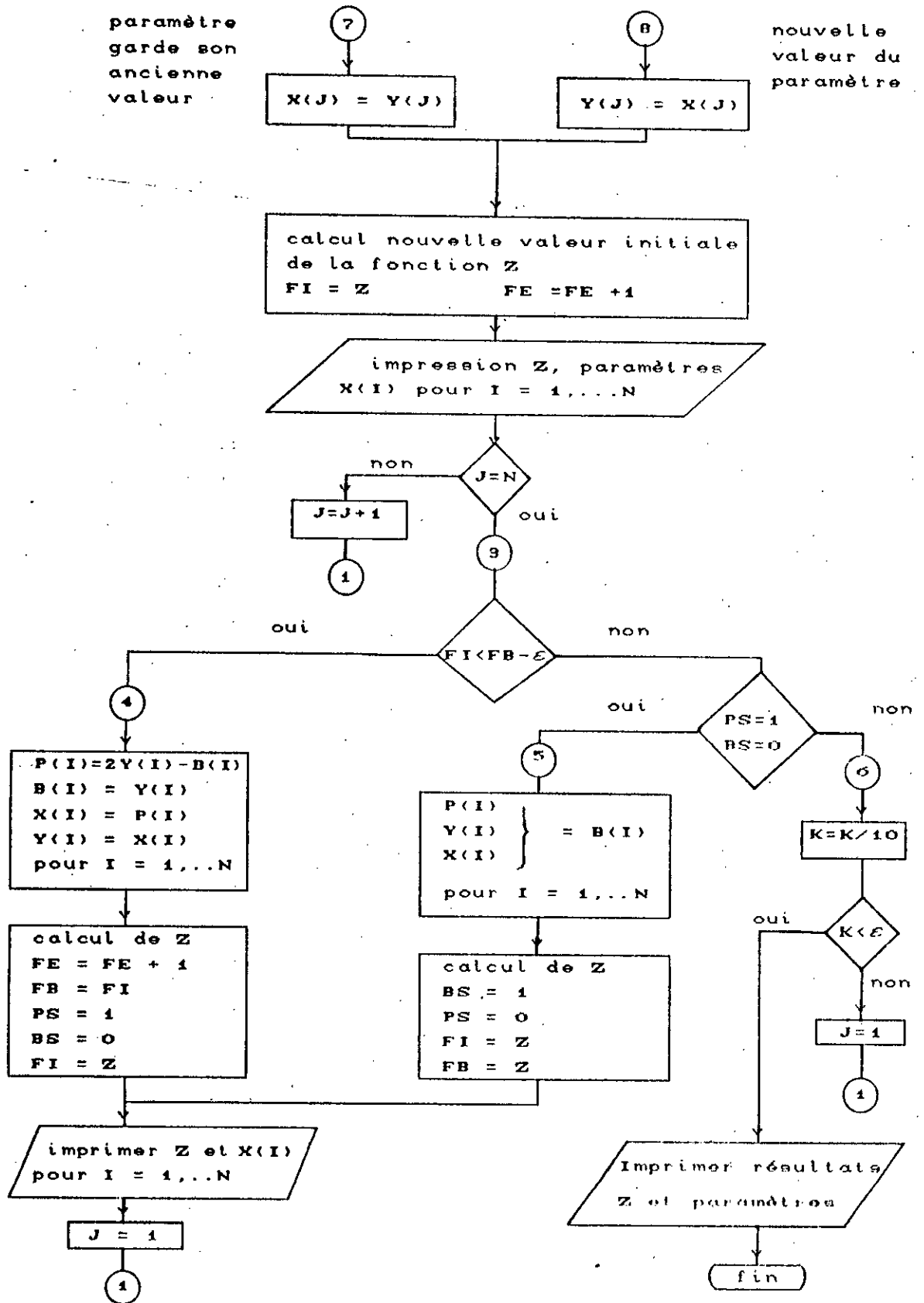
ANNEXE A

ALGORITHME DE HOOKE ET JEEVES

On cherche les valeurs des paramètres  $x_1, x_2, \dots, x_N$  qui minimisent la fonction  $Z = f(x_1, x_2, \dots, x_N)$ .

Organigramme :





On cherche les valeurs des paramètres  $x_1, x_2, \dots, x_N$  qui minimisent la fonction  $Z = f(x_1, x_2, \dots, x_N)$ .

Etape 1:

on choisit un pas  $K$  et on calcule  $f(x_1 + K, x_2, \dots, x_N)$  et on la compare avec  $f(x_1, x_2, \dots, x_N)$ .

Si  $Z$  a diminué, on garde la nouvelle valeur du paramètre et on passe au suivant.

