

**ABDER REZAK BENASKEUR**

**Aspects de l'application du backstepping adaptatif à la commande  
décentralisée des systèmes non linéaires**

Thèse  
présentée  
à la Faculté des Études Supérieures  
de l'Université Laval  
pour l'obtention  
du grade de Philosophiæ Doctor (Ph.D.)

Département de Génie Électrique et de Génie Informatique  
FACULTÉ DES SCIENCES ET DE GÉNIE  
UNIVERSITÉ LAVAL  
QUÉBEC

FÉVRIER 2000



National Library  
of Canada

Acquisitions and  
Bibliographic Services

395 Wellington Street  
Ottawa ON K1A 0N4  
Canada

Bibliothèque nationale  
du Canada

Acquisitions et  
services bibliographiques

395, rue Wellington  
Ottawa ON K1A 0N4  
Canada

*Your file Votre référence*

*Our file Notre référence*

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-48967-1

Canada

# Résumé court

La faiblesse de l'équivalence certaine explique le manque de résultats que connaît, dans la pratique, la commande adaptative. Cette faiblesse réside dans le fait que la dynamique de l'estimateur n'est pas considérée dans la procédure de design. La lenteur de la convergence de l'adaptation peut entraîner l'instabilité, notamment en présence de non-linéarités et/ou couplages. Dans de telles situations, la commande adaptative par Lyapunov, à travers le **backstepping**, constitue une alternative de choix. Dans ce projet, le **backstepping** a été modifié, afin de le rendre applicable aux procédés industriels. Ces modifications concernent, essentiellement, le comportement de la commande, les erreurs résiduelles, la présence des retards, les déphasages non minimaux, ainsi que le couplage. Pour ce dernier, une nouvelle approche, pour la commande décentralisée, est proposée. La méthode de l'erreur modifiée consiste à fragmenter, en un ensemble de contrôleurs locaux, l'unique contrôleur multivariable obtenu par le **backstepping**. Cette subdivision est rendue possible par la modification des erreurs régulées dans la procédure récursive. La stabilité et les conditions de la décentralisation sont garanties par une fonction de Lyapunov.

# Résumé long

La faiblesse des régulateurs adaptatifs basés sur l'équivalence certaine, qui sont ceux couramment employés, explique en partie, le manque de résultats que connaît, dans la pratique, la commande adaptative. Cette faiblesse réside, essentiellement, dans le fait que l'estimateur et le régulateur sont des entités indépendantes. Dans une telle approche, la dynamique de l'estimateur n'est pas considérée dans la procédure de conception. La lenteur de la convergence de l'adaptation peut entraîner, dans certains cas, une instabilisation irréversible de la boucle, notamment en présence de non-linéarités et/ou couplages. Dans la pratique, la nécessité de disposer d'outils plus élaborés, pour prouver la stabilité des boucles et la convergence des estimateurs, a longtemps freiné l'application de la commande adaptative, basée sur l'équivalence certaine, aux procédés industriels. C'est pourquoi, la commande adaptative par Lyapunov, de par ses propriétés de stabilité, constitue une alternative de choix. La méthode du backstepping offre, pour cette dernière, un outil de design très efficace qui permet de construire récursivement, d'une manière systématique et directe, la loi de commande, la dynamique d'adaptation et la fonction de Lyapunov qui assurent la stabilité de la boucle. Le backstepping a été modifié, dans ce projet, afin de le rendre applicable aux procédés industriels. Ces modifications concernent, essentiellement, le comportement de la commande, les erreurs résiduelles, la présence des retards, les déphasages non minimaux, ainsi que le couplage. Pour ce dernier, une nouvelle approche, pour la commande décentralisée, est proposée. L'idée principale de la méthode de l'erreur modifiée (MEM) consiste à trouver des conditions suffisantes, sous lesquelles, l'unique contrôleur multivariable, obtenu par le backstepping, peut être fragmenté en un ensemble de contrôleurs locaux. Cela passe par l'élimination, de la loi de commande multivariable, des composantes croisées. Cette subdivision, qui doit s'effectuer sans altérer la stabilité et les performances de la

boucle, est rendue possible par la modification des erreurs régulées dans la procédure du **backstepping**. Chacun des contrôleurs locaux obtenus dépend, alors, uniquement de l'information qui lui est disponible localement. La stabilité du système global est garantie par une fonction de **Lyapunov**. La **MEM** dispose de plusieurs variantes, qui vont de la commande par retour d'état complet à la commande par retour de sortie, en passant par un retour d'état partiel. Une version avec observateur est également proposée.

# Avant-propos

Mes remerciements s'adressent, en premier lieu, à mon Directeur de recherche, M. André Desbiens pour son soutien, ses encouragements et ses idées qui ont été très appréciées tout au long de cette recherche.

Je tiens, également, à remercier M. Michel Perrier, professeur au département de génie chimique à l'École Polytechnique et MM. André Pomerleau et Huynh Huu Tuê, professeurs au département de génie électrique et génie informatique à l'Université Laval, pour avoir accepté d'être membres du jury.

Sur le plan personnel, je voudrais remercier mon épouse Hakima qui a su m'endurer pendant ces trois longues années de recherche et de rédaction.

# Table des matières

<b>Résumé court</b>	<b>i</b>
<b>Résumé long</b>	<b>ii</b>
<b>Avant-propos</b>	<b>iv</b>
<b>Table des matières</b>	<b>v</b>
<b>Table des figures</b>	<b>x</b>
<b>Table des symboles</b>	<b>xiii</b>
<b>I Introduction générale</b>	<b>1</b>
<b>Introduction générale</b>	<b>2</b>
<b>II Aspects théoriques de la commande non linéaire</b>	<b>9</b>
<b>Introduction</b>	<b>10</b>
<b>1 Systèmes non linéaires</b>	<b>11</b>
1.1 Définitions . . . . .	12
1.2 Méthodes d'analyse . . . . .	16
1.2.1 Première méthode de Lyapunov . . . . .	17
1.2.2 Deuxième méthode de Lyapunov . . . . .	18

1.3	Conclusion . . . . .	22
<b>2</b>	<b>Commande des systèmes non linéaires</b>	<b>23</b>
2.1	Commande par linéarisation . . . . .	24
2.1.1	Linéarisation locale . . . . .	24
2.1.2	Linéarisation exacte . . . . .	26
2.1.3	Limitations de la linéarisation . . . . .	28
2.2	Commande par Lyapunov . . . . .	29
2.2.1	Application au design . . . . .	30
2.2.2	Choix de la fonction de Lyapunov . . . . .	32
2.3	Commande robuste par Lyapunov . . . . .	33
2.3.1	Stabilisation robuste . . . . .	33
2.3.2	Amortissement non linéaire . . . . .	35
2.4	Commande adaptative par Lyapunov . . . . .	37
2.4.1	Commande adaptative par modèle de référence . . . . .	39
2.4.2	Principe de l'équivalence certaine . . . . .	39
2.4.3	Commande adaptative basée sur Lyapunov . . . . .	40
2.5	Conclusion . . . . .	44
<b>3</b>	<b>Design par backstepping</b>	<b>45</b>
3.1	Algorithme de base . . . . .	48
3.2	Système d'ordre $n$ . . . . .	56
3.3	Backstepping Robuste . . . . .	57
3.3.1	Principe . . . . .	57
3.3.2	Applications . . . . .	62
3.4	Backstepping avec observateur . . . . .	62
3.4.1	Applicabilité . . . . .	63
3.4.2	Principe . . . . .	65
3.5	Conclusion . . . . .	69
<b>4</b>	<b>Backstepping Adaptatif</b>	<b>70</b>
4.1	Applicabilité . . . . .	70
4.2	Principe . . . . .	71



4.3	Cas général . . . . .	75
4.4	Limitations . . . . .	81
4.5	Méthode des fonctions de réglage . . . . .	81
4.5.1	Extended matching condition . . . . .	82
4.5.2	Cas général . . . . .	85
4.5.3	Système d'ordre $n$ . . . . .	91
4.5.4	Robustification . . . . .	92
4.6	Vitesse d'adaptation . . . . .	95
4.7	Backstepping modulaire . . . . .	95
4.8	Backstepping avec observateur adaptatif . . . . .	97
4.9	Conclusion . . . . .	102
	<b>Conclusion</b>	<b>104</b>
	<b>III Aspects pratiques du backstepping</b>	<b>105</b>
	<b>Introduction</b>	<b>106</b>
<b>5</b>	<b>Bruit de mesure</b>	<b>108</b>
5.1	Design récursif . . . . .	109
5.2	Observateur . . . . .	111
5.3	Filtrage par Lyapunov . . . . .	116
5.4	Conclusion . . . . .	120
<b>6</b>	<b>Perturbations à moyenne non nulle</b>	<b>124</b>
6.1	Backstepping avec action intégrale . . . . .	125
6.2	Filtrage par Lyapunov . . . . .	129
6.3	Version adaptative . . . . .	132
6.4	Amortissement non linéaire . . . . .	135
6.5	Simulation . . . . .	137
6.6	Équivalence certaine . . . . .	148
6.7	Conclusion . . . . .	148

<b>7 Déphasage non minimal/retard</b>	<b>151</b>
7.1 Principe . . . . .	151
7.2 Simulations . . . . .	152
7.3 Limitations . . . . .	155
7.4 Conclusion . . . . .	158
<b>Conclusion</b>	<b>159</b>
<b>IV Application à la commande décentralisée</b>	<b>160</b>
<b>Introduction</b>	<b>161</b>
<b>8 Commande décentralisée</b>	<b>162</b>
8.1 Principe . . . . .	164
8.2 Commande adaptative décentralisée . . . . .	166
8.3 Commande à gain fort . . . . .	167
8.4 Commande adaptative robuste . . . . .	168
8.5 Backstepping . . . . .	171
8.6 Conclusion . . . . .	173
<b>9 Méthode de l'erreur modifiée</b>	<b>175</b>
9.1 Design récursif . . . . .	177
9.2 Commande multivariable . . . . .	183
9.3 Décentralisation . . . . .	187
9.4 Application . . . . .	190
9.5 Extensions . . . . .	191
9.5.1 Approche adaptative . . . . .	191
9.5.2 Systèmes non linéaires . . . . .	198
9.6 Conclusion . . . . .	206
<b>10 Design avec observateur</b>	<b>208</b>
10.1 MEM avec observateur . . . . .	209
10.2 Décentralisation . . . . .	216

10.3 Simulation . . . . .	219
10.4 Conclusion . . . . .	219
<b>11 Commande par retour de sortie</b>	<b>223</b>
11.1 Modification directe . . . . .	223
11.2 Commande par retour d'état partiel . . . . .	227
11.3 Design avec action intégrale . . . . .	235
11.3.1 Commande par retour d'état . . . . .	235
11.3.2 Version adaptative . . . . .	238
11.3.3 Commande par retour de sortie . . . . .	247
11.4 Conclusion . . . . .	249
<b>Conclusion</b>	<b>252</b>
<b>V Conclusion générale</b>	<b>254</b>
<b>Conclusion générale</b>	<b>255</b>
<b>VI Annexes</b>	<b>258</b>
<b>A Commande adaptative robuste</b>	<b>259</b>
<b>B Définitions</b>	<b>266</b>
<b>Bibliographie</b>	<b>268</b>

# Table des figures

1.1	Trajectoire d'un système dans le plan de phase . . . . .	14
1.2	Stabilité simple selon <b>Lyapunov</b> . . . . .	15
1.3	Stabilité asymptotique selon <b>Lyapunov</b> . . . . .	15
1.4	Contours à énergie constante dans le plan de phase . . . . .	19
1.5	Stabilité par la fonction de <b>Lyapunov</b> . . . . .	21
2.1	Commande adaptative indirecte . . . . .	38
2.2	Commande adaptative directe . . . . .	38
2.3	Commande par Modèle de Référence . . . . .	39
2.4	Commande adaptative basée sur <b>Lyapunov</b> . . . . .	41
3.1	<b>Backstepping</b> vs Linéarisation . . . . .	55
5.1	Bruit de mesure : <b>backstepping</b> simple . . . . .	112
5.2	Bruit de mesure : <b>backstepping</b> avec observateur . . . . .	117
5.3	Bruit de mesure : <b>backstepping</b> avec filtre . . . . .	121
5.4	Perturbation en échelon : <b>backstepping</b> . . . . .	122
6.1	Perturbation en échelon : <b>backstepping</b> avec vs sans intégrateur . . . . .	130
6.2	<b>Backstepping</b> avec intégrateur : avec vs sans filtre . . . . .	133
6.3	<b>Backstepping</b> simple vs intégrateur : commandes $u$ . . . . .	138
6.4	<b>Backstepping</b> simple vs intégrateur : sorties $y$ . . . . .	139
6.5	<b>Backstepping</b> simple vs intégrateur : estimé de $a$ . . . . .	140
6.6	<b>Backstepping</b> simple vs intégrateur : estimé de $b$ . . . . .	141
6.7	<b>Backstepping</b> simple vs intégrateur : estimé de $c$ . . . . .	142

6.8	Perturbation à la sortie ( <b>backstepping simple vs intégrateur</b> ) : $u$ . . . . .	143
6.9	Perturbation à la sortie ( <b>backstepping simple vs intégrateur</b> ) : $y$ . . . . .	144
6.10	Perturbation à la sortie ( <b>backstepping simple vs intégrateur</b> ) : $a$ . . . . .	145
6.11	Perturbation à la sortie ( <b>backstepping simple vs intégrateur</b> ) : $b$ . . . . .	146
6.12	Perturbation à la sortie ( <b>backstepping simple vs intégrateur</b> ) : $c$ . . . . .	147
6.13	PID adaptatif : <b>backstepping vs équivalence certaine</b> . . . . .	149
6.14	PID adaptatif : <b>backstepping vs équivalence certaine</b> . . . . .	150
7.1	Prédicteur de <b>Smith</b> . . . . .	152
7.2	Prédicteur de <b>Smith</b> modifié . . . . .	153
7.3	Prédicteur de <b>Smith</b> : retard/phase non minimale . . . . .	154
7.4	Prédicteur de <b>Smith</b> : perturbation en échelon . . . . .	156
7.5	Prédicteur de <b>Smith</b> : bruit blanc . . . . .	157
8.1	Procédé à deux entrées et deux sorties . . . . .	166
9.1	Systèmes connectés . . . . .	185
9.2	Retour d'état complet — Sorties . . . . .	192
9.3	Retour d'état complet — Commandes . . . . .	193
9.4	Approche adaptative — Sorties . . . . .	199
9.5	Approche adaptative — Commandes . . . . .	200
9.6	Approche adaptative — Paramètres de $G_1$ . . . . .	201
9.7	Approche adaptative — Paramètres de $G_2$ . . . . .	202
9.8	Approche adaptative — Paramètres $c_1$ & $c_2$ . . . . .	203
9.9	Procédé non linéaire deux-par-deux . . . . .	204
10.1	Retour d'état avec observateur — Sorties . . . . .	220
10.2	Retour d'état avec observateur — Commandes . . . . .	221
11.1	Retour de sortie sans intégrateur — Sorties . . . . .	228
11.2	Retour de sortie sans intégrateur — Commandes . . . . .	229
11.3	Retour d'état partiel — Sorties . . . . .	233
11.4	Retour d'état partiel — Commandes . . . . .	234
11.5	Retour d'état avec intégrateur — Sorties . . . . .	239

11.6	Retour d'état avec intégrateur — Commandes . . . . .	240
11.7	Approche adaptative avec intégrateur — Sorties . . . . .	242
11.8	Approche adaptative avec intégrateur — Commandes . . . . .	243
11.9	Approche adaptative avec intégrateur — Paramètres de $G_1$ . . . . .	244
11.10	Approche adaptative avec intégrateur — Paramètres de $G_2$ . . . . .	245
11.11	Approche adaptative avec intégrateur — Paramètres $c_1$ & $c_2$ . . . . .	246
11.12	Retour de sortie avec intégrateur — Sorties . . . . .	250
11.13	Retour de sortie avec intégrateur — Commandes . . . . .	251

# Table des symboles

$a_i$	Coefficients du polynôme $A_i(s)$
$A_i(s)$	Dénominateur de la fonction de transfert $G_i(s)$
$\mathbf{A}$	Matrice d'évolution du système
$\mathbf{A}_0$	Matrice d'évolution du système des erreurs d'observation
$b_i$	Coefficients du polynôme $B_i(s)$
$B_i(s)$	Numérateur de la fonction de transfert $G_i(s)$
$\mathbf{B}$	Matrice d'application de l'entrée
$\mathbf{C}$	Matrice d'observation de la sortie
$\mathcal{D}$	Domaine d'intérêt
$\partial$	Dérivée partielle
$e$	Erreur de poursuite
$\mathcal{F}(s)$	Polynôme caractéristique de la matrice $\mathbf{K}$
$g_0$	Seuil de la zone morte
$G_c(s)$	Fonction de transfert du contrôleur
$G_m(s)$	Fonction de transfert du modèle du procédé ou de référence
$G_p(s)$	Fonction de transfert du procédé
$d$	Paramètre de design pour le backstepping avec observateur
$h$	Paramètre de design pour l'observateur
$\mathbf{H}$	Matrice des gains de l'observateur
$\mathbf{I}$	Matrice identité
$k$	Paramètre de design pour le backstepping

$\mathbf{K}$	Matrice des gains $k_i$
$\mathcal{L}_p(\cdot)$	Norme $\  \cdot \ _p$
$m$	Paramètre de design pour l'amortissement non linéaire
$\mathcal{M}$	Filtre stable de gain unitaire et sans résonance
$\mathcal{N}$	Nombre de composantes dans un système couplé
$M_0$	Borne sur les paramètres inconnus
$\mathbf{P}$	Matrice définie positive
$Q_i(s)$	Quotient d'interaction d'un procédé couplé
$r$	Consigne
$\mathcal{R}$	Corps des nombres réels
$\mathcal{R}_+$	Corps des nombres réels positifs ou nuls
$s$	Variable de Laplace
$S_i$	$i^{ieme}$ composante d'un système couplé
$t$	Variable temps
$u$	Variable manipulée (Commande)
$v$	Commande filtrée
$\mathcal{V}$	Fonction de Lyapunov
$\mathcal{V}^\dagger$	Fonction de Lyapunov pour la commande adaptative
$\mathcal{W}$	Fonction définie positive
$w$	Régresseur non linéaire
$x$	Variable d'état
$\mathbf{x}$	Vecteur d'état
$\hat{x}$	Estimé de la variable d'état $x$
$\tilde{x}$	Erreur d'estimation de $x$
$\mathbf{x}_e$	Position d'équilibre
$y$	Variable de sortie
$\hat{y}$	Estimé de la sortie $y$
$\tilde{y}$	Erreur d'estimation de $y$



$y_r$	Trajectoire de référence
$z$	Estimé de la partie de l'état indépendante des paramètres inconnus
$\mathbf{z}$	Vecteur des estimés $z$

## Caractères grecs

$\alpha$	Commande virtuelle dans le <b>backstepping</b>
$\chi$	Terme supplémentaire introduit dans la méthode de l'erreur modifiée
$\delta$	Incertitude
$\delta_0$	Borne supérieure sur la norme des incertitudes $\delta$
$\Delta$	Fonction non linéaire connue
$\varepsilon$	Variable d'erreur dans le <b>backstepping</b>
$\boldsymbol{\varepsilon}$	Vecteur d'erreur dans le <b>backstepping</b>
$\eta$	Fonction non linéaire connue
$\gamma$	Gain d'adaptation
$\Gamma$	Matrice des gains d'adaptation
$\kappa$	Fonction de classe $\mathcal{K}$ ou $\mathcal{K}_\infty$
$\mu$	Signal de normalisation pour les lois d'adaptation
$\varphi$	Fonction non linéaire connue
$\psi$	Fonction non linéaire connue
$\rho$	Degré relatif du procédé
$\rho$	Coefficients du polynôme caractéristique $\mathcal{F}$
$\zeta$	Amortissement non linéaire
$\sigma$	Terme constant de robustification des lois d'adaptation
$\sigma_s$	Terme variable de robustification des lois d'adaptation
$\tau$	Fonction de réglage
$\vartheta$	Paramètre connu du procédé
$\boldsymbol{\vartheta}$	Vecteur des paramètres connus
$\theta$	Paramètre inconnu du procédé
$\boldsymbol{\theta}$	Vecteur des paramètres inconnus

$\theta_p$	Paramètre du procédé
$\theta_c$	Paramètre du contrôleur
$\hat{\theta}$	Estimé du paramètre inconnu $\theta$
$\hat{\theta}$	Vecteur des estimés
$\bar{\theta}$	Erreur d'estimation du paramètre $\theta$
$\bar{\theta}$	Vecteur des erreurs d'estimation des paramètres
$\xi$	Partie déterministe d'une fonction contenant des incertitudes
$\zeta$	Estimé de la partie de l'état dépendant des paramètres inconnus
$\zeta$	Vecteur des estimés $\zeta$

**Première partie**  
**Introduction générale**

# Introduction générale

La commande des systèmes constitue l'un des domaines les plus riches en termes d'algorithmes, d'outils d'analyse et de techniques de design. Toutefois, toutes ces techniques sont basées sur la connaissance des paramètres du système à contrôler. Du degré de validité de cette hypothèse, dépend la qualité du comportement du système compensé obtenu. Si les paramètres du procédé sont connus avec exactitude<sup>1</sup> un contrôleur bien conçu donne, en général, de bons résultats. En présence d'incertitudes ou de paramètres inconnus, rien ne garantit un fonctionnement qui respecte les spécifications.

Le problème de la commande des systèmes sujets à de faibles incertitudes, peut être traité dans le cadre de la commande robuste (Morari & Zafiriou 1989, Green & Limebeer 1996, Skogestad & Postlethwaite 1996). En présence de fortes perturbations, incertitudes du modèle ou paramètres à forte variation dans le temps, les structures auto-régulantes présentent, souvent, un meilleur choix. Ces structures, communément connues sous le nom de "commande adaptative" (Landau 1979, Åström & Wittenmark 1995, Krstić et al. 1995, Ioannou & Sun 1996), connaissent un grand succès depuis plusieurs années. L'intérêt croissant, porté à cette approche, s'explique, surtout, par sa large applicabilité (procédés industriels, armement, robotique, instrumentation biomédicale, ... ). Cette propriété est attribuable à l'omniprésence des incertitudes, inhérentes à tout système physique, d'une part et les niveaux de fiabilité et/ou de performance, de plus en plus élevés, exigés des systèmes commandés, d'autre part. Ces différents facteurs ne cessent de gagner de l'importance avec la complexité croissante des systèmes considérés.

L'approche par l'équivalence certaine constituait, jusqu'à très récemment, l'essentiel

---

<sup>1</sup>Ce qui est rarement le cas.

des méthodes de commande adaptatives proposées dans la littérature. Dans la pratique, la faiblesse des régulateurs basés sur ce principe de l'équivalence certaine<sup>2</sup> limite l'application de la commande adaptative aux seuls procédés linéaires et monovariables. Cette faiblesse réside, essentiellement, dans le fait que l'estimateur et le régulateur sont des entités indépendantes. Par conséquent, la dynamique de l'estimateur, intrinsèquement non linéaire, n'est pas considérée dans la procédure de conception. La lenteur (relative) de la convergence de l'adaptation peut entraîner, dans certains cas, une instabilisation irréversible de la boucle. Dans le cas des systèmes non linéaires, ces effets peuvent être catastrophiques, à cause du "finite escape time" qui les caractérise. La nécessité de disposer d'outils plus élaborés, pour prouver la stabilité des boucles et la convergence des estimateurs, a longtemps freiné l'application de la commande adaptative aux procédés non linéaires.

Les contrôleurs basés sur la méthode directe de Lyapunov (chapitre 1) présentent une meilleure alternative à l'équivalence certaine. L'idée principale de l'utilisation de la fonction de Lyapunov dans la commande adaptative, consiste à calculer une loi de commande et une loi de mise à jour des paramètres, afin de garantir que la dérivée d'une certaine fonction définie positive, et bien choisie, est non positive. L'approche consiste donc à trouver un triplet (fonction de Lyapunov, loi de commande, loi d'adaptation) qui répond aux spécifications et tient compte de la dynamique de l'estimation. Le backstepping (chapitre 3 et 4) est l'algorithme qui a rendu cette approche applicable à une large classe de systèmes, indépendamment de leur ordre. C'est une méthode qui permet de construire récursivement, d'une manière systématique et directe, la loi de commande, la dynamique d'adaptation et la fonction de Lyapunov qui assurent la stabilité de l'ensemble.

## Problèmes

De par ces propriétés de stabilité, il est clair que la commande adaptative par la méthode de Lyapunov, apporte, du moins sur le plan théorique, une réponse aux problèmes que posent la lenteur de l'adaptation et la présence des non-linéarités. Dans

---

<sup>2</sup>Qui sont ceux couramment employés

la pratique, le **backstepping** pose certains problèmes qui limitent son applicabilité.

- En présence de bruit de mesure, le contrôleur par retour d'état résultant du **backstepping**, provoque un comportement très peu désirable de la variable manipulée. Ce dysfonctionnement s'explique par la présence d'une action dérivée et l'absence d'un filtre dans la branche de retour. L'ajout d'un filtre peut compromettre la stabilité de la boucle.
- La nature avance de phase du contrôleur obtenu par le **backstepping** ne lui permet pas, en présence de perturbations à moyenne non nulle à l'entrée<sup>3</sup> du procédé, d'éliminer les erreurs résiduelles.
- La présence, dans les procédés, des retards et/ou déphasages non minimaux est un autre phénomène qui rend le **backstepping**, dans sa forme originelle, inutilisable dans la pratique. D'une part, la nature retour d'état continu de la méthode exclue les retards, qui ne peuvent y être représentés. D'autre part, le **backstepping** étant une variante de la commande par **modèle de référence**, le procédé doit nécessairement être à phase minimale.
- Dans la pratique, les modèles scalaires sont incapables de fournir la précision suffisante garantissant le niveau de performance requis. Des modèles multivariables sont alors utilisés. Deux stratégies existent pour commander ce type de modèles. L'approche centralisée utilise une loi de commande multivariable, pour contrôler le système dans son intégralité. La deuxième approche remplace cette unique loi par un ensemble de contrôleurs locaux, assurant chacun la commande d'une partie du système. On parle de commande décentralisée. En plus de réduire la complexité du design, cette approche procure plus de flexibilité et de fiabilité aux boucles de contrôle que la méthode centralisée. À cause de ses propriétés fort intéressantes, l'approche décentralisée est très souvent adoptée comme stratégie de contrôle. Dans le contexte adaptatif, la présence de couplage dans les procédés, combiné à la lenteur de l'adaptation, peut entraîner

---

<sup>3</sup>Et à la sortie du procédé, si celui-ci ne contient pas d'intégrateur.

l'instabilité de la boucle de contrôle, rendant l'approche par équivalence certaine inappropriée pour de telles situations. Malheureusement, la méthode du **backstepping**, qui garantit de meilleures propriétés de stabilité, est fondamentalement de nature centralisée. Elle ne permet pas, dans sa version de base, la conception de lois de commande décentralisées.

## Objectif

L'objectif principal du projet de recherche, qui fait l'objet de cette thèse, est d'apporter les modifications nécessaires à la procédure récursive du **backstepping**, afin de répondre aux problèmes précédemment mentionnés et rendre la méthode applicable aux systèmes réels, en particulier aux procédés industriels. Le manque de résultats dans ce domaine, et surtout l'absence d'approches générales de design, ont été parmi nos principales motivations. Un aperçu des différentes solutions, qui seront abordées avec détail dans la troisième et quatrième parties de cette thèse, est présenté dans la section suivante.

## Solutions

- Afin d'améliorer le comportement de la variable manipulée, en présence de bruit de mesure, un nouveau type de filtre (chapitre 5) est inséré dans la branche dérivée du contrôleur. Ce filtre, tout en réduisant la variance de la commande, garantit des performances supérieures à celles données par la variante avec observateur du **backstepping** (chapitre 3). Son design, basé sur la théorie de Lyapunov, permet de préserver la stabilité de la boucle.
- Pour être en mesure d'éliminer les erreurs résiduelles, en présence de perturbations à moyenne non nulle, une action intégrale est introduite dans les contrôleurs générés par le **backstepping** (chapitre 6). Cette introduction nécessite une modification de la procédure de design. Outre ses performances en régime permanent, l'action intégrale assure de meilleures propriétés de convergence pour les estimés des paramètres inconnus.

- La solution proposée pour répondre aux problèmes, que pose la présence des retards et/ou déphasages non minimaux, consiste à combiner le **backstepping** avec le prédicteur de Smith (chapitre 7). Dans cette approche, les zéros dans le demi-plan de droite sont traités au même titre que les retards, i.e. sortis de la boucle interne.
- Le problème de l'application du **backstepping** à la commande décentralisée des systèmes couplés a bénéficié d'une attention particulière de notre part. Étant donné la nature centralisée de la méthode, il est clair que la conception de lois commande décentralisées passe par la modification de la procédure récursive du **backstepping**. Cette modification a donné lieu à une nouvelle approche, baptisée **Méthode de l'Erreur Modifiée (MEM)**, qui constitue la principale nouveauté de ce projet (Benaskeur & Desbiens 1999a,b,c,d,e). Le principe, ainsi que les différentes variantes, de la méthode sont brièvement présentés dans la section suivante.

### **Méthode de l'erreur modifiée**

L'idée principale de la **MEM** consiste, à partir de la méthode du **backstepping**, à concevoir un seul contrôleur multivariable, garantissant les performances requises du système global. Par la suite, des conditions sous lesquelles ce contrôleur centralisé peut être fragmenté en un ensemble de contrôleurs locaux, sans altérer la stabilité et les performances en boucle fermée, sont dérivées. La modification des erreurs régulées dans la procédure du **backstepping**, via l'introduction de termes supplémentaires appropriés, a rendu possible l'élimination des composantes croisées de la loi de commande multivariable. La stabilité est garantie par une version modifiée de la fonction de Lyapunov originelle, obtenue par le **backstepping**. La première version de la **MEM** (Benaskeur & Desbiens 1999c) a été élaborée sous l'hypothèse de la disponibilité à la mesure du vecteur d'état. Cette hypothèse, quand elle est réalisable, peut se traduire par des coûts exorbitants, limitant l'application de la méthode. Pour palier cette situation, une version à retour de sortie a été présentée dans (Benaskeur & Desbiens 1999d). Afin d'éviter la mesure des variables d'état, le contrôleur proposé combine la version à retour d'état avec la technique du **backstepping** à base d'observateur (Krstić & Kokotović 1994).



Toutefois, même pour un procédé scalaire, l'introduction d'un observateur nécessite la modification de toute la procédure de design. Une solution qui offre un bon compromis entre la complexité du design et la mesure des variables d'état est proposée dans (Benaskeur & Desbiens 1999a). En effet, sans accroître la complexité, ni modifier la procédure de design, la commande est, à toute fin pratique, réduite à un retour de sortie. Étant donné que certaines variables restent indispensables à l'implantation des contrôleurs obtenus, on parle de commande à retour d'état partiel. Cette amélioration est rendue possible grâce à un choix différent des variables d'erreur régulées par le backstepping. Sous certaines conditions additionnelles, et avec un choix différent des erreurs régulées, les mesures nécessaires se limitent aux sorties. Toutefois, dans le cadre général, ce choix donne lieu à des erreurs non nulles en régime permanent. Dans le but de surmonter cette limitation, i.e. assurer des erreurs nulles, une action intégrale est introduite dans chacun des régulateurs locaux. Avec cette importante modification, il est clair qu'un nouveau choix des variables régulées s'impose, garantissant à la fois la stabilité interne de la boucle et des erreurs résiduelles nulles.

## **Plan du document**

Cette thèse est subdivisée en six parties. Outre la première partie, qui se réduit au présent chapitre d'introduction générale, les cinq autres parties sont organisées comme suit.

**Aspects théoriques de la commande non linéaire (Partie II)** — Faisant l'objet des chapitres 1 à 4, cette partie consiste à explorer et rendre accessible un nouveau domaine de recherche, très actif, qui est la commande adaptative des systèmes non linéaires. Un intérêt particulier y est porté, via la technique du backstepping, à l'analyse et au design par la méthode de Lyapunov.

**Aspects pratiques du backstepping (Partie III)** — Cette partie de la thèse traite des différents aspects de l'utilisation du backstepping pour la commande de procédés réels. On y aborde, dans les chapitres 5 à 7, les problèmes des bruits de mesure, des perturbations à moyenne non nulle et des retards et/ou déphasages non minimaux.

Quelques modifications et solutions y sont proposées.

**Application à la commande décentralisée (Partie IV)** — Couvrant les chapitres 8 à 11, cette partie est entièrement dédiée la présentation de la **méthode de l'erreur modifiée**.

La cinquième partie vient conclure la présente thèse, alors que la sixième et dernière partie contient les annexes.

## **Deuxième partie**

### **Aspects théoriques de la commande non linéaire**

# Introduction

Cette partie est consacrée à la présentation des différents aspects théoriques de la commande adaptative des systèmes non linéaires. Étant donné leur importance pour les développements subséquents, la théorie de Lyapunov, et la méthode du **backstepping** en particulier, bénéficieront de beaucoup d'attention.

Le problème crucial des systèmes non linéaires est introduit dans le **chapitre 1**, où sont également données quelques définitions relatives à la stabilité. Ces dernières permettront d'aborder la question de l'analyse des systèmes dynamiques. Parmi les techniques présentées, l'accent sera particulièrement mis sur la **méthode directe de Lyapunov**. Celle-ci constitue un élément indispensable à la compréhension des chapitres subséquents. Le **chapitre 2** traite des problèmes de la commande des systèmes non linéaires. Les méthodes basées sur les fonctions de Lyapunov y sont présentées sous tous les angles. Dans ce chapitre sont également abordés, en plus des méthodes par linéarisation, les problèmes de la commande par Lyapunov en présence des incertitudes paramétriques (commande adaptative par Lyapunov) et/ou non paramétriques (commande robuste par Lyapunov). Ce chapitre constitue une bonne introduction aux chapitres **3** et **4**, entièrement consacrés à la méthode du **backstepping**. Le **chapitre 3** traite le cas du **backstepping** non adaptatif, alors que le **chapitre 4** présente la version adaptative de la méthode.

# Chapitre 1

## Systemes non linéaires

Le souci constant d'améliorer les performances des systèmes commandés conduit à des modélisations de plus en plus précises ; mais s'il y a un modèle qui rend compte du comportement d'un système dans une large plage de fonctionnement, il est malheureusement le plus souvent non linéaire. De ce fait, les outils d'analyse et/ou de synthèse de lois de commande, utilisés dans le domaine linéaire, deviennent caduques et absolument incapables de rendre compte de certains phénomènes dont les systèmes non linéaires sont le siège. D'autres méthodes, plus élaborées, deviennent alors nécessaires. Quelques exemples de ces phénomènes sont donnés (Khalil 1996) :

- Un système linéaire possède un seul point d'équilibre. Un système non linéaire peut en avoir plusieurs. Le système peut converger, en régime permanent, à l'un de ces points, suivant les conditions initiales (point d'opération).
- L'état d'un système linéaire instable peut prendre des valeurs infinies quand le temps devient infini. Un système non linéaire peut voir son état prendre des valeurs infinies pour des valeurs finies du temps (finite escape time). Cette propriété peut être à l'origine de sérieux problèmes de stabilité, notamment en commande adaptative.
- Pour qu'un système linéaire présente des oscillations entretenues, il faut qu'il ait une paire de pôles imaginaires. Il est presque impossible de maintenir de telles oscillations en présence de perturbations. Dans la pratique, les oscillations stables (d'amplitude et

fréquence fixes) doivent être réalisées par des systèmes non linéaires. Ce type d'oscillations est connu sous le nom de **cycles limites**.

- Un système linéaire, sous l'effet d'une entrée périodique, produit un signal périodique de même période à sa sortie. Un système non linéaire peut osciller avec des fréquences qui sont des sous-multiples et/ou des multiples de la fréquence d'entrée.
- Un système non linéaire peut avoir des régimes permanents plus compliqués que ceux cités précédemment. Ces régimes sont connus sous le nom de **chaos**. Ils présentent un aspect stochastique.

Contrairement au cas linéaire, les systèmes non linéaires, étant définis par négation, ne représentent pas un ensemble homogène. Une méthode qui donne de bons résultats sur une classe de systèmes, peut avoir des conséquences catastrophiques sur une autre classe, même très peu différente de la première (Gille 1988). Par conséquent, dans le cadre de la commande des systèmes non linéaires, la question de l'**applicabilité** est primordiale, et une méthode générale n'existe pas. Tous les efforts consentis ces dernières années visent à élargir, le plus possible, l'ensemble des systèmes auxquels les méthodes sont applicables. Ces ensembles sont définis en termes de contraintes imposées aux non-linéarités du système.

## 1.1 Définitions

Quelques définitions relatives à la stabilité des systèmes non linéaires seront données. Les méthodes d'analyse seront brièvement présentées. On parlera surtout des méthodes de Lyapunov, et l'accent sera, plus particulièrement, mis sur la deuxième méthode de Lyapunov, qui constitue l'élément central de la méthode design adoptée, i.e. le **backstepping**.

**Systèmes non linéaires** — On appelle systèmes linéaires les systèmes physiques représentés par des équations différentielles linéaires à coefficients constants. L'hypothèse de linéarité équivaut au principe de superposition. Les systèmes non linéaires,

par opposition aux systèmes linéaires, sont des systèmes physiques qui ne sont pas régis par des équations linéaires. Autrement dit, le principe de superposition ne peut leur être appliqué.

**Équilibre** — Physiquement, un système est en équilibre quand il conserve son état en absence de forces externes. Mathématiquement, cela équivaut à dire que la dérivée  $\dot{\mathbf{x}}$  de son état est nul. Pour un système

$$\dot{\mathbf{x}} = \varphi(\mathbf{x}) \quad (1.1)$$

l'état (ou les états) d'équilibre  $\mathbf{x}_e$  est la solution (sont les solutions) de l'équation algébrique

$$\varphi(\mathbf{x}) = \mathbf{0}$$

Dans le cas de systèmes linéaires, on a  $\varphi(\mathbf{x}) = \mathbf{A}\mathbf{x}$ , ce qui implique que  $\mathbf{x} = \mathbf{0}$  est un point équilibre pour tout système linéaire. Si  $\mathbf{A}$  est régulière, l'origine est le seul point d'équilibre. Dans le cas où la matrice  $\mathbf{A}$  est singulière, tout le sous-espace défini par  $\mathbf{A}\mathbf{x} = \mathbf{0}$  constitue une région d'équilibre. Pour les systèmes non linéaires, la solution est moins évidente et l'origine n'est pas forcément un point d'équilibre. En plus, en présence de plusieurs équilibres, ces derniers peuvent se présenter, comme dans le cas linéaire, sous forme de domaines continus, mais aussi de points isolés, voire même de combinaisons des deux.