Sinon, on calcule  $f(x_1 - K, x_2, \dots, x_N)$ . Si  $Z$  a diminué, le paramètre considéré garde sa nouvelle valeur. Si  $Z$  n'a pas diminué, il reprend son ancienne valeur et l'on passe au paramètre suivant pour lequel on procède de la même manière. Ainsi de suite jusqu'au dernier paramètre  $x_N$ .

Etape 2 :

on compare  $Z$  à sa valeur initiale (i.e. avec les paramètres d'avant l'étape 1) :

a) Si  $Z$  a suffisamment diminué (test 3), on essaie d'accélérer la recherche dans la direction indiquée par les résultats de l'étape 1 (branchement 4 :  $x(I) = 2 Y(I) - B(I)$  avec  $Y(I)$  valeurs les plus récentes des paramètres et  $B(I)$  valeurs précédentes) et on recommence la procédure de l'étape 1. Si  $Z$  a de nouveau diminué, on continue comme cela. Sinon on reprend les valeurs précédentes des paramètres (branchement 5) et on retourne à l'étape 1 (i.e. sans accélérer la recherche).

b) Si  $Z$  n'a pas suffisamment diminué (au test 3) et que l'on n'est pas en train d'accélérer la recherche dans une direction (i.e. quand  $PS \neq 1$  ou  $BS \neq 0$ ) on diminue le pas (branchement 6) pour affiner la recherche. Si le pas est inférieur à la valeur minimale fixée  $\epsilon$ , la procédure de recherche des paramètres optimaux est terminée. Si le pas est supérieur à  $\epsilon$ , on retourne à l'étape 1 mais avec la nouvelle valeur du pas.

## ANNEXE B

### TAUX DE CONVERGENCE

Cette annexe renferme des expressions utilisées dans le chapitre 7 de la partie A. Elles ont été obtenues lors de la dérivation de l'enveloppe de l'erreur de poursuite du MCS.

Coefficients de l'équation (7-22) :

$$A_1 = (b_{m1} r d_{21}) / [r_1 (r_1 - r_2)] \quad (B-1)$$

$$A_2 = -(b_{m1} r d_{21}) / [r_2 (r_1 - r_2)] \quad (B-2)$$

$$A_3 = \left[ (b_{m1} / (r_1 r_2)) + (a_{m1} / b_{m1}) \right] d_{21} r \quad (B-3)$$

$$A_4 = (b_{m1} r d_{22}) / [r_1 (r_1 - r_2)] \quad (B-4)$$

$$A_5 = -(b_{m1} r d_{22}) / [r_2 (r_1 - r_2)] \quad (B-5)$$

$$A_6 = \left[ (b_{m1} / (r_1 r_2)) + (a_{m1} / b_{m1}) \right] d_{22} r \quad (B-6)$$

$$A_7 = -d_{21} \quad (B-7)$$

$$A_8 = -d_{22} \quad (B-8)$$

$$A_9 = -1 / (a_{p1} / b_{p1}) \quad (B-9)$$

Bornes supérieures des intégrales de l'équation (7-22) :

$$|I_1| = \left| \int_0^t e_1 e^{r_1 \tau} d\tau \right|$$

$$|I_1| \leq \frac{K}{r_1 - r_2} \left[ \frac{1}{2r_1(\gamma - r_1)} + \frac{1}{(r_1 + r_2)(\gamma - r_2)} - \frac{1}{(\gamma + r_1)(\gamma - r_1)} - \frac{1}{(\gamma + r_1)(\gamma - r_2)} \right] \quad (\text{B-10})$$

$$|I_2| = \left| \int_0^t e_1 e^{r_2 \tau} d\tau \right| \leq \frac{K}{r_1 - r_2} \left[ -\frac{1}{\gamma + r_2} \left( \frac{1}{\gamma - r_1} + \frac{1}{\gamma - r_2} \right) + \frac{1}{(r_1 + r_2)(\gamma - r_1)} + \frac{1}{2r_2(\gamma - r_2)} \right] \quad (\text{B-11})$$

$$|I_3| = \left| \int_0^t e_1 d\tau \right| \leq \frac{K}{r_1 - r_2} \left[ -\frac{1}{\gamma} \left( \frac{1}{\gamma - r_1} + \frac{1}{\gamma - r_2} \right) + \frac{1}{r_1(\gamma - r_1)} + \frac{1}{r_2(\gamma - r_2)} \right] \quad (\text{B-12})$$

$$|I_4| = \left| \int_0^t e_2 e^{r_1 \tau} d\tau \right| \leq \frac{K}{r_1 - r_2} \left[ -\frac{1}{\gamma + r_1} \left( \frac{|r_1|}{\gamma - r_1} + \frac{|r_2|}{\gamma - r_2} \right) + \frac{|r_1|}{2r_1(\gamma - r_1)} + \frac{|r_2|}{(r_1 + r_2)(\gamma - r_2)} \right] \quad (\text{B-13})$$

$$|I_5| = \left| \int_0^t e_2 e^{r_2 \tau} d\tau \right| \leq \frac{K}{r_1 - r_2} \left[ \left( \frac{|r_1|}{\gamma - r_1} + \frac{|r_2|}{\gamma - r_2} \right) \frac{-1}{\gamma + r_2} + \frac{|r_1|}{(r_1 + r_2)(\gamma - r_1)} + \frac{|r_2|}{2r_2(\gamma - r_2)} \right] \quad (\text{B-14})$$

$$|I_6| = \left| \int_0^{t_0} e_2 d\tau \right|$$

$$|I_6| \leq \frac{K}{r_1 - r_2} \left[ \frac{-1}{\gamma} \left( \frac{|r_1|}{\gamma - r_1} + \frac{|r_2|}{\gamma - r_2} \right) + \frac{|r_1|}{r_1(\gamma - r_1)} + \frac{|r_2|}{r_2(\gamma - r_2)} \right] \quad (B-15)$$

$$|I_7| = \left| \int_0^{t_0} e_1^2 d\tau \right|$$

$$\begin{aligned} &\leq \left( \frac{K}{r_1 - r_2} \right)^2 \left[ \frac{-1}{2\gamma} \left( \frac{1}{\gamma - r_1} + \frac{1}{\gamma - r_2} \right)^2 \right. \\ &\quad + 2 \left( \frac{1}{\gamma - r_1} + \frac{1}{\gamma - r_2} \right) \left( \frac{1}{\gamma^2 - r_1^2} + \frac{1}{\gamma^2 - r_2^2} \right) \\ &\quad \left. - \frac{1}{2r_1(\gamma - r_1)^2} - \frac{1}{2r_2(\gamma - r_2)^2} - \frac{2}{(\gamma - r_1)(\gamma - r_2)(r_1 + r_2)} \right] \end{aligned}$$

(B-16)

$$|I_8| = \left| \int_0^{t_0} e_1 e_2 d\tau \right|$$

$$\begin{aligned} &\leq \left( \frac{K}{r_1 - r_2} \right)^2 \left[ \frac{-1}{2\gamma} \left( \frac{1}{\gamma - r_1} + \frac{1}{\gamma - r_2} \right) \left( \frac{|r_1|}{\gamma - r_1} + \frac{|r_2|}{\gamma - r_2} \right) \right. \\ &\quad + \left( |r_1| \left( \frac{1}{\gamma - r_1} + \frac{1}{\gamma - r_2} \right) + \frac{|r_1|}{\gamma - r_1} + \frac{|r_2|}{\gamma - r_2} \right) \frac{1}{\gamma^2 - r_1^2} \\ &\quad + \left( |r_2| \left( \frac{1}{\gamma - r_1} + \frac{1}{\gamma - r_2} \right) + \frac{|r_1|}{\gamma - r_1} + \frac{|r_2|}{\gamma - r_2} \right) \frac{1}{\gamma^2 - r_2^2} \\ &\quad \left. - \frac{|r_1|}{2r_1(\gamma - r_1)^2} - \frac{|r_2|}{2r_2(\gamma - r_2)^2} - \frac{|r_1| + |r_2|}{(r_1 + r_2)(\gamma - r_1)(\gamma - r_2)} \right] \end{aligned}$$

(B-17)

Coefficients de l'équation (7-24) :

$$B_1 = C |A_1| + |r_1| |A_4| / (2r_1) + C |A_3| + |r_1| |A_6| / r_1 + C |A_2| + |r_1| |A_5| / (r_1 + r_2) \quad (B-18)$$

$$B_2 = C |A_1| + |r_2| |A_4| / (r_1 + r_2) + C |A_2| + |r_2| |A_5| / (2r_2) + C |A_3| + |r_2| |A_6| / r_2 \quad (B-19)$$

$$B_3 = - |A_2| - |r_1| |A_5| \quad (B-20)$$

$$B_4 = - |A_1| - |r_2| |A_4| \quad (B-21)$$

$$B_5 = - |A_1| - |r_1| |A_4| \quad (B-22)$$

$$B_6 = - |A_2| - |r_2| |A_5| \quad (B-23)$$

$$B_7 = - |A_3| - |r_1| |A_6| \quad (B-24)$$

$$B_8 = - |A_3| - |r_2| |A_6| \quad (B-25)$$

$$B_9 = - \frac{K}{2(r_1 - r_2)} ( |A_7| + |r_1| |A_8| ) \quad (B-26)$$

$$B_{10} = - \frac{K}{2(r_1 - r_2)} ( |A_7| + |r_2| |A_8| ) \quad (B-27)$$

$$B_{11} = \frac{K}{r_1 - r_2} ( - |A_7| - ( |r_1| + |r_2| ) |A_8| / 2 ) \quad (B-28)$$