**Plan de phase** — Pour comprendre le comportement d'un système non linéaire, on utilise souvent une représentation de ses trajectoires dans l'espace de phase (figure 1.1). Ces trajectoires sont un ensemble de courbes qui représentent l'évolution de l'état du système dans le temps. L'obtention de ces trajectoires passe, toutefois, par la résolution de l'équation différentielles (1.1), qui peut s'avérer une tâche difficile. C'est pourquoi, les outils permettant l'analyse du comportement du système, sans avoir à résoudre les équations le décrivant, ont connu un grand succès. Les techniques basées sur la deuxième méthode de Lyapunov font partie de cette classe.

**Stabilité (Définition intuitive)** — On dit qu'un système est stable si, déplacé de sa position d'équilibre, il tend à y revenir ; instable, s'il tend à s'en écarter davantage.

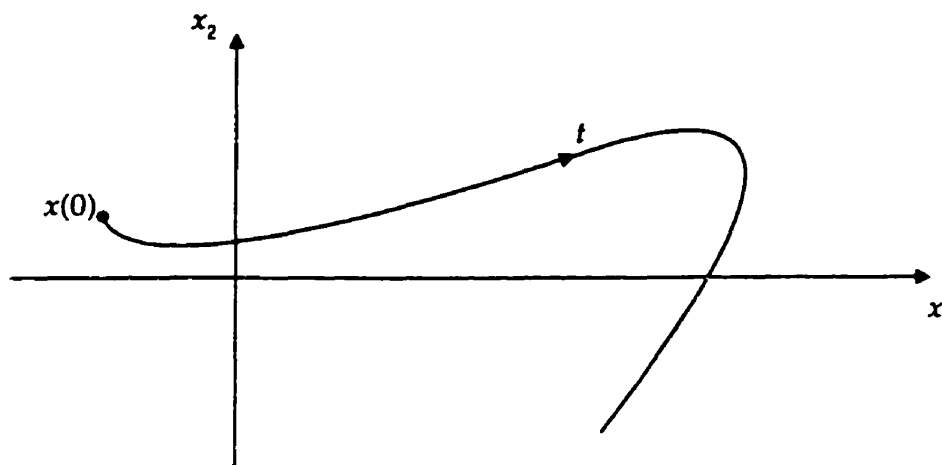


FIG. 1.1: Trajectoire d'un système dans le plan de phase

**Stabilité (Selon Lyapunov)** — Soit un système dont l'état est défini par le vecteur  $\mathbf{x}$  qui possède la position d'équilibre  $\mathbf{x}_e$ . Écarté de cette position, et abandonné à lui-même au temps  $t = t_0$  avec les conditions initiales  $\mathbf{x}(t_0)$ , le système aura comme état  $\mathbf{x}(t)$ . On dit que la position d'équilibre du système est stable (figure 1.2), si pour tout  $\epsilon > 0$ , il existe  $\delta(\epsilon) > 0$  tel que si

$$\|\mathbf{x}(t_0) - \mathbf{x}_e\|^2 < \delta,$$

on soit certain qu'on aura après un certain temps  $t$  et pour toutes les valeurs  $t > t_0$

$$\|\mathbf{x}(t) - \mathbf{x}_e\|^2 < \epsilon,$$

Dans le cas contraire (i.e. s'il existe au moins un  $\epsilon$  tel que l'on ne puisse pas trouver  $\delta$  correspondant qui satisfasse aux inégalités) on dit que l'équilibre est instable.

La stabilité (simple) ne requiert pas que le vecteur d'état  $\mathbf{x}(t)$  tendent vers  $\mathbf{x}_e$  lorsque  $t$  augmente indéfiniment. Si le système, écarté de la position d'équilibre, finit par y revenir (i.e.  $\mathbf{x}(t)$  tend vers le point d'équilibre  $\mathbf{x}_e$  pour  $t$  infini) on dit qu'il y a stabilité asymptotique (figure 1.3).



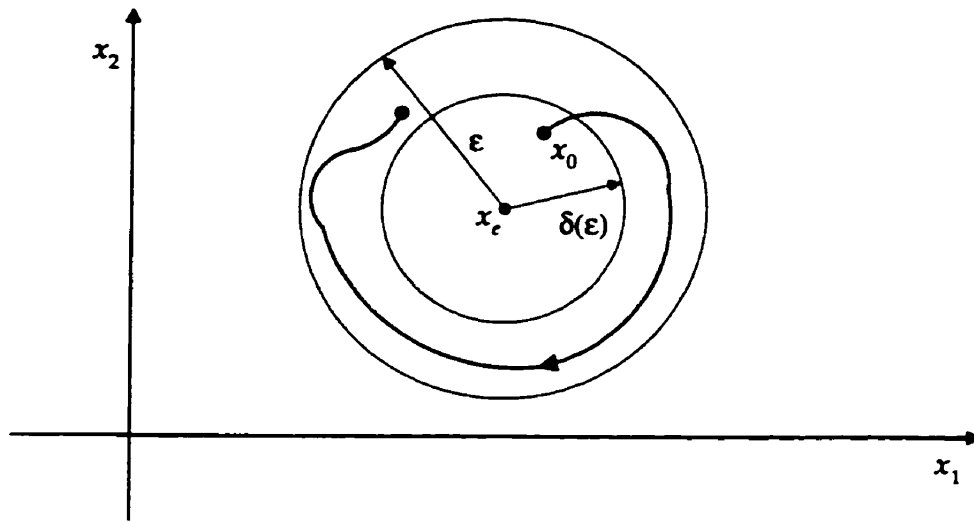


FIG. 1.2: Stabilité simple selon Lyapunov

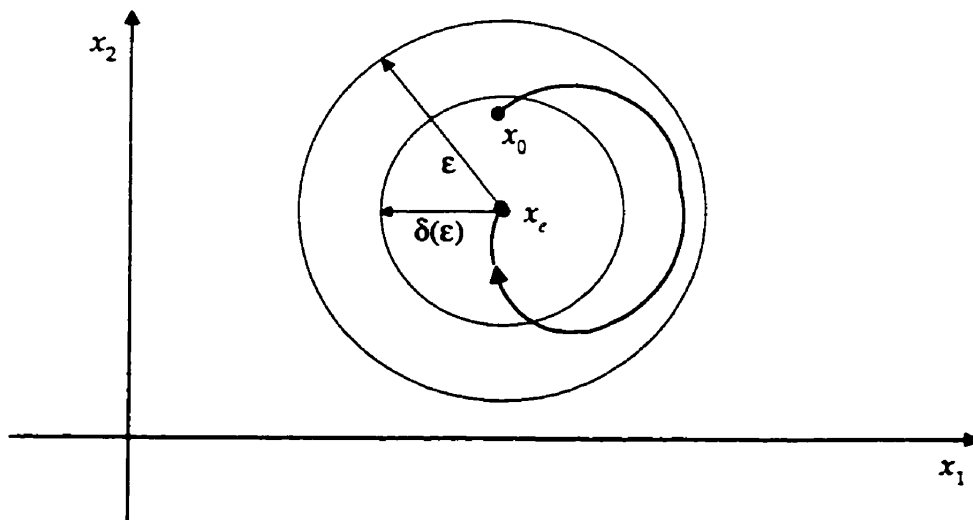


FIG. 1.3: Stabilité asymptotique selon Lyapunov

## Stabilité locale et Stabilité globale

On peut prédire le comportement d'un système linéaire à partir de l'analyse de sa position d'équilibre. Un système dont le point d'équilibre est stable (instable) est stable (instable). Il n'en est plus de même pour un système non linéaire. Étant donné que celui-ci peut avoir plusieurs positions d'équilibre, la stabilité de l'une de ces positions d'équilibre ne suffit pas à elle seule à prédire la stabilité du système. Afin de quantifier l'influence de la stabilité d'un point d'équilibre sur la stabilité du système, de nouvelles définitions de la stabilité sont introduites ; on parle de **stabilité locale**, **stabilité globale** et **région d'attraction**.

**Stabilité locale** — La stabilité locale concerne simplement la position d'équilibre considérée, sans rien préjuger sur le domaine de validité de cette stabilité. C'est une condition nécessaire, mais non suffisante à la stabilité du système dans un certain domaine  $\mathcal{D}$ , contenant cette position d'équilibre.

**Stabilité globale** — On parle de stabilité globale lorsque le système est stable pour toutes les valeurs que peuvent prendre les variables du système. La stabilité globale possède un intérêt pratique beaucoup plus considérable que la stabilité locale. Elle ne dépend pas seulement du système, mais aussi des valeurs que peuvent prendre les variables dans le problème considéré. Ainsi, le même système est stable ou instable globalement, suivant le domaine de variables auquel on s'intéresse.

**Région d'attraction** — La région autour de la position d'équilibre, à l'intérieur de laquelle toutes les trajectoires approchent le point d'équilibre est appelée région ou domaine d'attraction. Sa taille est souvent un facteur très important dans l'évaluation des performances des systèmes non linéaires.

## 1.2 Méthodes d'analyse

Dans la théorie des systèmes, la stabilité est un élément très important. Celle-ci passe au premier plan dans le cadre des systèmes non linéaires, pour devenir le facteur clé. Cela explique le regain d'activité que connaît ce domaine depuis plusieurs

années. Du degré d'importance des non-linéarités présentes dans un système dépend la complexité des méthodes permettant de se prononcer sur sa stabilité. Pour une grande classe de systèmes non linéaires, les méthodes d'analyse linéaire donnent en général, des résultats acceptables. Dans certains cas, des méthodes approximatives (l'équivalent harmonique ou la première méthode de Lyapunov<sup>1</sup>) (Gille 1988, Benaskeur 1997b) sont utilisées afin de palier aux limitations des méthodes linéaires. Malheureusement pour une grande majorité de systèmes, ces deux approches restent insuffisantes et ne donnent que des conditions nécessaires de stabilité. Des techniques plus générales, et reposant sur une théorie plus rigoureuse, ont été proposées. Contrairement aux méthodes linéaires, ces méthodes (adaptées au cas non linéaire) permettent d'expliquer les phénomènes énumérés précédemment et d'en tenir compte dans l'analyse des systèmes. Elles reposent toutes (ou presque) sur la deuxième méthode (directe) de Lyapunov. Cette dernière fournit un outil très puissant pour tester (et trouver des conditions suffisantes à) la stabilité des systèmes dynamiques, sans avoir à résoudre explicitement les équations différentielles les décrivant.

### 1.2.1 Première méthode de Lyapunov

Pour un système non linéaire, on s'intéresse souvent à son comportement au voisinage des points singuliers. Si la dynamique est linéarisée au tour d'un point d'équilibre, peut-on se prononcer sur la stabilité locale du système ? La réponse est donnée par le théorème de stabilité locale de Lyapunov, connu sous le nom de première méthode. Soit le cas du système décrit par

$$\begin{aligned}\dot{x}_1 &= ax_1 + bx_2 + \Pi_1(x_1, x_2) \\ \dot{x}_2 &= cx_1 + dx_2 + \Pi_2(x_1, x_2)\end{aligned}$$

D'après Lyapunov, la stabilité de la position d'équilibre de ce système peut s'étudier sur la version linéarisée, obtenue en négligeant les termes  $\Pi_1$  et  $\Pi_2$  qui contiennent des puissances supérieures ou égales à deux en  $x_1$  et  $x_2$ .

---

<sup>1</sup>La première méthode de Lyapunov apporte une validité théorique à la technique de linéarisation, qui reste l'un des outils d'analyse (et de design) les plus utilisés dans le cas des systèmes non linéaires.

### **Théorème 1.1 (Lyapunov)**

- Si le système linéarisé est asymptotiquement stable, il y a stabilité asymptotique.
- Si le système linéarisé est instable, il y a instabilité.
- Si le système linéarisé est stable sans l'être asymptotiquement, on ne peut se prononcer. C'est le cas critique de Lyapunov. La stabilité ou l'instabilité dans ce cas dépend des termes de degré supérieur à un, négligés dans l'approximation.



**Remarque 1.1 (Cas critique)** — Il faut noter qu'en présence d'un cas critique, il n'est pas possible d'esquiver la difficulté en changeant **extrêmement peu** les valeurs de certains paramètres du problème (Gille 1988).



Du point de vue pratique, ce théorème a cependant une importance limitée. D'une part, il ne permet d'étudier que la stabilité du point singulier (stabilité locale), et ne donne aucune information sur le domaine de stabilité (stabilité globale). D'autre part, il suppose que l'approximation du premier degré existe, autrement dit, que les développements en séries des seconds membres des équations comportent des termes du premier degré. Cette hypothèse exclut un certain nombre de cas importants (organe avec zone morte, plus-ou-moins, ... ).

### **1.2.2 Deuxième méthode de Lyapunov**

Cette méthode découle du concept d'énergie dans un système. Pour un système physique, l'énergie est une fonction définie positive de son état. Si le système est conservatif l'énergie reste constante ; pour un système dissipatif, elle décroît. Pour ces deux cas, le système est stable. Si l'énergie croît, il est clair que le système est instable.

L'idée ici est d'analyser la stabilité du système, sans avoir à résoudre explicitement les équations différentielles non linéaires le régissant. On se contente dans cette méthode d'étudier les variations (signe de la dérivée) de l'énergie (ou une fonction qui lui est équivalente) le long de la trajectoire du système (figures 1.4 & 1.5). Comme outil, on

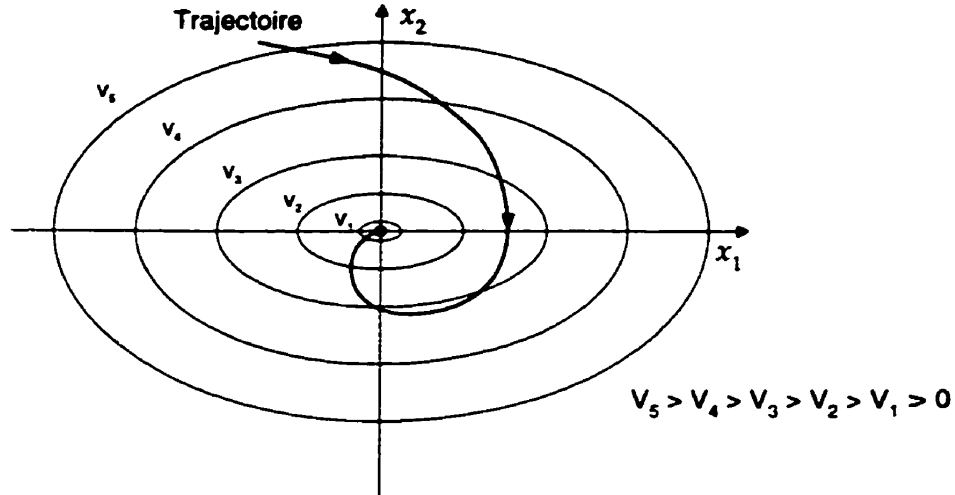


FIG. 1.4: Contours à énergie constante dans le plan de phase

utilise les théorèmes suivants, que l'on doit à A. M. Lyapunov (1966), qui permettent de se prononcer sur la stabilité (ou l'instabilité) d'un système.

### **Théorème 1.2 (Stabilité asymptotique)**

S'il est possible de trouver une fonction  $\mathcal{V}(\mathbf{x})$  de signe défini<sup>2</sup> (avec  $\mathcal{V}(\mathbf{0}) = 0$ ), dans un domaine  $\mathcal{D}$  comprenant la position d'équilibre, et dont la dérivée totale par rapport au temps  $\dot{\mathcal{V}}$  soit définie et de signe opposé dans le même domaine, l'équilibre sera asymptotiquement stable dans ce domaine.

■

### **Théorème 1.3 (Instabilité)**

S'il est possible de trouver une fonction  $\mathcal{V}$  dont la dérivée est de signe défini dans un domaine  $\mathcal{D}$  comprenant l'origine et que  $\mathcal{V}$  soit

- définie de même signe que  $\dot{\mathcal{V}}$ , ou
- indéfinie en signe,

l'équilibre est instable.

■

---

<sup>2</sup>Voir la définition B.2 dans l'annexe.

### **Théorème 1.4 (Stabilité simple)**

S'il est possible de trouver une fonction  $\mathcal{V}$  de signe défini dans un domaine  $\mathcal{D}$  et dont la dérivée totale  $\dot{\mathcal{V}}$  soit semi-définie et de signe opposé dans le même domaine, l'équilibre est (simplement) stable dans ce domaine.



Dans le cas d'une stabilité simple, l'asymptoticité de la stabilité (si le système est asymptotiquement stable) peut être prouvée en utilisant le théorème de Barbašin-Krasovskij (Gille 1988), qui vient compléter le théorème de Lyapunov pour la stabilité simple.

### **Théorème 1.5 (Barbašin et Krasovskij)**

S'il est possible de trouver une fonction  $\mathcal{V}$  de signe défini dans un domaine  $\mathcal{D}$  et dont la dérivée totale  $\dot{\mathcal{V}}$  soit semi-définie et de signe opposé dans le même domaine, l'équilibre est asymptotiquement stable dans ce domaine, si l'ensemble des points où  $\dot{\mathcal{V}}$  s'annule ne contient pas l'une des trajectoires possibles du système<sup>3</sup>.



Ces théorèmes offrent l'avantage de ne pas se limiter à la prédiction de la stabilité des points d'équilibre, étant donné qu'ils permettent aussi l'estimation de la région d'attraction au tour des équilibres stables. Cette dernière est donnée par l'intérieur de la plus grande surface (ou hypersurface) à l'intérieur de laquelle les conditions de signe sont vérifiées. Comme toute étude de stabilité globale, le domaine de stabilité prédit est à l'intérieur du domaine de stabilité réel, qui est souvent beaucoup plus grand, mais difficile à trouver la fonction de Lyapunov qui permet sa prédiction.

### **Application à l'analyse**

Ces théorèmes présentent une condition suffisante à la stabilité. Ainsi, un point d'équilibre d'un système non linéaire est stable, si une fonction de Lyapunov peut être trouvée et qui vérifie, au voisinage de ce point, les conditions imposées. Pour l'étude de la stabilité d'un système caractérisé par un vecteur d'état  $\mathbf{x}$ , la méthode directe

---

<sup>3</sup>Ceci est équivalent à dire que les équations des courbes (ou des surfaces) pour lesquelles  $\dot{\mathcal{V}} = 0$ , substituées dans les équations du système ne les vérifient en aucun point, autre que l'origine.

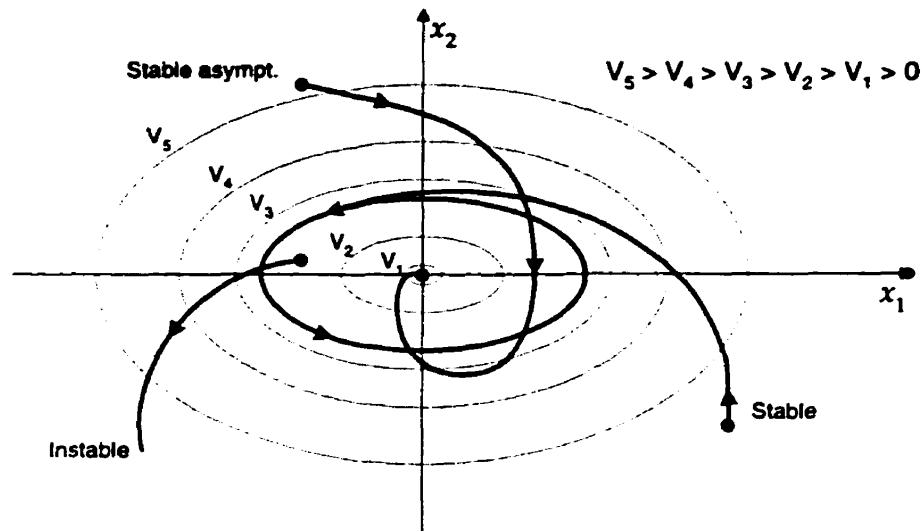


FIG. 1.5: Stabilité par la fonction de Lyapunov

de Lyapunov consiste, alors, à chercher une fonction  $\mathcal{V}(\mathbf{x})$  (représentative de l'énergie) de signe défini qui se prête à l'application de l'un des théorèmes cités précédemment. Pour les systèmes linéaires, il existe des méthodes systématiques (Olas 1990) pour construire une fonction de Lyapunov permettant de conclure à sa stabilité. Il n'en est pas de même pour les systèmes non linéaires, pour lesquels on est réduit à essayer des types de fonctions pour telle ou telle classe de systèmes. Il n'y a aucune règle générale permettant de trouver une fonction de Lyapunov pour n'importe quel système. Il existe néanmoins des approches qui conduisent, en général, à des résultats acceptables (Gille 1988, Khalil 1996, Benaskeur 1997b). Des exemples de fonctions qui réussissent souvent comme fonctions de Lyapunov, sont données par

- Fonction quadratique (Lyapunov)

$$\mathcal{V}(\mathbf{x}) = \mathbf{x}^T \mathbf{P} \mathbf{x}$$

où  $\mathbf{P}$  est une matrice symétrique définie positive.

- Fonction quadratique plus intégrale (Lur'e)

$$\mathcal{V}(\mathbf{x}) = \mathbf{x}^T \mathbf{P} \mathbf{x} + \int_0^{\mathbf{x}} \varphi(u) du$$

avec  $\varphi$  est assujettie à certaines contraintes.

### 1.3 Conclusion

La multitude des approches, utilisées pour l'analyse des systèmes non linéaires, reflète la grande diversité des non-linéarités que l'on peut y rencontrer. Les méthodes approximatives, très largement utilisées, ne sont d'aucune utilité en présence de non-linéarités essentielles, i.e. qui ne peuvent être négligées. Dans de telles situations, des méthodes exactes, telle que la deuxième méthode de Lyapunov, constituent des outils incontournables.

N'étant à l'origine qu'un outil d'analyse, la méthode de Lyapunov a suscité beaucoup d'intérêts ces dernières années dans le domaine du design ; ce qui a donné lieu à des techniques systématiques de conception de contrôleurs pour les systèmes non linéaires. Le backstepping, qui constitue l'élément central de ce travail, est l'un des résultats majeurs en cette matière. Mais, avant d'aborder cette méthode en détail (chapitres 3 et 4), le problème de la commande des systèmes non linéaires sera d'abord présenté dans un cadre plus général (chapitre 2).



# Chapitre 2

## Commande des systèmes non linéaires

Du degré d'importance des non-linéarités présentes dans un système, dépend la technique à utiliser pour le commander. Dans certains cas<sup>1</sup>, la présence des non-linéarités rend la tâche plus ardue, étant donné les problèmes théoriques et pratiques à résoudre. Mais il existe, néanmoins, certaines méthodes qui donnent des solutions intéressantes. Si le système à commander présente des faibles non-linéarités, la démarche naturelle pour l'ingénieur est de les ignorer, en les traitant comme des perturbations affectant un modèle linéaire du système. Cette approche, qui peut s'appliquer à une grande classe de systèmes, permet de profiter de l'arsenal de la théorie du contrôle des systèmes linéaires qui est maintenant assez complet et maîtrisé. Pour beaucoup de systèmes non linéaires qui font partie de cette classe, l'utilisation de versions modifiées des techniques de contrôle linéaire donne des résultats satisfaisants. Reste à noter que cette méthode n'est pas générale et sa validation doit être faite pour chaque cas.

Malheureusement, dans beaucoup de situations, les non-linéarités ne peuvent être négligées, on parle alors de non-linéarités essentielles. De tels systèmes ne peuvent être directement traités dans le cadre de la commande linéaire, et des méthodes mieux adaptées à ce type de problèmes sont à considérer. Deux approches sont à distinguer pour aborder ce type de systèmes. La première vise à linéariser le système à comman-

---

<sup>1</sup>Le plus souvent.

der, afin de profiter des techniques linéaires. Cette linéarisation est réalisée, moyennant des approximations ou des transformations géométriques dans l'espace de phases. L'inconvénient avec ces méthodes réside dans l'annulation, quand cela est possible, de toutes les non-linéarités, même les non nuisibles.

La deuxième approche<sup>2</sup>, quant à elle, se base sur l'idée de concevoir une commande de façon à pouvoir trouver une Fonction de Commande de Lyapunov (fcl) garantissant certaines performances pour le système en boucle fermée. Trouver une telle fcl, pour un système non linéaire d'ordre élevé, peut s'avérer très difficile. La technique du backstepping permet de réduire cette complexité. Elle consiste à fragmenter le système en un ensemble de sous-systèmes imbriqués, d'ordre décroissant. Le calcul de la fcl s'effectue récursivement en partant de l'intérieur de la boucle. À chaque étape, l'ordre du système est augmenté et la partie non stabilisée lors de l'étape précédente est traitée. La dernière étape correspond au système global. L'ordre réduit des systèmes considérés à chaque étape procure une flexibilité dans le design, ce qui permet de résoudre plusieurs problèmes de commande sous moins de contraintes qu'avec la première approche.

## 2.1 Commande par linéarisation

Les méthodes basées sur la linéarisation<sup>3</sup> constituaient, jusqu'à très récemment, l'essentiel des techniques utilisées pour la commande des systèmes non linéaires. Elles permettent, moyennant des approximations et/ou des transformations, de ramener les équations du système sous une forme linéaire. Deux techniques de linéarisation sont utilisées : la linéarisation locale et la linéarisation exacte.

### 2.1.1 Linéarisation locale

Cette approche repose sur la première méthode de Lyapunov. Elle consiste à déduire le comportement d'un système non linéaire à partir de son système linéaire associé, obtenu par une linéarisation jacobienne autour du point de fonctionnement désiré (Gille

---

<sup>2</sup>La plus intéressante.

<sup>3</sup>Pour plus de détail, consulter Khalil (1996) et/ou Benaskeur (1997a).

1988, Khalil 1996). Une variante de cette méthode est donnée par la technique du modèle linéaire tangent (**gain scheduling**), où plusieurs linéarisations sont effectuées à différents points d'opération, donnant lieu à une famille de systèmes linéaires pour lesquels un seul contrôleur à paramètres variables est conçu.

### Principe

Soit le système non linéaire

$$\dot{\mathbf{x}} = \varphi(\mathbf{x}, u)$$

où  $(\mathbf{x} = \mathbf{0}, u = 0)$  est le point de fonctionnement considéré. Son système linéaire associé est donné par

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u$$

où

$$\mathbf{A} = \left. \frac{\partial \varphi}{\partial \mathbf{x}}(\mathbf{x}, u) \right|_{\mathbf{x}=\mathbf{0}, u=0} \quad \text{et} \quad \mathbf{B} = \left. \frac{\partial \varphi}{\partial u}(\mathbf{x}, u) \right|_{\mathbf{x}=\mathbf{0}, u=0}$$

Le design s'effectue sur le système linéaire ; une fois les objectifs atteints, la commande est appliquée au système non linéaire dont on évalue les performances. Le fait que le système linéaire répond aux exigences, ne garantit en rien le bon comportement du système non linéaire. C'est une condition nécessaire mais pas suffisante au bon fonctionnement de ce dernier. Une amélioration peut, toutefois, être apportée à cette approche, en considérant les variations dues à la présence des non-linéarités comme des perturbations affectant le modèle linéaire (modèle nominal). Par la suite, une loi de commande robuste qui tient compte de ces perturbations est conçue.

**Remarque 2.1 (Équivalent linéaire)** Le choix de l'équivalent linéaire n'est pas unique. D'autres techniques de linéarisation existent, elles peuvent être basées sur la minimisation d'un critère d'optimisation. Il faut toutefois noter que les modèles linéarisés ne définissent de façon intrinsèque un système que si la linéarisation est effectuée en un point d'équilibre (Fossard 1993).

□

## **Modèle linéaire tangent**

Quand la connaissance du modèle linéarisé en un point n'est plus suffisante pour satisfaire les objectifs du contrôle, une des solutions consiste à linéariser le système autour de différents points de fonctionnement. L'ensemble de tous les modèles linéarisés autour de tous les points de fonctionnement possibles forme ce que l'on appelle le **modèle linéaire tangent**. Sa connaissance, bien que constituant une information partielle sur le comportement non linéaire du système, est souvent amplement suffisante pour calculer des lois de contrôle répondant aux spécifications. La synthèse des commandes de vol et des pilotes automatiques pour les avions est réalisée à partir de tels modèles (Fossard 1993, Friedland 1996, Khalil 1996). Cette technique qui est une généralisation de la méthode par linéarisation jacobienne, est aussi connue sous le nom de **commande au premier ordre**.

La validité de la représentation de l'organe non linéaire par le modèle linéaire tangent dépend de l'importance de la non-linéarité et du nombre de points utilisés dans l'approximation. Souvent, le design se limite à un nombre réduit de points. Par la suite les paramètres discrets du contrôleur sont interpolés pour les points intermédiaires, afin d'obtenir une loi de commande non linéaire dont la linéarisation donnerait le contrôleur calculé. Le choix de la technique d'interpolation est un élément très important dans la phase de design. La méthode la plus simple et plus rapide est l'interpolation d'ordre zéro (BOZ) ; elle manque cependant de souplesse lors des commutations entre les différentes valeurs des paramètres. Il existe d'autres méthodes fondées sur l'intégration continue des commandes obtenues localement et qui conduisent à de meilleures performances (Johansen & Foss 1993).

### **2.1.2 Linéarisation exacte**

Cette approche consiste à linéariser le système en boucle fermée, par une série de transformations, sans faire d'approximations. La commande choisie est composée de deux parties. La première a pour rôle d'annuler les différents termes non linéaires du procédé, alors que la deuxième est choisie pour satisfaire les exigences en boucle fermée du système linéarisé.

## Principe

La classe de systèmes non linéaires de la forme

$$\begin{aligned}\dot{\mathbf{x}} &= \varphi(\mathbf{x}) + \psi(\mathbf{x})u \\ \mathbf{y} &= \eta(\mathbf{x})\end{aligned}$$

est considérée, où l'objectif est de trouver une loi de commande

$$u = \sigma(\mathbf{x}) + \beta(\mathbf{x})v$$

et une transformation

$$z = \mathcal{T}(\mathbf{x})$$

qui transforment le système non linéaire en un système linéaire équivalent.  $v$  est la commande pour le nouveau système. Deux cas sont à distinguer.

### Linéarisation entrée/état

Cette approche consiste à choisir la loi de commande  $u$  afin que toutes les équations d'état du système compensé aient une forme linéaire. Cette solution revient à annuler les différentes non-linéarités présentes dans les équations, en insérant des termes absorbants dans l'expression de la commande, afin d'obtenir des relations linéaires entre la nouvelle entrée  $v$  et les différentes variables d'état. Certaines propriétés structurales du système peuvent faciliter la linéarisation. Il est évident que pour annuler une non-linéarité  $\sigma(\mathbf{x})$  par soustraction, cette dernière doit toujours apparaître au même endroit que la commande  $u$  et vient s'ajouter à celle-ci ( $u + \sigma(\mathbf{x})$ ). De même, afin de pouvoir annuler une non-linéarité  $\mu(\mathbf{x})$  par division,  $u$  et  $\mu(\mathbf{x})$  doivent toujours apparaître comme un produit ( $u\mu(\mathbf{x})$ ). Si  $\mu(\mathbf{x})$  est non singulière dans le domaine d'intérêt  $\mathcal{D}$ , alors elle peut être annulée par une commande  $u = \beta(\mathbf{x})v$ , où  $\beta(\mathbf{x}) = \mu^{-1}(\mathbf{x})$ .

Pour être capable de ramener le système non linéaire à une forme linéaire contrôlable, il suffit alors que son équation ait la structure suivante<sup>4</sup>

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\beta^{-1}(\mathbf{x}) \left[ u - \sigma(\mathbf{x}) \right] \quad (2.1)$$

---

<sup>4</sup> $\beta^{-1}(\mathbf{x})$  étant la matrice inverse de  $\beta(\mathbf{x})$  pour chaque  $\mathbf{x}$  et non la transformation inverse.

Dans ce cas, on peut transformer le système par la commande

$$u = \sigma(\mathbf{x}) + \beta(\mathbf{x})v$$

pour obtenir un système linéaire équivalent

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}v$$

pour lequel les techniques linéaires permettent le choix de la loi de commande adéquate  $v$  permettant d'obtenir les performances requises.

Le fait que le système n'ait pas la structure (2.1), ne signifie pas forcément que la linéarisation n'est pas possible. Le modèle d'état d'un système n'étant pas unique, si le système n'a pas la structure requise pour un choix des variables d'état, il peut l'avoir pour un autre choix. Il suffit de trouver un difféomorphisme<sup>5</sup> qui transforme les équations d'état sous la forme requise, dans un nouveau système de coordonnées.

### Linéarisation entrée/sortie

Le modèle d'un système est souvent représenté par les équations d'état et une équation de sortie. La linéarisation des équations d'état ne garantit pas forcément celle de l'équation de sortie. Quand on s'intéresse aux variables de sortie (problème de poursuite par exemple), il est plus intéressant de linéariser la relation entrée/sortie au prix de laisser une partie des équations d'état non linéaire. C'est le problème de la linéarisation entrée/sortie<sup>6</sup>.

### 2.1.3 Limitations de la linéarisation

Bien que largement utilisées, les méthodes basées sur la linéarisation présentent quelques inconvénients. Leur utilisation doit, souvent, s'accompagner de beaucoup de précaution. La linéarisation locale (ou jacobienne) n'est valable qu'à un point d'opération, alors que la linéarisation exacte n'est pas toujours possible. Même si celle-ci existe, il n'est pas toujours intéressant d'éliminer toutes les non-linéarités présentes

<sup>5</sup>Voir la définition B.5 dans l'annexe.

<sup>6</sup>Il faut toutefois faire attention à ne pas confondre la dynamique de la boucle (fermée) qui est linéaire avec celle du système qui reste, quant à elle, toujours non linéaire.

dans un système. Certaines non-linéarités aident à préserver la stabilité du système, et leur élimination ne fait qu'augmenter, considérablement et inutilement, l'effort que doit fournir l'actionneur (Krstić et al. 1995).

C'est pourquoi, les recherches se sont orientées, ces dernières années, vers des méthodes mieux adaptées à la nature non linéaire des systèmes. Le design basé sur la méthode directe (ou deuxième) de Lyapunov (1966) constitue l'un des axes majeurs de cette orientation.

## 2.2 Commande par Lyapunov

Introduite au départ comme un outil d'analyse, la méthode (deuxième) de Lyapunov s'est très vite avérée fort utile dans les problèmes de design. L'idée consiste à calculer une loi de commande, afin de garantir que la dérivée d'une certaine fonction –de Lyapunov– définie positive, et bien choisie, soit non positive. Cette propriété doit être vérifiée le long de toutes les trajectoires possibles du système en boucle fermée. En présence d'incertitudes et/ou de paramètres inconnus, une loi d'adaptation devient nécessaire. Le design adaptatif consiste, alors, à trouver un triplet (fonction de Lyapunov, loi de commande, loi d'adaptation) qui répond aux spécifications. L'avantage avec cette approche réside dans la prise en compte de la dynamique de l'estimation, afin d'éviter tout effet destructeur dû à celle-ci. Pour des raisons de stabilité, ces méthodes conviennent mieux aux systèmes non linéaires. Les solutions proposées sont spécialement conçues pour ce type de systèmes<sup>7</sup>.

Avant de détailler ces techniques de design, rappelons la définition de la fonction de contrôle de Lyapunov (fcl), et le théorème de stabilité de Lasalle-Yoshizawa (Lasalle & Lefschetz 1961. Krstić et al. 1995, Khalil 1996), sur lequel elle repose.

---

<sup>7</sup>Avant d'être, éventuellement, appliquées aux systèmes linéaires.

### **Théorème 2.1 (Lasalle -Yoshizawa)**

Soit le système non linéaire décrit par

$$\dot{\mathbf{x}} = \varphi(\mathbf{x}, t) \quad (2.2)$$

qui a comme équilibre  $\mathbf{x}_e = 0$ , et soit  $\mathcal{V} : \mathcal{R}^n \times \mathcal{R}_+ \rightarrow \mathcal{R}_+$  une fonction telle que

$$\kappa_1(\|\mathbf{x}\|) \leq \mathcal{V}(\mathbf{x}, t) \leq \kappa_2(\|\mathbf{x}\|)$$

$$\dot{\mathcal{V}} = \frac{\partial \mathcal{V}}{\partial t} + \frac{\partial \mathcal{V}}{\partial \mathbf{x}} \varphi(\mathbf{x}, t) \leq -\mathcal{W}(\|\mathbf{x}\|) \leq 0$$

$\forall t \geq 0, \forall \mathbf{x} \in \mathcal{R}^n$ , où  $\kappa_1, \kappa_2$  et  $\mathcal{W}$  sont des fonctions de classe<sup>8</sup>  $\mathcal{K}_\infty$  et  $\mathcal{W}$  est continue. Si ces conditions sont vérifiées, toutes les solutions de (2.2) sont globalement uniformément bornées (stabilité simple), avec

$$\lim_{t \rightarrow \infty} \mathcal{W} = 0$$

En plus, si  $\mathcal{W}$  est définie positive, le point d'équilibre  $\mathbf{x}_e = 0$  possède une stabilité asymptotique globale et uniforme. ■

#### **2.2.1 Application au design**

Soit le cas du système non linéaire autonome suivant

$$\dot{\mathbf{x}} = \varphi(\mathbf{x}, u) \quad (2.3)$$

Avec  $\varphi(0, 0) = 0$ . Le but est de concevoir une loi de commande  $u = \alpha(\mathbf{x})$  tel que l'équilibre ( $\mathbf{x}_e = 0$ ) du système résultant en boucle fermée

$$\dot{\mathbf{x}} = \varphi(\mathbf{x}, \alpha(\mathbf{x}))$$

soit asymptotiquement stable. Pour garantir cette stabilité, un bon choix des fonctions  $\mathcal{V}(\mathbf{x})$  et  $\mathcal{W}(\mathbf{x})$  est nécessaire. La commande  $u = \alpha(\mathbf{x})$  doit vérifier

$$\dot{\mathcal{V}} = \frac{\partial \mathcal{V}}{\partial \mathbf{x}} \varphi(\mathbf{x}, \alpha(\mathbf{x})) \leq -\mathcal{W}(\mathbf{x}) \leq 0 \quad (2.4)$$

---

<sup>8</sup>Voir la définition B.3 dans l'annexe.



À cause d'un mauvais choix de  $\mathcal{V}(\mathbf{x})$  et  $\mathcal{W}(\mathbf{x})$ , une telle commande peut stabiliser le système (2.3) sans pour autant vérifier (2.4). Un système pour lequel un bon choix de ces deux fonctions existe, est dit possédant une **Fonction de Commande de Lyapunov** (fcl). Pour de tels systèmes l'équilibre est globalement asymptotiquement stabilisable.