$$B_{12} = \frac{2K}{r_1 - r_2} ( |A_7| + |r_1| |A_8| ) \quad (B-29)$$

$$B_{13} = \frac{K}{r_1 - r_2} ( 2|A_7| + (|r_1| + |r_2|) |A_8| ) \quad (B-30)$$

$$B_{14} = \frac{K}{r_1 - r_2} ( 2|A_7| + (|r_1| + |r_2|) |A_8| ) \quad (B-31)$$

$$B_{15} = \frac{2K}{r_1 - r_2} ( |A_7| + |r_2| |A_8| ) \quad (B-32)$$

$$B_{16} = \frac{-K}{2(r_1 - r_2)r_1} ( |A_7| + |r_1| |A_8| ) \quad (B-33)$$

$$B_{17} = \frac{-K}{2(r_1 - r_2)r_2} ( |A_7| + |r_2| |A_8| ) \quad (B-34)$$

$$B_{18} = \frac{-K}{(r_1 - r_2)(r_1 + r_2)} ( 2|A_7| + (|r_1| + |r_2|) |A_8| ) \quad (B-35)$$

Polynômes  $P_i(\gamma)$  de l'équation (7-24) :

$$P_1(\gamma) = \gamma - r_1 \quad (B-36)$$

$$P_2(\gamma) = \gamma - r_2 \quad (B-37)$$

$$P_3(\gamma) = (\gamma - r_1)(\gamma + r_2) \quad (B-38)$$

$$P_4(\gamma) = (\gamma + r_1)(\gamma - r_2) \quad (B-39)$$

$$P_5(\gamma) = \gamma^2 - r_1^2 \quad (B-40)$$

$$P_6(\gamma) = \gamma^2 - r_2^2 \quad (B-41)$$

$$P_7(\gamma) = \gamma(\gamma - r_1) \quad (B-42)$$



$$P_8(\gamma) = \gamma (\gamma - r_2) \quad (B-43)$$

$$P_9(\gamma) = \gamma (\gamma - r_1)^2 \quad (B-44)$$

$$P_{10}(\gamma) = \gamma (\gamma - r_2)^2 \quad (B-45)$$

$$P_{11}(\gamma) = \gamma (\gamma - r_1)(\gamma - r_2) \quad (B-46)$$

$$P_{12}(\gamma) = (\gamma - r_1)(\gamma^2 - r_1^2) \quad (B-47)$$

$$P_{13}(\gamma) = (\gamma - r_2)(\gamma^2 - r_1^2) \quad (B-48)$$

$$P_{14}(\gamma) = (\gamma - r_1)(\gamma^2 - r_2^2) \quad (B-49)$$

$$P_{15}(\gamma) = (\gamma - r_2)(\gamma^2 - r_2^2) \quad (B-50)$$

$$P_{16}(\gamma) = (\gamma - r_1)^2 \quad (B-51)$$

$$P_{17}(\gamma) = (\gamma - r_2)^2 \quad (B-52)$$

$$P_{18}(\gamma) = (\gamma - r_1)(\gamma - r_2) \quad (B-53)$$

Coefficients du polynôme  $P(\gamma)$  de l'équation (7-25) :

$$z = \frac{r_1 - r_2}{K} |A_9| \quad (B-54)$$

$$C_0 = -r_1 r_2 [r_1 r_2 (B_{11} - r_2 B_7 - r_1 B_8) + r_2^2 B_9 + r_1^2 B_{10}] / z \quad (B-55)$$

$$\begin{aligned}
C_1 = & r_1^8 r_2^9 - [ r_1 r_2 (r_1 r_2 (-r_2 B_1 - r_1 B_2 + B_{10} + B_7 + B_8 - B_9 - B_4) + r_2^2 B_{16} \\
& + r_1^2 B_{17}) - r_2^2 B_5 - r_1 B_6 + r_2 B_{13} + r_1 B_{14} \\
& + (r_2 - r_1)(r_2^2 B_9 - r_1^2 B_9) + r_1^3 B_{15} + r_2^3 B_{12} ] / z \quad (B-56)
\end{aligned}$$