**Définition 2.1 (Fonction de Commande de Lyapunov)** — Une fonction  $\mathcal{V} : \mathcal{R}^n \rightarrow \mathcal{R}_+$  définie positive et non bornée radialement est appelée une fonction de commande de Lyapunov (fcl) pour le système (2.3) si

$$\inf_{u \in \mathcal{R}} \left\{ \frac{\partial \mathcal{V}}{\partial \mathbf{x}}(\mathbf{x}) \boldsymbol{\varphi}(\mathbf{x}, u) \right\} < 0, \quad \forall \mathbf{x} \neq 0$$

□

**Exemple 2.1** — Soit à stabiliser l'origine ( $x_1 = 0$ ) du système scalaire

$$\dot{x}_1 = \boldsymbol{\varphi}_1(x_1)^T \boldsymbol{\vartheta} + \psi_1(x_1)u \quad (2.5)$$

où  $\boldsymbol{\varphi}_1$  et  $\psi_1$  sont des fonctions non linéaires, et  $\boldsymbol{\vartheta}$  est un vecteur de paramètres connus. Pour ce faire, une fcl  $\mathcal{V}(x_1)$  doit être choisie et une commande  $u$  qui annule sa dérivée le long de la trajectoire, doit être calculée. Pour un système scalaire,

$$\mathcal{V}(x_1) = \frac{1}{2}x_1^2$$

représente souvent un bon choix (Gille 1988, Krstić et al. 1995, Khalil 1996). Sa dérivée le long de la solution de (2.5) donne

$$\dot{\mathcal{V}}(x_1) = x_1 \dot{x}_1 = x_1 \left[ \boldsymbol{\varphi}_1(x_1)^T \boldsymbol{\vartheta} + \psi_1(x_1)u \right]$$

Un choix judicieux de  $u$  rend  $\dot{\mathcal{V}}(x_1)$  négative et assure la stabilité asymptotique de l'origine du système. Un exemple de commande est donné par le choix de  $u$  tel que

$$\boldsymbol{\varphi}_1(x_1)^T \boldsymbol{\vartheta} + \psi_1(x_1)u = -k_1 x_1, \quad k_1 > 0$$

Ce qui donne

$$u = -\frac{1}{\psi_1(x_1)} \left[ k_1 x_1 + \boldsymbol{\varphi}_1(x_1)^T \boldsymbol{\vartheta} \right]$$

La dérivée s'écrit alors

$$\dot{V}(x_1) = -k_1 x_1^2 \leq 0 \quad (2.6)$$

d'où la stabilité asymptotique de l'origine. Le fait que, dans (2.6),  $\dot{V}$  soit semi-définie négative n'implique pas forcément une stabilité simple. L'ensemble des points où la dérivée s'annule ne constitue pas une trajectoire possible du système, puisqu'elle ne s'annule qu'à l'origine. On peut donc, selon le théorème de Barbašin-Krasovskij (Gille 1988, Benaskeur 1997b), affirmer la stabilité asymptotique.

**Remarque 2.2 (Choix de la commande)** — Le choix de  $u$  n'est pas unique. Un bon choix permet de rendre négative la dérivée, sans supprimer les non-linéarités utiles dans le système, ni augmenter inutilement l'effort fourni par l'actionneur.

□

### 2.2.2 Choix de la fonction de Lyapunov

Même pour des systèmes simples et en l'absence d'incertitudes, le choix de la fonction de Lyapunov, et de la loi de commande<sup>9</sup>, n'est pas toujours facile. Aucune règle générale n'existe à ce jour quant au choix d'une telle fonction. Et quand on sait l'influence de ce choix sur le comportement général du système, on comprend l'intérêt qu'a suscité ce problème ces dernières années. Un bon choix de la fonction de Lyapunov permet d'assurer une stabilité dans une large plage de fonctionnement, voire même globale. Différentes approches ont été présentées dans Gille (1988) et Benaskeur (1997b), concernant la construction des fonctions de Lyapunov dans le cadre de l'analyse des systèmes simples.

Dans ce qui suit, seront présentées des techniques qui répondent aux problèmes que peut poser le choix d'une telle fonction et/ou de la commande, pour des systèmes non linéaires présentant :

- Des incertitudes paramétriques et/ou non paramétriques : commande robuste par Lyapunov.

---

<sup>9</sup>Qui dépend directement de la fonction de Lyapunov.

- Des paramètres inconnus : commande adaptative par Lyapunov.
- Un ordre élevé : commande par **backstepping**. Cette méthode, qui sera présentée en détail dans le **chapitre 3**, offre l'avantage de répondre simultanément à ces trois problèmes.

## 2.3 Commande robuste par Lyapunov

Une solution intéressante au problème des incertitudes (paramétriques et/ou non) est donnée par la **resynthèse par Lyapunov**. Cette technique utilise la fonction de Lyapunov d'un système nominal pour calculer une commande qui vient s'ajouter à la commande initiale, afin de robustifier le système en présence d'incertitudes qui satisfont la **matching condition**<sup>10</sup>. Ce problème de la resynthèse peut être abordé de différentes façons, dépendamment du niveau de connaissance que l'on a, concernant les incertitudes affectant le système. Les objectifs de la commande peuvent également changer en fonction de cette connaissance. Deux cas sont présentés.

### 2.3.1 Stabilisation robuste

Dans ce cas les incertitudes sont relativement bien connues, à savoir qu'elles sont bornées. Cette borne supérieure, qui n'a pas à être petite, doit être connue. Si ces conditions sont vérifiées, une commande qui stabilise le système perturbé, peut être calculée (Khalil 1996). Soit à stabiliser le système

$$\dot{\mathbf{x}} = \varphi(t, \mathbf{x}) + \psi(t, \mathbf{x}) \left[ u + \delta(t, \mathbf{x}, u) \right] \quad (2.7)$$

$\delta$  est une fonction inconnue qui représente les incertitudes, et qui vérifie la **matching condition**. Le système nominal est donné par

$$\dot{\mathbf{x}} = \varphi(t, \mathbf{x}) + \psi(t, \mathbf{x})u$$

---

<sup>10</sup>Les paramètres inconnus se trouvent dans la même équation d'état que la commande.

On suppose que l'on connaît une loi de commande  $u = \alpha(t, \mathbf{x})$ , telle que l'origine du système en boucle fermée

$$\dot{\mathbf{x}} = \varphi(t, \mathbf{x}) + \psi(t, \mathbf{x})\alpha(t, \mathbf{x}) \quad (2.8)$$

soit uniformément asymptotiquement stable. On suppose en plus que l'on connaît une fonction de Lyapunov  $\mathcal{V}(t, \mathbf{x})$  pour le système (2.8) vérifiant

$$\kappa_1(\|\mathbf{x}\|) \leq \mathcal{V}(t, \mathbf{x}) \leq \kappa_2(\|\mathbf{x}\|)$$

et

$$\frac{\partial \mathcal{V}(\mathbf{x}, t)}{\partial t} + \frac{\partial \mathcal{V}(t, \mathbf{x})}{\partial \mathbf{x}} \dot{\mathbf{x}} = \frac{\partial \mathcal{V}}{\partial t} + \frac{\partial \mathcal{V}}{\partial \mathbf{x}} \left[ \varphi(t, \mathbf{x}) + \psi(t, \mathbf{x})\alpha(t, \mathbf{x}) \right] \leq -\mathcal{W}(\|\mathbf{x}\|)$$

où  $\kappa_1, \kappa_2, \mathcal{W}$  sont des fonctions de classe  $\mathcal{K}_\infty$ . On suppose qu'avec la commande

$$u = \alpha(t, \mathbf{x}) + \varsigma$$

le terme incertain satisfait l'inégalité

$$\|\delta(t, \mathbf{x}, \alpha(t, \mathbf{x}) + \varsigma)\| \leq \lambda(t, \mathbf{x}) + k\|\varsigma\|, \quad 0 \leq k < 1 \quad (2.9)$$

où  $\lambda$  est une fonction continue non négative qui représente le niveau de l'incertitude. L'inégalité (2.9) est la seule information que l'on a besoin de connaître sur  $\delta$ . Il faut noter que  $\lambda$  n'a pas besoin d'être petit, la seule condition est qu'il soit connu. Le but, ici, est de montrer que, connaissant  $\mathcal{V}$ ,  $\lambda$  et la constante  $k$ , on soit en mesure de trouver une loi de commande  $\varsigma$ , tel que la commande totale

$$u = \alpha(t, \mathbf{x}) + \varsigma$$

stabilise le système en présence d'incertitudes. Le design de  $\varsigma$  est connu sous le nom de resynthèse par Lyapunov (Lyapunov redesign). Considérons le système

$$\dot{\mathbf{x}} = \varphi + \psi\alpha + \psi(\varsigma + \delta)$$

qui est une version perturbée du système nominal (2.8). Le calcul de la dérivée de ce système le long de la trajectoire donne

$$\dot{\mathcal{V}} = \frac{\partial \mathcal{V}}{\partial t} + \frac{\partial \mathcal{V}}{\partial \mathbf{x}}(\varphi + \psi\alpha) + \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi(\varsigma + \delta) \leq -\mathcal{W} + \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi(\varsigma + \delta)$$

À cause de la **matching condition**, le terme incertain apparaît au même endroit que la commande, ce qui permet de choisir  $\varsigma$  afin d'annuler l'effet déstabilisant de  $\delta$  sur  $\dot{\mathcal{V}}$ . Avec la condition (2.9), on a

$$\begin{aligned} \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi(\varsigma + \delta) &\leq \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi\varsigma + \left\| \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi \right\| \|\delta\| \\ &\leq \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi\varsigma + \left\| \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi \right\| (\lambda + k\|\varsigma\|) \end{aligned}$$

Le choix

$$\varsigma = -\frac{h(t, \mathbf{x})}{1-k} \operatorname{sgn} \left( \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi \right)$$

avec  $h(t, \mathbf{x}) \geq \lambda(t, \mathbf{x})$ , donne

$$\begin{aligned} \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi(\varsigma + \delta) &\leq -\frac{h}{1-k} \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi \operatorname{sgn} \left( \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi \right) + \lambda \left\| \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi \right\| + \frac{kh}{1-k} \left\| \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi \right\| \\ &= \frac{-h}{1-k} \left\| \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi \right\| + \frac{kh}{1-k} \left\| \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi \right\| + \lambda \left\| \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi \right\| \\ &= (\lambda - h) \left\| \frac{\partial \mathcal{V}}{\partial \mathbf{x}}\psi \right\| \leq 0 \end{aligned}$$

Ainsi la dérivée  $\dot{\mathcal{V}}$ , le long de la trajectoire du système en boucle fermée est négative.

### 2.3.2 Amortissement non linéaire

Cette méthode s'applique dans le cas où tout ce que l'on sait de la perturbation est qu'elle est le produit d'une fonction connue  $\Delta(t, \mathbf{x})$  et d'un terme inconnu  $\delta(t, \mathbf{x}, u)$ . Ce dernier est borné, mais sa borne n'est pas connue. Cette situation ne permet pas de faire le design d'une commande assurant la stabilité du système. Néanmoins, on est en mesure de trouver une loi de commande qui assure, sans aucune information sur la borne (sauf qu'elle existe), des trajectoires bornées du système en boucle fermée (Krstić et al. 1995). Soit, à nouveau, le système

$$\dot{\mathbf{x}} = \varphi(t, \mathbf{x}) + \psi(t, \mathbf{x}) \left[ u + \Delta(t, \mathbf{x})\delta(t, \mathbf{x}, u) \right] \quad (2.10)$$

avec comme seule condition l'existence d'une borne supérieure  $\delta_0$  pour  $\|\delta\|$ . On considère que l'on connaît une loi de commande  $\alpha$  et une fonction de contrôle de Lyapunov  $\mathcal{V}$  (ayant les mêmes propriétés que précédemment), qui garantissent la stabilité du système nominal. Même si la borne supérieure de  $\|\delta\|$  n'est pas connue, il est possible de trouver une loi de commande ( $u = \alpha + \varsigma$ ) qui borne les trajectoires du système perturbé (en boucle fermée). En effet, si on calcule la dérivée de  $\mathcal{V}$  le long de la solution de (2.10), on obtient

$$\dot{\mathcal{V}} = \frac{\partial \mathcal{V}}{\partial t} + \frac{\partial \mathcal{V}}{\partial \mathbf{x}} \left[ \varphi + \psi \alpha \right] + \frac{\partial \mathcal{V}}{\partial \mathbf{x}} \psi \left[ \varsigma + \Delta \delta \right] \leq -\mathcal{W} + \frac{\partial \mathcal{V}}{\partial \mathbf{x}} \psi \left[ \varsigma + \Delta \delta \right]$$

Avec le choix

$$\varsigma = -m \frac{\partial \mathcal{V}}{\partial \mathbf{x}} \psi \|\Delta\|_2^2, \quad m > 0 \quad (2.11)$$

la dérivée devient

$$\begin{aligned} \dot{\mathcal{V}} &\leq -\mathcal{W} - m \left\| \frac{\partial \mathcal{V}}{\partial \mathbf{x}} \psi \right\|_2^2 \|\Delta\|_2^2 + \frac{\partial \mathcal{V}}{\partial \mathbf{x}} \psi \Delta \delta \\ &\leq -\mathcal{W} - m \left\| \frac{\partial \mathcal{V}}{\partial \mathbf{x}} \psi \right\|_2^2 \|\Delta\|_2^2 + \left\| \frac{\partial \mathcal{V}}{\partial \mathbf{x}} \psi \right\|_2 \|\Delta\|_2 \delta_0 \end{aligned}$$

où  $\delta_0$  est la borne supérieure inconnue de  $\|\delta\|$ . Le terme

$$-m \left\| \frac{\partial \mathcal{V}}{\partial \mathbf{x}} \psi \right\|_2^2 \|\Delta\|_2^2 + \left\| \frac{\partial \mathcal{V}}{\partial \mathbf{x}} \psi \right\|_2 \|\Delta\|_2 \delta_0$$

a pour maximum  $\delta_0/4m$ , qui correspond à

$$\left\| \frac{\partial \mathcal{V}}{\partial \mathbf{x}} \psi \right\|_2 \|\Delta\|_2 = \frac{\delta_0}{2m}$$

Cela donne

$$\dot{\mathcal{V}} \leq -\mathcal{W} + \frac{\delta_0}{4m}$$

$\mathcal{W}$  étant de classe  $\mathcal{K}_\infty$ , on est sûr d'avoir  $\dot{\mathcal{V}}$  négative à l'extérieur de la sphère définie par

$$\|\mathbf{x}\|_2 > \mathcal{W}^{-1} \left( \frac{\delta_0}{4m} \right)$$

Ce qui garantit que la solution du système, ainsi compensé, est bornée pour toutes les conditions initiales  $\mathbf{x}(t_0)$ . La commande donnée par l'équation (2.11) est connue sous le nom d'amortissement non linéaire.

## 2.4 Commande adaptative par Lyapunov

La commande adaptative est une commande dans laquelle le contrôleur est automatiquement ajusté afin de compenser des changements dans le procédé et/ou son environnement. Ce genre de problème peut être traité dans le cadre de la commande robuste (section 2.3), mais en présence de fortes perturbations, incertitudes du modèle ou paramètres à forte variation dans le temps, la commande adaptative présente un meilleur choix. Ce domaine, qui a vu le jour dans les années 50, a connu un grand essor ces trois dernières décennies et continue de représenter un domaine de recherche très actif.

Un contrôleur adaptatif n'est autre que la combinaison, d'un module d'estimation en ligne de certains paramètres inconnus, et d'un contrôleur qui se sert de ces paramètres. Cette combinaison donne lieu à deux approches différentes de conception et de commande : la commande adaptative indirecte et la commande adaptative directe.

### Commande adaptative indirecte

Dans cette approche (figure 2.1) ce sont les paramètres du procédé qui sont estimés en ligne, pour être ensuite utilisés dans le calcul de ceux du contrôleur. Le modèle<sup>11</sup> du procédé  $G_p(s, \theta_p)$  est paramétrisé par rapport à certains paramètres inconnus  $\theta_p$ . Un estimateur (en ligne) génère des estimés  $\hat{\theta}_p$  de  $\theta_p$  à chaque instant  $t$ , en utilisant les signaux d'entrée et de sortie ( $u$  et  $y$ ) du procédé. Dans la procédure de design, pour calculer les paramètres  $\theta_c$  du contrôleur, le modèle estimé du procédé  $G_p(s, \hat{\theta}_p)$  est considéré comme étant le vrai modèle, à chaque instant  $t$ .

### Commande adaptative directe

Contrairement à la méthode précédente, les paramètres du contrôleur sont directement estimés (figure 2.2), et on n'a nul besoin d'estimer ceux du procédé. Le modèle de ce dernier est directement paramétrisé en fonction des paramètres inconnus  $\theta_c$  du contrôleur. Les estimés  $\hat{\theta}_c$  servent directement à mettre à jour le contrôleur, sans aucun calcul intermédiaire.

---

<sup>11</sup>Afin de simplifier la présentation, des modèles linéaires seront considérés.

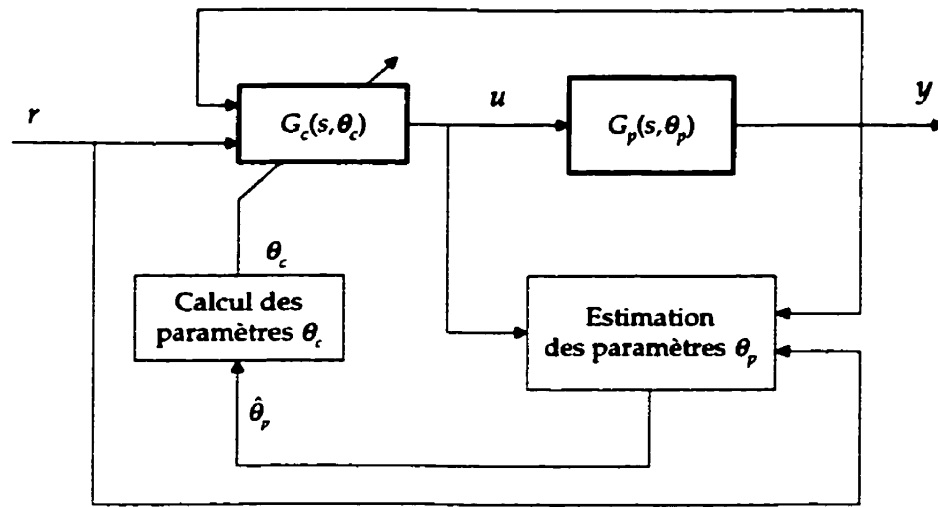


FIG. 2.1: Commande adaptative indirecte

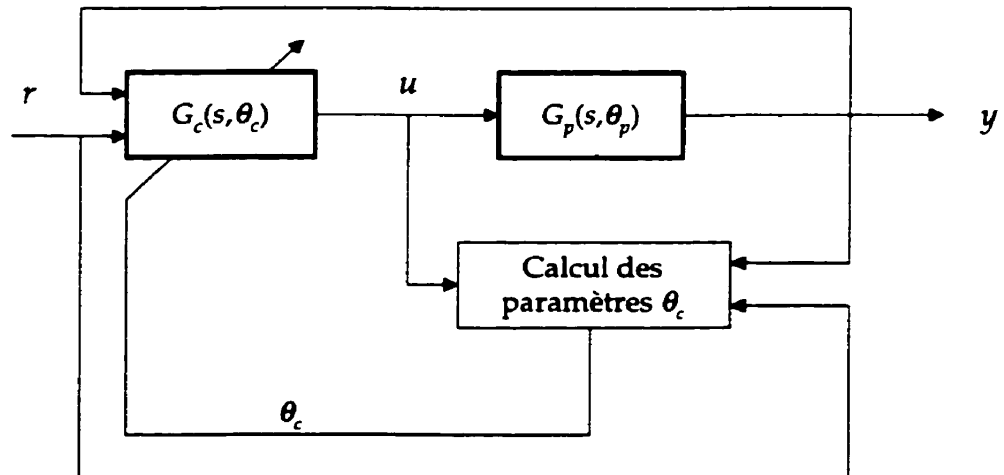


FIG. 2.2: Commande adaptative directe



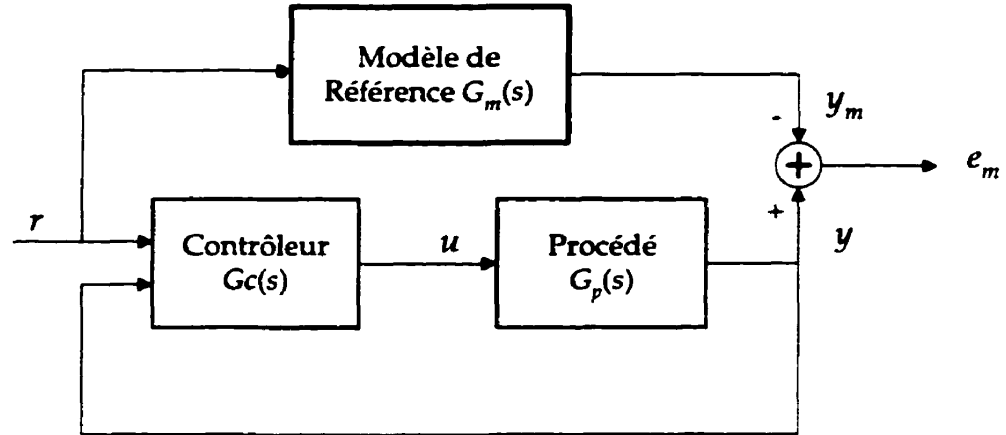


FIG. 2.3: Commande par Modèle de Référence

### 2.4.1 Commande adaptative par modèle de référence

Le principe d'une commande par modèle de référence (MRC) est illustré par la figure 2.3. L'objectif d'une telle commande est de trouver une loi de contrôle  $u$  qui change la structure et la dynamique du procédé (à paramètres connus) de façon à ce que ses propriétés entrée/sortie ( $r(t)$  vs  $y(t)$ ), en boucle fermée, coïncident avec celles d'un modèle de référence  $G_m(s)$  préalablement choisi. Si les paramètres du procédé sont inconnus, une loi d'identification est greffée au régulateur MRC. On parle, dans ce cas, de commande MRAC<sup>12</sup> (Parks 1966, Landau 1979, Åström & Wittenmark 1995, Ioannou & Sun 1996). Cette dernière, qui peut être directe ou indirecte, constitue l'une des structures les plus utilisées dans le domaine de la commande adaptative.

### 2.4.2 Principe de l'équivalence certaine

Le principe de deux approches (directe et indirecte) telles que présentées est conceptuellement simple. Le contrôleur est conçu pour un modèle à paramètres connus. Par la suite les paramètres inconnus du procédé sont estimés en ligne. L'idée consiste à

<sup>12</sup>Model Reference Adaptive Control.

considérer que l'on dispose du bon modèle en tout temps. À chaque instant  $t$ , la procédure de design du contrôleur  $G_c(s, \theta_c)$  considère les estimés,  $\hat{\theta}_p$  (dans le cas indirect) ou  $\hat{\theta}_c$  (dans le cas direct), comme s'ils étaient les vrais paramètres. Cette approche, largement dominante en commande adaptative, est connue sous le nom de "principe de l'équivalence certaine". Le libre choix, aussi bien de la méthode de design que celle de l'estimation, constitue le point fort de cette approche. On peut combiner différents estimateurs avec différentes lois de commande, pour obtenir un vaste choix de structures adaptatives. Toutefois, étant donné que la stabilité de ces structures est prouvée pour un procédé à paramètres connus<sup>13</sup>, le bon fonctionnement de la boucle n'est pas garanti dans une large plage. Par conséquent, les régulateurs adaptatifs, basés sur le principe de l'équivalence certaine<sup>14</sup>, peuvent rendre instable la plupart des structures adaptatives qui les utilisent (Ioannou & Kokotovic 1984, Ioannou 1986, Ioannou & Sun 1996).

La faiblesse des régulateurs adaptatifs basés sur le principe de l'équivalence certaine réside dans le fait que l'estimateur et le régulateur sont des entités indépendantes. Par conséquent, la dynamique de l'estimateur n'est pas considérée dans la procédure de conception. Si le système présente un comportement explosif (instable ou non linéaire) ou un fort couplage (agissant comme une perturbation), la lenteur de la convergence de l'adaptation, qui caractérise cette approche, peut entraîner une instabilisation irréversible du système (Krstić et al. 1995, Khalil 1996). Les contrôleurs basés sur la méthode directe de Lyapunov présentent une meilleure alternative.

### 2.4.3 Commande adaptative basée sur Lyapunov

L'idée principale de l'utilisation de la fonction de Lyapunov dans la commande adaptative, consiste à calculer une loi de commande et une loi de mise à jour des paramètres, afin de garantir que la dérivée d'une certaine fonction, définie positive et bien choisie, est non positive. L'approche consiste donc à trouver un triplet (fonction de Lyapunov, loi de commande, loi d'adaptation) (figure 2.4) qui répond aux spécifications. Les tâches

<sup>13</sup>Donc valable uniquement après convergence de l'estimation.

<sup>14</sup>Qui sont ceux couramment utilisés.

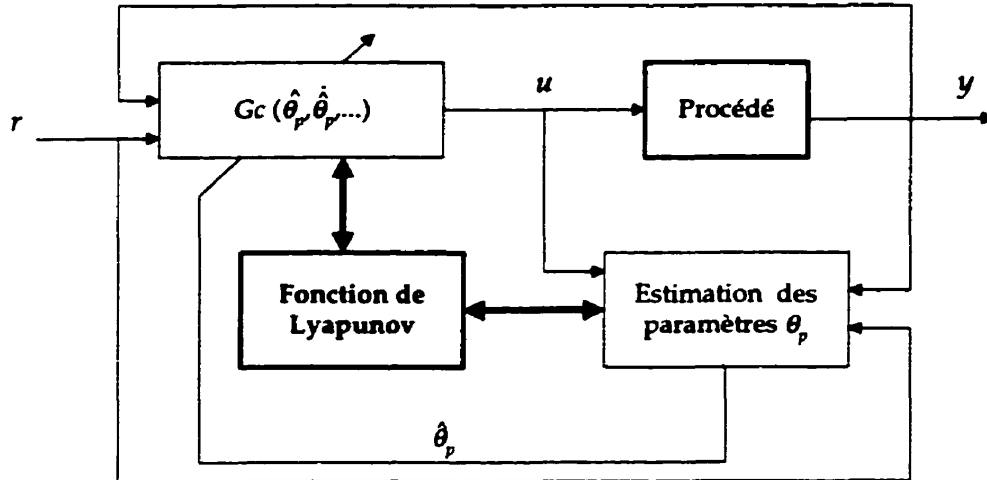


FIG. 2.4: Commande adaptative basée sur Lyapunov

s'effectuent simultanément, et la dynamique de l'estimation est prise en compte et/ou maîtrisée, afin d'éviter tout effet destructeur. C'est grâce à cette propriété que ce type de méthodes présente un intérêt pour la commande des systèmes non linéaires.

Malheureusement, bien qu'il soit l'un des premiers résultats en commande adaptative (Butchart & Shackloth 1965, Parks 1966), le design par Lyapunov est resté, jusqu'à récemment, limité aux systèmes linéaires. Cette limitation était principalement due à l'absence, pour les systèmes non linéaires, d'une méthode systématique de calcul de "bonnes" fonctions de Lyapunov.

Le design adaptatif par Lyapunov sera présenté ici comme une introduction (ou plutôt un cas particulier) de la méthode du **backstepping**. En effet, comme le verra dans le chapitre 3, le **backstepping** se réduit, pour les systèmes scalaires, à la procédure présentée ci-dessous.

### Principe

Il s'agit, pour un système scalaire, d'utiliser la fonction de Lyapunov pour calculer une loi de commande et une dynamique d'adaptation qui assurent la stabilité, en

présence d'incertitudes. On considère le cas simple du système non linéaire (scalaire) décrit par l'équation

$$\dot{x} = u + \varphi(x)^T \theta \quad (2.12)$$

où  $\theta$  est un vecteur de paramètres constants et inconnus. On désire trouver la loi de commande  $u(x, \theta)$  qui assure la stabilité de l'origine du système (2.12). Pour ce faire, on prend comme fonction de contrôle de Lyapunov (fcl), la forme quadratique

$$V_1(x) = \frac{1}{2}x^2 \quad (2.13)$$

qui est définie positive. Sa dérivée, évaluée le long de la solution de (2.12), s'écrit

$$\dot{V}_1 = x\dot{x} = x \left[ u + \varphi(x)^T \theta \right]$$

Le choix de la loi de commande

$$u(x, \theta) = -\varphi(x)^T \theta - k_1 x \quad (2.14)$$

où  $k_1 > 0$ , permet de rendre  $\dot{V}_1$  négative. En effet, avec un tel choix, on a

$$\dot{V}_1 = -k_1 x^2 \leq 0$$

Deux cas se présentent alors :

- $\theta$  est connu : la loi de commande (2.14) peut être réalisée, ce qui permet de stabiliser l'origine du système. L'objectif du design est ainsi atteint.
- $\theta$  est inconnu : le contrôleur décrit par (2.14) ne peut être réalisé. On se propose de le remplacer par son équivalent (basé sur le principe de l'équivalence certaine), où, au lieu de  $\theta$ , on utilise son estimé  $\hat{\theta}$  dans (2.14)

$$u = -\varphi(x)^T \hat{\theta} - k_1 x \quad (2.15)$$

Remplaçons (2.15) dans (2.12), on obtient

$$\begin{aligned} \dot{x} &= -k_1 x + \varphi(x)^T (\theta - \hat{\theta}) \\ &= -k_1 x + \varphi(x)^T \tilde{\theta} \end{aligned}$$

où,  $\tilde{\theta}$  représente l'erreur d'estimation ( $\theta - \hat{\theta}$ ). La dérivée de la fcl donnée par (2.13) s'écrit avec ces choix

$$\begin{aligned}\dot{V}_1 &= x \left[ -k_1 x + \varphi(x)^T \tilde{\theta} \right] \\ &= -k_1 x^2 + x \varphi(x)^T \tilde{\theta}\end{aligned}$$

Étant donné que l'expression contient le terme inconnu  $\tilde{\theta}$ , son signe est indéfini, et aucune conclusion ne peut être tirée quant à la stabilité du système. Afin d'en savoir plus sur cette stabilité, on construit un contrôleur dynamique, en augmentant (2.15) d'une loi de mise à jour pour l'estimé  $\hat{\theta}$ . Un bon choix de cette loi, qui doit assurer la stabilité de l'ensemble, est à déterminer. On définit alors une nouvelle fonction Lyapunov en ajoutant, à la fonction initiale donnée par (2.13), un terme quadratique en l'erreur d'estimation  $\tilde{\theta}$

$$V_2(x, \tilde{\theta}) = \frac{1}{2} x^2 + \frac{1}{2} \tilde{\theta}^T \Gamma^{-1} \tilde{\theta}$$

où,  $\Gamma$  est une matrice (définie positive) qui représente le gain d'adaptation. La dérivée de cette fonction devient alors

$$\begin{aligned}\dot{V}_2 &= x \dot{x} + \tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}} \\ &= -k_1 x^2 + x \varphi(x)^T \tilde{\theta} + \tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}} \\ &= -k_1 x^2 + \tilde{\theta}^T \Gamma^{-1} \left[ \dot{\tilde{\theta}} + \tau \right]\end{aligned}$$

où,

$$\tau = \Gamma x \varphi(x)$$

La dérivée reste toujours indéfinie, mais cette fois, le degré de liberté, offert par le libre choix de la dynamique de mise à jour, permet de choisir cette dernière de façon à annuler le second terme de l'équation. Le choix

$$\dot{\tilde{\theta}} = -\tilde{\theta} = \tau \tag{2.16}$$

permet d'obtenir

$$\dot{V}_2 = -k_1 x^2 \leq 0$$

Le système résultant est donné par (2.12) avec la loi de commande (2.15), et la loi de mise à jour (2.16)

$$\begin{aligned}\dot{x} &= -k_1 x + \varphi(x)^T (\theta - \hat{\theta}) \\ \dot{\hat{\theta}} &= \Gamma x \varphi(x)\end{aligned}$$

## 2.5 Conclusion

À la diversité des méthodes d'analyse des systèmes non linéaires, correspond, en toute évidence, une diversité des méthodes de design. En fonction du type et de l'importance du phénomène non linéaire présent dans le système, ces méthodes peuvent être approximatives ou exactes, aboutissant à conditions de "bon fonctionnement" nécessaires ou suffisantes.

Encore une fois, tout comme pour l'analyse, le design par Lyapunov se démarque par sa supériorité, notamment dans le contexte adaptatif. Cette méthode a été présentée, dans ce chapitre, pour le cas simple des systèmes scalaires. Sa généralisation aux systèmes d'ordre arbitraire a été rendue possible, grâce à la technique récursive du backstepping. Pour des systèmes non linéaires de tout ordre, à paramètres connus ou inconnus, le backstepping constitue "la solution". À cause de son importance pour les chapitres subséquents, cette méthode sera présentée, en détail, dans les chapitres 3 et 4.

# Chapitre 3

## Design par backstepping

Le design d'un contrôleur pour un système non linéaire de la forme

$$\dot{\mathbf{x}} = \varphi(\mathbf{x}, \boldsymbol{\theta}, u, t)$$

où le vecteur d'état  $\mathbf{x}$  est de dimension élevée, peut souvent s'avérer une tâche difficile, voire impossible. La technique du **backstepping** offre une méthode systématique pour répondre à ce type de problème. Elle combine la notion de fonction de contrôle de Lyapunov (fcl) avec une procédure récursive de design. Cela permet de surmonter l'obstacle de la dimension et d'exploiter la souplesse de conception dans le cas scalaire pour résoudre les problèmes de commande pour des systèmes d'ordre plus élevé. Ne faisant pas nécessairement appel à la linéarisation, le **backstepping** permet, quand il y en a, de conserver les non-linéarités utiles qui, souvent, aident à conserver des valeurs finies du vecteur d'état. Cette technique suppose que l'on est en mesure de trouver, au moins pour un système scalaire, une loi de commande  $u$  et une fonction de contrôle de Lyapunov  $\mathcal{V}(\mathbf{x})$  qui stabilisent son origine.

Le **backstepping** a été développé par Kanellakopoulos et al. (1991) et inspiré par les travaux de Feurer & Morse (1978) d'une part et Tsinias (1989) et Kokotović & Sussmann (1989) d'autre part. L'arrivée de cette méthode a donné un nouveau souffle à la commande adaptative des systèmes non linéaires, qui malgré les grands progrès réalisés, manquait d'approches générales. Le **backstepping** présente une alternative prometteuse aux méthodes basées sur l'équivalence certaine. Il se base sur la deuxième méthode de

Lyapunov, dont il combine le choix de la fonction avec celui des lois de commande et d'adaptation. Ceci lui permet, en plus de la tâche pour laquelle le contrôleur est conçu (poursuite et/ou régulation), de garantir, en tout temps, la stabilité globale du système compensé.

Cette approche n'est pas la seule dans le domaine de la commande adaptative des systèmes non linéaires. D'autres méthodes, notamment celles basées sur la linéarisation (Singh & Rugh 1972, Isidori 1981, Sastry & Isidori 1989), ainsi que celles s'inspirant de la commande robuste (Corless & Leitmann 1981, Barmish & Leitmann 1982, Qu 1993) ont été proposées. Toutefois, toutes ces méthodes se trouvent confrontées au problème de l'applicabilité. Une méthode qui donne de bons résultats sur une classe de systèmes, peut avoir des conséquences catastrophiques sur une autre classe, même très peu différente de la première.

Dans le design par **backstepping**, aucune contrainte n'est imposée à la caractéristique non linéaire du système<sup>1</sup>. Ce dernier doit, cependant, se présenter sous la forme dite **paramétrique pure**. Les équations d'un tel système sont données par

$$\begin{aligned}
 \dot{x}_1 &= \varphi_1(x_1)^T \boldsymbol{\theta} + \psi_1(x_1)x_2 \\
 \dot{x}_2 &= \varphi_2(x_1, x_2)^T \boldsymbol{\theta} + \psi_2(x_1, x_2)x_3 \\
 &\vdots \\
 \dot{x}_{n-1} &= \varphi_{n-1}(x_1, \dots, x_{n-1})^T \boldsymbol{\theta} + \psi_{n-1}(x_1, \dots, x_{n-1})x_n \\
 \dot{x}_n &= \varphi_n(x_1, \dots, x_{n-1}, x_n)^T \boldsymbol{\theta} + \psi_n(x_1, \dots, x_{n-1}, x_n)u \\
 y &= x_1
 \end{aligned} \tag{3.1}$$

où  $\boldsymbol{\theta}$  est un vecteur de paramètres constants (connus ou inconnus). Les  $\psi_i$  et les  $\varphi_i$  sont des fonctions non linéaires connues, avec  $\varphi_i(\mathbf{0}) = \mathbf{0}$  et  $\psi_n(\mathbf{x}) \neq 0, \forall \mathbf{x} \in \mathcal{R}^n$ .

Dans le cas où le système à commander fait partie de la classe plus restrictive des systèmes dits à forme **paramétrique stricte** ( $\psi_i = 1$ ), les propriétés de poursuite et de régulation obtenues sont globales (Kanellakopoulos et al. 1991, Krstić et al. 1995). Pour

<sup>1</sup>Ce qui n'est pas toujours le cas avec les autres méthodes.



les systèmes à forme **paramétrique pure**, l'étendue de la validité de ces propriétés dépend du domaine de définition des transformations géométriques (difféomorphismes) qui permettent de ramener le système sous la forme **stricte**. Dans le cas où ce domaine est  $\mathcal{R}^n$ , les propriétés sont également globales. Il faut noter que cette restriction (structure **paramétrique pure** ou **stricte**) n'est nullement due à l'aspect adaptatif de la méthode. Elle est imposée par la procédure récursive de design et reste en vigueur même dans le cas non adaptatif.

La version de base du **backstepping** concerne la commande non adaptative par retour d'état, en l'absence d'incertitudes. Ce cas très particulier servira à introduire la procédure récursive de design. Par la suite, différentes variantes de la méthode seront présentées. La robustification du **backstepping**, qui consiste à l'augmenter de termes d'amortissements non linéaires, sera abordée dans la section 3.3, alors que le problème de la commande par retour de sortie (**backstepping** avec observateur) sera présenté dans la section 3.4. La commande non linéaire adaptative (**backstepping** adaptatif) sera détaillée dans le chapitre 4, qui lui sera entièrement consacré.

**Remarque 3.1 (Simplification)** — Pour toutes les méthodes qui seront présentées, la procédure de design restera la même, pour les systèmes d'ordre  $n \geq 3$ . Afin de simplifier les expressions, les systèmes utilisés seront d'ordre 3. Les résultats généraux (ordre  $n$ ) seront toutefois donnés. Il faut noter que le nombre des étapes, nécessaires à la construction de la commande, de la fcl et éventuellement de la loi d'adaptation, est égal à l'ordre du système (Kanellakopoulos et al. 1991, Åström & Wittenmark 1995, Krstić et al. 1995, Khalil 1996).

□

### 3.1 Algorithme de base

Afin d'illustrer le principe de la méthode du **backstepping**, on considère le cas des systèmes non linéaires de la forme

$$\dot{x}_1 = \varphi_1^T(x_1)\vartheta + \psi_1(x_1)x_2 \quad (3.2)$$

$$\dot{x}_2 = \varphi_2^T(x_1, x_2)\vartheta + \psi_2(x_1, x_2)x_3 \quad (3.3)$$

$$\dot{x}_3 = \varphi_3^T(x_1, x_2, x_3)\vartheta + \psi_3(x_1, x_2, x_3)u \quad (3.4)$$

Le vecteur des paramètres  $\vartheta$  est supposé connu<sup>2</sup>. On désire faire suivre à la sortie  $y = x_1$  le signal de référence  $y_r$ , où  $y_r$ ,  $\dot{y}_r$ ,  $\ddot{y}_r$  et  $y_r^{(3)}$  sont supposées connues et uniformément bornées. Le système étant du troisième ordre, le design s'effectue en trois étapes.

**Étape 1** — On considère d'abord l'équation (3.2), où la variable d'état  $x_2$  est traitée comme une commande<sup>3</sup> et l'on définit la première valeur désirée

$$(x_1)_d \triangleq \alpha_0 = y_r$$

La première variable d'erreur se définit par

$$\varepsilon_1 = x_1 - \alpha_0 \quad (3.5)$$

Avec ces variables, le système (3.2) s'écrit

$$\begin{aligned} \dot{\varepsilon}_1 &= \dot{x}_1 - \dot{\alpha}_0 \\ &= \varphi_1^T\vartheta + \psi_1x_2 - \dot{\alpha}_0 \end{aligned} \quad (3.6)$$

Pour un tel système, il a été montré (**Exemple 2.1**) que la fonction quadratique

$$\mathcal{V}_1(\varepsilon_1) = \frac{1}{2}\varepsilon_1^2$$

constitue un bon choix de fcl. Sa dérivée, le long de la solution de (3.6), est donnée par

$$\begin{aligned} \dot{\mathcal{V}}_1 &= \varepsilon_1\dot{\varepsilon}_1 \\ &= \varepsilon_1 \left[ \varphi_1^T\vartheta + \psi_1x_2 - \dot{\alpha}_0 \right] \end{aligned}$$

<sup>2</sup>Les paramètres seront notés  $\vartheta$  quand ils sont connus et  $\theta$  quand ils sont inconnus

<sup>3</sup>Il faut se rappeler que  $x_2$  n'est guère une commande, mais une variable d'état interne, qui ne peut être forcée à prendre instantanément une valeur désirée. L'appellation "commande" n'est qu'un subterfuge de design.

Un choix judicieux de  $x_2$  rendrait  $\dot{\mathcal{V}}_1$  négative et assurerait la stabilité de l'origine du sous-système décrit par (3.6). Prenons comme valeur de  $x_2$ , la fonction  $\alpha_1$ , telle que

$$\psi_1 \alpha_1 + \varphi_1^T \boldsymbol{\vartheta} - \dot{\alpha}_0 = -k_1 \varepsilon_1$$

où  $k_1 > 0$  est un paramètre de design. Cela donne

$$(x_2)_d \triangleq \alpha_1 = \frac{1}{\psi_1} \left[ -k_1 \varepsilon_1 - \varphi_1^T \boldsymbol{\vartheta} + \dot{\alpha}_0 \right] \quad (3.7)$$

et la dérivée s'écrit

$$\dot{\mathcal{V}}_1 = -k_1 \varepsilon_1^2 \leq 0$$

d'où la stabilité asymptotique de l'origine de (3.6).

**Étape 2** — On considère le sous-système (3.2)–(3.3) et l'on définit la nouvelle variable d'erreur

$$\varepsilon_2 = x_2 - \alpha_1 \quad (3.8)$$

qui représente l'écart entre la variable d'état  $x_2$  et sa valeur désirée  $\alpha_1$ . À cause du fait que  $x_2$  ne peut être forcée à prendre instantanément une valeur désirée, en l'occurrence  $\alpha_1$ , l'erreur  $\varepsilon_2$  n'est pas, instantanément, nulle. Le design dans cette étape consiste, alors, à la forcer à s'annuler avec une certaine dynamique, choisie au préalable.

Les équations du système à commander, dans l'espace  $(\varepsilon_1, \varepsilon_2)$ , s'écrivent

$$\dot{\varepsilon}_1 = \varphi_1^T \boldsymbol{\vartheta} - \dot{\alpha}_0 + \psi_1 (\varepsilon_2 + \alpha_1) \quad (3.9)$$

$$\dot{\varepsilon}_2 = \varphi_2^T \boldsymbol{\vartheta} - \dot{\alpha}_1 + \psi_2 x_3 \quad (3.10)$$

pour lequel on choisit comme fonction de Lyapunov

$$\mathcal{V}_2(\varepsilon_1, \varepsilon_2) = \mathcal{V}_1 + \frac{1}{2} \varepsilon_2^2 \quad (3.11)$$

Cette dernière a pour dérivée, le long de la solution de (3.9)–(3.10)

$$\begin{aligned}
\dot{V}_2(\varepsilon_1, \varepsilon_2) &= \dot{V}_1 + \varepsilon_2 \dot{\varepsilon}_2 \\
&= \varepsilon_1 \left[ \boldsymbol{\varphi}_1^T \boldsymbol{\vartheta} + \psi_1(\alpha_1 + \varepsilon_2) - \dot{\alpha}_0 \right] + \varepsilon_2 \left[ \boldsymbol{\varphi}_2^T \boldsymbol{\vartheta} + \psi_2 x_3 - \dot{\alpha}_1 \right] \\
&= \varepsilon_1 \left[ \boldsymbol{\varphi}_1^T \boldsymbol{\vartheta} + \psi_1 \alpha_1 - \dot{\alpha}_0 \right] + \varepsilon_2 \left[ \boldsymbol{\varphi}_2^T \boldsymbol{\vartheta} + \psi_1 \varepsilon_1 + \psi_2 x_3 - \dot{\alpha}_1 \right] \\
&= -k_1 \varepsilon_1^2 + \varepsilon_2 \left[ \boldsymbol{\varphi}_2^T \boldsymbol{\vartheta} + \psi_1 \varepsilon_1 + \psi_2 x_3 - \dot{\alpha}_1 \right]
\end{aligned}$$

Le choix de la valeur désirée de  $x_3$  devient évident. Ce dernier est donné par

$$(x_3)_d \triangleq \alpha_2 = \frac{1}{\psi_2} \left[ \dot{\alpha}_1 - \psi_1 \varepsilon_1 - \boldsymbol{\varphi}_2^T \boldsymbol{\vartheta} - k_2 \varepsilon_2 \right] \quad (3.12)$$

où  $k_2 > 0$ , avec  $\dot{\alpha}_1$  calculée analytiquement

$$\dot{\alpha}_1 = \frac{\partial \alpha_1}{\partial x_1} \dot{x}_1 + \frac{\partial \alpha_1}{\partial y_r} \dot{y}_r + \frac{\partial \alpha_1}{\partial \dot{y}_r} \ddot{y}_r$$

Un tel choix permet de réduire la dérivée à

$$\begin{aligned}
\dot{V}_2 &\leq -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 \\
&\leq 0
\end{aligned}$$

ce qui assure la stabilité asymptotique de l'origine de (3.9)–(3.10).

**Étape 3** — Le système (3.2)–(3.4) est maintenant considéré dans sa globalité. La variable d'erreur

$$\varepsilon_3 = x_3 - \alpha_2 \quad (3.13)$$

est définie, ce qui permet d'écrire les équations du système, dans l'espace des erreurs  $(\varepsilon_1, \varepsilon_2, \varepsilon_3)$

$$\dot{\varepsilon}_1 = \boldsymbol{\varphi}_1^T \boldsymbol{\vartheta} - \dot{\alpha}_0 + \psi_1(\varepsilon_2 + \alpha_1) \quad (3.14)$$

$$\dot{\varepsilon}_2 = \boldsymbol{\varphi}_2^T \boldsymbol{\vartheta} - \dot{\alpha}_1 + \psi_2(\varepsilon_3 + \alpha_2) \quad (3.15)$$

$$\dot{\varepsilon}_3 = \boldsymbol{\varphi}_3^T \boldsymbol{\vartheta} - \dot{\alpha}_2 + \psi_3 u \quad (3.16)$$

Avec comme fonction de Lyapunov

$$\mathcal{V}_3(\varepsilon_1, \varepsilon_2, \varepsilon_3) = \mathcal{V}_2 + \frac{1}{2}\varepsilon_3^2 \quad (3.17)$$

la dérivée, le long de la solution de (3.14)–(3.16), devient

$$\begin{aligned} \dot{\mathcal{V}}_3(\varepsilon_1, \varepsilon_2, \varepsilon_3) &= \dot{\mathcal{V}}_2 + \varepsilon_3 \dot{\varepsilon}_3 \\ &= -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 + \varepsilon_3 \left[ \psi_3 u + \psi_2 \varepsilon_2 + \varphi_3^T \vartheta - \dot{\alpha}_2 \right] \end{aligned}$$

À présent, on est en présence de la vraie commande<sup>4</sup>. Un bon choix de celle-ci est donné par

$$u = \frac{1}{\psi_3} \left[ \dot{\alpha}_2 - \psi_2 \varepsilon_2 - \varphi_3^T \vartheta - k_3 \varepsilon_3 \right] \quad (3.18)$$

où  $k_3 > 0$  et  $\dot{\alpha}_2$  est également calculée analytiquement,

$$\dot{\alpha}_2 = \frac{\partial \alpha_2}{\partial x_1} \dot{x}_1 + \frac{\partial \alpha_2}{\partial x_2} \dot{x}_2 + \frac{\partial \alpha_2}{\partial y_r} \dot{y}_r + \frac{\partial \alpha_2}{\partial \dot{y}_r} \ddot{y}_r + \frac{\partial \alpha_2}{\partial \ddot{y}_r} y_r^{(3)}$$

Avec ce choix, on a

$$\begin{aligned} \dot{\mathcal{V}}_3(\varepsilon_1, \varepsilon_2, \varepsilon_3) &\leq -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - k_3\varepsilon_3^2 \\ &\leq 0 \end{aligned}$$

d'où la stabilité asymptotique de l'origine de (3.14)–(3.16). Ceci se traduit par la stabilité, en boucle fermée, du système originel (3.2)–(3.4) et la régulation à zéro de l'erreur de poursuite  $y - y_r$ . Les deux principaux objectifs du design sont alors atteints.

**Remarque 3.2 (Paramètres  $k_i$ )** — Les paramètres de design  $k_i$  sont directement liés à position de pôles de la boucle fermée. Leur choix permet de faire un placement des pôles, fixant ainsi la dynamique en régulation de cette boucle.

□

---

<sup>4</sup>Qui, contrairement à  $x_2$  et  $x_3$ , peut être instantanément forcée à prendre n'importe quelle valeur désirée (physiquement réalisable).

**Exemple 3.1** — Soit à stabiliser l'origine du système

$$\dot{x}_1 = x_1^2 - x_1^3 + x_2 \quad (3.19)$$

$$\dot{x}_2 = x_3 \quad (3.20)$$

$$\dot{x}_3 = u \quad (3.21)$$

**Étape 1** — Le sous-système (3.19) est considéré en premier lieu. Étant donné que  $\alpha_0 = 0$ , on prend comme fonction de contrôle de Lyapunov

$$V_1(x_1) = \frac{1}{2}x_1^2$$

Sa dérivée le long de la trajectoire (3.19) est donnée par

$$\dot{V}_1(x_1) = x_1(x_1^2 - x_1^3 + x_2)$$

Avec le choix (qui préserve les non-linéarités utiles, i.e.  $-x_1^3$ )

$$(x_2)_d \triangleq \alpha_1 = -x_1^2 - x_1$$

la dérivée s'écrit

$$\begin{aligned} \dot{V}_1(x_1) &= x_1(x_1^2 - x_1^3 - x_1^2 - x_1) \\ &= -x_1^4 - x_1^2 \\ &\leq -x_1^2 \end{aligned}$$

Ce qui implique que l'origine de (3.19) est globalement asymptotiquement stable.

**Étape 2** — Cette fois, on considère (3.19)–(3.20) et l'on définit la nouvelle variable

$$\begin{aligned} \varepsilon_2 &= x_2 - \alpha_1 \\ &= x_2 + x_1^2 + x_1 \end{aligned}$$

Le sous-système (3.19)–(3.20) s'écrit alors

$$\dot{x}_1 = -x_1 - x_1^3 + \varepsilon_2 \quad (3.22)$$

$$\begin{aligned} \dot{\varepsilon}_2 &= \dot{x}_2 - \dot{\alpha}_1 \\ &= x_3 - \dot{\alpha}_1 \end{aligned} \quad (3.23)$$

Si l'on prend pour fcl

$$\mathcal{V}_2(x_1, \varepsilon_2) = \frac{1}{2}x_1^2 + \frac{1}{2}\varepsilon_2^2$$

sa dérivée le long de la solution de (3.22)–(3.23) est donnée par

$$\begin{aligned}\dot{\mathcal{V}}_2(x_1, \varepsilon_2) &= x_1\dot{x}_1 + \varepsilon_2\dot{\varepsilon}_2 \\ &= -x_1^4 - x_1^2 + \varepsilon_2 \left[ x_3 - \dot{\alpha}_1 + x_1 \right]\end{aligned}$$

Il suffit, à présent, de choisir la valeur désirée  $\alpha_2$  de  $x_3$  pour rendre négative cette dérivée. Un tel choix est donné par

$$(x_3)_d \triangleq \alpha_2 = \dot{\alpha}_1 - x_1 - k_2\varepsilon_2$$

où

$$\dot{\alpha}_1 = -(2x_1 + 1)\dot{x}_1 = -(2x_1 + 1)(x_1^2 - x_1^3 + x_2)$$

Ce qui donne (pour  $k_2 = 1$ )

$$\alpha_2 = 2x_1^4 - x_1^3 - 2x_1^2 - 2x_1 - 2x_1x_2 - 2x_2$$

**Étape 3** — Tout le système (3.19)–(3.21) est maintenant considéré. On définit la nouvelle variable d'erreur

$$\varepsilon_3 = x_3 - \alpha_2$$

Ce qui permet d'écrire le système (3.19)–(3.21) sous la forme

$$\dot{x}_1 = -x_1^2 - x_1^3 + \alpha_1 + \varepsilon_2 \tag{3.24}$$

$$\dot{\varepsilon}_2 = \alpha_2 + \varepsilon_3 - \dot{\alpha}_1 \tag{3.25}$$

$$\dot{\varepsilon}_3 = u - \dot{\alpha}_2 \tag{3.26}$$

Le choix de la fcl

$$\mathcal{V}_3(x_1, \varepsilon_2, \varepsilon_3) = \frac{1}{2}x_1^2 + \frac{1}{2}\varepsilon_2^2 + \frac{1}{2}\varepsilon_3^2$$

et le calcul de sa dérivée le long de la trajectoire de (3.24)–(3.26), de la même manière qu'à l'étape 2, permet d'obtenir la commande qui assure la stabilité asymptotique de l'origine du système. Après calcul et simplification (avec  $k_1 = k_2 = k_3 = 1$ ), cette commande est donnée par

$$\begin{aligned}
 u &= \frac{\partial \alpha_2}{\partial x_1} (x_1^2 - x_1^3 + x_2) + \frac{\partial \alpha_2}{\partial x_2} x_3 - (x_2 - \alpha_1) - (x_3 - \alpha_2) \\
 &= (8x_1^3 - 3x_1^2 - 2x_2 - 2)(x_1^2 - x_1^3 + x_2) - (2x_1 + 1)x_3 \\
 &\quad - (x_2 + x_1^2 + x_1) - (x_3 - 2x_1^4 + x_1^3 + 2x_1^2 + 2x_1 + 2x_1x_2 + 2x_2) \\
 &= -(8x_1^5 - 11x_1^4 + x_1^3 - x_1^2 + 5x_1 + 3)x_1 \\
 &\quad + (10x_1^3 - 5x_1^2 - 2x_1 - 2x_2 - 5)x_2 - 2(x_1 + 1)x_3
 \end{aligned}$$

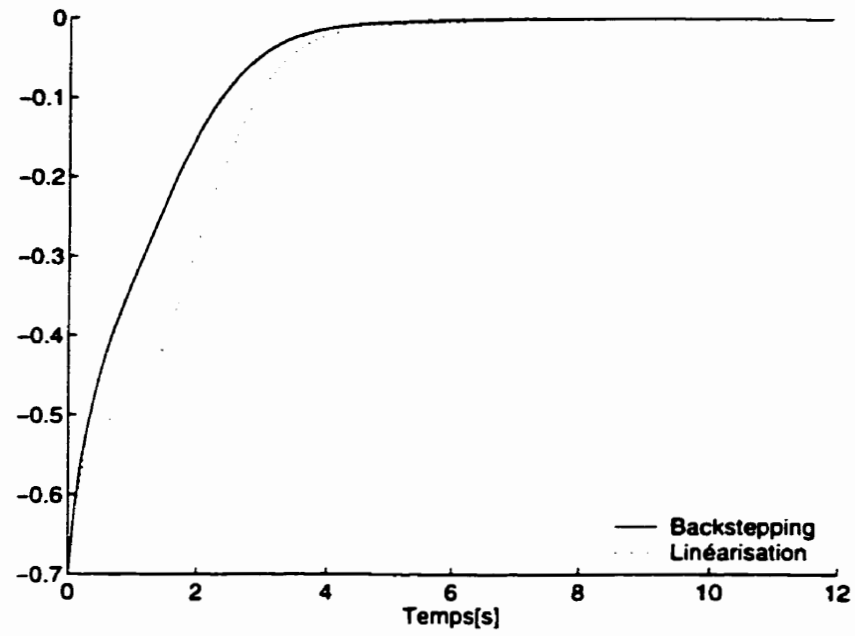
Avec comme fonction de Lyapunov

$$\mathcal{V}_3 = \frac{1}{2}x_1^2 + \frac{1}{2} \left[ x_2 + x_1^2 + x_1 \right]^2 + \frac{1}{2} \left[ x_3 - 2x_1^4 + x_1^3 + 2x_1^2 + 2x_1 + 2x_1x_2 + 2x_2 \right]^2$$

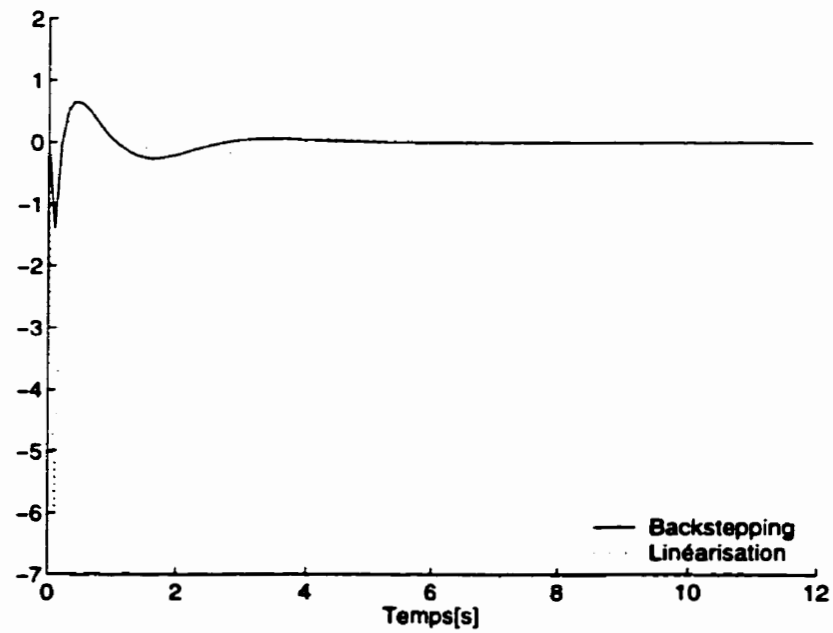
### Simulation

Une comparaison entre le contrôleur obtenu par la technique du **backstepping** et celui que générerait la méthode par linéarisation exacte (section 2.1.2) est donnée par les figures 3.1 (a) & (b). Partant d'une position initiale non nulle  $x_1(0) = .7$ , le but du design consiste à ramener le système à sa position d'équilibre  $x_1 = 0$  et l'y maintenir. On constate que, pour l'approche par linéarisation, l'élimination systématique de toutes les non-linéarités se traduit par un effort inutilement élevé de l'actionneur, puisque le **backstepping** permet, avec une commande beaucoup plus douce, d'obtenir des performances nettement supérieures.





(a) Sorties



(b) Commandes

FIG. 3.1: Backstepping vs Linéarisation

### 3.2 Système d'ordre $n$

L'application récursive du **backstepping** permet l'extension de la procédure de design aux systèmes triangulaires de la forme

$$\begin{aligned}\dot{x}_1 &= \varphi_1^T(x_1)\boldsymbol{\vartheta} + \psi_1(x_1)x_2 \\ \dot{x}_2 &= \varphi_2^T(x_1, x_2)\boldsymbol{\vartheta} + \psi_2(x_1, x_2)x_3 \\ &\vdots \\ \dot{x}_{n-1} &= \varphi_{n-1}^T(x_1, x_2, \dots, x_{n-1})\boldsymbol{\vartheta} + \psi_{n-1}(x_1, x_2, \dots, x_{n-1})x_n \\ \dot{x}_n &= \varphi_n^T(x_1, x_2, \dots, x_{n-1}, x_n)\boldsymbol{\vartheta} + \psi_n(x_1, x_2, \dots, x_{n-1}, x_n)u\end{aligned}$$

où

$$\varphi_i(\mathbf{0}) = \mathbf{0} \quad \text{et} \quad \psi_i \neq 0, \quad 1 \leq i \leq n$$

dans le domaine d'intérêt  $\mathcal{D}$ . La procédure de design commence avec la première équation. Le changement de variable adéquat à chaque étape  $i$  permet d'appliquer le **backstepping** récursivement, en rajoutant l'équation  $i + 1$ . Partant de  $\alpha_0$ , on construit les différents  $\alpha_i$  et  $\mathcal{V}_i$ . Ce qui résulte en

$$\begin{aligned}(x_1)_d &\triangleq \alpha_0 = y_r \\ (x_{i+1})_d &\triangleq \alpha_i = \frac{1}{\psi_i} \left[ \sum_{k=1}^{i-1} \left( \frac{\partial \alpha_{i-1}}{\partial x_k} \psi_k x_{k+1} + \frac{\partial \alpha_{i-1}}{\partial y_r^{(k-1)}} y_r^{(k)} \right) - \psi_{i-1} \varepsilon_{i-1} - k_i \varepsilon_i - \mathbf{w}_i^T \boldsymbol{\vartheta} \right]\end{aligned}$$

où

$$\begin{aligned}i &= 1, \dots, n \\ \varepsilon_i &= x_i - \alpha_{i-1} \\ \mathbf{w}_i &= \boldsymbol{\varphi}_i - \sum_{k=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial x_k} \boldsymbol{\varphi}_k\end{aligned}$$

Les différentes fonctions de Lyapunov sont données par

$$\mathcal{V}_i = \frac{1}{2} \sum_{j=1}^i \left[ x_j - \alpha_{j-1} \right]^2$$

La commande  $u$ , qui permet d'atteindre les objectifs du design pour le système global, est donnée par la dernière commande virtuelle  $\alpha_n$ .

**Remarque 3.3 (Incertitudes)** — Malgré sa souplesse et son efficacité, le **backstepping**, dans sa forme de base, souffre, en présence des incertitudes, des mêmes limitations que les méthodes "classiques". Il offre, cependant, l'avantage de se combiner facilement avec des techniques de robustification (tel que l'amortissement non linéaire), afin d'améliorer la stabilité des boucles perturbées.

□

### 3.3 Backstepping Robuste

L'efficacité de la méthode du **backstepping** se manifeste en présence d'incertitudes. Dans de telles circonstances, aucune autre méthode systématique de design n'existe (Krstić et al. 1995). L'idée consiste à combiner la méthode de l'amortissement non linéaire (section 2.3.2), pratique pour les systèmes simples, avec la procédure récursive du **backstepping** (section 3.1), afin d'augmenter la robustesse des systèmes complexes perturbés. Cette approche permet de surmonter la contrainte de la **matching condition**, que doivent vérifier les incertitudes dans le cas de l'amortissement non linéaire simple (section 2.3.2). Cette propriété rend le **backstepping robuste** applicable à une plus large classe de problèmes.

#### 3.3.1 Principe

Soit à commander le système

$$\dot{x}_1 = \varphi_1(x_1)^T \boldsymbol{\vartheta} + \psi_1(x_1) \left[ x_2 + \Delta_1(t, x_1) \delta_1 \right] \quad (3.27)$$

$$\dot{x}_2 = \varphi_2(x_1, x_2)^T \boldsymbol{\vartheta} + \psi_2(x_1, x_2) \left[ x_3 + \Delta_2(t, x_1, x_2) \delta_2 \right] \quad (3.28)$$

$$\dot{x}_3 = \varphi_3(x_1, x_2, x_3)^T \boldsymbol{\vartheta} + \psi_3(x_1, x_2, x_3) \left[ u + \Delta_3(t, x_1, x_2, x_3) \delta_3 \right] \quad (3.29)$$

qui est une version perturbée de (3.2)–(3.4).  $\Delta_1$ ,  $\Delta_2$  et  $\Delta_3$  sont des fonctions non linéaires connues, et  $\delta_1$ ,  $\delta_2$  et  $\delta_3$  sont des termes non linéaires incertains, mais bornés<sup>5</sup>.

<sup>5</sup>Pour simplifier les expressions, on suppose qu'ils ont une borne commune  $\delta_0$ .

**Étape 1** — On considère d'abord (3.27), où  $x_2$  est prise pour la commande. La technique du **backstepping** en l'absence des perturbations (section 3.1) donne comme première commande virtuelle

$$(x_2)_d \triangleq \alpha_1 = -\frac{1}{\psi_1} \left[ k_1 \varepsilon_1 + \varphi_1^T \boldsymbol{\vartheta} - \boldsymbol{\xi}_0 \right] \quad (3.30)$$

où  $\boldsymbol{\xi}_0 = \dot{\alpha}_0$ . La méthode de l'amortissement non linéaire permet de calculer la commande qui assure des trajectoires bornées de (3.27), même en présence des incertitudes. L'équation (2.11) donne

$$(x_2)_d = \alpha_1 + \varsigma_1 \quad (3.31)$$

avec comme amortissement non linéaire

$$\varsigma_1 = -m_1 \varepsilon_1 \psi_1 \|\Delta_1\|_2^2$$

où  $m_1 > 0$  est un paramètre de design et  $\alpha_1$  est la commande (virtuelle) qui assure la stabilité de (3.2), qui est la version non perturbée de (3.27), avec comme fcl  $\mathcal{V}_1$ . La dérivée de celle-ci vérifie

$$\dot{\mathcal{V}}_1 \leq -k_1 \varepsilon_1^2$$

le long de la solution non perturbée. Avec la commande (3.31), on est sûr que cette même dérivée vérifie

$$\dot{\mathcal{V}}_1 \leq -k_1 \varepsilon_1^2 + \frac{\delta_0}{4m_1}$$

le long de la trajectoire perturbée (3.27).

**Étape 2** — On considère le sous-système (3.27)–(3.28) et l'on définit la variable  $\varepsilon_2$

$$\varepsilon_2 = x_2 - (\alpha_1 + \varsigma_1)$$

qui permet d'écrire (3.27)–(3.28) sous la forme

$$\dot{\varepsilon}_1 = \varphi_1^T \boldsymbol{\vartheta} + \psi_1 \left[ \varepsilon_2 + \alpha_1 + \varsigma_1 + \Delta_1 \delta_1 \right] - \dot{\alpha}_0 \quad (3.32)$$

$$\dot{\varepsilon}_2 = \varphi_2^T \boldsymbol{\vartheta} + \psi_2 \left[ x_3 + \Delta_2 \delta_2 \right] - (\dot{\alpha}_1 + \dot{\varsigma}_1) \quad (3.33)$$

Si l'on prend pour fcl

$$\begin{aligned} \mathcal{V}_2(\varepsilon_1, \varepsilon_2) &= \mathcal{V}_1 + \frac{1}{2}\varepsilon_2^2 \\ &= \frac{1}{2}\varepsilon_1^2 + \frac{1}{2}\varepsilon_2^2 \end{aligned}$$

sa dérivée le long de la solution de (3.32)–(3.33) sera donnée par

$$\begin{aligned} \dot{\mathcal{V}}_2 &= \varepsilon_1 \dot{\varepsilon}_1 + \varepsilon_2 \dot{\varepsilon}_2 \\ &= \varepsilon_1 \left[ \boldsymbol{\varphi}_1^T \boldsymbol{\vartheta} + \psi_1(\alpha_1 + \varsigma_1 + \Delta_1 \delta_1) - \dot{\alpha}_0 \right] \\ &\quad + \varepsilon_2 \left[ \boldsymbol{\varphi}_2^T \boldsymbol{\vartheta} + \psi_2(x_3 + \Delta_2 \delta_2) - (\dot{\alpha}_1 + \dot{\varsigma}_1) + \varepsilon_1 \psi_1 \right] \\ &\leq -k_1 \varepsilon_1^2 + \frac{\delta_0}{4m_1} + \varepsilon_2 \left[ \boldsymbol{\varphi}_2^T \boldsymbol{\vartheta} + \psi_2 x_3 + \varepsilon_1 \psi_1 - \boldsymbol{\xi}_1 \right] + \varepsilon_2 \left[ \psi_2 \Delta_2 \delta_2 - \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1} \Delta_1 \delta_1 \right] \end{aligned}$$

où

$$\boldsymbol{\xi}_1 = \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1} (\boldsymbol{\varphi}_1^T \boldsymbol{\vartheta} + \psi_1 x_2) + \frac{\partial(\alpha_1 + \varsigma_1)}{\partial y_r} \dot{y}_r + \frac{\partial(\alpha_1 + \varsigma_1)}{\partial \dot{y}_r} \ddot{y}_r$$

est la partie déterministe de  $\dot{\alpha}_1 + \dot{\varsigma}_1$ . La valeur désirée de  $x_3$  est fixée à

$$(x_3)_d = \alpha_2 + \varsigma_2$$

où

$$\alpha_2 = \frac{1}{\psi_2} \left[ \boldsymbol{\xi}_1 - \boldsymbol{\varphi}_2^T \boldsymbol{\vartheta} - \varepsilon_1 \psi_1 - k_2 \varepsilon_2 \right]$$

avec  $k_2 > 0$ . Avec un tel choix, la dérivée  $\dot{\mathcal{V}}_2$  vérifie

$$\dot{\mathcal{V}}_2 \leq -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 + \frac{\delta_0}{4m_1} + \varepsilon_2 \left[ \psi_2 \varsigma_2 + \psi_2 \Delta_2 \delta_2 - \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1} \Delta_1 \delta_1 \right]$$

L'application de l'amortissement non linéaire à ce problème permet de choisir  $\varsigma_2$ . Si l'on prend

$$\varsigma_2 = -m_2 \frac{\varepsilon_2}{\psi_2} \left[ \|\psi_2 \Delta_2\|^2 + \left\| \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1} \Delta_1 \right\|^2 \right]$$

où  $m_2 > 0$ , la dérivée  $\dot{\mathcal{V}}_2$  peut être réécrite sous la forme

$$\begin{aligned}
\dot{\mathcal{V}}_2 &\leq -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 + \frac{\delta_0}{4m_1} \\
&\quad - m_2\varepsilon_2^2 \left[ \|\psi_2\Delta_2\|^2 + \left\| \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1} \Delta_1 \right\|^2 \right] + \varepsilon_2 \left[ \psi_2\Delta_2\delta_2 - \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1} \Delta_1\delta_1 \right] \\
&\leq -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 + \frac{\delta_0}{4m_1} \\
&\quad - m_2 \left[ \|\varepsilon_2\psi_2\Delta_2\|^2 + \left\| \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1} \Delta_1 \right\|^2 \right] + \|\varepsilon_2\psi_2\Delta_2\| \|\delta_2\| + \|\varepsilon_2 \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1} \Delta_1\| \|\delta_1\| \\
&\leq -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 + \frac{\delta_0}{4m_1} + \frac{\delta_0}{4m_2} + \frac{\delta_0}{4m_2} \\
&\leq -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 + \frac{3\delta_0}{4m_{12}}
\end{aligned}$$

où

$$m_{12} = \min(m_1, m_2)$$

**Étape 3** — On considère à présent le système (3.27)–(3.29) dans sa globalité. On définit la nouvelle variable

$$\varepsilon_3 = x_3 - (\alpha_2 + \varsigma_2)$$

et la fonction de Lyapunov

$$\begin{aligned}
\mathcal{V}_3(\varepsilon_1, \varepsilon_2, \varepsilon_3) &= \mathcal{V}_2 + \frac{1}{2}\varepsilon_3^2 \\
&= \frac{1}{2}\varepsilon_1^2 + \frac{1}{2}\varepsilon_2^2 + \frac{1}{2}\varepsilon_3^2
\end{aligned}$$

Les équations (3.27)–(3.29) s'écrivent avec ces variables

$$\dot{\varepsilon}_1 = \varphi_1^T \boldsymbol{\vartheta} + \psi_1 \left[ \varepsilon_2 + \alpha_1 + \varsigma_1 + \Delta_1\delta_1 \right] - \dot{\alpha}_0 \quad (3.34)$$

$$\dot{\varepsilon}_2 = \varphi_2^T \boldsymbol{\vartheta} + \psi_2 \left[ \varepsilon_3 + \alpha_2 + \varsigma_2 + \Delta_2\delta_2 \right] - (\dot{\alpha}_1 + \dot{\varsigma}_1) \quad (3.35)$$

$$\dot{\varepsilon}_3 = \varphi_3^T \boldsymbol{\vartheta} + \psi_3 \left[ u + \Delta_3\delta_3 \right] - (\dot{\alpha}_2 + \dot{\varsigma}_2) \quad (3.36)$$

Le long des trajectoires de ce système, la dérivée de la fcl est donnée par

$$\begin{aligned}\dot{V}_3 &= \varepsilon_1 \dot{\varepsilon}_1 + \varepsilon_2 \dot{\varepsilon}_2 + \varepsilon_3 \dot{\varepsilon}_3 \\ &\leq -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 + \frac{3\delta_0}{4m_{12}} + \varepsilon_3 \left[ \varepsilon_2 \psi_2 + \varphi_3^T \vartheta + \psi_3 (u + \Delta_3 \delta_3) - \dot{\alpha}_2 - \dot{\varsigma}_2 \right] \\ &= -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 + \frac{3\delta_0}{4m_{12}} + \varepsilon_3 \left[ \varepsilon_2 \psi_2 + \varphi_3^T \vartheta + \psi_3 u - \xi_2 \right] \\ &\quad + \varepsilon_3 \left[ \psi_3 \Delta_3 \delta_3 - \frac{\partial(\alpha_2 + \varsigma_2)}{\partial x_2} \psi_2 \Delta_2 \delta_2 - \frac{\partial(\alpha_2 + \varsigma_2)}{\partial x_1} \psi_1 \Delta_1 \delta_1 \right]\end{aligned}$$

où

$$\begin{aligned}\xi_2 &= \frac{\partial(\alpha_2 + \varsigma_2)}{\partial y_r} \dot{y}_r + \frac{\partial(\alpha_2 + \varsigma_2)}{\partial \dot{y}_r} \ddot{y}_r + \frac{\partial(\alpha_2 + \varsigma_2)}{\partial \ddot{y}_r} y_r^{(3)} \\ &\quad + \frac{\partial(\alpha_2 + \varsigma_2)}{\partial x_1} (\varphi_1^T \vartheta + \psi_1 x_2) + \frac{\partial(\alpha_2 + \varsigma_2)}{\partial x_2} (\varphi_2^T \vartheta + \psi_2 x_3)\end{aligned}$$

est la partie déterministe de  $\dot{\alpha}_2 + \dot{\varsigma}_2$ . Si l'on choisit comme commande

$$u = \alpha_3 + \varsigma_3$$

où

$$\alpha_3 = \frac{1}{\psi_3} \left[ \xi_2 - \varphi_3^T \vartheta - \varepsilon_2 \psi_2 - k_3 \varepsilon_3 \right]$$

la dérivée  $\dot{V}_3$  se réduira à

$$\begin{aligned}\dot{V}_3 &\leq -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 - k_3 \varepsilon_3^2 + \frac{3\delta_0}{4m_{12}} \\ &\quad + \varepsilon_3 \left[ \psi_3 \varsigma_3 + \psi_3 \Delta_3 \delta_3 - \frac{\partial(\alpha_2 + \varsigma_2)}{\partial x_2} \psi_2 \Delta_2 \delta_2 - \frac{\partial(\alpha_2 + \varsigma_2)}{\partial x_1} \psi_1 \Delta_1 \delta_1 \right]\end{aligned}$$

Une fois encore, l'application de l'amortissement non linéaire permet de choisir  $\varsigma_3$ . Il suffit, à présent, de prendre

$$\varsigma_3 = -m_3 \frac{\varepsilon_3}{\psi_3} \left[ \|\psi_3 \Delta_3\|^2 + \left\| \frac{\partial(\alpha_2 + \varsigma_2)}{\partial x_2} \Delta_2 \right\|^2 + \left\| \frac{\partial(\alpha_2 + \varsigma_2)}{\partial x_1} \Delta_1 \right\|^2 \right]$$

où  $m_3 > 0$ , pour obtenir

$$\begin{aligned}\dot{V}_3 &\leq -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 - k_3 \varepsilon_3^2 + \frac{3\delta_0}{4m_{12}} + \frac{3\delta_0}{4m_3} \\ &\leq -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 - k_3 \varepsilon_3^2 + \frac{2\delta_0}{3m_{123}}\end{aligned}$$

où

$$m_{123} = \min(m_{12}, m_3) = \min(m_1, m_2, m_3)$$

Ce résultat implique que  $\dot{V}_3$  est négative à l'extérieur d'une certaine sphère, et que  $\varepsilon_1$ ,  $\varepsilon_2$  et  $\varepsilon_3$  sont bornées (la borne dépend de  $\delta_0$  et des  $m_i$ ), malgré la présence des incertitudes.

### 3.3.2 Applications

Une application directe de la version robuste du **backstepping** est donnée par la méthode du **backstepping** avec observateur (Krstić et al. 1995, Krstić & Kokotović 1994). Cette dernière, en aboutissant à des contrôleurs par retour de sortie, permet de s'affranchir de la nécessité de la mesure de toutes les variables d'état du procédé, puisque seule la sortie reste nécessaire à l'implantation de la loi de commande résultante.

Une autre application intéressante, consiste à utiliser la robustification par amortissements non linéaires dans le contexte adaptatif. Cette approche permet (comme il sera montré dans la section 4.5.4) de conserver la stabilité des boucles, même en l'absence d'adaptation (i.e. erreurs d'identification non nulles). Cette propriété a donné lieu à une nouvelle approche de conception des contrôleurs adaptatifs par **backstepping**, connue sous le nom de **Modular Design Method** (Krstić et al. 1995, Krstić & Kokotović 1995, 1996).