$$\begin{aligned}
C_2 = & -r_1^2 r_2^2 (r_1 + r_2) - [ r_1^2 r_2^2 (B_1 + B_2) + (r_2 - r_1)(r_2^2 B_{16} - r_1^2 B_{17}) \\
& + (r_2 B_7 + r_1 B_8 - B_{11})(r_1^2 + r_2^2) + (r_1 + r_2)(-r_2 B_9 + r_2^2 B_5 + r_1^2 B_6) \\
& - r_1 B_{10} (r_1 + r_2) - r_2^2 (B_{12} + B_{13}) - r_1^2 (B_{14} + B_{15}) \\
& + 2r_1 r_2 (r_1 B_9 + r_2 B_4) ] / z \quad (B-57)
\end{aligned}$$

$$\begin{aligned}
C_3 = & r_1 r_2 (-r_1^2 - r_2^2 + r_1 r_2) - [ (r_2 B_1 + r_1 B_2 - B_7 - B_8 - B_{10})(r_1^2 + r_2^2) \\
& (-r_2 B_{16} - r_1 B_{17} + (B_4 - B_9)(r_1 - r_2))(r_1 + r_2) - r_1 (B_{13} + B_{15}) \\
& - r_2 (B_{12} + B_{14}) + (r_1 - r_2)(B_9 - B_{10} + r_2 B_5 - r_1 B_{16}) ] / z \quad (B-58)
\end{aligned}$$

$$\begin{aligned}
C_4 = & (r_1^2 + r_2^2)(r_1 + r_2) - [ -(r_1^2 + r_2^2)(B_1 + B_2) + (r_1 - r_2)(B_{16} - B_{17}) \\
& - r_2 (B_7 + 2B_9) - r_1 (B_8 + 2B_4) - (r_1 + r_2)(B_5 + B_6) + \sum_{i=9}^{15} B_i ] / z \quad (B-59)
\end{aligned}$$

$$\begin{aligned}
C_5 = & (-r_1^2 - r_2^2 + r_1 r_2) - (-r_2 B_1 - r_1 B_2 + \sum_{i=9}^8 B_i + \sum_{i=16}^{10} B_i) / z \quad (B-60)
\end{aligned}$$

$$C_6 = - (r_1 + r_2) - (B_1 + B_2) / z \quad (B-61)$$

## ANNEXE C

### STABILITE AU SENS DE LYAPUNOV

Soit un système dynamique autonome décrit par une équation différentielle vectorielle :

$$\dot{x} = f(x, t) \quad -\infty < t < +\infty \quad (C-1)$$

où  $x$  représente le vecteur d'état du système.

Soit  $\phi(t, x_0, t_0)$  une solution du système (C-1), fonction unique et différentiable en  $t$ , vérifiant pour tous  $x_0, t_0$ , les équations :

$$\phi(t_0, x_0, t_0) = x_0 \quad (C-2)$$

$$\dot{\phi}(t, x_0, t_0) = f(\phi(t, x_0, t_0), t) \quad (C-3)$$

Un état  $x_0$  du système dynamique autonome donné par l'équation (C-1) est appelé état d'équilibre si :

$$f(x_0, t) = 0 \quad , \text{ pour tout } t \quad (C-4)$$

#### Définition 1 : Stabilité

Un état d'équilibre  $x_0$  du système donné par l'équation (C-1) est stable si pour tout nombre réel  $\varepsilon > 0$ , il existe un nombre réel  $\delta(\varepsilon, t_0) > 0$  tel que :

$$\|x_0 - x_0\| \leq \delta \Rightarrow \|\phi(t, x_0, t_0) - x_0\| \leq \varepsilon \quad (C-5)$$

pour tout  $t \geq t_0$ .

## Définition 2 : Stabilité asymptotique

Un état d'équilibre  $x_0$  du système donné par l'équation (C-1) est asymptotiquement stable si :

1.  $x_0$  est stable
2. il existe un nombre réel  $\delta(t_0)$  tel que :

$$\|x_0 - x_0\| \leq \delta(t_0) \rightarrow \lim_{t \rightarrow \infty} \|\phi(t, x_0, t_0) - x_0\| = 0 \quad (\text{C-6})$$

(i.e. toute solution  $\phi(t, x_0, t_0)$  qui démarre suffisamment près de  $x_0$  converge vers  $x_0$ , quand  $t$  tend vers l'infini).