## 3.4 Backstepping avec observateur

La méthode du **backstepping**, telle que présentée jusqu'ici, suppose la disponibilité à la mesure de l'état du procédé (au complet). Dans le cas où une partie de cet état n'est pas mesurable<sup>6</sup>, le **backstepping** (dans sa forme de base) ne peut être appliqué. Une solution consiste alors à remplacer l'état par un estimé, fourni par un observateur. Cette solution, largement utilisée en commande linéaire, n'est pas sans poser des problèmes (notamment de stabilité) dans le contexte non linéaire. Pour un système

<sup>6</sup>Soit physiquement non réalisable, soit se traduisant par des coûts trop élevés.



linéaire, la stabilité de la boucle (procédé, contrôleur, observateur) est garantie par le principe de **séparation**, qui stipule que si l'observateur et le couple (procédé, contrôleur) sont chacun stable de son côté, alors l'ensemble est stable. La vitesse de convergence de l'observateur (erreurs d'estimation des variables d'état) n'affecte en rien la stabilité de la boucle. Son seul effet réside dans la modification des performances en boucle fermée.

Il est évident, qu'à cause du "finite escape time", le principe de **séparation** ne peut s'appliquer aux systèmes non linéaires. La lenteur de la convergence de l'observateur peut engendrer une instabilité irréversible de la boucle. D'où, l'idée d'utiliser la version robuste du **backstepping**, où les erreurs d'estimation de variables d'état constituent la partie stochastique des équations. Des termes d'amortissements non linéaires peuvent alors être introduits, afin de garantir la bornétude du vecteur d'état.

En réalité, cette méthode ne se limite pas à garantir la bornétude, puisque les propriétés de l'observateur (stabilité et convergence exponentielle) sont exploitées pour s'assurer de la régulation à zéro de toutes les variables d'erreurs  $\varepsilon_i$  définies dans la procédure récursive de design.

### 3.4.1 Applicabilité

L'existence et la difficulté du design d'observateurs (à convergence exponentielle) pour les systèmes non linéaires, constitue un handicap majeur à l'application de cette méthode. C'est pourquoi, le **backstepping** avec observateur n'est applicable qu'aux systèmes dont les non-linéarités sont fonctions de la sortie. De tels systèmes sont décrits par

$$\begin{aligned}
 \dot{x}_1 &= \varphi_1(y) + x_2 \\
 \dot{x}_2 &= \varphi_2(y) + x_3 \\
 &\vdots \\
 \dot{x}_{n-1} &= \varphi_{n-1}(y) + x_n \\
 \dot{x}_n &= \varphi_n(y) + u \\
 y &= x_1
 \end{aligned} \tag{3.37}$$

Sous forme matricielle, ces équations s'écrivent

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \boldsymbol{\varphi}(y) + \mathbf{B}u \\ y &= \mathbf{C}^T \mathbf{x}\end{aligned}$$

où

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} [0]_{(n-1) \times 1} \\ 1 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 \\ [0]_{(n-1) \times 1} \end{bmatrix},$$

$$\mathbf{x} = [x_1 \quad x_2 \quad \cdots \quad x_n]^T \quad \text{et} \quad \boldsymbol{\varphi} = [\varphi_1 \quad \varphi_2 \quad \cdots \quad \varphi_n]^T$$

Afin d'être en mesure d'estimer l'état du procédé, un observateur (non linéaire) de Luenberger (Luenberger 1964, 1971) est utilisé. Son équation est donnée par

$$\begin{aligned}\dot{\hat{\mathbf{x}}} &= \mathbf{A}\hat{\mathbf{x}} + \mathbf{H}(y - \hat{y}) + \boldsymbol{\varphi}(y) + \mathbf{B}u \\ \hat{y} &= \mathbf{C}^T \hat{\mathbf{x}}\end{aligned}$$

où le vecteur des gains

$$\mathbf{H} = [h_1 \quad h_2 \quad \cdots \quad h_n]^T$$

est choisi afin de rendre hurwitzienne<sup>7</sup> la matrice

$$\mathbf{A}_0 = \mathbf{A} - \mathbf{H}\mathbf{C}^T$$

L'équation de l'erreur d'estimation résultante s'écrit alors

$$\dot{\bar{\mathbf{x}}} = \mathbf{A}_0 \bar{\mathbf{x}} \tag{3.38}$$

avec  $\bar{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$  assuré (par le choix adéquat de  $\mathbf{H}$ ) de converger vers zéro.

<sup>7</sup>Ses valeurs propres sont à partie réelle négative.

### 3.4.2 Principe

La disponibilité d'une estimation de l'état du procédé permet, à présent, d'appliquer l'algorithme récursif. Comme pour la version par retour d'état, le design sera développé pour le système de troisième ordre

$$\begin{aligned}\dot{x}_1 &= \varphi_1(y) + x_2 \\ \dot{x}_2 &= \varphi_2(y) + x_3 \\ \dot{x}_3 &= \varphi_3(y) + u \\ y &= x_1\end{aligned}$$

pour lequel on construit l'observateur<sup>8</sup>

$$\begin{aligned}\dot{\hat{x}}_1 &= \varphi_1(y) + h_1(y - \hat{y}) + \hat{x}_2 \\ \dot{\hat{x}}_2 &= \varphi_2(y) + h_2(y - \hat{y}) + \hat{x}_3 \\ \dot{\hat{x}}_3 &= \varphi_3(y) + h_3(y - \hat{y}) + u \\ \hat{y} &= \hat{x}_1\end{aligned}$$

**Étape 1** — On définit la première variable d'erreur

$$\varepsilon_1 = y - \alpha_0$$

où  $\alpha_0 = y_r$ . On choisit comme première fonction de contrôle de Lyapunov

$$\mathcal{V}_1 = \frac{1}{2}\varepsilon_1^2 + \frac{1}{4m_1}\tilde{\mathbf{x}}^T \mathbf{P}\tilde{\mathbf{x}} \quad (3.39)$$

où  $m_1 > 0$  est un paramètre de design et  $\mathbf{P}$  est une matrice symétrique définie positive, solution de l'équation de Lyapunov

$$\mathbf{P}\mathbf{A}_0 + \mathbf{A}_0^T \mathbf{P} = -\mathbf{I}$$

L'existence et la positivité de  $\mathbf{P}$  sont garanties par la stabilité du système d'erreurs (3.38). La dérivée de (3.39) est alors donnée par

$$\begin{aligned}\dot{\mathcal{V}}_1 &= \varepsilon_1 \dot{\varepsilon}_1 + \frac{1}{4m_1}\tilde{\mathbf{x}}^T \left[ \mathbf{P}\mathbf{A}_0 + \mathbf{A}_0^T \mathbf{P} \right] \tilde{\mathbf{x}} \\ &= \varepsilon_1 \left[ x_2 + \varphi_1(y) - \dot{\alpha}_0 \right] - \frac{1}{4m_1}\tilde{\mathbf{x}}^T \tilde{\mathbf{x}}\end{aligned}$$

---

<sup>8</sup>Les détails de cette construction sont volontairement omis.

Étant donné que la variable  $x_2$  n'est pas disponible pour la mesure, elle ne peut être choisie comme commande virtuelle. L'équation de la dérivée est réécrite sous la forme

$$\dot{\mathcal{V}}_1 = \varepsilon_1 \left[ \hat{x}_2 + \bar{x}_2 + \varphi_1(y) - \dot{\alpha}_0 \right] - \frac{1}{4m_1} \bar{\mathbf{x}}^T \bar{\mathbf{x}}$$

où l'on fait apparaître l'estimé  $\hat{x}_2$ , qui sera prise comme commande virtuelle. Sa valeur désirée est donnée par

$$\begin{aligned} (\hat{x}_2)_d &= \alpha_1 + \varsigma_1 \\ &= -k_1 \varepsilon_1 - \varphi_1(y) + \xi_0 + \varsigma_1 \end{aligned}$$

où  $\xi_0 = \dot{\alpha}_0$  et  $\varsigma_1$  est le terme d'amortissement non linéaire, dont la valeur reste à déterminer. L'expression de la dérivée devient avec ces choix et notations

$$\begin{aligned} \dot{\mathcal{V}}_1 &= -k_1 \varepsilon_1^2 + \varepsilon_1 (\varsigma_1 + \bar{x}_2) - \frac{1}{4m_1} \bar{\mathbf{x}}^T \bar{\mathbf{x}} \\ &\leq -k_1 \varepsilon_1^2 + \varepsilon_1 (\varsigma_1 + \bar{x}_2) - \frac{1}{4m_1} \bar{x}_2^2 \\ &= -k_1 \varepsilon_1^2 + \varepsilon_1 (\varsigma_1 + m_1 \varepsilon_1) - m_1 \left[ \varepsilon_1 - \frac{\bar{x}_2}{2m_1} \right]^2 \\ &\leq -k_1 \varepsilon_1^2 + \varepsilon_1 (\varsigma_1 + m_1 \varepsilon_1) \end{aligned}$$

Il suffit, à présent, de prendre

$$\varsigma_1 = -m_1 \varepsilon_1$$

pour assurer la négativité de la dérivée  $\dot{\mathcal{V}}_1$ . Ceci aura pour conséquence de garantir la stabilité de la première équation d'erreur

$$\dot{\varepsilon}_1 = -(k_1 + m_1) \varepsilon_1 + \bar{x}_2$$

malgré la présence du terme inconnu  $\bar{x}_2$ .

**Étape 2** — La deuxième variable d'erreur est définie par

$$\varepsilon_2 = \hat{x}_2 - (\alpha_1 + \varsigma_1)$$

et la nouvelle fcl est une version augmentée de la précédente

$$V_2 = V_1 + \frac{1}{2}\varepsilon_2^2 + \frac{1}{4m_2}\tilde{\mathbf{x}}^T \mathbf{P}\tilde{\mathbf{x}} \quad (3.40)$$

Sa dérivée est donnée par

$$\begin{aligned} \dot{V}_2 &= \dot{V}_1 + \varepsilon_2 \dot{\varepsilon}_2 - \frac{1}{4m_2}\tilde{\mathbf{x}}^T \dot{\tilde{\mathbf{x}}} \\ &\leq -k_1\varepsilon_1^2 + \varepsilon_2 \left[ \varepsilon_1 + \dot{\varepsilon}_2 \right] - \frac{1}{4m_2}\tilde{\mathbf{x}}^T \tilde{\mathbf{x}} \end{aligned}$$

où

$$\dot{\varepsilon}_2 = \hat{x}_3 + \varphi_2(y) + h_2(y - \hat{y}) - (\dot{\alpha}_1 + \dot{\varsigma}_1)$$

et

$$\begin{aligned} \dot{\alpha}_1 + \dot{\varsigma}_1 &= \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1}(\varphi_1 + \hat{x}_2 + \tilde{x}_2) + \frac{\partial(\alpha_1 + \varsigma_1)}{\partial y_r}\dot{y}_r + \frac{\partial(\alpha_1 + \varsigma_1)}{\partial \dot{y}_r}\ddot{y}_r \\ &= \boldsymbol{\xi}_1 + \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1}\tilde{x}_2 \end{aligned}$$

où l'on note  $\boldsymbol{\xi}_1$  la partie connue (et calculable analytiquement) de  $\dot{\alpha}_1 + \dot{\varsigma}_1$ . La deuxième commande virtuelle peut maintenant être choisie comme

$$\begin{aligned} (\hat{x}_3)_d &= \alpha_2 + \varsigma_2 \\ &= -k_2\varepsilon_2 - \varepsilon_1 - \varphi_2(y) - h_2(y - \hat{y}) + \boldsymbol{\xi}_1 + \varsigma_2 \end{aligned}$$

Avec ce choix, la dérivée de la fcl (3.40) devient

$$\begin{aligned} \dot{V}_2 &\leq -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - \frac{1}{4m_2}\tilde{\mathbf{x}}^T \tilde{\mathbf{x}} + \left[ \varsigma_2 - \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1}\tilde{x}_2 \right] \varepsilon_2 \\ &\leq -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - \frac{1}{4m_2}\tilde{x}_2^2 + \left[ \varsigma_2 - \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1}\tilde{x}_2 \right] \varepsilon_2 \\ &= -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - m_2 \left[ \frac{1}{2m_2}\tilde{x}_2 + \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1}\varepsilon_2 \right]^2 + \left[ \varsigma_2 + m_2 \left( \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1} \right)^2 \varepsilon_2 \right] \varepsilon_2 \\ &\leq -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 + \left[ \varsigma_2 + m_2 \left( \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1} \right)^2 \varepsilon_2 \right] \varepsilon_2 \end{aligned}$$

Il suffit à présent de choisir comme deuxième amortissement non linéaire

$$\varsigma_2 = -m_2 \left[ \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1} \right]^2 \varepsilon_2$$

pour assurer de la stabilité des deux équations considérées (même en présence des erreurs d'estimation).

**Étape 3** — La dernière variable d'erreur est donnée par

$$\varepsilon_3 = \hat{x}_3 - (\alpha_2 + \varsigma_2)$$

Sa fonction de Lyapunov associée est définie par

$$\mathcal{V}_3 = \mathcal{V}_2 + \frac{1}{2}\varepsilon_3^2 + \frac{1}{4m_3}\tilde{\mathbf{x}}^T \mathbf{P}\tilde{\mathbf{x}}$$

Le calcul de la dérivée de cette dernière, le long des trajectoires du système global, donne

$$\begin{aligned}\dot{\mathcal{V}}_3 &= \dot{\mathcal{V}}_2 + \varepsilon_3 \dot{\varepsilon}_3 - \frac{1}{4m_3}\tilde{\mathbf{x}}^T \tilde{\mathbf{x}} \\ &\leq -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 + \varepsilon_3 \left[ \varepsilon_2 + \dot{\varepsilon}_3 \right] - \frac{1}{4m_3}\tilde{x}_2^2\end{aligned}$$

Avec

$$\dot{\varepsilon}_3 = u + \varphi_3(y) + h_3(y - \hat{y}) - (\dot{\alpha}_2 + \dot{\varsigma}_2)$$

et

$$\begin{aligned}\dot{\alpha}_2 + \dot{\varsigma}_2 &= \frac{\partial(\alpha_2 + \varsigma_2)}{\partial y_r} \dot{y}_r + \frac{\partial(\alpha_2 + \varsigma_2)}{\partial \dot{y}_r} \ddot{y}_r + \frac{\partial(\alpha_2 + \varsigma_2)}{\partial \ddot{y}_r} y_r^{(3)} \\ &\quad + \frac{\partial(\alpha_2 + \varsigma_2)}{\partial x_1} (\varphi_1 + \hat{x}_2 + \tilde{x}_2) + \frac{\partial(\alpha_2 + \varsigma_2)}{\partial \hat{x}_2} \left[ \varphi_2 + \hat{x}_3 + h_2(y - \hat{y}) \right] \\ &= \boldsymbol{\xi}_2 + \frac{\partial(\alpha_2 + \varsigma_2)}{\partial x_1} \tilde{x}_2\end{aligned}$$

où  $\boldsymbol{\xi}_2$  est la partie connue de  $\dot{\alpha}_2 + \dot{\varsigma}_2$ . Avec ces notations, la vraie commande s'écrit

$$\begin{aligned}u &= \alpha_3 + \varsigma_3 \\ &= -k_3\varepsilon_3 - \varepsilon_2 - \varphi_3(y) - h_3(y - \hat{y}) + \boldsymbol{\xi}_2 + \varsigma_3\end{aligned}$$

Ce qui permet d'écrire la dérivée de la fcl  $\mathcal{V}_3$  sous la forme

$$\dot{\mathcal{V}}_3 \leq -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - k_3\varepsilon_3^2 - \frac{1}{4m_3}\tilde{x}_2^2 + \left[ \varsigma_3 - \frac{\partial(\alpha_2 + \varsigma_2)}{\partial x_1} \tilde{x}_2 \right] \varepsilon_3$$

Un réarrangement de cette équation, identique à celui effectué à l'étape 2, permet de choisir le terme d'amortissement non linéaire  $\varsigma_3$  qui assure la stabilité du système global (procédé, contrôleur, observateur). Ce choix est donné par

$$\varsigma_3 = -m_3 \left[ \frac{\partial(\alpha_2 + \varsigma_2)}{\partial x_1} \right]^2 \varepsilon_3$$

La dérivée devient avec ces choix

$$\dot{V}_3 \leq -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 - k_3 \varepsilon_3^2$$

D'où la stabilité de la boucle et la régulation à zéro des erreurs de poursuite  $\varepsilon_i$  et d'estimation  $\tilde{x}_i$ .

### 3.5 Conclusion

La technique du **backstepping** offre un outil efficace qui permet, pour les systèmes non linéaires de tout ordre, de construire récursivement, et d'une manière systématique et directe, la loi de commande et la fonction de Lyapunov qui assurent la stabilité de la boucle. Dans sa version originelle, le **backstepping** aboutit à des commandes par retour d'état. Pour obtenir des contrôleurs par retour de sortie, une variante robuste de la méthode permet, en utilisant un observateur, de s'affranchir de la nécessité de la mesure de l'état. On parle, dans ce cas, de **backstepping avec observateur**. Ce dernier fait appel à la technique des amortissements non linéaires qui garantit, même en présence des incertitudes, la bornétude l'état du procédé.

Dans le cas où ces incertitudes sont de nature paramétrique, le **backstepping** garantit plus que la bornétude d'état. En effet, sans avoir à utiliser les amortissements non linéaires<sup>9</sup>, la version adaptative du **backstepping**, qui sera présentée dans le chapitre 4, offre, en plus de la loi de commande et la fonction de Lyapunov, un mécanisme de mise à jour des paramètres inconnus. Ce dernier assure, sous certaines conditions, la convergence vers zéro des erreurs d'identification.

---

<sup>9</sup>Qui peuvent, toutefois, être utilisés comme robustification supplémentaire.

# Chapitre 4

## Backstepping Adaptatif

La version adaptative du **backstepping** offre une méthode itérative et systématique, qui permet, pour des systèmes non linéaires de tout ordre, de construire récursivement les trois parties indispensables à une commande adaptative basée sur Lyapunov :

- La loi de commande : permet de répondre aux spécifications désirées, quant au comportement du système à commander.
- La loi d'adaptation : détermine la dynamique d'estimation des paramètres inconnus. Elle doit garantir leur convergence vers leurs valeurs respectives, sans affecter le bon fonctionnement, surtout la stabilité, de l'ensemble.
- La fonction de Lyapunov : permet le choix adéquat des deux précédentes lois et garantit la convergence et la stabilité de la structure adaptative, en tout temps.

La construction de ce triplet s'effectue simultanément. Les trois opérations sont entrelacées, ce qui permet de tenir compte des différents effets destructeurs, afin de préserver la stabilité du système.

### 4.1 Applicabilité

En plus des contraintes imposées à la partie déterministe dans le cas non adaptatif<sup>1</sup>, quelques restrictions portent également sur les incertitudes paramétriques, dans

---

<sup>1</sup>Structure paramétrique pure ou stricte.



le contexte adaptatif. À ce sujet, le **backstepping** est, de loin, la moins sévère des méthodes (Kanellakopoulos et al. 1991, Krstić et al. 1995, Khalil 1996). Il impose aux termes incertains de se présenter linéairement dans les équations d'état du système, sous forme de paramètres constants inconnus. C'est une contrainte que l'on retrouve dans la plupart des méthodes. Le **backstepping** a permis, cependant, d'éliminer la contrainte de la **matching condition**<sup>2</sup>, qui représente une barrière infranchissable pour beaucoup de méthodes. Ce résultat a été obtenu au prix d'une sur-paramétrisation.

Cette sur-paramétrisation peut être évitée dans la situation très particulière de l'**extended matching condition**<sup>3</sup> (Jiang & Praly 1991), mais reste inévitable dans le cadre général. Une variante du **backstepping** connue sous le nom de la méthode des fonctions de réglage (**tuning functions design**), proposée par Krstić et al. (1992) a permis de surmonter cette limitation. Cette technique requiert le nombre minimal d'estimés, i.e. égal au nombre de paramètres inconnus.

En plus d'assurer la stabilité du système et la convergence vers zéro des erreurs (poursuite et/ou régulation, estimation), le **backstepping** offre le moyen d'améliorer la qualité du régime transitoire de cette convergence. Ces améliorations s'obtiennent par l'utilisation de l'amortissement non linéaire et de l'initialisation des trajectoires (Kanellakopoulos et al. 1993, Krstić et al. 1994). La combinaison de ces deux méthodes permet de garder le vecteur d'erreur borné, même en l'absence d'adaptation, et d'avoir le contrôle sur sa borne. Avec ces améliorations, les fonctions de réglage permettent d'obtenir des contrôleurs dont les performances sont supérieures à celles obtenues par les contrôleurs adaptatifs basés sur l'équivalence certaine (Krstić et al. 1995).

## 4.2 Principe

Le **backstepping** adaptatif est la méthode qui résulte de la fusion du design adaptatif par Lyapunov (section 2.4.3) et la technique récursive du **backstepping** non adaptatif (section 3.1). Toutefois, la combinaison directe de ces deux méthodes conduit, en général, à

<sup>2</sup>Paramètres inconnus et commande dans la même équation.

<sup>3</sup>Un seul intégrateur sépare la commande des paramètres inconnus.

des contrôleurs d'ordre trop élevé. On parle alors de sur-paramétrisation, i.e. plusieurs estimés par paramètre inconnu. Dans le cas où le système présente certaines propriétés structurales (**matching condition** ou l'**extended matching condition**), le contrôleur obtenu est d'ordre minimal. Le cas simple des systèmes vérifiant la **matching condition**, servira à introduire le principe de la méthode. Par la suite, sera abordé le cas général des systèmes ne vérifiant pas cette condition.

Soit à commander le système non linéaire

$$\dot{x}_1 = \varphi_1^T \vartheta + \psi_1 x_2 \quad (4.1)$$

$$\dot{x}_2 = \varphi_2^T \vartheta + \psi_2 x_3 \quad (4.2)$$

$$\dot{x}_3 = \varphi_3^T \theta + \psi_3 u \quad (4.3)$$

$$y = x_1 \quad (4.4)$$

où le vecteur des paramètres  $\theta$  dans l'équation (4.3) n'est pas connu. L'objectif est de faire suivre à la trajectoire de référence  $y_r$  la sortie  $y = x_1$ . Pour ce faire, on procède récursivement, comme dans le cas non adaptatif (section 3.1).

Le sous-système(4.1)–(4.2) est identique au sous-système (3.2)–(3.3). Sa commande virtuelle  $\alpha_2$ , valeur désirée de  $x_3$ , peut être réalisée de la même façon que dans le cas non adaptatif. Celle-ci est donnée par l'équation (3.12). La différence, par rapport au cas non adaptatif, réside dans la troisième étape de la procédure, à cause de la présence du terme inconnu  $\theta$  dans (4.3). Par conséquent, seule l'étape 3 de la procédure du design sera présentée. Les deux premières restent inchangées.

**Étape 3** — On conserve la même approche récursive, où l'on considère tout le système (4.1)–(4.3). La fonction de Lyapunov est choisie telle que

$$\mathcal{V}_3(\varepsilon_1, \varepsilon_2, \varepsilon_3) = \frac{1}{2}\varepsilon_1^2 + \frac{1}{2}\varepsilon_2^2 + \frac{1}{2}\varepsilon_3^2 \quad (4.5)$$

où la nouvelle variable d'erreur  $\varepsilon_3$  est définie par

$$\varepsilon_3 = x_3 - \alpha_2$$

Les équations du système s'écrivent alors

$$\dot{\varepsilon}_1 = \varphi_1^T \boldsymbol{\vartheta} + \psi_1(\varepsilon_2 + \alpha_1) - \dot{\alpha}_0 \quad (4.6)$$

$$\dot{\varepsilon}_2 = \varphi_2^T \boldsymbol{\vartheta} + \psi_2(\varepsilon_3 + \alpha_2) - \dot{\alpha}_1 \quad (4.7)$$

$$\dot{\varepsilon}_3 = \varphi_3^T \boldsymbol{\theta} + \psi_3 u - \dot{\alpha}_2 \quad (4.8)$$

La dérivée de (4.5), le long de (4.6)–(4.8), se met sous la forme

$$\begin{aligned} \dot{\mathcal{V}}_3(\varepsilon_1, \varepsilon_2, \varepsilon_3) &= \varepsilon_1 \dot{\varepsilon}_1 + \varepsilon_2 \dot{\varepsilon}_2 + \varepsilon_3 \dot{\varepsilon}_3 \\ &= -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 + \varepsilon_3 \left[ \psi_2 \varepsilon_2 + \varphi_3^T \boldsymbol{\theta} + \psi_3 u - \dot{\alpha}_2 \right] \end{aligned}$$

Si le vecteur  $\boldsymbol{\theta}$  était connu, la commande

$$u = \frac{1}{\psi_3} \left[ \dot{\alpha}_2 - \psi_2 \varepsilon_2 - \varphi_3^T \boldsymbol{\theta} - k_3 \varepsilon_3 \right] \quad (4.9)$$

où  $k_3 > 0$ , aurait permis de rendre  $\dot{\mathcal{V}}_3$  négative; ce qui aurait stabilisé l'origine du système d'erreurs global. Étant donné que  $\boldsymbol{\theta}$  est inconnu, une approche adaptative s'impose. Le vecteur  $\boldsymbol{\theta}$  est alors remplacé par son estimé  $\hat{\boldsymbol{\theta}}$ . Le contrôleur (4.9) s'écrit, dans ces conditions,

$$u = \frac{1}{\psi_3} \left[ \dot{\alpha}_2 - \psi_2 \varepsilon_2 - \varphi_3^T \hat{\boldsymbol{\theta}} - k_3 \varepsilon_3 \right] \quad (4.10)$$

et la dérivée prend la forme

$$\dot{\mathcal{V}}_3(\varepsilon_1, \varepsilon_2, \varepsilon_3) = -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 - k_3 \varepsilon_3^2 + \varepsilon_3 \varphi_3^T \tilde{\boldsymbol{\theta}}$$

La présence du terme d'erreur  $\tilde{\boldsymbol{\theta}}$  dans l'expression de la dérivée ne permet pas la détermination de son signe. La stabilité du système ne peut être affirmée.

**Remarque 4.1 (Equivalence certaine)** — On remarque que l'équivalent<sup>4</sup> adaptatif (4.10) du contrôleur (4.9), n'est pas en mesure de garantir la stabilité du système. Cela est dû aux effets instabilisants causés par la dynamique d'adaptation. D'où, l'idée d'en tenir compte dans la procédure de design, afin de préserver la stabilité.

□

---

<sup>4</sup>Basé sur le principe de l'équivalence certaine.

Afin de surmonter cet obstacle, on augmente la fonction de Lyapunov (4.5) d'un terme quadratique en  $\tilde{\theta}$

$$\begin{aligned}\mathcal{V}_3^\dagger(\varepsilon_1, \varepsilon_2, \varepsilon_3, \tilde{\theta}) &= \mathcal{V}_3 + \frac{1}{2}\tilde{\theta}^T \Gamma^{-1} \tilde{\theta} \\ &= \frac{1}{2}\varepsilon_1^2 + \frac{1}{2}\varepsilon_2^2 + \frac{1}{2}\varepsilon_3^2 + \frac{1}{2}\tilde{\theta}^T \Gamma^{-1} \tilde{\theta}\end{aligned}$$

Sa dérivée le long de la trajectoire du système (4.1)–(4.3), s'écrit (après quelques simplifications)

$$\begin{aligned}\dot{\mathcal{V}}_3^\dagger(\varepsilon_1, \varepsilon_2, \varepsilon_3, \tilde{\theta}) &= -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - k_3\varepsilon_3^2 + \varepsilon_3\varphi_3^T \tilde{\theta} + \tilde{\theta}^T \Gamma^{-1} \dot{\tilde{\theta}} \\ &= -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - k_3\varepsilon_3^2 + \tilde{\theta}^T \Gamma^{-1} \left[ \Gamma\varepsilon_3\varphi_3 + \dot{\tilde{\theta}} \right] \\ &= -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - k_3\varepsilon_3^2 + \tilde{\theta}^T \Gamma^{-1} \left[ \tau - \dot{\tilde{\theta}} \right]\end{aligned}$$

où  $\tau = \Gamma\varepsilon_3\varphi_3$ . Le signe de la dérivée reste toujours indéfini, mais un choix judicieux de la loi de mise à jour de l'estimé, permet d'annuler le terme incertain. Avec le choix

$$\dot{\tilde{\theta}} = \tau \tag{4.11}$$

la dérivée devient

$$\begin{aligned}\dot{\mathcal{V}}_3^\dagger(\varepsilon_1, \varepsilon_2, \varepsilon_3, \tilde{\theta}) &= -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - k_3\varepsilon_3^2 \\ &\leq 0\end{aligned}$$

La négativité de la dérivée de la fonction de Lyapunov implique la stabilité asymptotique de l'origine du système d'erreur de régulation et la convergence vers zéro de l'erreur d'estimation. Le système obtenu en boucle fermée est décrit par

$$\begin{bmatrix} \dot{\varepsilon}_1 \\ \dot{\varepsilon}_2 \\ \dot{\varepsilon}_3 \end{bmatrix} = \begin{bmatrix} -k_1 & 1 & 0 \\ -1 & -k_2 & 1 \\ 0 & -1 & -k_3 \end{bmatrix} \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \varphi_3^T \end{bmatrix} (\theta - \hat{\theta})$$

avec comme loi d'adaptation

$$\dot{\hat{\theta}} = \Gamma\varphi_3^T \varepsilon_3$$

### 4.3 Cas général

On considère toujours le même problème de poursuite, appliqué au système

$$\dot{x}_1 = \varphi_1^T \theta + \psi_1 x_2 \quad (4.12)$$

$$\dot{x}_2 = \varphi_2^T \theta + \psi_2 x_3 \quad (4.13)$$

$$\dot{x}_3 = \varphi_3^T \theta + \psi_3 u \quad (4.14)$$

$$y = x_1 \quad (4.15)$$

où cette fois, le vecteur des paramètres inconnus  $\theta$  est présent dans les trois équations.

**Étape 1** — On considère le sous-système (4.12), où  $x_2$  est traitée comme une commande et l'on introduit la variable d'erreur donnée par (3.5). L'équation (4.12) s'écrit alors

$$\dot{\varepsilon}_1 = \varphi_1^T \theta + \psi_1 x_2 - \dot{\alpha}_0 \quad (4.16)$$

où  $\alpha_0$  représente la trajectoire de référence  $y_r$ . On prend comme fonction de Lyapunov

$$V_1(\varepsilon_1, \bar{\theta}_1) = \frac{1}{2} \varepsilon_1^2 + \frac{1}{2} \bar{\theta}_1^T \Gamma^{-1} \bar{\theta}_1 \quad (4.17)$$

où  $\bar{\theta}_1$  est l'erreur  $(\theta - \hat{\theta}_1)$ . Le vecteur  $\hat{\theta}_1$  étant l'estimé<sup>5</sup> à l'étape 1 de  $\theta$ . La dérivée de (4.17), le long des trajectoires possibles du système, est donnée par

$$\begin{aligned} \dot{V}_1 &= \varepsilon_1 \dot{\varepsilon}_1 + \bar{\theta}_1^T \Gamma^{-1} \dot{\bar{\theta}}_1 \\ &= \varepsilon_1 \left[ \varphi_1^T \theta + \psi_1 x_2 - \xi_0 \right] + \bar{\theta}_1^T \Gamma^{-1} \dot{\bar{\theta}}_1 \\ &= \varepsilon_1 \left[ w_1^T \theta + \psi_1 x_2 - \xi_0 \right] + \bar{\theta}_1^T \Gamma^{-1} \dot{\bar{\theta}}_1 \end{aligned}$$

où, pour conserver la même notation, on introduit

$$\xi_0 = \dot{\alpha}_0$$

$$w_1 = \varphi_1$$

---

<sup>5</sup>Le vecteur des paramètres  $\theta$  aura un estimé par étape, on parle de sur-paramétrisation.

Le choix de la valeur désirée de  $x_2$

$$(x_2)_d \triangleq \alpha_1(\varepsilon_1, \hat{\theta}_1) = \frac{1}{\psi_1} \left[ \xi_0 - \mathbf{w}_1^T \hat{\theta}_1 - k_1 \varepsilon_1 \right]$$

permet d'écrire la dérivée sous la forme

$$\begin{aligned} \dot{\mathcal{V}}_1 &= \varepsilon_1 \left[ \mathbf{w}_1^T (\boldsymbol{\theta} - \hat{\theta}_1) - k_1 \varepsilon_1 \right] + \tilde{\boldsymbol{\theta}}_1^T \Gamma^{-1} \dot{\tilde{\boldsymbol{\theta}}}_1 \\ &= -k_1 \varepsilon_1^2 + \tilde{\boldsymbol{\theta}}_1^T \Gamma^{-1} \left[ \Gamma \mathbf{w}_1 \varepsilon_1 + \dot{\tilde{\boldsymbol{\theta}}}_1 \right] \\ &= -k_1 \varepsilon_1^2 + \tilde{\boldsymbol{\theta}}_1^T \Gamma^{-1} \left[ \boldsymbol{\tau}_1 - \dot{\tilde{\boldsymbol{\theta}}}_1 \right] \end{aligned}$$

où  $\boldsymbol{\tau}_1 = \Gamma \mathbf{w}_1 \varepsilon_1$ . La loi de mise à jour

$$\dot{\tilde{\boldsymbol{\theta}}}_1 = \boldsymbol{\tau}_1$$

assure la négativité de la dérivée de la fonction (4.17), qui s'écrit

$$\dot{\mathcal{V}}_1 = -k_1 \varepsilon_1^2 \leq 0$$

**Étape 2** — On considère le sous-système (4.12)-(4.13). On utilise les variables d'erreurs données par (3.5) et (3.8). Le système se met sous la forme

$$\dot{\varepsilon}_1 = \boldsymbol{\varphi}_1^T \boldsymbol{\theta} + \psi_1 (\varepsilon_2 + \alpha_1) - \dot{\alpha}_0 \quad (4.18)$$

$$\dot{\varepsilon}_2 = \boldsymbol{\varphi}_2^T \boldsymbol{\theta} + \psi_2 x_3 - \dot{\alpha}_1 \quad (4.19)$$

On prend la fonction de Lyapunov donnée par l'équation (4.17). Sa dérivée le long de la solution de (4.18)-(4.19) s'écrit, après simplification,

$$\dot{\mathcal{V}}_1 = -k_1 \varepsilon_1^2 + \tilde{\boldsymbol{\theta}}_1^T \Gamma^{-1} \left[ \boldsymbol{\tau}_1 - \dot{\tilde{\boldsymbol{\theta}}}_1 \right] + \psi_1 \varepsilon_1 \varepsilon_2$$

Avec le choix  $\dot{\tilde{\boldsymbol{\theta}}}_1 = \boldsymbol{\tau}_1$ , la dérivée se réduit à

$$\dot{\mathcal{V}}_1 = -k_1 \varepsilon_1^2 + \psi_1 \varepsilon_1 \varepsilon_2$$

Le signe de  $\dot{\mathcal{V}}_1$  étant indéterminé, afin d'en savoir plus sur la stabilité du système, on construit la fcl

$$\mathcal{V}_2(\varepsilon_1, \hat{\theta}_1, \varepsilon_2) = \mathcal{V}_1(\varepsilon_1, \hat{\theta}_1) + \frac{1}{2} \varepsilon_2^2$$

Sa dérivée s'écrit (toujours avec le choix  $\dot{\hat{\theta}}_1 = \tau_1$ )

$$\begin{aligned}
 \dot{V}_2 &= \dot{V}_1 + \varepsilon_2 \dot{\varepsilon}_2 \\
 &= -k_1 \varepsilon_1^2 + \psi_1 \varepsilon_1 \varepsilon_2 + \varepsilon_2 \dot{\varepsilon}_2 \\
 &= -k_1 \varepsilon_1^2 + \varepsilon_2 \left[ \psi_1 \varepsilon_1 + \dot{\varepsilon}_2 \right] \\
 &= -k_1 \varepsilon_1^2 + \varepsilon_2 \left[ \psi_2 x_3 + \psi_1 \varepsilon_1 + \varphi_2^T \theta - \dot{\alpha}_1 \right]
 \end{aligned}$$

où le terme  $\dot{\alpha}_1$  se calcule analytiquement

$$\begin{aligned}
 \dot{\alpha}_1 &= \frac{\partial \alpha_1}{\partial x_1} \dot{x}_1 + \frac{\partial \alpha_1}{\partial \hat{\theta}_1} \dot{\hat{\theta}}_1 \\
 &= \frac{\partial \alpha_1}{\partial x_1} \left[ \varphi_1^T \theta + \psi_1 x_2 \right] + \frac{\partial \alpha_1}{\partial \hat{\theta}_1} \tau_1 \\
 &= \xi_1 + \frac{\partial \alpha_1}{\partial x_1} \varphi_1^T \theta
 \end{aligned} \tag{4.20}$$

où  $\xi_1$  est la partie connue de  $\dot{\alpha}_1$ . Sa dérivée s'écrit, avec ces notations,

$$\dot{V}_2 = -k_1 \varepsilon_1^2 + \varepsilon_2 \left[ \psi_2 x_3 + \psi_1 \varepsilon_1 - \xi_1 + w_2^T \theta \right]$$

où

$$w_2 = \varphi_2 - \frac{\partial \alpha_1}{\partial x_1} \varphi_1$$

On choisit, à présent, la valeur désirée  $\alpha_2$  de  $x_3$ , afin d'éliminer les termes connus dans l'expression de la dérivée. Étant donné que  $\theta$  n'est pas connu, on le remplace par son estimé à l'étape 1, i.e.  $\hat{\theta}_1$ ,

$$(x_3)_d \triangleq \alpha_2 = \frac{1}{\psi_2} \left[ \xi_1 - \psi_1 \varepsilon_1 - k_2 \varepsilon_2 - w_2^T \hat{\theta}_1 \right]$$

et la dérivée qui en résulte est donnée par

$$\dot{V}_2 = -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 + \varepsilon_2 w_2^T \left[ \theta - \hat{\theta}_1 \right]$$

Avec ces choix, on ne dispose plus d'aucune liberté pour annuler le terme incertain. Pour surmonter ce handicap, au lieu d'utiliser  $\hat{\theta}_1$  comme estimé de  $\theta$ , on utilise un

nouvel estimé  $\hat{\theta}_2$  (estimé à l'étape 2). Après quelques simplifications, l'expression de la dérivée s'écrit

$$\dot{\mathcal{V}}_2 = -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 + \varepsilon_2 \mathbf{w}_2^T \left[ \boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_2 \right]$$

Afin de se débarrasser du terme incertain, on utilise la fonction de Lyapunov suivante

$$\mathcal{V}_2^\dagger = \mathcal{V}_2 + \frac{1}{2} \bar{\boldsymbol{\theta}}_2^T \Gamma^{-1} \bar{\boldsymbol{\theta}}_2 \quad (4.21)$$

où  $\bar{\boldsymbol{\theta}}_2$  représente l'erreur d'estimation à l'étape 2 ( $\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_2$ ). La dérivée est donnée par

$$\begin{aligned} \dot{\mathcal{V}}_2^\dagger &= \dot{\mathcal{V}}_2 + \bar{\boldsymbol{\theta}}_2^T \Gamma^{-1} \dot{\bar{\boldsymbol{\theta}}}_2 \\ &= -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 + \bar{\boldsymbol{\theta}}_2^T \Gamma^{-1} \left[ \dot{\bar{\boldsymbol{\theta}}}_2 + \Gamma \varepsilon_2 \mathbf{w}_2 \right] \\ &= -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 + \bar{\boldsymbol{\theta}}_2^T \Gamma^{-1} \left[ \boldsymbol{\tau}_2 - \dot{\bar{\boldsymbol{\theta}}}_2 \right] \end{aligned}$$

où  $\boldsymbol{\tau}_2 = \Gamma \varepsilon_2 \mathbf{w}_2$ . Le choix de la dynamique de mise à jour de l'estimé  $\hat{\boldsymbol{\theta}}_2$  permet de lever l'indétermination du signe de la dérivée. Avec la loi

$$\dot{\bar{\boldsymbol{\theta}}}_2 = \boldsymbol{\tau}_2$$

la dérivée se réduit à

$$\begin{aligned} \dot{\mathcal{V}}_2^\dagger &= -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 \\ &\leq 0 \end{aligned}$$

**Étape 3** — On considère à présent tout le système (4.12)–(4.14). Avec les variables d'erreurs définies précédemment, ce système s'écrit

$$\begin{aligned} \dot{\varepsilon}_1 &= \boldsymbol{\varphi}_1^T \boldsymbol{\theta} + \psi_1(\varepsilon_2 + \alpha_1) - \dot{\alpha}_0 \\ \dot{\varepsilon}_2 &= \boldsymbol{\varphi}_2^T \boldsymbol{\theta} + \psi_2(\varepsilon_3 + \alpha_2) - \dot{\alpha}_1 \\ \dot{\varepsilon}_3 &= \boldsymbol{\varphi}_3^T \boldsymbol{\theta} + \psi_3 u - \dot{\alpha}_2 \end{aligned}$$

On commence par la fcl donnée par l'équation (4.21). Sa dérivée évaluée le long de la trajectoire du système vaut<sup>6</sup>,

$$\dot{\mathcal{V}}_2^\dagger = -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 + \psi_2\varepsilon_2\varepsilon_3$$

---

<sup>6</sup>Après simplification et avec le choix  $\dot{\bar{\boldsymbol{\theta}}}_2 = \boldsymbol{\tau}_2$ .



Afin de venir à bout du terme de signe indéfini, on introduit la nouvelle fcl

$$V_3 = V_2^{\dagger} + \frac{1}{2}\varepsilon_3^2$$

dont la dérivée est donnée par

$$\begin{aligned}\dot{V}_3 &= \dot{V}_2^{\dagger} + \varepsilon_3 \dot{\varepsilon}_3 \\ &= -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 + \psi_2 \varepsilon_2 \varepsilon_3 + \varepsilon_3 \dot{\varepsilon}_3 \\ &= -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 + \varepsilon_3 \left[ \psi_2 \varepsilon_2 + \dot{\varepsilon}_3 \right] \\ &= -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 + \varepsilon_3 \left[ \varphi_3^T \theta + \psi_3 u + \psi_2 \varepsilon_2 - \dot{\alpha}_2 \right]\end{aligned}$$

On calcule  $\dot{\alpha}_2$  analytiquement,

$$\begin{aligned}\dot{\alpha}_2 &= \frac{\partial \alpha_2}{\partial x_1} \dot{x}_1 + \frac{\partial \alpha_2}{\partial x_2} \dot{x}_2 + \frac{\partial \alpha_2}{\partial \hat{\theta}_1} \dot{\hat{\theta}}_1 + \frac{\partial \alpha_2}{\partial \hat{\theta}_2} \dot{\hat{\theta}}_2 \\ &= \frac{\partial \alpha_2}{\partial x_1} \left[ \varphi_1^T \theta + \psi_1 x_2 \right] + \frac{\partial \alpha_2}{\partial x_2} \left[ \varphi_2^T \theta + \psi_2 x_3 \right] + \frac{\partial \alpha_2}{\partial \hat{\theta}_1} \tau_1 + \frac{\partial \alpha_2}{\partial \hat{\theta}_2} \tau_2 \\ &= \xi_2 + \frac{\partial \alpha_2}{\partial x_1} \varphi_1^T \theta + \frac{\partial \alpha_2}{\partial x_2} \varphi_2^T \theta\end{aligned}$$

où  $\xi_2$  est la partie calculable de  $\dot{\alpha}_2$ . Ceci donne pour la dérivée

$$\dot{V}_3 = -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 + \varepsilon_3 \left[ \psi_3 u + \psi_2 \varepsilon_2 - \xi_2 + w_3^T \theta \right]$$

où

$$w_3 = \varphi_3 - \frac{\partial \alpha_2}{\partial x_1} \varphi_1 - \frac{\partial \alpha_2}{\partial x_2} \varphi_2$$

Pour choisir la commande  $u$ , étant donné que  $\theta$  n'est pas connu, on utilise son plus récent estimé disponible, en l'occurrence  $\hat{\theta}_2$ . La loi de commande est choisie à fin d'éliminer les termes certains dans l'expression de la dérivée

$$u = \frac{1}{\psi_3} \left[ \xi_2 - \psi_2 \varepsilon_2 - w_3^T \hat{\theta}_2 \right]$$

On obtient alors comme dérivée

$$\dot{V}_3 = -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 - k_3 \varepsilon_3^2 + \varepsilon_3 w_3^T \left[ \theta - \hat{\theta}_2 \right]$$

Pour pouvoir éliminer le terme résiduel, on définit, encore une fois, un nouvel estimé  $\hat{\theta}_3$  (estimé à l'étape 3 de  $\theta$ ), qui permet, après quelques simplifications, d'écrire l'expression de la dérivée sous la forme

$$\dot{\mathcal{V}}_3 = -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - k_3\varepsilon_3^2 + \varepsilon_3\mathbf{w}_3^T \left[ \theta - \hat{\theta}_3 \right]$$

On exploite le libre choix de la loi de mise à jour de  $\hat{\theta}_3$  pour éliminer le terme incertain. Pour ce faire, on construit la nouvelle fcl

$$\mathcal{V}_3^\dagger = \mathcal{V}_3 + \frac{1}{2}\bar{\theta}_3^T \Gamma^{-1} \bar{\theta}_3 \quad (4.22)$$

où  $\bar{\theta}_3$  représente l'erreur d'estimation à l'étape 3 ( $\theta - \hat{\theta}_3$ ). La dérivée de la fonction (4.22) s'écrit alors

$$\begin{aligned} \dot{\mathcal{V}}_3^\dagger &= \dot{\mathcal{V}}_3 + \bar{\theta}_3^T \Gamma^{-1} \dot{\bar{\theta}}_3 \\ &= -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - k_3\varepsilon_3^2 + \bar{\theta}_3^T \Gamma^{-1} \left[ \dot{\bar{\theta}}_3 + \Gamma \varepsilon_3 \mathbf{w}_3 \right] \\ &= -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - k_3\varepsilon_3^2 + \bar{\theta}_3^T \Gamma^{-1} \left[ \tau_3 - \dot{\bar{\theta}}_3 \right] \end{aligned}$$

où  $\tau_3 = \Gamma \varepsilon_3 \mathbf{w}_3$ . À présent le choix de la dynamique de mise à jour de l'estimé  $\hat{\theta}_3$  est simple. En effet, avec le choix

$$\dot{\bar{\theta}}_3 = \tau_3$$

l'indétermination du signe de la dérivée est levée. On obtient alors

$$\begin{aligned} \dot{\mathcal{V}}_3^\dagger &= -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - k_3\varepsilon_3^2 \\ &\leq 0 \end{aligned}$$

d'où la stabilité asymptotique globale. Le système compensé est donné, dans l'espace des erreurs, par

$$\begin{bmatrix} \dot{\varepsilon}_1 \\ \dot{\varepsilon}_2 \\ \dot{\varepsilon}_3 \end{bmatrix} = \begin{bmatrix} -k_1 & 1 & 0 \\ -1 & -k_2 & 1 \\ 0 & -1 & -k_3 \end{bmatrix} \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{bmatrix} + \begin{bmatrix} \tau_1 & 0 & 0 \\ 0 & \tau_2 & 0 \\ 0 & 0 & \tau_3 \end{bmatrix} \begin{bmatrix} \theta - \hat{\theta}_1 \\ \theta - \hat{\theta}_2 \\ \theta - \hat{\theta}_3 \end{bmatrix}$$

et les trois lois de mise à jour s'écrivent

$$\dot{\hat{\theta}}_i = \tau_i, \quad i = 1, 2, 3.$$

## 4.4 Limitations

Le **backstepping** adaptatif a permis de résoudre le problème de la stabilisation des systèmes non linéaires, en présence d'incertitudes paramétriques. Bien que très intéressante sur le plan théorique, cette technique reste très lourde d'usage dans le pratique (Krstić et al. 1995). Cette limitation vient essentiellement de la sur-paramétrisation, nécessaire, sauf dans certains cas très particuliers, pour venir à bout des incertitudes qui ne vérifient pas ladite **matching condition**. Une amélioration majeure a été apportée à cette méthode par Krstić et al. (1992). Il s'agit de la technique des fonctions de réglage (**tuning functions design**).

## 4.5 Méthode des fonctions de réglage

Cette méthode offre l'avantage de conduire à des contrôleurs dont l'ordre est réduit à son minimum, i.e. le nombre des estimés à calculer est égal au nombre des paramètres inconnus. La loi de mise à jour des estimés est construite récursivement, comme pour la fonction de **Lyapunov**. Mais contrairement aux méthodes qui viennent d'être présentées, les lois intermédiaires ne seront pas réalisées et serviront uniquement à la construction de la dernière (et vraie) loi d'adaptation et des différentes commandes virtuelles  $\alpha_i$ . Cette minimisation de l'ordre des contrôleurs n'est pas uniquement d'un intérêt d'ordre pratique (facilité de réalisation); elle garantit, également, de meilleures propriétés de stabilité et de convergence que celles obtenues par le **backstepping**, tel qu'il a été présenté jusqu'ici (Krstić et al. 1995). Ces propriétés seront présentées à la fin du présent chapitre.

Avant de présenter la méthode des fonctions de réglage, une extension du cas de la **matching condition**, servira d'exemple d'introduction. Il s'agit des systèmes dans lesquels, le terme incertain ne se trouve pas directement dans la même équation que la commande, mais séparé de celle-ci par un intégrateur. On parle, dans ce cas, de l'**extended matching condition**.

### 4.5.1 Extended matching condition

L'idée dans cette solution consiste à retarder, d'une étape, le choix de la loi de mise à jour des estimés, afin de n'en avoir qu'une seule à la fin de la procédure. Cette idée sera généralisée, mais avec une approche différente, par la méthode des fonctions de réglage.

Soit, encore une fois, le problème de poursuite de  $y_r$  par la sortie  $y = x_1$  du système

$$\dot{x}_1 = \varphi_1^T \boldsymbol{\vartheta} + \psi_1 x_2 \quad (4.23)$$

$$\dot{x}_2 = \varphi_2^T \boldsymbol{\theta} + \psi_2 x_3 \quad (4.24)$$

$$\dot{x}_3 = \varphi_3^T \boldsymbol{\vartheta} + \psi_3 u \quad (4.25)$$

$$y = x_1 \quad (4.26)$$

On remarque la présence du terme inconnu  $\boldsymbol{\theta}$ , que l'on suppose constant, dans l'équation (4.24). On reprend les mêmes étapes de design que précédemment.

**Étape 1** — Cette étape est identique au cas non adaptatif (section 3.1). Elle ne sera pas développée.

**Étape 2** — On considère le sous-système (4.23)–(4.24). Toujours avec les mêmes variables d'erreurs  $(\varepsilon_1, \varepsilon_2)$ , ce système se met sous la forme

$$\dot{\varepsilon}_1 = \varphi_1^T \boldsymbol{\vartheta} + \psi_1(\varepsilon_2 + \alpha_1) - \dot{\alpha}_0 \quad (4.27)$$

$$\dot{\varepsilon}_2 = \varphi_2^T \boldsymbol{\theta} + \psi_2 x_3 - \dot{\alpha}_1 \quad (4.28)$$

On considère à nouveau une fonction de Lyapunov de la forme

$$V_2(\varepsilon_1, \varepsilon_2) = \frac{1}{2}\varepsilon_1^2 + \frac{1}{2}\varepsilon_2^2$$

Avec les résultats de l'étape 1 de la section 3.1, sa dérivée s'écrit

$$\dot{V}_2 = -k_1 \varepsilon_1^2 + \varepsilon_2 \left[ \varphi_2^T \boldsymbol{\theta} + \psi_2 x_3 + \psi_1 \varepsilon_1 - \boldsymbol{\xi}_1 \right]$$

où, pour conserver la même notation,

$$\begin{aligned} \boldsymbol{\xi}_1 &= \dot{\alpha}_1 \\ &= \frac{\partial \alpha_1}{\partial x_1} (\varphi_1^T \boldsymbol{\vartheta} + \psi_1 x_2) + \frac{\partial \alpha_1}{\partial y_r} \dot{y}_r + \frac{\partial \alpha_1}{\partial \dot{y}_r} \ddot{y}_r \end{aligned}$$

Afin d'éliminer les termes de signe incertain, on choisit, à présent, la valeur désirée  $\alpha_2$  de  $x_3$ . Le vecteur inconnu  $\theta$  est remplacé par son estimé  $\hat{\theta}$ .

**Remarque 4.2 (Un seul estimé)** — On remarque que, contrairement à ce qui a été fait dans la section 4.3, on utilise comme estimé  $\hat{\theta}$  et non pas  $\hat{\theta}_2$ . La raison est qu'un seul estimé sera utilisé tout au long de la procédure de design. □

L'équation (3.7) devient avec ce choix

$$\alpha_2 = \frac{1}{\psi_2} \left[ \xi_1 - \psi_1 \varepsilon_1 - k_2 \varepsilon_2 - \varphi_2^T \hat{\theta} \right]$$

Ceci donne pour la dérivée

$$\dot{\nu}_2 = -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 + \varepsilon_2 \varphi_2^T (\theta - \hat{\theta})$$

Afin de se débarrasser du terme incertain, on augmente la fcl d'un terme quadratique en l'erreur d'estimation

$$\nu_2^\dagger = \nu_2 + \frac{1}{2} \bar{\theta}^T \Gamma^{-1} \bar{\theta}$$

Après quelques simplifications, l'expression de la dérivée s'écrit

$$\dot{\nu}_2^\dagger = -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 + \bar{\theta}^T \Gamma^{-1} (\tau_2 - \dot{\bar{\theta}})$$

où  $\tau_2 = \Gamma \varepsilon_2 \varphi_2$ . La différence, par rapport à la section 4.3, réside dans le fait que le choix de la dynamique de mise à jour de l'estimé  $\hat{\theta}$  est différé à l'étape suivante.

**Étape 3** — On considère à présent tout le système (4.12)–(4.14) qui s'écrit, dans l'espace des erreurs  $(\varepsilon_1, \varepsilon_2, \varepsilon_3)$ , sous la forme

$$\dot{\varepsilon}_1 = \varphi_1^T \vartheta + \psi_1 (\varepsilon_2 + \alpha_1) - \dot{\alpha}_0$$

$$\dot{\varepsilon}_2 = \varphi_2^T \theta + \psi_2 (\varepsilon_3 + \alpha_2) - \dot{\alpha}_1$$

$$\dot{\varepsilon}_3 = \varphi_3^T \vartheta + \psi_3 u - \dot{\alpha}_2$$

La fcl

$$\nu_3 = \nu_2^\dagger + \frac{1}{2} \varepsilon_3^2$$

a pour dérivée, le long de la solution des ces équations,

$$\begin{aligned}\dot{V}_3 &= -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 + \psi_2\varepsilon_2\varepsilon_3 + \bar{\theta}^T\Gamma^{-1}(\tau_2 - \dot{\hat{\theta}}) + \varepsilon_3\dot{\varepsilon}_3 \\ &= -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 + \varepsilon_3\left[\varphi_3^T\vartheta + \psi_3u + \psi_2\varepsilon_2 - \dot{\alpha}_2\right] + \bar{\theta}^T\Gamma^{-1}(\tau_2 - \dot{\hat{\theta}})\end{aligned}$$

où  $\dot{\alpha}_2$  est donnée par

$$\begin{aligned}\dot{\alpha}_2 &= \frac{\partial\alpha_2}{\partial x_1}\dot{x}_1 + \frac{\partial\alpha_2}{\partial x_2}\dot{x}_2 + \frac{\partial\alpha_2}{\partial\hat{\theta}}\dot{\hat{\theta}} \\ &= \frac{\partial\alpha_2}{\partial x_1}(\varphi_1^T\vartheta + \psi_1x_2) + \frac{\partial\alpha_2}{\partial x_2}(\varphi_2^T\hat{\theta} + \psi_2x_3) + \frac{\partial\alpha_2}{\partial\hat{\theta}}\dot{\hat{\theta}} \\ &= \frac{\partial\alpha_2}{\partial x_1}(\varphi_1^T\vartheta + \psi_1x_2) + \frac{\partial\alpha_2}{\partial x_2}\left[\varphi_2^T(\hat{\theta} + \bar{\theta}) + \psi_2x_3\right] + \frac{\partial\alpha_2}{\partial\hat{\theta}}\dot{\hat{\theta}} \\ &= \xi_2 + \frac{\partial\alpha_2}{\partial\hat{\theta}}\dot{\hat{\theta}} + \frac{\partial\alpha_2}{\partial x_2}\varphi_2^T\bar{\theta}\end{aligned}$$

avec

$$\xi_2 = \frac{\partial\alpha_2}{\partial x_1}(\varphi_1^T\vartheta + \psi_1x_2) + \frac{\partial\alpha_2}{\partial x_2}(\varphi_2^T\hat{\theta} + \psi_2x_3) + \frac{\partial\alpha_1}{\partial y_r}\dot{y}_r + \frac{\partial\alpha_1}{\partial\dot{y}_r}\ddot{y}_r + \frac{\partial\alpha_1}{\partial\ddot{y}_r}y_r^{(3)}$$

représentant la partie connue de  $\dot{\alpha}_2$ . Ceci permet d'écrire la dérivée sous la forme

$$\dot{V}_3 = -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 + \varepsilon_3\left[\varphi_3^T\vartheta + \psi_3u + \psi_2\varepsilon_2 - \xi_2 - \frac{\partial\alpha_2}{\partial\hat{\theta}}\dot{\hat{\theta}}\right] + \bar{\theta}^T\Gamma^{-1}(\tau_2 - \dot{\hat{\theta}})$$

où

$$\begin{aligned}\tau_3 &= \tau_2 - \Gamma\varepsilon_3\frac{\partial\alpha_2}{\partial x_2}\varphi_2 \\ &= \Gamma\left[\varepsilon_2 - \varepsilon_3\frac{\partial\alpha_2}{\partial x_2}\right]\varphi_2\end{aligned}$$

À présent, le choix d'une seule et unique dynamique d'adaptation permet d'annuler le terme incertain. En effet, il suffit de prendre

$$\dot{\hat{\theta}} = \tau_3$$

pour ne conserver que les termes connus dans l'expression de la dérivée. Le choix de la loi de commande  $u$  rend possible l'élimination des termes résiduels, assurant ainsi la

négativité de la dérivée. Un bon choix de  $u$  est donné par

$$\begin{aligned} u &= \frac{1}{\psi_3} \left[ \xi_2 - \psi_2 \varepsilon_2 - k_3 \varepsilon_3 - \varphi_3^T \vartheta + \frac{\partial \alpha_2}{\partial \hat{\theta}} \dot{\hat{\theta}} \right] \\ &= \frac{1}{\psi_3} \left[ \xi_2 - \psi_2 \varepsilon_2 - k_3 \varepsilon_3 - \varphi_3^T \vartheta + \frac{\partial \alpha_2}{\partial \hat{\theta}} \tau_3 \right] \end{aligned}$$

Ce qui permet d'écrire la dérivée de la fcl

$$\begin{aligned} \dot{\mathcal{V}}_3 &= -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 - k_3 \varepsilon_3^2 \\ &\leq 0 \end{aligned}$$

d'où la stabilité du système.

**Remarque 4.3 (Différence)** — La différence par rapport à l'algorithme originel réside dans le fait que seule la fonction de Lyapunov finale (et globale)  $\mathcal{V}_3$  est assurée d'être négative. Le signe de la fonction intermédiaire  $\mathcal{V}_2^\dagger$  est sacrifié afin d'éviter une sur-paramétrisation. Ceci ne constitue nullement un handicap, étant donné que  $\mathcal{V}_3$  assure la stabilité du système global.

□

Retarder le choix de la dynamique d'adaptation d'une étape a permis d'obtenir un contrôleur d'ordre minimal. Une seule loi de mise à jour est à réaliser. Malheureusement, cette méthode ne peut être étendue aux cas où le terme inconnu est séparé de la commande par plus d'un intégrateur. Le fait de différer le choix de plus d'une étape fait apparaître des dérivées de  $\hat{\theta}$  d'ordre supérieur à 1, dont les expressions analytiques ne sont pas disponibles (Krstić et al. 1995). La méthode des fonctions de réglage contourne ce problème des dérivées d'ordre supérieur, tout en conservant l'idée de différer le choix de l'adaptation.

## 4.5.2 Cas général (Fonctions de réglage)

L'idée consiste à construire autant de lois de mise à jour qu'il y a d'étapes. La construction se fait récursivement. À chaque étape  $i$ , une nouvelle (éventuelle) loi de mise à jour  $\tau_i$ , appelée **fonction de réglage**, est construite à partir de la loi  $\tau_{i-1}$  et des nouvelles données disponibles à cette étape. Contrairement aux méthodes directes, il

n'y aura pas de problème de sur-paramétrisation, étant donné que seule la dernière de ces fonctions sera réellement utilisée pour l'estimation. Les fonctions intermédiaires serviront, uniquement, à éviter les termes de dérivées d'ordre supérieur dans les différentes expressions.

### Principe

Soit à nouveau le système (3.1), avec  $n = 3$

$$\dot{x}_1 = \varphi_1^T \theta + \psi_1 x_2 \quad (4.29)$$

$$\dot{x}_2 = \varphi_2^T \theta + \psi_2 x_3 \quad (4.30)$$

$$\dot{x}_3 = \varphi_3^T \theta + \psi_3 u \quad (4.31)$$

$$y = x_1 \quad (4.32)$$

On se propose de faire suivre à la trajectoire de référence  $\alpha_0 = y_r$  la sortie  $y = x_1$ . Comme précédemment, à chaque étape  $i$  une nouvelle variable d'erreur  $\varepsilon_i$  est introduite. Elle définit l'écart entre la variable d'état  $x_i$  et sa valeur désirée  $\alpha_{i-1}$ , déterminée à l'étape  $i - 1$ . Ces variables d'erreur sont données par

$$\varepsilon_1 = x_1 - \alpha_0$$

$$\varepsilon_2 = x_2 - \alpha_1$$

$$\varepsilon_3 = x_3 - \alpha_2$$

Avec ces variables, le système s'écrit

$$\dot{\varepsilon}_1 = \varphi_1^T \theta + \psi_1(\varepsilon_2 + \alpha_1) - \dot{\alpha}_0 \quad (4.33)$$

$$\dot{\varepsilon}_2 = \varphi_2^T \theta + \psi_2(\varepsilon_3 + \alpha_2) - \dot{\alpha}_1 \quad (4.34)$$

$$\dot{\varepsilon}_3 = \varphi_3^T \theta + \psi_3 u - \dot{\alpha}_2 \quad (4.35)$$

**Étape 1** — On considère l'équation (4.33) et la fonction de Lyapunov

$$V_1^\dagger(\varepsilon_1, \tilde{\theta}) = \frac{1}{2} \varepsilon_1^2 + \frac{1}{2} \tilde{\theta}^T \Gamma^{-1} \tilde{\theta}$$

dont la dérivée est donnée par

$$\dot{V}_1^\dagger = \varepsilon_1 \left[ \psi_1 \varepsilon_2 + \psi_1 \alpha_1 + w_1^T \hat{\theta} - \xi_0 \right] + \tilde{\theta}^T \Gamma^{-1} (\tau_1 - \dot{\tilde{\theta}})$$



où

$$\begin{aligned}\xi_0 &= \dot{\alpha}_0 \\ \tau_1 &= \Gamma \mathbf{w}_1 \varepsilon_1 \\ \mathbf{w}_1 &= \boldsymbol{\varphi}_1\end{aligned}$$

Afin de rendre cette dérivée négative, on considère la partie connue, en éliminant tout ce qui peut l'être. On ne se préoccupe pas, du moins pour l'instant, de la partie incertaine. Le choix

$$(x_2)_d \triangleq \alpha_1 = \frac{1}{\psi_1} \left[ \xi_0 - k_1 \varepsilon_1 - \mathbf{w}_1^T \hat{\boldsymbol{\theta}} \right] \quad (4.36)$$

permet d'écrire la dérivée sous la forme

$$\dot{\mathcal{V}}_1^\dagger = -k_1 \varepsilon_1^2 + \psi_1 \varepsilon_1 \varepsilon_2 + \bar{\boldsymbol{\theta}}^T \Gamma^{-1} \left[ \tau_1 - \dot{\hat{\boldsymbol{\theta}}} \right]$$

Si  $x_2$  était la vraie commande,  $\varepsilon_2$  pourrait être forcé à 0, en prenant  $x_2 = \alpha_1$ . Dans ce cas, avec les choix (4.36) comme commande et

$$\dot{\hat{\boldsymbol{\theta}}} = \tau_1$$

comme loi d'adaptation, la stabilité asymptotique globale serait garantie. Étant donné que  $x_2$  n'est pas la commande,  $\varepsilon_2$  ne peut être parfaitement annulé et le signe de la dérivée reste indéterminé. La loi  $\tau_1$  ne sera pas utilisée pour l'estimation des paramètres, et constituera la première fonction de réglage. Le choix de la vraie loi d'adaptation est différé à une étape ultérieure (la dernière).

**Étape 2** — Avec le choix (4.36), l'équation (4.34) s'écrit

$$\begin{aligned}\dot{\varepsilon}_2 &= \dot{x}_2 - \dot{\alpha}_1 \\ &= \boldsymbol{\varphi}_2^T \boldsymbol{\theta} + \psi_2 x_3 - \dot{\alpha}_1 \\ &= \psi_2 x_3 + \mathbf{w}_2^T \boldsymbol{\theta} - \xi_1 - \frac{\partial \alpha_1}{\partial \hat{\boldsymbol{\theta}}} \dot{\hat{\boldsymbol{\theta}}}\end{aligned}$$

où

$$\xi_1 = \frac{\partial \alpha_1}{\partial x_1} \psi_1 x_2 + \frac{\partial \alpha_1}{\partial y_r} \dot{y}_r + \frac{\partial \alpha_1}{\partial \dot{y}_r} \ddot{y}_r$$

est la partie déterministe de  $\dot{\alpha}_1$  et

$$\mathbf{w}_2 = \boldsymbol{\varphi}_2 - \frac{\partial \alpha_1}{\partial x_1} \boldsymbol{\varphi}_1$$

est le nouveau régresseur. Le sous-système (4.33)–(4.34) se met, avec ces notations, sous la forme

$$\begin{aligned} \dot{\varepsilon}_1 &= -k_1 \varepsilon_1 + \psi_1 \varepsilon_2 + \mathbf{w}_1^T \bar{\boldsymbol{\theta}} \\ \dot{\varepsilon}_2 &= \psi_2 x_3 + \mathbf{w}_2^T \boldsymbol{\theta} - \boldsymbol{\xi}_1 - \frac{\partial \alpha_1}{\partial \bar{\boldsymbol{\theta}}} \dot{\bar{\boldsymbol{\theta}}} \end{aligned}$$

Si l'on prend comme fcl

$$\mathcal{V}_2^\dagger = \mathcal{V}_1^\dagger + \frac{1}{2} \varepsilon_2^2$$

sa dérivée est donnée par

$$\dot{\mathcal{V}}_2^\dagger = -k_1 \varepsilon_1^2 + \varepsilon_2 \left[ \psi_1 \varepsilon_1 + \psi_2 x_3 - \boldsymbol{\xi}_1 + \mathbf{w}_2^T \hat{\boldsymbol{\theta}} - \frac{\partial \alpha_1}{\partial \hat{\boldsymbol{\theta}}} \dot{\hat{\boldsymbol{\theta}}} \right] + \bar{\boldsymbol{\theta}}^T \Gamma^{-1} \left[ \boldsymbol{\tau}_2 - \dot{\hat{\boldsymbol{\theta}}} \right]$$

où

$$\begin{aligned} \boldsymbol{\tau}_2 &= \boldsymbol{\tau}_1 + \Gamma \mathbf{w}_2 \varepsilon_2 \\ &= \Gamma \left[ \mathbf{w}_1 \varepsilon_1 + \mathbf{w}_2 \varepsilon_2 \right] \end{aligned}$$

De la même façon que dans le cas précédent, on considère uniquement la partie certaine.

On élimine les termes connus par un choix judicieux de  $\alpha_2$  (valeur désirée de  $x_3$ )

$$(x_3)_d \triangleq \alpha_2 = \frac{1}{\psi_2} \left[ \boldsymbol{\xi}_1 - \psi_1 \varepsilon_1 - k_2 \varepsilon_2 - \mathbf{w}_2^T \hat{\boldsymbol{\theta}} + \frac{\partial \alpha_1}{\partial \hat{\boldsymbol{\theta}}} \boldsymbol{\tau}_2 \right] \quad (4.37)$$

Avec ce choix, la dérivée se réduit à

$$\dot{\mathcal{V}}_2 = -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 + \psi_2 \varepsilon_2 \varepsilon_3 + \frac{\partial \alpha_1}{\partial \hat{\boldsymbol{\theta}}} (\boldsymbol{\tau}_2 - \dot{\hat{\boldsymbol{\theta}}}) + \bar{\boldsymbol{\theta}}^T \Gamma^{-1} (\boldsymbol{\tau}_2 - \dot{\hat{\boldsymbol{\theta}}})$$

Le terme incertain pourrait être éliminé par le choix de la loi d'adaptation

$$\dot{\hat{\boldsymbol{\theta}}} = \boldsymbol{\tau}_2$$

Cependant, pour les mêmes raisons que dans l'étape 1, i.e.  $x_3$  n'est pas la vraie commande,  $\boldsymbol{\tau}_2$  ne sera pas prise comme loi de mise à jour et sera considérée comme la deuxième fonction de réglage.

**Étape 3** — À présent, la commande (4.37) permet d'écrire l'équation (4.35) sous la forme

$$\begin{aligned}\dot{\varepsilon}_3 &= \psi_3 u + \varphi_3^T \theta - \dot{\alpha}_2 \\ &= \psi_3 u - \xi_2 + \mathbf{w}_3^T \theta - \frac{\partial \alpha_2}{\partial \hat{\theta}} \dot{\hat{\theta}}\end{aligned}$$

où

$$\xi_2 = \frac{\partial \alpha_2}{\partial x_1} \psi_1 x_2 + \frac{\partial \alpha_2}{\partial x_2} \psi_2 x_3 + \frac{\partial \alpha_1}{\partial y_r} \dot{y}_r + \frac{\partial \alpha_1}{\partial \dot{y}_r} \ddot{y}_r + \frac{\partial \alpha_1}{\partial \ddot{y}_r} y_r^{(3)}$$

est la partie calculable de  $\dot{\alpha}_2$  et

$$\mathbf{w}_3 = \varphi_3 - \frac{\partial \alpha_2}{\partial x_2} \mathbf{w}_2 - \frac{\partial \alpha_2}{\partial x_1} \mathbf{w}_1$$

le nouveau régresseur. Le système (4.33)–(4.35) s'écrit

$$\begin{aligned}\dot{\varepsilon}_1 &= -k_1 \varepsilon_1 + \psi_1 \varepsilon_2 + \mathbf{w}_1^T \bar{\theta} \\ \dot{\varepsilon}_2 &= -\psi_1 \varepsilon_1 - k_2 \varepsilon_2 + \psi_2 \varepsilon_3 + \mathbf{w}_2^T \bar{\theta} + \frac{\partial \alpha_1}{\partial \hat{\theta}} (\tau_2 - \dot{\hat{\theta}}) \\ \dot{\varepsilon}_3 &= \psi_3 u - \xi_2 + \mathbf{w}_3^T \theta - \frac{\partial \alpha_2}{\partial \hat{\theta}} \dot{\hat{\theta}}\end{aligned}$$

La nouvelle fcl est donnée par

$$\mathcal{V}_3^\dagger = \mathcal{V}_2^\dagger + \frac{1}{2} \varepsilon_3^2$$

et a, comme dérivée le long des trajectoires possibles du système,

$$\begin{aligned}\dot{\mathcal{V}}_3^\dagger &= -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 + \varepsilon_2 \frac{\partial \alpha_1}{\partial \hat{\theta}} (\tau_2 - \dot{\hat{\theta}}) \\ &\quad + \varepsilon_3 \left[ \psi_2 \varepsilon_2 + \psi_3 u + \mathbf{w}_3^T \hat{\theta} - \xi_2 - \frac{\partial \alpha_2}{\partial \hat{\theta}} \dot{\hat{\theta}} \right] + \bar{\theta}^T \Gamma^{-1} \left[ \tau_3 - \dot{\hat{\theta}} \right]\end{aligned}$$

où

$$\begin{aligned}\tau_3 &= \tau_2 + \Gamma \mathbf{w}_3 \varepsilon_3 \\ &= \Gamma \left[ \mathbf{w}_1 \varepsilon_1 + \mathbf{w}_2 \varepsilon_2 + \mathbf{w}_3 \varepsilon_3 \right]\end{aligned}$$

À présent, on est en présence de la vraie commande  $u$ . Son choix, combiné à celui de la loi d'adaptation, va permettre de rendre la dérivée négative. Un bon choix de la commande est donné par

$$u = \frac{1}{\psi_3} \left[ \xi_2 - \psi_2 \varepsilon_2 - k_3 \varepsilon_3 - \mathbf{w}_3^T \hat{\boldsymbol{\theta}} + \frac{\partial \alpha_2}{\partial \hat{\boldsymbol{\theta}}} \boldsymbol{\tau}_3 + v \right] \quad (4.38)$$

Le terme  $v$ , qui sera fixé par la suite, servira à éliminer des éventuels termes résiduels indésirables. Avec cette commande, la dérivée s'écrit

$$\begin{aligned} \dot{V}_3^\dagger &= -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 - k_3 \varepsilon_3^2 + \varepsilon_3 v + \\ &\quad \varepsilon_3 \frac{\partial \alpha_2}{\partial \hat{\boldsymbol{\theta}}} (\boldsymbol{\tau}_3 - \dot{\hat{\boldsymbol{\theta}}}) + \varepsilon_2 \frac{\partial \alpha_1}{\partial \hat{\boldsymbol{\theta}}} (\boldsymbol{\tau}_2 - \dot{\hat{\boldsymbol{\theta}}}) + \tilde{\boldsymbol{\theta}}^T \Gamma^{-1} (\boldsymbol{\tau}_3 - \dot{\hat{\boldsymbol{\theta}}}) \end{aligned}$$

Afin d'être en mesure d'éliminer les termes gênants, cette expression est réarrangée, pour s'écrire

$$\begin{aligned} \dot{V}_3^\dagger &= -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 - k_3 \varepsilon_3^2 \\ &\quad + \varepsilon_3 v + \varepsilon_3 \frac{\partial \alpha_2}{\partial \hat{\boldsymbol{\theta}}} (\boldsymbol{\tau}_3 - \dot{\hat{\boldsymbol{\theta}}}) + \varepsilon_2 \frac{\partial \alpha_1}{\partial \hat{\boldsymbol{\theta}}} (\boldsymbol{\tau}_3 - \Gamma \varepsilon_3 \mathbf{w}_3 - \dot{\hat{\boldsymbol{\theta}}}) + \tilde{\boldsymbol{\theta}}^T \Gamma^{-1} (\boldsymbol{\tau}_3 - \dot{\hat{\boldsymbol{\theta}}}) \\ &= -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 - k_3 \varepsilon_3^2 \\ &\quad + \varepsilon_3 \left[ v - \frac{\partial \alpha_1}{\partial \hat{\boldsymbol{\theta}}} \Gamma \mathbf{w}_3 \varepsilon_2 \right] + \left[ \varepsilon_2 \frac{\partial \alpha_1}{\partial \hat{\boldsymbol{\theta}}} + \varepsilon_3 \frac{\partial \alpha_2}{\partial \hat{\boldsymbol{\theta}}} \right] (\boldsymbol{\tau}_3 - \dot{\hat{\boldsymbol{\theta}}}) + \tilde{\boldsymbol{\theta}}^T \Gamma^{-1} (\boldsymbol{\tau}_3 - \dot{\hat{\boldsymbol{\theta}}}) \end{aligned}$$

Avec le choix

$$v = \frac{\partial \alpha_1}{\partial \hat{\boldsymbol{\theta}}} \Gamma \mathbf{w}_3 \varepsilon_2$$

la commande donnée par (4.38) devient

$$u = \frac{1}{\psi_3} \left[ \xi_2 - \psi_2 \varepsilon_2 - k_3 \varepsilon_3 - \mathbf{w}_3^T \hat{\boldsymbol{\theta}} + \frac{\partial \alpha_2}{\partial \hat{\boldsymbol{\theta}}} \boldsymbol{\tau}_3 + \frac{\partial \alpha_1}{\partial \hat{\boldsymbol{\theta}}} \Gamma \mathbf{w}_3 \varepsilon_2 \right]$$

La dérivée s'écrit alors

$$\dot{V}_3^\dagger = -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 - k_3 \varepsilon_3^2 + \left[ \varepsilon_2 \frac{\partial \alpha_1}{\partial \hat{\boldsymbol{\theta}}} + \varepsilon_3 \frac{\partial \alpha_2}{\partial \hat{\boldsymbol{\theta}}} \right] (\boldsymbol{\tau}_3 - \dot{\hat{\boldsymbol{\theta}}}) + \tilde{\boldsymbol{\theta}}^T \Gamma^{-1} (\boldsymbol{\tau}_3 - \dot{\hat{\boldsymbol{\theta}}})$$

En plus des trois premiers termes négatifs, seuls restent les termes qui peuvent être annulés. Le choix adéquat d'une seule loi d'adaptation permet de s'en débarrasser et

de stabiliser le système dans son ensemble. Un tel choix est donné par

$$\begin{aligned}\dot{\hat{\theta}} &= \tau_3 \\ &= \Gamma \left[ \varepsilon_1 \varphi_1 + \varepsilon_2 \left( \varphi_2 - \frac{\partial \alpha_1}{\partial x_1} \varphi_1 \right) + \varepsilon_3 \left( \varphi_3 - \frac{\partial \alpha_2}{\partial x_2} \left( \varphi_2 - \frac{\partial \alpha_1}{\partial x_1} \varphi_1 \right) - \frac{\partial \alpha_2}{\partial x_1} \varphi_1 \right) \right]\end{aligned}$$

Cela permet de mettre la dérivée sous la forme

$$\begin{aligned}\dot{V}_3 &= -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 - k_3 \varepsilon_3^2 \\ &\leq 0\end{aligned}$$

d'où la stabilité.