## Définition 3 : Stabilité asymptotique globale

Un état d'équilibre  $x_0$  du système donné par l'équation (C-1) est globalement asymptotiquement stable si pour tout  $x_0 \in \mathbb{R}^n$  :

1.  $x_0$  est stable
2.  $\lim_{t \rightarrow \infty} \|\phi(t, x_0, t_0) - x_0\| = 0$

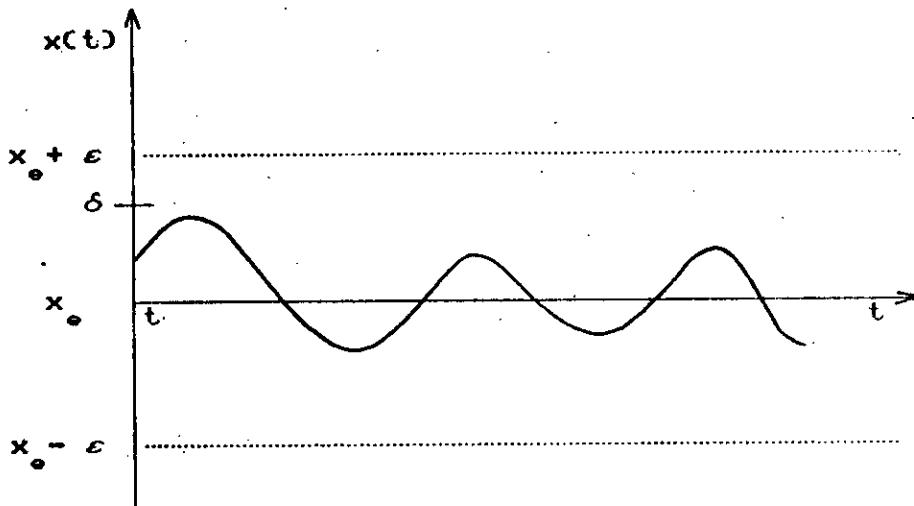


Fig C-1 Exemple d'évolution d'une variable d'état pour un système stable au sens de Lyapunov.

ANNEXE D

HYPERSTABILITE

Soit un système à CR de la forme suivante :

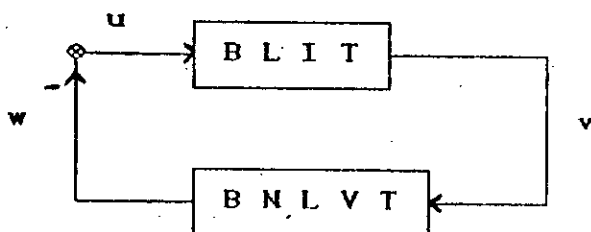


Fig D-1 Système en boucle fermée

BLIT : bloc linéaire invariant dans le temps.

BNLVT : bloc non linéaire variant dans le temps.

et soit l'inégalité intégrale de Popov donnée ci-dessous :

$$\int_0^{t_1} w^T v dt \geq -\gamma_0^2 \quad \forall t_1 > 0 \quad (D-1)$$

$\gamma_0^2$  constante finie positive.

Un système de la forme ci-dessus est dit (asymptotiquement) hyperstable lorsqu'il est (asymptotiquement) stable globalement pour tous les blocs de CR satisfaisant l'inégalité (D-1).

Puisque le bloc de CR est défini par la relation (D-1), les propriétés d'hyperstabilité du système à CR dépendent uniquement des caractéristiques du bloc direct. Un bloc direct qui garantit que le système en boucle fermée considéré est hyperstable est appelé bloc hyperstable.

Considérons le système en boucle fermée de la figure (D-1) ayant un bloc direct représenté par :

$$\begin{cases} \dot{x} = Ax + Bu = Ax - Bw & (D-2) \\ v = Cx + Ju = Cx - Jw & (D-3) \end{cases}$$

où  $x$  est un vecteur de dimension  $n$  et  $u, v, w$  des vecteurs de dimension  $m$ . Le système (D-2), (D-3) est supposé complètement commandable et observable.

Définition 1 :

Le bloc direct défini par les équations (D-2), (D-3) est hyperstable s'il existe une constante  $\delta > 0$  et une constante  $\gamma \geq 0$ , telles que toutes les solutions du système (D-2), (D-3) vérifient l'inégalité :

$$\|x(t)\| < \delta \|x(0)\| + \gamma, \quad \forall t > 0 \quad (D-4)$$

pour tous les blocs de CR vérifiant l'inégalité (D-1).

Définition 2 :

Le bloc direct (D-2), (D-3) est asymptotiquement hyperstable s'il est hyperstable et si

$$\lim_{t \rightarrow \infty} x(t) = 0 \quad (D-5)$$

pour tous les blocs de CR vérifiant l'inégalité (D-1).

La théorie de l'hyperstabilité développée par Popov étudie les conditions nécessaires et suffisantes d'hyperstabilité.

Théorème 1 :

Le bloc linéaire (D-2), (D-3) est hyperstable dans le sens de la définition 1 si et seulement si sa matrice de transfert

$$H(s) = C(sI - A)^{-1} B + J \quad (D-6)$$

est réelle positive.

Théorème 2 :

Le bloc linéaire (D-2), (D-3) est asymptotiquement hyperstable dans le sens de la définition 2, si et seulement si sa matrice de transfert, donnée par l'équation (D-6), est strictement réelle positive.

SOLUTIONS DE L'INEGALITE INTEGRALE DE POPOV

La synthèse des systèmes adaptatifs avec MR, à l'aide de la théorie de l'hyperstabilité, nous conduit à chercher les solutions d'une inégalité intégrale. Ce problème apparaît à la deuxième étape de la méthode lorsqu'il faut synthétiser la partie non linéaire du système équivalent à contre-réaction.

Soit un bloc de CR ayant une entrée  $v$  et une sortie  $w$ , où  $v$  et  $w$  sont des vecteurs réels de dimension  $m$ ,  $v$  étant une fonction vectorielle continue par morceaux.

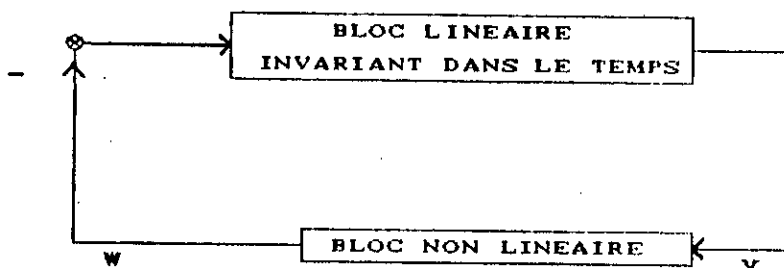


Fig E-1 Système à CR équivalent

L'inégalité intégrale de Popov :

$$\int_{t_0}^{t_1} w^T(t) v(t) dt \geq -\gamma_0^2, \quad \forall t_1 \geq t_0 \quad (E-1)$$

a été solutionnée par Landau [16][17] pour des blocs de CR caractérisés par des relations entrée-sortie de la forme :

$$w(t) = \left( \int_0^t \phi_1(v, t, \tau) d\tau + \phi_2(v, t) + K_1 \right) x(t) \quad (E-2)$$

où  $\phi_1$ ,  $\phi_2$  sont des fonctionnelles matricielles de  $v$ ,

$K_1$  est une matrice constante,

$\phi_1$ ,  $\phi_2$ ,  $K_1$  sont de dimension  $m \times n$  et  $x(t)$  est une fonction



vectorielle continue par morceaux de dimension  $n$ .

Ces solutions se présentent sous la forme

$$\phi_1(v(\tau), t) = F(t-\tau)v(\tau)(Gx(\tau))^T \quad (E-3)$$

$$\forall \tau < t$$

$$\phi_2(v(t), t) = \tilde{F}(t)v(t)(\tilde{G}(t)x(t))^T \quad (E-4)$$

où  $F(t-\tau)$  est un noyau matriciel défini positif dont la transformée de Laplace est une matrice de transfert réelle positive ayant un pôle nul.

$G$  est une matrice constante définie positive.

$\tilde{F}(t)$ ,  $\tilde{G}(t)$  sont des matrices variables définies positives ou semi-définies positives.

Une forme particulière de  $\phi_1$  et  $\phi_2$  conduit à la loi d'adaptation "proportionnelle + intégrale". Elle correspond à

$$F(t-\tau) = F \quad \forall (t-\tau) \quad (E-5)$$

$$\tilde{F}(t) = \tilde{F} \quad (E-6)$$

$$\tilde{G}(t) = G \quad (E-7)$$

REMARQUE :

Pour

$$\tilde{G} = G = I \quad (E-8)$$

$$\tilde{F} = \beta \quad (E-9)$$

$$F = \alpha \quad (E-10)$$

et  $K_1 = 0 \quad (E-11)$

on obtient :

$$\int_0^t \phi_1(v, t, \tau) d\tau + \phi_2(v, t) + K_1 = \int_0^t \alpha v(\tau) x^T(\tau) d\tau + \beta v(t) x^T(t) \quad (E-12)$$

Les gains de la loi de commande de l'algorithme MRAC (cf. page 25) ont la forme ci-dessus.

ANNEXE F

FILTRAGE DE KALMAN

Filtrage linéaire :

Soit le système décrit par le modèle linéaire :

$$\dot{x}(t) = F(t) x(t) + B(t) u(t) + G(t) w(t) \quad (F-1)$$

et soient les observations :

$$z(t) = H(t) x(t) + v(t) \quad (F-2)$$

où  $w$  et  $v$  sont des bruits blancs gaussiens centrés tels que

$$E \langle w(t) w^T(t+\tau) \rangle = Q(t) \delta(\tau) \quad (F-3)$$

$$E \langle v(t) v^T(t+\tau) \rangle = R(t) \delta(\tau) \quad (F-4)$$

La condition initiale  $x(t_0)$  est supposée gaussienne de moyenne  $\hat{x}_0$  et de covariance  $P_0$ .