### 4.5.3 Système d'ordre $n$

Les résultats obtenus pour le système d'ordre trois peuvent facilement s'étendre aux systèmes d'ordre arbitraire. Étant donné que la procédure de design est identique, seuls les résultats finals seront donnés. Ces résultats peuvent être résumés comme suit

$$\begin{aligned}\alpha_0 &= y_r \\ \alpha_i &= \frac{1}{\psi_i} \left[ \xi_{i-1} - \psi_{i-1} \varepsilon_{i-1} - k_i \varepsilon_i - \mathbf{w}_i^T \hat{\theta} + \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}} \tau_i + \sum_{k=2}^{i-1} \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \Gamma \mathbf{w}_i \varepsilon_k \right] \\ \tau_i &= \Gamma \sum_{k=1}^i \mathbf{w}_i \varepsilon_k \\ \dot{\hat{\theta}} &= \tau_n\end{aligned}$$

où

$$\begin{aligned}i &= 1, \dots, n \\ \varepsilon_i &= x_i - \alpha_{i-1} \\ \xi_{i-1} &= \sum_{k=1}^{i-1} \left( \frac{\partial \alpha_{i-1}}{\partial x_k} \psi_k x_{k+1} + \frac{\partial \alpha_{i-1}}{\partial y_r^{(k-1)}} y_r^{(k)} \right) + y_r^{(i)} \\ \mathbf{w}_i &= \varphi_i - \sum_{k=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial x_k} \varphi_k \\ u &= \alpha_n\end{aligned}$$

#### 4.5.4 Robustification

Certaines modifications peuvent être apportées à la méthode des fonctions de réglages, afin d'améliorer la qualité du régime transitoire et éviter que le vecteur d'état ne prenne des valeurs excessives, pouvant nuire à la stabilité du système. Deux solutions sont présentées.

##### Amortissement non linéaire

Cette technique (Krstić et al. 1995, Khalil 1996, Benaskeur 1998) permet d'obtenir des contrôleurs adaptatifs d'une grande robustesse. Ces contrôleurs garantissent la bornétude du vecteur d'erreur, même en l'absence de l'adaptation ( $\Gamma = 0$ ). L'utilisation de la technique de l'amortissement non linéaire dans le backstepping non adaptatif a été présentée dans la section 3.3. Son extension au cas adaptatif est directe, c'est pourquoi seuls les résultats finals sont donnés. Il faut noter que seuls les  $\alpha_i$  sont affectés par cette modification. Leur expression est donnée par

$$\alpha_i = \frac{1}{\psi_i} \left[ \xi_{i-1} - \psi_{i-1} \varepsilon_{i-1} - k_i \varepsilon_i - \mathbf{w}_i^T \hat{\boldsymbol{\theta}} - m_i \varepsilon_i^2 |\mathbf{w}_i|^2 + \frac{\partial \alpha_{i-1}}{\partial \hat{\boldsymbol{\theta}}} \boldsymbol{\tau}_i + \sum_{k=2}^{i-1} \frac{\partial \alpha_{k-1}}{\partial \hat{\boldsymbol{\theta}}} \Gamma \mathbf{w}_i \varepsilon_k \right]$$

Avec ce choix la dérivée de la fonction de Lyapunov

$$\mathcal{V}_i^\dagger = \frac{1}{2} \sum_{j=1}^i \varepsilon_j^2 + \bar{\boldsymbol{\theta}}^T \Gamma^{-1} \bar{\boldsymbol{\theta}}$$

vérifie, en l'absence de l'adaptation ( $\Gamma = 0$ ),

$$\dot{\mathcal{V}}_i^\dagger \leq -k_0 |\boldsymbol{\varepsilon}|^2 + \frac{|\bar{\boldsymbol{\theta}}|^2}{4m_0}$$

où

$$\boldsymbol{\varepsilon} = [\varepsilon_1, \varepsilon_2, \dots, \varepsilon_i]^T, \quad k_0 = \min(k_1, k_2, \dots, k_i) \quad \text{et} \quad \frac{1}{m_0} = \sum_{j=1}^i \frac{1}{m_j}$$

Ceci garantit que  $\dot{\mathcal{V}}_i^\dagger$  est négative pour

$$|\boldsymbol{\varepsilon}| > \frac{|\bar{\boldsymbol{\theta}}|}{2\sqrt{k_0 m_0}}$$

ce qui se traduit par

$$|\boldsymbol{\varepsilon}| < \frac{|\bar{\boldsymbol{\theta}}|}{2\sqrt{k_0 m_0}} \quad (4.39)$$

Le module  $|\boldsymbol{\varepsilon}|$  du vecteur d'erreur est donc borné, ce qui implique que l'état du système l'est également<sup>7</sup>. Cette propriété est valable pour toutes les valeurs positives de  $k_0$  et  $m_0$ .

On peut penser, à partir de (4.39), qu'augmenter  $k_0$  (ou  $m_0$ ) permettrait de réduire la borne. Ceci est loin d'être évident. Le fait d'augmenter ces paramètres pourrait accroître rapidement la valeur initiale  $\boldsymbol{\varepsilon}(0)$  de l'erreur, ce qui réduit la qualité du régime transitoire. Ce problème peut être contourné par l'utilisation d'une initialisation de la trajectoire.

### Initialisation de la trajectoire

Cette méthode permet de réduire la norme  $\mathcal{L}_\infty$  du vecteur d'erreur, en effectuant un bon choix des valeurs initiales de la trajectoire de référence. Cela permet d'augmenter la qualité du régime transitoire, en réduisant la borne des erreurs. Afin de montrer l'effet de  $k_0$  et  $m_0$  sur la borne du vecteur d'erreur, on considère la fonction de Lyapunov

$$\mathcal{V}_i^\dagger = \frac{1}{2} \sum_{j=1}^i \varepsilon_j^2 + \bar{\boldsymbol{\theta}}^T \Gamma^{-1} \bar{\boldsymbol{\theta}}$$

que l'on peut écrire, en introduisant le facteur temps et les normes (pour simplifier, on prend  $\Gamma = \gamma I$ )

$$\begin{aligned} \mathcal{V}_i^\dagger(t) &= \frac{1}{2} |\boldsymbol{\varepsilon}(t)|^2 + \frac{1}{2\gamma} |\bar{\boldsymbol{\theta}}(t)|^2 \\ &\geq \frac{1}{2\gamma} |\bar{\boldsymbol{\theta}}(t)|^2 \end{aligned}$$

Étant donné que  $\mathcal{V}_i^\dagger$  est non croissante, on en déduit

$$\begin{aligned} \frac{1}{2\gamma} |\bar{\boldsymbol{\theta}}(t)|^2 &\leq \mathcal{V}_i^\dagger(t) \\ &\leq \mathcal{V}_i^\dagger(0) \\ &= \frac{1}{2} |\boldsymbol{\varepsilon}(0)|^2 + \frac{1}{2\gamma} |\bar{\boldsymbol{\theta}}(0)|^2 \end{aligned}$$

---

<sup>7</sup>Puisque le signal de référence  $y_r$ , et ses dérivées, sont supposés bornés.

On remarque que, en plus du gain  $\gamma$ , la borne dépend des valeurs initiales  $\tilde{\theta}(0)$  et  $\epsilon(0)$ . Étant donné que l'on ne dispose d'aucun contrôle sur  $\tilde{\theta}(0)$ , une solution, pour réduire cette borne, consiste à choisir  $\epsilon(0) = 0$ . Les variables d'erreurs étant données par

$$\epsilon_i = x_i - \alpha_{i-1}$$

On peut démontrer (Krstić et al. 1995) que forcer les  $\epsilon_i(0)$  à 0, est équivalent à placer les différents  $y_r^{(i)}(0)$  aux valeurs estimées de la sortie et ses dérivées à  $t = 0$ , i.e.  $y^{(i)}(0)$ . Il faut noter que cette initialisation est indépendante des paramètres  $k_i$ ,  $m_i$  et  $\gamma$ . La sortie du procédé<sup>8</sup> étant donnée par

$$y = x_1$$

le signal de référence  $y_r$  doit donc satisfaire

$$\frac{d^i y_r}{dt^i}(0) = x_{i+1}(0), \quad i = 0, \dots, n-1$$

Or, les valeurs initiales  $y_r(0), \dot{y}_r(0), \dots, y_r^{(n)}(0)$  sont dictées par le signal de référence  $y_r$  et n'ont pas à être fixées. Toutefois, étant donné que l'objectif de poursuite ne peut jamais être atteint instantanément, mais asymptotiquement, on peut se permettre de superposer à la consigne initiale  $y_{r0}(t)$ , un signal  $z(t)$  qui tend vers zéro asymptotiquement et dont les valeurs initiales viendront modifier celles de la consigne et ses dérivées, afin d'obtenir les  $y_r(0), \dot{y}_r(0), \dots, y_r^{(n)}(0)$  désirés. Cette modification (forcer les  $\epsilon_i(0)$  à zéro) permet d'obtenir, en utilisant le fait que la fonction  $\mathcal{V}_i^\dagger$  est non croissante,

$$\frac{1}{2}|\epsilon(t)|^2 \leq \mathcal{V}_i^\dagger(0) = \frac{1}{2\gamma}|\tilde{\theta}(0)|^2$$

Ce qui donne,

$$|\epsilon(t)| \leq \frac{1}{\sqrt{\gamma}}|\tilde{\theta}(0)|$$

L'équation (4.39) donne une deuxième borne pour le vecteur  $\epsilon$ . En combinant ces deux bornes, dans la même équation, on obtient

$$|\epsilon(t)| \leq \min\left(\frac{1}{2\sqrt{k_0 m_0}}, \frac{1}{\sqrt{\gamma}}\right)|\tilde{\theta}(0)| \quad (4.40)$$

Ce qui montre que la borne  $\mathcal{L}_\infty$  peut être, systématiquement, réduite par l'augmentation de l'un des trois paramètres  $k_0$ ,  $m_0$  et  $\gamma$ .

<sup>8</sup>On considère le cas dans un procédé à forme paramétrique stricte, i.e. chaîne d'intégrateurs.



## 4.6 Vitesse d'adaptation

La nature explosive des phénomènes instables, présents dans les systèmes non linéaires (finite escape time), exige beaucoup de rigueur dans leur analyse et/ou commande, en particulier dans un environnement adaptatif. La raison est, qu'en plus des non-linéarités propres au procédé, le module d'identification étant intrinsèquement non linéaire, rend l'ensemble très vulnérable.

À cause de la faiblesse de l'équivalence certaine, ce problème a été, jusqu'ici, abordé dans le cadre de la commande adaptative par Lyapunov. L'idée consiste à choisir, simultanément, une commande et une loi d'adaptation, afin de rendre non positive la dérivée d'une fonction de Lyapunov. Le secret de la réussite de cette approche réside dans la combinaison contrôleur stabilisateur/loi d'adaptation rapide. En effet, la vitesse de l'adaptation est dictée par celle du phénomène non linéaire le plus rapide, présent dans le procédé (Krstić et al. 1995). Cette propriété est à double tranchant. D'une part, elle permet d'éviter toute instabilité irréversible de la boucle; de l'autre, elle réduit les possibilités pour le choix des lois d'identification utilisées et utilisables. Celles-ci sont imposées par la méthode de conception et ne peuvent être modifiées sans affecter la stabilité.

Afin pallier cette limitation, des variantes modulaires du backstepping, basées sur le concept de la stabilité entrée/état (Input to State Stability – ISS), ont été proposées (Krstić et al. 1995, Krstić & Kokotović 1995, 1996). Ces méthodes assurent une séparation entre la conception du contrôleur et celle du module d'identification et élargissent, par conséquent, le choix de ce dernier.

## 4.7 Backstepping modulaire

Afin de prévenir l'instabilité des boucles adaptatives, construites autour de procédés non linéaires, deux approches sont disponibles. Les fonctions de réglage, qui est l'approche utilisée jusqu'ici, consiste à choisir des lois d'adaptation suffisamment rapides, afin d'aider le contrôleur à préserver l'intégrité de la boucle. Toutefois, toute la tâche

de stabilisation est assurée par ce dernier, d'où la complexité des expressions obtenues (section 4.5.3). Contrairement à celle-ci, l'approche par design modulaire aborde le problème d'un point de vue **stabilité entrée/état**. Cette notion, qui peut facilement s'exprimer en termes de fonctions de Lyapunov<sup>9</sup>, consiste à concevoir des contrôleurs robustes, qui préservent la bornétude du vecteur d'état, pour des entrées bornées, i.e. entrées bornées/état borné. Un cas particulier de cette approche concerne les entrées se présentant sous forme d'incertitudes paramétriques.

Le design modulaire mise plus sur la robustesse du couple contrôleur/identificateur que sur leurs performances. L'erreur d'identification  $\tilde{\theta}$ , ainsi que la dérivée  $\dot{\tilde{\theta}}$ , sont traitées comme des perturbations externes par le contrôleur. Bien qu'elle n'apparaisse pas directement dans les équations, la dérivée  $\dot{\tilde{\theta}}$  est considérée dans le design, à cause de l'aspect dynamique de  $\tilde{\theta}$ .

L'idée consiste, sous l'hypothèse de la bornétude de ces deux pseudo- perturbations, à concevoir un contrôleur robuste ( $u = \alpha(\mathbf{x}, \hat{\theta})$ ) qui conserve un vecteur d'état borné. En absence des incertitudes, i.e.  $\tilde{\theta} = 0$ , ce contrôleur doit garantir le bon fonctionnement de la boucle (commande non adaptative). Un tel contrôleur peut être réalisé par la méthode du **backstepping robuste**, en utilisant des amortissements non linéaires (section 3.3).

Dans le cadre de la stabilité entrée/état (Sontag & Sussmann 1990, Krstić et al. 1995, Khalil 1996), cette approche s'écrit

$$|\mathbf{x}(t)| \leq \kappa_1(|\mathbf{x}(0)|, t) + \kappa_2 \left( \sup_{0 \leq \tau \leq t} \left\| \begin{bmatrix} \tilde{\theta}(\tau) \\ \dot{\tilde{\theta}}(\tau) \end{bmatrix} \right\| \right)$$

où  $\kappa_1$  est fonction de classe  $\mathcal{KL}$  et  $\kappa_2$  est fonction de classe  $\mathcal{K}$ . Le design consiste, utilisant la notion de l'**ISS-Lyapunov function**, à trouver une commande  $u = \alpha(\mathbf{x}, \hat{\theta})$ ,

<sup>9</sup>On parle alors de fonction de Lyapunov pour la stabilité entrée/état (**ISS-Lyapunov function**).

une fonction de Lyapunov  $\mathcal{V}(\mathbf{x}, \hat{\boldsymbol{\theta}})$  et deux fonctions de classe  $\mathcal{K}$ ,  $\kappa_1$  et  $\kappa_2$ , telles que

$$|\mathbf{x}| \geq \kappa_2 \left( \left\| \begin{bmatrix} \bar{\boldsymbol{\theta}} \\ \dot{\hat{\boldsymbol{\theta}}} \end{bmatrix} \right\| \right)$$

↓

$$\frac{\partial \mathcal{V}}{\partial \mathbf{x}} \left[ \boldsymbol{\varphi}^T \hat{\boldsymbol{\theta}} + \psi \alpha \right] + \frac{\partial \mathcal{V}}{\partial \mathbf{x}} \boldsymbol{\varphi}^T \bar{\boldsymbol{\theta}} + \frac{\partial \mathcal{V}}{\partial \hat{\boldsymbol{\theta}}} \dot{\hat{\boldsymbol{\theta}}} \leq -\kappa_1(|\mathbf{x}|)$$

Ceci revient à dire, qu'à l'extérieur de l'hypersurface définie par  $\kappa_2$ , la fonction de Lyapunov est négative, ce qui implique la bornétude de  $|\mathbf{x}|$  (étant donnée celle de  $\bar{\boldsymbol{\theta}}$  et  $\dot{\hat{\boldsymbol{\theta}}}$ ).

Cette méthode se prête bien à un design récursif, pour la construction systématique du triplet recherché, pour des systèmes d'ordres élevés. Toutefois, à cause de sa grande similitude avec la méthode du **backstepping** robuste, présentée dans la section 3.3, cette méthode ne sera pas détaillée. Le lecteur intéressé est référé à (Krstić et al. 1995).

### Adaptation

La partie adaptation, consiste à choisir un module d'identification, dont le design est indépendant du contrôleur, qui garantit la bornétude des deux "perturbations"  $\bar{\boldsymbol{\theta}}$  et  $\dot{\hat{\boldsymbol{\theta}}}$  et leur convergence vers zéro. Le libre choix de la loi de mise à jour, qui peut être sélectionnée parmi la panoplie des méthodes disponibles (moindres carrés, gradient, Lyapunov, ... ), constitue le point fort de cette approche.

## 4.8 Backstepping avec observateur adaptatif

Comme pour sa version non adaptative, le **backstepping** adaptatif possède une variante à retour de sortie. L'idée consiste à remplacer l'observateur statique, utilisé dans le cas non adaptatif (section 3.4), par un observateur adaptatif. Ce dernier permet d'estimer les états du procédé, en utilisant des estimés des paramètres inconnus, fournis par le module d'adaptation. Le principe du design récursif, basé sur l'utilisation des

amortissements non linéaires, reste le même, avec comme seule différence, le choix de l'observateur adaptatif et des commandes virtuelles. Ces dernières ne doivent directement dépendre d'aucune grandeur inconnue, i.e. ni de l'état du procédé ni des ses paramètres.

### Principe

Pour simplifier les différentes expressions, le principe de la méthode sera illustré sur le procédé de deuxième ordre donné par

$$\begin{aligned}\dot{x}_1 &= \theta\varphi_1(y) + x_2 \\ \dot{x}_2 &= \theta\varphi_2(y) + u \\ y &= x_1\end{aligned}$$

Sous forme matricielle, ces équations s'écrivent

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \boldsymbol{\varphi}(y) + \mathbf{B}u \\ y &= \mathbf{C}^T \mathbf{x}\end{aligned}$$

où

$$\begin{aligned}\mathbf{A} &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, & \mathbf{B} &= \begin{bmatrix} 0 \\ 1 \end{bmatrix}, & \mathbf{C} &= \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \\ \mathbf{x} &= [x_1 \quad x_2]^T & \text{et} & \boldsymbol{\varphi} &= [\varphi_1 \quad \varphi_2]^T\end{aligned}$$

On suppose que seule la sortie  $y = x_1$  est mesurée. La variable d'état  $x_2$  n'est donc pas disponible pour la réalisation d'une loi de commande par retour d'état. On se propose de construire un observateur adaptatif, afin de rendre disponible un estimé  $\hat{x}_2$  de la variable  $x_2$ . Le vecteur d'état estimé s'écrit

$$\hat{\mathbf{x}} = \mathbf{z} + \theta\boldsymbol{\zeta}$$

où  $\mathbf{z}$  est la partie de l'estimé indépendante du paramètre inconnu  $\theta$  et  $\boldsymbol{\zeta}$  est la partie qui en dépend. Ces deux parties constituent les sorties de deux bancs de filtres, dont la combinaison donne l'observateur adaptatif recherché. Ces filtres sont donnés par

$$\begin{aligned}\dot{\mathbf{z}} &= \mathbf{A}_0\mathbf{z} + \mathbf{H}y + \mathbf{B}u \\ \dot{\boldsymbol{\zeta}} &= \mathbf{A}_0\boldsymbol{\zeta} + \boldsymbol{\varphi}(y)\end{aligned}$$

où le vecteur des gains

$$\mathbf{H} = [h_1 \quad h_2]^T$$

est choisi afin de rendre hurwitzienne la matrice

$$\mathbf{A}_0 = \mathbf{A} - \mathbf{H}\mathbf{C}^T$$

L'équation de l'erreur d'estimation résultante s'écrit alors

$$\begin{aligned} \dot{\bar{\mathbf{x}}} &= \dot{\mathbf{x}} - (\dot{\mathbf{z}} + \theta\dot{\boldsymbol{\zeta}}) \\ &= \mathbf{A}\mathbf{x} + \mathbf{B}u + \theta\boldsymbol{\varphi}(y) - \mathbf{A}_0\mathbf{z} - \mathbf{H}y - \mathbf{B}u - \theta\mathbf{A}_0\boldsymbol{\zeta} - \theta\boldsymbol{\varphi}(y) \\ &= \mathbf{A}\mathbf{x} - \mathbf{H}y - \mathbf{A}_0(\mathbf{z} + \theta\boldsymbol{\zeta}) \\ &= \mathbf{A}_0(\mathbf{x} - \mathbf{z} - \theta\boldsymbol{\zeta}) \\ &= \mathbf{A}_0\bar{\mathbf{x}} \end{aligned} \tag{4.41}$$

Ainsi, par le choix adéquat de  $\mathbf{H}$ ,  $\bar{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$  est assuré de converger vers zéro. Le design récursif peut, à présent, être appliqué.

**Étape 1** — On définit la première variable d'erreur

$$\varepsilon_1 = x_1 - y_r$$

sa dérivée est donnée par

$$\begin{aligned} \dot{\varepsilon}_1 &= \dot{x}_1 - \dot{y}_r \\ &= x_2 + \theta\varphi_1(y) - \dot{\alpha}_0 \\ &= (z_2 + \theta\zeta_2) + \bar{x}_2 + \theta\varphi_1(y) - \dot{\alpha}_0 \\ &= z_2 + \theta w_1 + \bar{x}_2 - \xi_0 \end{aligned}$$

où

$$\xi_0 = \dot{\alpha}_0$$

$$w_1 = \zeta_2 + \varphi_1(y)$$

Si l'on considère la fonction de Lyapunov

$$\mathcal{V}_1^\dagger = \frac{1}{2}\varepsilon_1^2 + \frac{1}{2\gamma}\bar{\theta}^2 + \frac{1}{4m_1}\bar{\mathbf{x}}^T \mathbf{P}\bar{\mathbf{x}}$$

où  $\gamma > 0$  et  $m_1 > 0$ , sa dérivée est donnée par

$$\begin{aligned}\dot{\mathcal{V}}_1^\dagger &= \varepsilon_1 \dot{\varepsilon}_1 + \frac{1}{\gamma}\bar{\theta}\dot{\bar{\theta}} - \frac{1}{4m_1}\bar{\mathbf{x}}^T \dot{\bar{\mathbf{x}}} \\ &= \varepsilon_1 \left[ z_2 + \theta w_1 - \xi_0 + \bar{x}_2 \right] + \frac{1}{\gamma}\bar{\theta}\dot{\bar{\theta}} - \frac{1}{4m_1}\bar{\mathbf{x}}^T \dot{\bar{\mathbf{x}}}\end{aligned}$$

On choisit, à présent, parmi les variables disponibles, celle qui sera prise comme commande virtuelle. Or, la seule variable connue et ne dépendant pas du paramètre inconnu  $\theta$  est  $z_2$ . Sa valeur désirée est donnée par

$$\begin{aligned}(z_2)_d &= \alpha_1 + \varsigma_1 \\ &= -k_1 \varepsilon_1 + \xi_0 - \hat{\theta} w_1 + \varsigma_1\end{aligned}$$

où  $k_1 > 0$ . Avec ce choix, la dérivée se réduit à

$$\begin{aligned}\dot{\mathcal{V}}_1^\dagger &= -k_1 \varepsilon_1^2 + \varepsilon_1 \bar{\theta} w_1 + \varepsilon_1 (\bar{x}_2 + \varsigma_1) + \frac{1}{\gamma}\bar{\theta}\dot{\bar{\theta}} - \frac{1}{4m_1}\bar{\mathbf{x}}^T \dot{\bar{\mathbf{x}}} \\ &= -k_1 \varepsilon_1^2 + \frac{1}{\gamma}\bar{\theta}(\dot{\bar{\theta}} + \tau_1) + \varepsilon_1 (\bar{x}_2 + \varsigma_1) - \frac{1}{4m_1}\bar{\mathbf{x}}^T \dot{\bar{\mathbf{x}}}\end{aligned}$$

où  $\tau_1 = \gamma \varepsilon_1 w_1$ . En utilisant l'amortissement non linéaire (section 3.3)

$$\varsigma_1 = -m_1 \varepsilon_1$$

la dérivée vérifie

$$\begin{aligned}\dot{\mathcal{V}}_1^\dagger &\leq -k_1 \varepsilon_1^2 + \frac{1}{\gamma}\bar{\theta}(\dot{\bar{\theta}} + \tau_1) \\ &= -k_1 \varepsilon_1^2 + \frac{1}{\gamma}\bar{\theta}(\tau_1 - \dot{\bar{\theta}})\end{aligned}$$

Afin d'éviter une sur-paramétrisation, la loi de mise à jour  $\dot{\bar{\theta}}$  ne sera pas choisie à cette étape. Son choix est reporté à l'étape suivante.

**Étape 2** — On définit la variable d'erreur

$$\varepsilon_2 = z_2 - (\alpha_1 + \varsigma_1)$$

dont dérivée est donnée par

$$\begin{aligned}\dot{\varepsilon}_2 &= \dot{z}_2 - (\dot{\alpha}_1 + \dot{\varsigma}_1) \\ &= h_2(y - z_1) + u - (\dot{\alpha}_1 + \dot{\varsigma}_1)\end{aligned}$$

La dérivée de  $\alpha_1 + \varsigma_1$  est calculée analytiquement

$$\begin{aligned}\dot{\alpha}_1 + \dot{\varsigma}_1 &= \frac{\partial(\alpha_1 + \varsigma_1)}{\partial y_r} \dot{y}_r + \frac{\partial(\alpha_1 + \varsigma_1)}{\partial \dot{y}_r} \ddot{y}_r + \frac{\partial(\alpha_1 + \varsigma_1)}{\partial \hat{\theta}} \dot{\hat{\theta}} \\ &\quad + \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1} (z_2 + \theta w_1 + \bar{x}_2) + \frac{\partial(\alpha_1 + \varsigma_1)}{\partial \varsigma_2} (\varphi_2 - h_2 \varsigma_2) \\ &= \xi_1 - \bar{\theta} w_2 + \frac{\partial(\alpha_1 + \varsigma_1)}{\partial \hat{\theta}} \dot{\hat{\theta}} + \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1} \bar{x}_2\end{aligned}$$

où  $\xi_1$  est sa partie connue et

$$w_2 = -\frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1} w_1$$

On considère la fonction de Lyapunov

$$V_2^{\dagger} = V_1^{\dagger} + \frac{1}{2} \varepsilon_2^2 + \frac{1}{4m_2} \bar{\mathbf{x}}^T \mathbf{P} \bar{\mathbf{x}}$$

Sa dérivée vérifie

$$\begin{aligned}\dot{V}_2^{\dagger} &\leq -k_1 \varepsilon_1^2 + \frac{1}{\gamma} \tilde{\theta} (\tau_1 - \dot{\hat{\theta}}) - \frac{1}{4m_2} \bar{\mathbf{x}}^T \bar{\mathbf{x}} \\ &\quad + \varepsilon_2 \left[ \varepsilon_1 + h_2(y - z_1) + u - \xi_1 + \bar{\theta} w_2 - \frac{\partial(\alpha_1 + \varsigma_1)}{\partial \hat{\theta}} \dot{\hat{\theta}} - \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1} \bar{x}_2 \right]\end{aligned}$$

La présence de  $\bar{x}_2$  peut être compensée par l'utilisation d'un deuxième amortissement non linéaire, alors que celle de  $\tilde{\theta}$  nécessite le choix de la dynamique d'adaptation. Pour ce faire, on prend comme commande

$$u = \alpha_2 + \varsigma_2$$

où le terme d'amortissement non linéaire  $\varsigma_2$  est choisi tel que

$$\varsigma_2 = -m_2 \left[ \frac{\partial(\alpha_1 + \varsigma_1)}{\partial x_1} \right]^2 \varepsilon_2$$

La dérivée vérifie, avec ce choix,

$$\dot{V}_2^{\dagger} \leq -k_1 \varepsilon_1^2 + \frac{1}{\gamma} \bar{\theta} (\tau_2 - \dot{\bar{\theta}}) + \varepsilon_2 \left[ \varepsilon_1 + h_2(y - z_1) + \alpha_2 - \xi_1 - \frac{\partial(\alpha_1 + \varsigma_1)}{\partial \bar{\theta}} \dot{\bar{\theta}} \right]$$

où la deuxième fonction de réglage  $\tau_2$  est donnée par

$$\tau_2 = \tau_1 + \gamma w_2 \varepsilon_2$$

Celle-ci peut, étant donné que l'on est à dernière étape, être choisie comme loi de mise à jour. En effet, avec le choix

$$\dot{\bar{\theta}} = \tau_2$$

la dérivée de la fonction de Lyapunov se réduit à

$$\dot{V}_2^{\dagger} \leq -k_1 \varepsilon_1^2 + \varepsilon_2 \left[ \varepsilon_1 + h_2(y - z_1) + \alpha_2 - \xi_1 - \frac{\partial(\alpha_1 + \varsigma_1)}{\partial \bar{\theta}} \tau_2 \right]$$

Il suffit à présent de prendre

$$\alpha_2 = -k_2 \varepsilon_2 - \varepsilon_1 - h_2(y - z_1) + \xi_1 + \frac{\partial(\alpha_1 + \varsigma_1)}{\partial \bar{\theta}} \tau_2$$

pour assurer la stabilité de la boucle et la convergence vers zéro des erreurs d'identification, d'observation et de commande (poursuite et/ou régulation).

## 4.9 Conclusion

La version adaptative du **backstepping** permet, en plus de la commande et de la fonction de Lyapunov, de choisir une loi de mise à jour des paramètres inconnus du procédé. Toutefois, l'application directe la procédure récursive présente certains inconvénients. En effet, l'usage prématuré du seul degré de liberté dont nous disposons, en l'occurrence le choix de la loi de mise à jour, nous pousse à en créer d'autres, virtuellement. Cet artifice permet d'assurer la stabilité des sous-systèmes analysés à chaque étape, y compris la dernière, où le système global est considéré. La sur-paramétrisation est le prix à payer pour garantir cette stabilité à chaque étape. Afin de l'éviter, la méthode des **fonctions de réglages** diffère le choix de loi d'adaptation jusqu'à la dernière étape. Étant donné que l'objectif est de stabiliser le système dans son ensemble, cette



méthode ne considère pas la stabilité des sous-systèmes comme une finalité, mais plutôt comme qu'un subterfuge permettant d'atteindre l'objectif principal du design. Ainsi, à chaque étape, on élimine tous les termes instabilisants qui peuvent l'être, sans pour autant fixer la loi d'adaptation. À la dernière étape, le choix de cette loi, combiné avec celui de la commande, permet d'éliminer les éventuels termes résiduels certains à effet néfaste, ainsi que les termes inconnus. La méthode des fonctions de réglages possède, comme le **backstepping** non adaptatif, des variantes robustes et à retour de sortie. Elles se basent sur l'introduction, d'un amortissement non linéaire pour la version robuste, et d'un observateur plus un amortissement non linéaire pour celle à retour de sortie.

# Conclusion

Dans la commande des systèmes non linéaires, la stabilité est un élément très important, auquel la théorie de Lyapunov apporte une solution à la fois sûre et efficace. Dans un contexte adaptatif, cette stabilité passe au premier plan, pour devenir l'élément clé. La dynamique de mise à jour utilisée est beaucoup plus rapide que celle obtenue par l'approche de l'identification. Cette dynamique est prise en compte dans la loi de commande, ce qui permet d'éviter tout effet instabilisant. Les deux lois (adaptation et commande), ainsi que la fonction de Lyapunov qui garantit la stabilité et les performances du système global, sont conçues simultanément. Cette tâche est rendue possible, grâce à l'algorithme du **backstepping**. À cause de sa supériorité, notamment en présence de non-linéarités, le **backstepping** a été adopté comme outil de design. De par sa robustesse inhérente, cette méthode apporte une amélioration substantielle aux performances des contrôleurs proposés.

L'étape suivante dans le développement du **backstepping** concerne la robustification des lois d'adaptation (Ikhouane & Krstic 1998*a,b*) et l'application du **backstepping** à la commande décentralisée (Wen 1994, Jain & Khorrami 1995, Wen & Soh 1997, Benaskeur & Desbiens 1999*a,b,c,d,e*). Ces aspects seront abordés en détail dans la quatrième partie de cette thèse (chapitres 8 à 11). La troisième partie (chapitres 5 à 7), quant à elle, traite des aspects pratiques de l'application du **backstepping** aux procédés réels. Certaines modifications y sont apportées à la procédure du **backstepping**.

## **Troisième partie**

### **Aspects pratiques du backstepping**

# Introduction

Dans cette partie, différents aspects pratiques du **backstepping** sont considérés et des modifications y sont apportées, afin de le rendre utilisable sur des procédés réels. En présence de bruit de mesure, le contrôleur par retour d'état, généré par le **backstepping**, provoque un comportement peu désirable de la variable manipulée. Cet inconvénient s'explique par l'absence de filtrage dans la branche de retour. Une solution, qui consiste à introduire un nouveau type de filtre dans la branche dérivée du contrôleur, est alors proposée. Toutefois, la nature avance de phase de ce type de contrôleurs pose certains problèmes en présence de perturbations externes à moyenne non nulle, appliquées à la sortie et/ou l'entrée du procédé. En régime permanent, des erreurs statiques persistent. Pour une utilisation pratique, une action intégrale est introduite dans le **backstepping**. Cette introduction nécessite une modification de la procédure de design et donne lieu, pour les systèmes linéaires, à des contrôleurs de type **PID**.

Le dernier problème adressé dans cette partie concerne les systèmes avec retards et/ou déphasages non minimaux. Le **backstepping** étant une méthode de commande par retour d'état continu, les retards ne peuvent être représentés. En plus, cette méthode est une variante de la commande par modèle de référence, les systèmes commandés doivent être à phase minimale. Afin de surmonter, simultanément, ces deux limitations, une structure de **Smith** est utilisée.

Le chapitre 5 est consacré au problème du bruit de mesure et son incidence sur le comportement de la variable manipulée. La question des perturbations à moyenne non nulle est soulevée et résolue dans le chapitre 6, où sont également discutés les problèmes du filtrage et de l'adaptation. Le chapitre 7 traite de l'utilisation du **backstepping** dans le

**prédicteur de Smith, pour répondre aux difficultés que pose la commande des systèmes avec retards et/ou déphasages non minimaux.**

# Chapitre 5

## Bruit de mesure

La présence du bruit de mesure constitue l'un des éléments inévitables avec lesquels doivent composer toutes les boucles de contrôle (ou presque). Même si l'amplitude d'un tel bruit est faible, l'utilisation d'action dérivée dans les contrôleurs peut se traduire par un comportement fort indésirable de la variable manipulée<sup>1</sup>, si un filtrage adéquat n'est pas appliqué. Pour les méthodes classiques, l'application d'un tel filtrage ne change pas la procédure de design. La dynamique du filtre, si elle n'est pas directement incluse dans la fonction de transfert du procédé, est quantifiée en terme de perte de phase. Il en va tout autrement pour le **backstepping**. Étant donné que cette méthode découle de la théorie de **Lyapunov**, toute modification, i.e. ajout et/ou suppression d'éléments, doit être justifiée en termes de fonctions de **Lyapunov**. Comme il est montré en détail dans la section 5.3, cette justification est loin d'être directe.

Pour réduire la variance de la commande, en présence de bruit, le **backstepping** avec observateur reste, grâce à l'effet filtre de ce dernier, la seule solution disponible dans le cadre de la théorie de **Lyapunov**. Dans ce chapitre, une alternative à cette méthode est proposée. Il s'agit d'une version du **backstepping** qui, au lieu d'un observateur, utilise un nouveau type de filtre dont le design est basé sur les fonctions de **Lyapunov**. Les détails du design de ce filtre sont donnés dans la section 5.3. Mais avant, l'algorithme de base du **backstepping** est d'abord brièvement rappelé dans la section 5.1. À titre

---

<sup>1</sup>Forte variance.

de comparaison, la technique du **backstepping** avec observateur est également présentée (section 5.2). Par soucis de simplicité de la présentation, les développements s'effectuent sur un procédé linéaire de deuxième ordre. L'extension aux systèmes non linéaires est directe et sera donnée plus loin dans le chapitre.

## 5.1 Design récursif

La fonction de transfert du procédé linéaire à commander est donnée par

$$G_p(s) = \frac{1}{c(s^2 + bs + a)} \quad (5.1)$$

qui correspond, dans le domaine temporel, à l'équation différentielle

$$\ddot{y}(t) + b\dot{y}(t) + ay(t) = \frac{u(t)}{c} \quad (5.2)$$

Le choix

$$\begin{aligned} x_1 &= y(t) \\ x_2 &= \dot{y}(t) \end{aligned}$$

donne comme représentation d'état du système

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -ax_1 - bx_2 + \frac{u}{c} \end{aligned}$$

**Étape 1** — La première variable d'erreur est toujours définie par

$$\varepsilon_1 = y - y_r = x_1 - y_r$$

où  $y_r$  est la trajectoire de référence à suivre. Si l'on prend comme fonction de contrôle de Lyapunov (fcl)

$$\mathcal{V}_1 = \frac{1}{2}\varepsilon_1^2$$

sa dérivée sera donnée par

$$\dot{\mathcal{V}}_1 = \varepsilon_1 \dot{\varepsilon}_1 = \varepsilon_1(x_2 - \dot{y}_r)$$

Pour la rendre négative, il suffit de prendre

$$(x_2)_d \triangleq \alpha_1 = \dot{y}_r - k_1 \varepsilon_1$$

où  $k_1 > 0$  est un paramètre de design.

**Étape 2** — On définit la nouvelle variable d'erreur

$$\varepsilon_2 = x_2 - \alpha_1 = x_2 - \dot{y}_r + k_1 \varepsilon_1$$

et la fcl

$$V_2 = \frac{1}{2} \varepsilon_1^2 + \frac{1}{2} \varepsilon_2^2 \quad (5.3)$$

Sa dérivée, le long des trajectoires du système, est donnée par

$$\begin{aligned} \dot{V}_2 &= -k_1 \varepsilon_1^2 + \varepsilon_2 \left[ \varepsilon_1 + \dot{\varepsilon}_2 \right] \\ &= -k_1 \varepsilon_1^2 + \varepsilon_2 \left[ \frac{u}{c} - ax_1 - bx_2 - \ddot{y}_r + (1 - k_1^2) \varepsilon_1 + k_1 \varepsilon_2 \right] \end{aligned}$$

Pour la rendre négative, il suffit de choisir comme commande

$$\frac{u}{c} = ax_1 + bx_2 + \ddot{y}_r - (1 - k_1^2) \varepsilon_1 - (k_1 + k_2) \varepsilon_2 \quad (5.4)$$

où  $k_2 > 0$  est un nouveau paramètre de design.