Connaissant les caractéristiques des bruits et le modèle du système et des observations, le filtre de Kalman donne un estimé  $\hat{x}(t)$  optimal (au sens de la variance minimale) de l'état  $x(t)$ . Les équations du filtre sont :

$$\dot{\hat{x}}(t) = F(t) \hat{x}(t) + B(t) u(t) + K(t) \left[ z(t) - H(t) \hat{x}(t) \right] \quad (F-5)$$

$$K(t) = P(t) H^T(t) R^{-1}(t) \quad (F-6)$$

$$\begin{aligned} \dot{P}(t) = & F(t) P(t) + P(t) F^T(t) + G(t) Q(t) G^T(t) \\ & - P(t) H^T(t) R^{-1}(t) H(t) P(t) \end{aligned} \quad (F-7)$$

avec les conditions initiales  $\hat{x}(t_0) = \hat{x}_0$  et  $P(t_0) = P_0$ .  
 $K(t)$  est le gain du filtre de Kalman et  $P(t)$  est la (matrice

de) covariance de l'erreur d'estimation.

Les équations (F-5) et (F-6) montrent que :

- si  $P(t)$  est grand (i.e. a de grandes valeurs propres) alors la différence entre la mesure réelle  $z(t)$  et son estimé  $H(t)\hat{x}(t)$  est fortement pondérée. Autrement dit, si l'estimé  $\hat{x}(t)$  est très incertain alors le filtre accorde une grande importance à l'information apportée par les mesures.
- si  $R(t)$  est faible, c'est-à-dire si les mesures sont très précises alors l'information fournie par les mesures est fortement pondérée.

Dans l'expression de la covariance (F-7) :

- les deux premiers termes indiquent l'effet du système homogène (généralement stabilisant).
- le troisième terme reflète l'effet d'augmentation de la covariance dû au bruit  $w$ .
- le quatrième terme représente l'effet de diminution de la covariance dû à l'incorporation de l'information apportée par les mesures.

L'équation (F-7) est une équation différentielle matricielle de Riccati.

Lorsque le système considéré peut être décrit par un modèle linéaire (équation (F-1)), et lorsque les bruits  $v$  et  $w$  sont blancs et gaussiens, le filtre de Kalman est le meilleur estimateur de l'état  $x$ . Il est optimal pour la classe des estimateurs à variance minimale.

Si les bruits ne sont pas gaussiens, le filtre de Kalman est l'estimateur optimal pour la classe des estimateurs linéaires à variance minimale.

#### Filtrage non linéaire :

Soit le système décrit par le modèle non linéaire :

$$\dot{x}(t) = f \left[ x(t), u(t), t \right] + G(t) w(t) \quad (F-8)$$

$$z(t) = h[x, t] + v(t) \quad (\text{F-9})$$

où  $w$  et  $v$  sont des bruits blancs gaussiens centrés, non corrélés caractérisés par (F-3) et (F-4).

Le filtre de Kalman peut être étendu au système non linéaire (F-8), (F-9) en utilisant les développements en séries de Taylor de  $f$  et de  $h$  autour de  $\hat{x}(t)$  et à l'ordre 1. L'idée de base du filtre de Kalman étendu est de linéariser les équations par rapport à l'estimé courant  $\hat{x}(t)$  et d'utiliser les résultats du filtrage linéaire.

Le filtre de Kalman étendu est donné par :

$$\dot{\hat{x}}(t) = f[\hat{x}(t), u(t), t] + K(t) (z(t) - h[\hat{x}(t), t]) \quad (\text{F-10})$$

où le gain est :

$$K(t) = P(t) H^T [t, \hat{x}(t)] R^{-1}(t) \quad (\text{F-11})$$

et  $P$  satisfait :

$$\begin{aligned} \dot{P}(t) = & F [t, \hat{x}(t)] P(t) + P(t) F^T [t, \hat{x}(t)] \\ & + G(t) Q(t) G^T(t) - P(t) H^T [t, \hat{x}(t)] R^{-1}(t) H [t, \hat{x}(t)] P(t) \end{aligned} \quad (\text{F-12})$$

où  $F [t, \hat{x}(t)]$  et  $H [t, \hat{x}(t)]$  sont les dérivées partielles de  $f$  et de  $h$  par rapport à  $x$ , évaluées à  $x = \hat{x}(t)$ .

$$F [t, \hat{x}(t)] = \frac{\partial f(x, u, t)}{\partial x} \Big|_{x = \hat{x}(t)} \quad (\text{F-13})$$

$$H [t, \hat{x}(t)] = \frac{\partial h(x, t)}{\partial x} \Big|_{x = \hat{x}(t)} \quad (\text{F-14})$$