### Espace originel

Afin d'être en mesure de réaliser la commande obtenue sous la forme de retour de sortie, les erreurs  $\varepsilon_1$  et  $\varepsilon_2$  sont réécrites dans l'espace  $(x_1, x_2)$ . Cela permet d'obtenir

$$\begin{aligned} u &= c \left[ ax_1 + bx_2 + \ddot{y}_r - (1 - k_1^2) \varepsilon_1 - (k_1 + k_2)(x_2 - \dot{y}_r + k_1 \varepsilon_1) \right] \\ &= c \left[ ax_1 + bx_2 + \ddot{y}_r - (1 + k_1 k_2) \varepsilon_1 - (k_1 + k_2) \dot{\varepsilon}_1 \right] \\ &= c \left[ (a + bs)y + \ddot{y}_r \right] + c \left[ (1 + k_1 k_2) + (k_1 + k_2)s \right] (y_r - y) \\ &= c \left[ a + bs + s^2 \right] y_r + \left[ K_c + T_d s \right] (y_r - y) \end{aligned} \quad (5.5)$$



qui prend la forme d'un contrôleur proportionnelle-dérivée (PD) classique, où

$$K_c = c(1 + k_1 k_2 - a)$$

$$T_d = c(k_1 + k_2 - b)$$

### Performance en présence du bruit

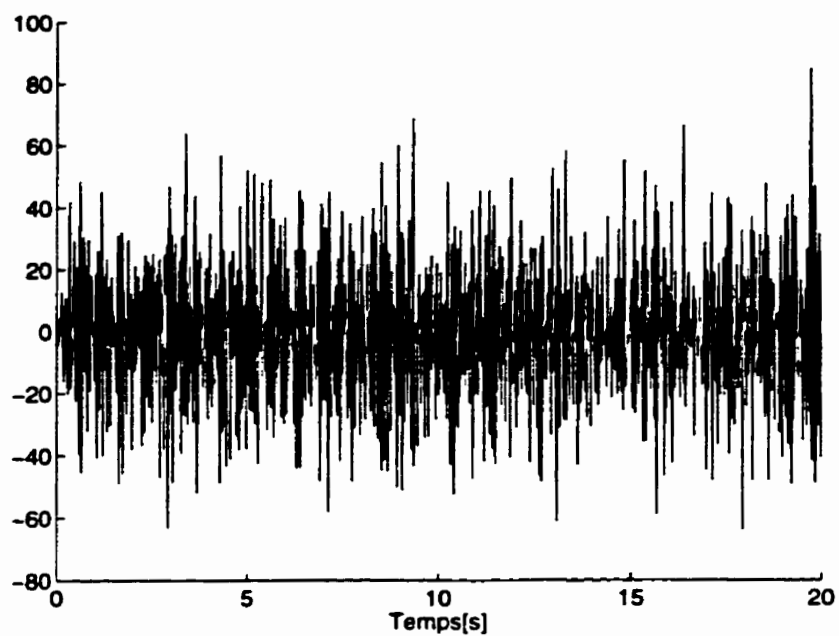
À titre d'illustration, le contrôleur obtenu (5.5) est appliqué, en présence de bruit de mesure (bruit blanc), au procédé instable dont la fonction de transfert est donnée par

$$G_p(s) = \frac{-1}{(1+s)(1-.5s)} = \frac{2}{s^2 - s - 2} \quad (5.6)$$

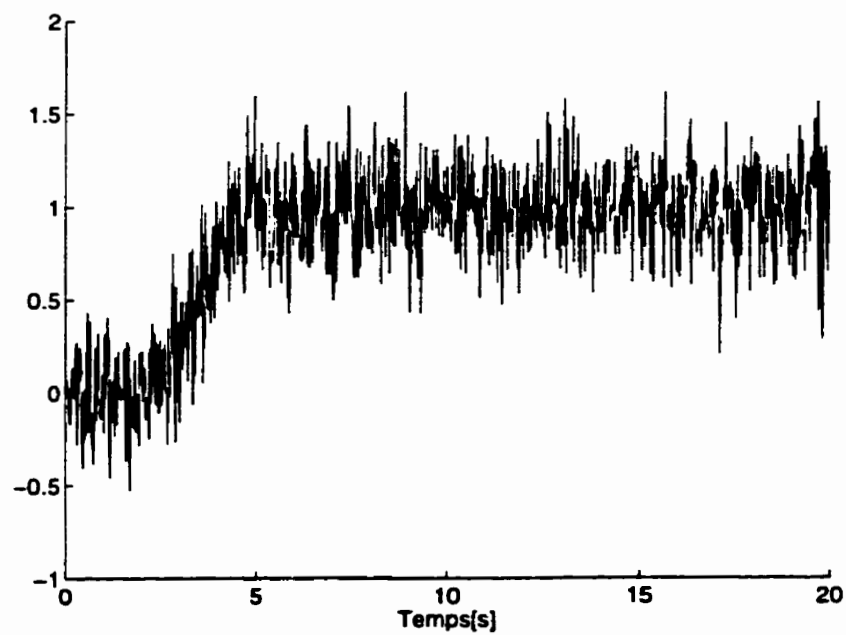
Les résultats des simulations sont donnés par les figures 5.1 (a) & (b). La grande variance de la variable manipulée s'explique par la présence de la fonction dérivée dans le contrôleur. Le bruit de mesure s'en voit, à cause de l'absence de filtrage dans la branche de retour, considérablement amplifié. Les variations de la commande de la figure 5.1 (a), quand elles sont admissibles par l'actionneur, se traduisent par une usure prématurée de celui-ci. Face à de telles situations, et afin d'adoucir la commande, un réflexe naturel consiste à introduire un filtre dans le contrôleur. Une telle démarche n'est cependant pas compatible avec le principe même du design par Lyapunov. Toute modification doit y être justifiée en termes de fonctions de Lyapunov. Cette justification, comme il est montré en détail dans la section 5.3, est loin d'être directe. Mais avant d'aborder ces problèmes de filtrage par Lyapunov, une autre solution, bien connue, sera d'abord présentée dans la section 5.2. Il s'agit de la version avec observateur du backstepping, dont l'effet filtre permet de réduire la variance de la commande.

## 5.2 Observateur

La méthode du backstepping avec observateur a été présentée, en détail, dans la section 3.4. Pour des fins de comparaison, elle sera appliquée dans cette section au procédé de deuxième ordre donnée par (5.6). Pour ce faire, l'équation (5.1) est d'abord



(a) Commande



(b) Réponse

FIG. 5.1: Bruit de mesure : backstepping simple

réécrite sous la forme matricielle

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}u \\ y &= \mathbf{C}^T \mathbf{x}\end{aligned}$$

où

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -a & -b \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1/c \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \text{et} \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

On construit l'observateur

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{H}(y - \mathbf{C}^T \hat{\mathbf{x}}) + \mathbf{B}u$$

où la matrice des gains

$$\mathbf{H} = [h_1 \quad h_2]^T$$

est choisie pour rendre stable l'équation de l'erreur d'estimation

$$\begin{aligned}\dot{\tilde{\mathbf{x}}} &= (\mathbf{A} - \mathbf{H}\mathbf{C}^T)\tilde{\mathbf{x}} \\ &= \mathbf{A}_0 \tilde{\mathbf{x}}\end{aligned}$$

où  $\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$ . La stabilité de la matrice  $\mathbf{A}_0$  garantit l'existence d'une matrice symétrique définie positive  $\mathbf{P}$  telle que

$$\mathbf{P}\mathbf{A}_0 + \mathbf{A}_0^T \mathbf{P} = -\mathbf{I}$$

Cette dernière servira, par la suite, à la construction des différentes fonctions de Lyapunov. On définit, pour ce faire, les deux variables d'erreur

$$\begin{aligned}\varepsilon_1 &= x_1 - y_r \\ \varepsilon_2 &= \hat{x}_2 - (\alpha_1 + \varsigma_1)\end{aligned}$$

**Étape 1** — On choisit comme première fonction de Lyapunov

$$\mathcal{V}_1 = \frac{1}{2}\varepsilon_1^2 + \frac{1}{4m_1}\tilde{\mathbf{x}}^T \mathbf{P}\tilde{\mathbf{x}}$$

dont la dérivée est donnée par

$$\begin{aligned}\dot{\mathcal{V}}_1 &= \varepsilon_1 \dot{\varepsilon}_1 + \frac{1}{4m_1} \tilde{\mathbf{x}}^T \left[ \mathbf{P} \mathbf{A}_0 + \mathbf{A}_0^T \mathbf{P} \right] \tilde{\mathbf{x}} \\ &= \varepsilon_1 (\alpha_1 + \varsigma_1 + \varepsilon_2 + \tilde{x}_2 - \dot{y}_r) - \frac{1}{4m_1} |\tilde{\mathbf{x}}|^2 \\ &\leq \varepsilon_1 (\alpha_1 + \varsigma_1 + \varepsilon_2 + \tilde{x}_2 - \dot{y}_r) - \frac{1}{4m_1} \tilde{x}_2^2\end{aligned}$$

Pour la rendre négative, on prend comme commande virtuelle

$$\alpha_1 = \dot{y}_r - k_1 \varepsilon_1$$

et comme amortissement non linéaire

$$\varsigma_1 = -m_1 \varepsilon_1$$

Avec ces choix, la dérivée s'écrit

$$\begin{aligned}\dot{\mathcal{V}}_1 &\leq -(k_1 + m_1) \varepsilon_1^2 + \varepsilon_1 (\varepsilon_2 + \tilde{x}_2) - \frac{1}{4m_1} \tilde{x}_2^2 \\ &= -k_1 \varepsilon_1^2 + \varepsilon_1 \varepsilon_2 - m_1 \left[ \varepsilon_1 - \frac{1}{2m_1} \tilde{x}_2 \right]^2\end{aligned}$$

**Étape 2** — la dérivée de l'erreur  $\varepsilon_2$  est donnée par

$$\begin{aligned}\dot{\varepsilon}_2 &= \dot{\tilde{x}}_2 - (\dot{\alpha}_1 + \dot{\varsigma}_1) \\ &= \frac{u}{c} + h_2(y - \hat{x}_1) - a\hat{x}_1 - b\tilde{x}_2 - (\dot{\alpha}_1 + \dot{\varsigma}_1)\end{aligned}$$

où

$$\begin{aligned}\dot{\alpha}_1 + \dot{\varsigma}_1 &= \ddot{y}_r - k_1 \dot{\varepsilon}_1 - m_1 \dot{\varepsilon}_1 \\ &= \ddot{y}_r - d_1 (\varepsilon_2 + \tilde{x}_2 - d_1 \varepsilon_1) \\ &= \xi_1 - d_1 \tilde{x}_2\end{aligned}$$

avec  $d_1 = k_1 + m_1$ , et

$$\xi_1 = \ddot{y}_r - d_1 \varepsilon_2 + d_1^2 \varepsilon_1$$

est la partie déterministe de  $\dot{\alpha}_1 + \dot{\varsigma}_1$ . Si l'on prend comme nouvelle fcl

$$\mathcal{V}_2 = \mathcal{V}_1 + \frac{1}{2} \varepsilon_2^2 + \frac{1}{4m_2} \tilde{\mathbf{x}}^T \mathbf{P} \tilde{\mathbf{x}}$$

sa dérivée sera donnée par

$$\begin{aligned}\dot{V}_2 &= \dot{V}_1 + \varepsilon_2 \dot{\varepsilon}_2 - \frac{1}{4m_2} |\bar{\mathbf{x}}|^2 \\ &\leq -k_1 \varepsilon_1^2 + \varepsilon_2 \left[ \varepsilon_1 - \boldsymbol{\xi}_1 + d_1 \bar{x}_2 + h_2 (y - \hat{x}_1) - a \hat{x}_1 - b \hat{x}_2 + \frac{u}{c} \right] - \frac{1}{4m_2} |\bar{\mathbf{x}}|^2 \\ &\leq -k_1 \varepsilon_1^2 + \varepsilon_2 \left[ \varepsilon_1 - \boldsymbol{\xi}_1 + d_1 \bar{x}_2 + h_2 (y - \hat{x}_1) - a \hat{x}_1 - b \hat{x}_2 + \frac{u}{c} \right] - \frac{1}{4m_2} \bar{x}_2^2\end{aligned}$$

Le choix de la commande

$$u = c \left[ a \hat{x}_1 + b \hat{x}_2 - h_2 (y - \hat{x}_1) + \boldsymbol{\xi}_1 - \varepsilon_1 - k_2 \varepsilon_2 + \varsigma_2 \right] \quad (5.7)$$

où l'amortissement non linéaire  $\varsigma_2$  est donné par

$$\varsigma_2 = -m_2 d_1^2 \varepsilon_2$$

permet de réduire la dérivée de la fonction de Lyapunov à

$$\begin{aligned}\dot{V}_2^* &\leq -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 - m_2 (d_1 \varepsilon_2)^2 - \frac{1}{4m_2} \bar{x}_2^2 + d_1 \varepsilon_2 \bar{x}_2 \\ &= -k_1 \varepsilon_1^2 - k_2 \varepsilon_2^2 - \frac{1}{m_1} |\bar{\mathbf{x}}|^2 - m_2 \left[ d_1 \varepsilon_2 - \frac{1}{2m_2} \bar{x}_2 \right]^2 \leq 0\end{aligned}$$

### Espace originel

Le retour à l'espace  $(x_1, x_2)$  s'effectue en remplaçant les variables d'erreur  $\varepsilon_1$  et  $\varepsilon_2$  par leurs valeurs respectives. Cela donne comme expression de la commande finale

$$\begin{aligned}u &= c \left[ a \hat{x}_1 + b \hat{x}_2 - h_2 (y - \hat{x}_1) + \dot{y}_r - (1 - d_1^2) \varepsilon_1 - (d_1 + d_2) \varepsilon_2 \right] \\ &= c \left[ (s^2 + bs - h_2) y_r + (a + h_2) \hat{x}_1 + K_c (y_r - x_1) + T_d (\dot{y}_r - \hat{x}_2) \right]\end{aligned}$$

où

$$K_c = 1 + d_1 d_2 + h_2$$

$$T_d = d_1 + d_2 - b$$

$$d_2 = m_2 d_1^2 + k_2$$

## Simulation

Les résultats de l'application du contrôleur obtenu au problème du bruit de mesure sont donnés par les figures 5.2 (a) & (b). Comparée au **backstepping** de base, dont les réponses sont données par les figures 5.1 (a) & (b), la version avec observateur permet une nette réduction de la variance de la commande. Cette amélioration s'explique par l'effet filtre de l'observateur.

Une autre solution, qui permet d'améliorer le comportement de la variable manipulée, peut être obtenue par un filtrage appliqué sur la branche dérivée du contrôleur. Cette idée, malgré sa simplicité, nécessite des preuves de stabilité beaucoup plus élaborées. En effet, l'introduction d'un tel filtrage dans le **backstepping** n'est pas aussi directe qu'avec les méthodes "classiques".

### 5.3 Filtrage par Lyapunov

Si l'on modifie la structure de la commande (5.5), en introduisant un filtrage sur la variable  $x_2$ , on obtient

$$u^* = c(a + bs + s^2)y_r + K_c(y_r - x_1) + T_d(\dot{y}_r - \hat{x}_2)$$

où  $\hat{x}_2$  est la version filtrée de  $x_2 = \dot{y}$ , qui peut être réécrite

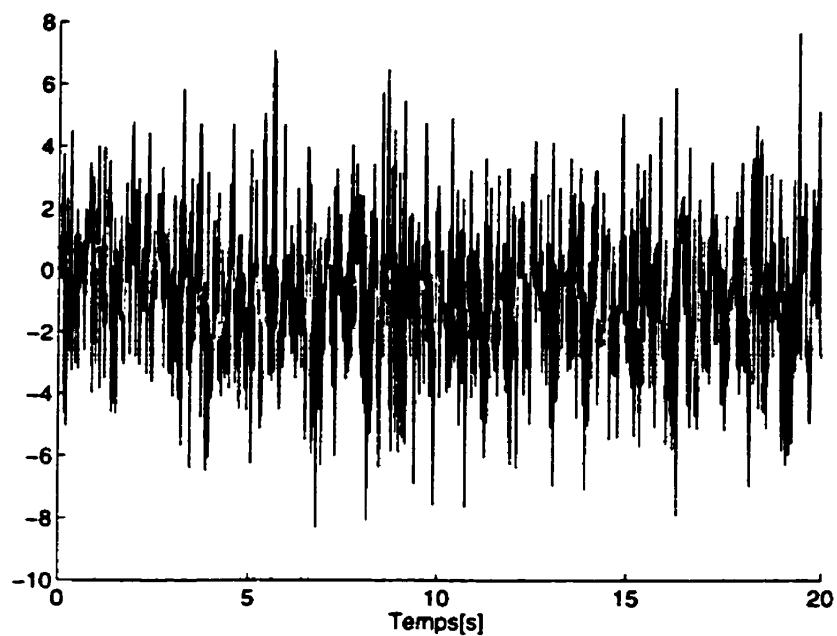
$$\hat{x}_2 = x_2 + \tilde{x}_2$$

où  $\tilde{x}_2$  est l'erreur d'estimation de  $x_2$ . La commande  $u^*$  s'écrit avec ces variables

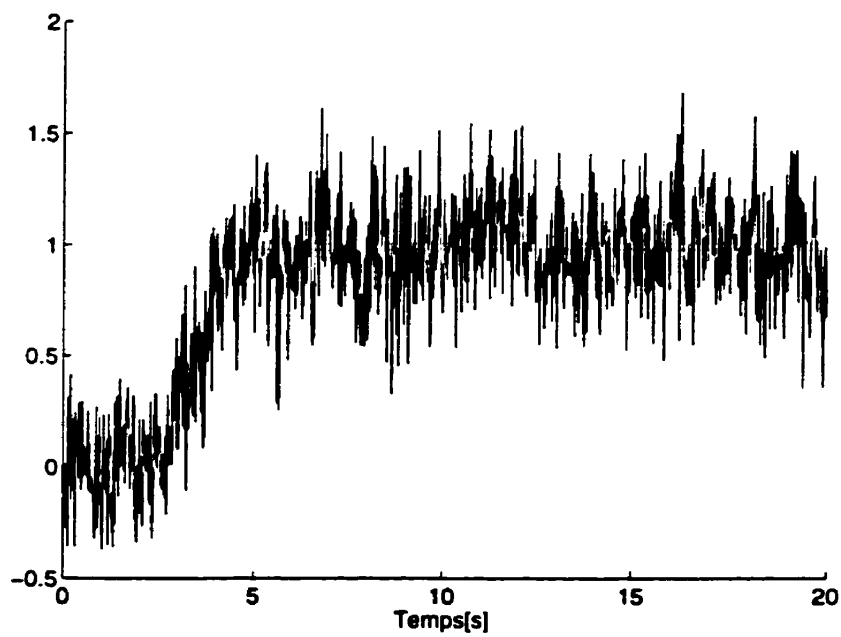
$$\begin{aligned} u^* &= c(a + bs + s^2)y_r + K_c(y_r - x_1) + T_d(\dot{y}_r - x_2) - T_d\tilde{x}_2 \\ &= u - T_d\tilde{x}_2 \end{aligned} \tag{5.8}$$

où  $u$  est la commande non filtrée, donnée par (5.5). L'application de (5.8) au procédé (5.1) donne comme dérivée de la fonction de Lyapunov (5.3)

$$\dot{V}_2 = -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - \frac{T_d}{c}\varepsilon_2\tilde{x}_2$$



(a) Commande



(b) Réponse

FIG. 5.2: Bruit de mesure : backstepping avec observateur

Cette dérivée n'étant pas négative, la stabilité n'est pas garantie. Pour se débarrasser du terme de signe incertain, on définit la nouvelle fonction de Lyapunov

$$V_2^* = \frac{1}{2}\varepsilon_1^2 + \frac{1}{2}\varepsilon_2^2 + \frac{1}{2\gamma}\tilde{x}_2^2$$

Sa dérivée, le long de la trajectoire du système commandé par (5.8), vaut

$$\begin{aligned}\dot{V}_2^* &= -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - \frac{T_d}{c}\varepsilon_2\tilde{x}_2 + \frac{1}{\gamma}\tilde{x}_2\dot{\tilde{x}}_2 \\ &= -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 + \frac{1}{\gamma}\tilde{x}_2\left[\dot{\tilde{x}}_2 - \frac{\gamma T_d}{c}\varepsilon_2\right]\end{aligned}$$

Avec cette notation, un choix intéressant de la dynamique de filtrage devient évident. En effet, il suffit, à présent, de prendre

$$\dot{\tilde{x}}_2 = \frac{\gamma T_d}{c}\varepsilon_2$$

pour assurer la négativité de la dérivée. Or ce que l'on cherche est une dynamique pour  $\hat{x}_2$  et non pas pour  $\tilde{x}_2$ . La relation entre les deux est donnée par

$$\begin{aligned}\dot{\hat{x}}_2 &= \dot{x}_2 + \dot{\tilde{x}}_2 \\ &= \frac{u^*}{c} - ax_1 - bx_2 + \frac{\gamma T_d}{c}\varepsilon_2\end{aligned}$$

où l'erreur  $\varepsilon_2$  vaut

$$\begin{aligned}\varepsilon_2 &= x_2 - \dot{y}_r + k_1\varepsilon_1 \\ &= k_1x_1 + x_2 - (s + k_1)y_r\end{aligned}$$

La loi de mise à jour de l'estimé  $\hat{x}_2$  peut alors être réécrite sous la forme

$$\begin{aligned}\dot{\hat{x}}_2 &= \frac{u^*}{c} - ax_1 - bx_2 + \frac{\gamma T_d}{c}\left[k_1x_1 + x_2 - (s + k_1)y_r\right] \\ &= -ax_1 - bx_2 + (a + bs + s^2)y_r \\ &\quad + \frac{K_c}{c}(y_r - x_1) + \frac{T_d}{c}(\dot{y}_r - \hat{x}_2) + \frac{\gamma T_d}{c}\left[k_1x_1 + x_2 - (s + k_1)y_r\right] \\ &= \left[s^2 + (k_1 + k_2 - \frac{\gamma T_d}{c})s\right]y_r \\ &\quad + (1 + k_1k_2 - k_1\frac{\gamma T_d}{c})(y_r - x_1) - (k_1 + k_2 - b)\hat{x}_2 + (\frac{\gamma T_d}{c} - b)x_2\end{aligned}$$



Afin d'avoir un gain unitaire entre  $x_2$  et  $\hat{x}_2$ , les paramètres de design  $k_1$  et  $k_2$  doivent vérifier

$$k_1 + k_2 = \frac{\gamma T_d}{c}$$

ce qui correspond à

$$\gamma = \frac{c(k_1 + k_2)}{T_d} = \frac{k_1 + k_2}{k_1 + k_2 - b}$$

Étant donné que  $\gamma$  doit être positif, les paramètres  $k_1$  et  $k_2$  doivent également vérifier

$$k_1 + k_2 > b$$

L'équation de  $\hat{x}_2$  peut s'écrire dans ces conditions

$$(s + k_1 + k_2 - b)\hat{x}_2 = (k_1 + k_2 - b)x_2 + (1 - k_1^2)(y_r - x_1) + s^2 y_r$$

où encore

$$\hat{x}_2 = \frac{s}{1 + T_f s} x_1 + \frac{T_f(1 - k_1^2)}{1 + T_f s} (y_r - x_1) + \frac{T_f s^2}{1 + T_f s} y_r$$

où la dynamique du filtre est définie par

$$T_f = \frac{1}{k_1 + k_2 - b}$$

qui est assuré d'être positif.

**Remarque 5.1 (Choix de  $y_r$ )** — On peut penser que l'estimé  $\hat{x}_2$  peut prendre des valeurs extrêmement élevées, à cause de la présence de  $s^2$  au numérateur du dernier terme. Étant donné que la trajectoire de référence  $y_r$  représente la sortie d'un filtre (modèle de référence), il suffit de choisir ce dernier avec un ordre  $\geq 2$ , pour compenser l'effet de  $s^2$ .

□

## Systemes non linéaires

La méthode présentée, développée jusqu'ici sur un système linéaire, se généralise facilement aux systèmes non linéaires. Si les non-linéarités du système dépendent uniquement de la sortie de celui-ci ( $y = x_1$ ), l'extension est directe et aucun traitement supplémentaire n'est requis. Il faut rappeler que le même type de restriction est imposé par la technique du **backstepping** avec observateur. Les calculs de filtres similaires, pour les non-linéarités dépendantes de  $x_2$ , bien que n'ayant abouti, jusqu'ici, à aucun résultat exploitable, continuent de bénéficier de beaucoup d'attention de notre part.

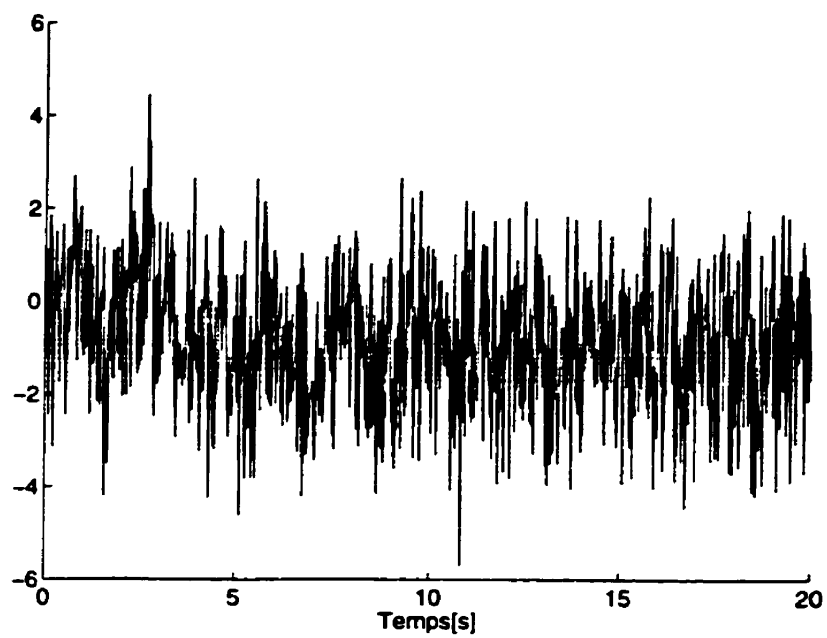
## Simulation

Le contrôleur avec filtre obtenu est appliqué au problème du bruit de mesure. Les résultats de la simulation sont donnés par les figures 5.3 (a) & (b). Comparée au **backstepping** de base (figures 5.1) et au **backstepping** avec observateur (figures 5.2), la version filtrée se démarque par ses bonnes performances. L'utilisation du filtre se traduit par une réduction considérable de la variance de la variable manipulée.

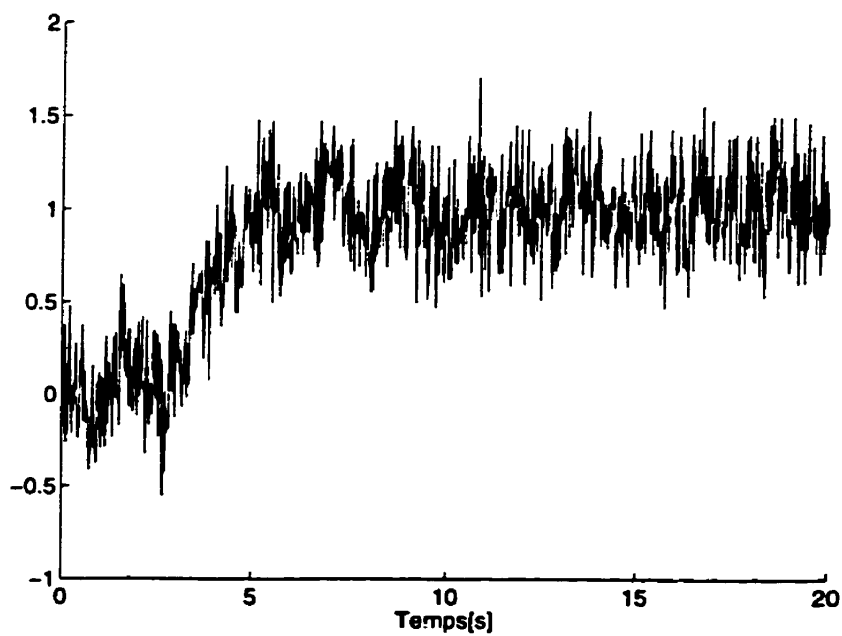
Afin de mieux illustrer la supériorité du contrôleur proposé, ce dernier est également comparé aux deux précédents (avec et sans observateur), pour une perturbation en échelon, appliquée à la sortie du procédé. Les résultats des simulations, donnés par les figures 5.4 (a) & (b), se passent de commentaires. D'une amplitude de -65 pour le **backstepping** simple, le pic de la commande est réduit à -40 pour la version avec observateur et -10 pour celle avec le filtre. L'absence d'actions intégrales ne permet, cependant, à aucun de ces contrôleurs d'éliminer les erreurs, en régime permanent, engendrées par la présence d'une telle perturbation. Cela constitue l'une des limitations majeures du **backstepping**. Ce problème est abordé en détail dans le chapitre 6.

## 5.4 Conclusion

Les problèmes liés à la présence, courante et inévitable, du bruit de mesure sont considérés. Le comportement des contrôleurs générés par la procédure du **backstepping** y est analysé et les limitations soulignées. Une nouvelle solution, utilisant un filtre basé

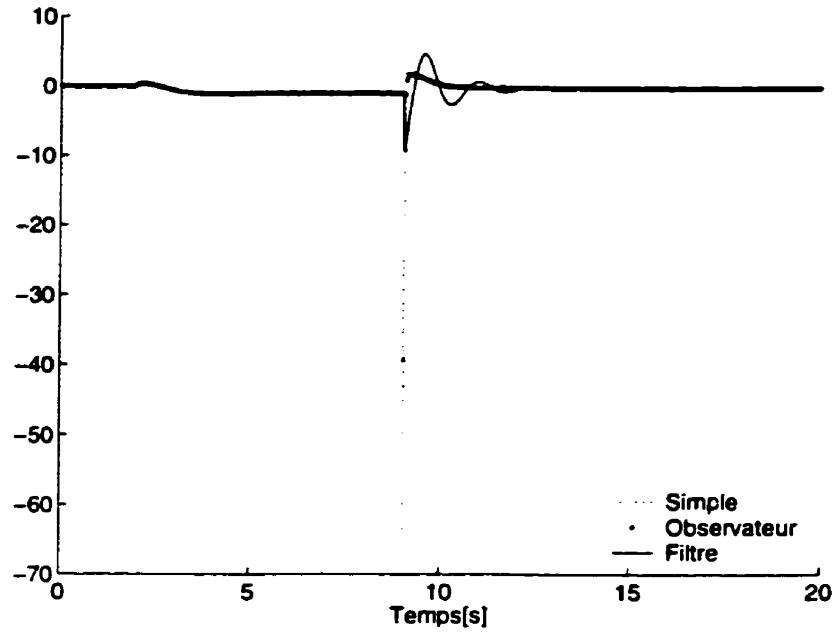


(a) Commande

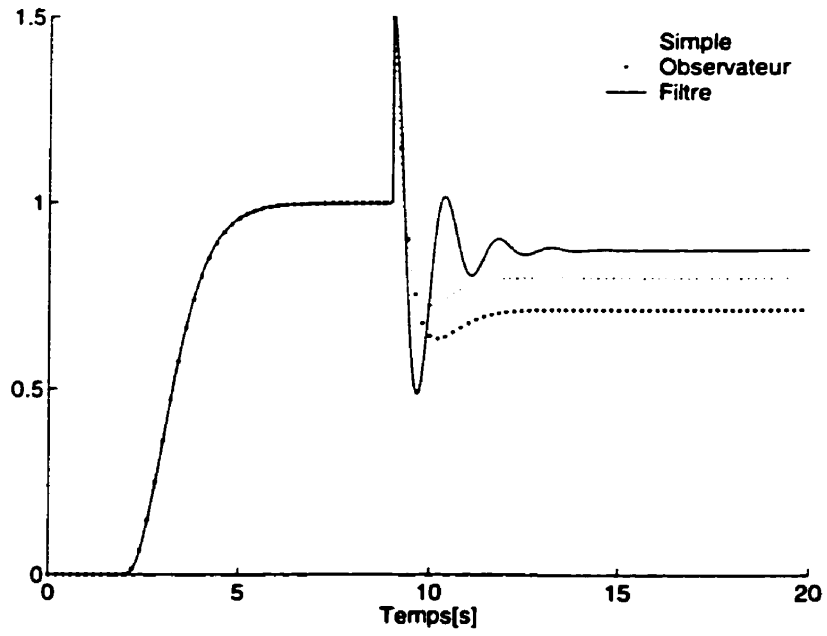


(b) Réponse

FIG. 5.3: Bruit de mesure : backstepping avec filtre



(a) Commandes



(b) Réponses

FIG. 5.4: Perturbation en échelon : backstepping

sur la méthode Lyapunov, est proposée. Cette dernière, qui génère une commande par retour de sortie, surpasse la version avec observateur du backstepping, qui reste la seule variante à retour de sortie disponible de cette méthode.

# Chapitre 6

## Perturbations à moyenne non nulle

Dans ce chapitre, le problème des perturbations externes à moyenne non nulle, appliquées à la sortie et/ou l'entrée du procédé, est considéré. Il a été constaté (section 5.3) qu'en présence de telles perturbations, aucun, des trois contrôleurs proposés, ne garantit des erreurs nulles en régime permanent. Ce dysfonctionnement s'explique, évidemment, par l'absence, dans ces contrôleurs, de l'action intégrale indispensable pour venir à bout des erreurs résiduelles.

Afin d'être en mesure d'éliminer ces erreurs, une solution consiste à doter les régulateurs obtenus d'une action intégrale (Benaskeur & Desbiens 2000). L'idée principale, de la méthode proposée ici, se résume à introduire, d'une manière virtuelle, des intégrateurs dans la fonction de transfert du procédé  $G_p(s)$ . Par la suite, cet intégrateur sera transféré dans le régulateur, afin de garantir des erreurs nulles. Pour rendre possible une telle manipulation, une représentation d'état quasi minimale ( $n + 1$  états) est utilisée pour la fonction  $G_p(s)$ . À cause de ses performances, et surtout de son importance pour la méthode de l'erreur modifiée<sup>1</sup>, cette méthode est traitée en détail dans le présent chapitre.

---

<sup>1</sup>Présentée dans la troisième partie de la thèse, qui lui sera entièrement consacrée.

## 6.1 Backstepping avec action intégrale

La procédure du **backstepping** est modifiée afin de permettre l'introduction d'action intégrale dans les contrôleurs obtenus. Outre les performances en régime permanent, i.e. erreurs nulles, que procure ce type de modification, une amélioration substantielle dans les performances de l'adaptation a été constatée.

Afin de simplifier la présentation, la méthode est, encore une fois, élaborée sur des systèmes linéaires. Son extension aux systèmes non linéaires est directe et ne nécessite aucune modification majeure. Les résultats de cette extension seront toutefois donnés à la fin de la présente section. Pour les systèmes linéaires, cette modification donne lieu à des régulateurs du type PID.

### Principe

L'équation différentielle (5.2) est, à nouveau, considérée. Au lieu de l'utiliser directement dans la procédure du design, cette dernière est remplacée par sa dérivée. Cela donne comme nouvelle équation

$$y^{(3)}(t) + b\ddot{y}(t) + a\dot{y}(t) = \frac{\dot{u}(t)}{c} \\ \triangleq v(t)$$

Le choix des variables d'état

$$x_1 = y(t)$$

$$x_2 = \dot{y}(t)$$

$$x_3 = \ddot{y}(t)$$

permet d'obtenir comme représentation non minimale du système

$$\dot{x}_1 = x_2 \tag{6.1}$$

$$\dot{x}_2 = x_3 \tag{6.2}$$

$$\dot{x}_3 = -ax_2 - bx_3 + v \tag{6.3}$$

pour laquelle l'application du **backstepping** donne comme commande intermédiaire  $v$

$$v = ax_2 + bx_3 + y_r^{(3)} - (k_1^3 - 2k_1 - k_2)\varepsilon_1 - (2 - k_1^2 - k_2^2 - k_1k_2)\varepsilon_2 - (k_1 + k_2 + k_3)\varepsilon_3 \quad (6.4)$$

où  $k_1$ ,  $k_2$  et  $k_3$  sont des paramètres positifs de design et

$$\varepsilon_1 = x_1 - y_r$$

$$\varepsilon_2 = x_2 - \alpha_1$$

$$\varepsilon_3 = x_3 - \alpha_2$$

Les valeurs désirées des variables sont données par

$$\alpha_1 = (x_2)_d = \dot{y}_r - k_1\varepsilon_1$$

$$\alpha_2 = (x_3)_d = \ddot{y}_r - (1 - k_1^2)\varepsilon_1 - (k_1 + k_2)\varepsilon_2$$

La commande (6.4) rend négative la dérivée de la fonction de Lyapunov

$$\mathcal{V}_3 = \frac{1}{2}\varepsilon_1^2 + \frac{1}{2}\varepsilon_2^2 + \frac{1}{2}\varepsilon_3^2 \quad (6.5)$$

qui s'écrit avec ce choix<sup>2</sup>

$$\dot{\mathcal{V}}_3 = -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - k_3\varepsilon_3^2 \leq 0$$

### Espace originel

Pour permettre d'exprimer  $v$  dans l'espace originel  $(x_1, x_2, x_3)$ , on commence par la dernière erreur, dont on remplace la valeur

$$\varepsilon_3 = x_3 - \ddot{y}_r + (1 - k_1^2)\varepsilon_1 + (k_1 + k_2)\varepsilon_2$$

Substituée dans l'expression de  $v$ , cela donne

$$\begin{aligned} v &= y_r^{(3)} + ax_2 + bx_3 - (k_1^3 - 2k_1 - k_2)\varepsilon_1 \\ &\quad - (2 - k_1^2 - k_2^2 - k_1k_2)\varepsilon_2 - (k_1 + k_2 + k_3) \left[ \ddot{y}_r + (1 - k_1^2)\varepsilon_1 + (k_1 + k_2)\varepsilon_2 + x_3 \right] \\ &= ax_2 + bx_3 + y_r^{(3)} \\ &\quad - (k_3 - k_1 - k_1^2k_2 - k_1^2k_3)\varepsilon_1 - (k_1 + k_2 + k_3)\ddot{y}_r - (2 + k_1k_2 + k_1k_3 + k_2k_3)\varepsilon_2 \end{aligned}$$

<sup>2</sup>Les détails sont volontairement omis (voir les chapitres précédents).



La deuxième variable d'erreur étant donnée par

$$\begin{aligned}\varepsilon_2 &= -\dot{y}_r + k_1\varepsilon_1 + x_2 \\ &= (s + k_1)\varepsilon_1\end{aligned}$$

on obtient, pour la dérivée de  $u$ ,

$$\begin{aligned}\dot{u} &= cv \\ &= c(ax_2 + bx_3 + y_r^{(3)}) - c(k_1 + k_2 + k_3)\ddot{\varepsilon}_1 \\ &\quad - c(k_3 - k_1 - k_1^2k_2 - k_1^2k_3)\varepsilon_1 - c(2 + k_1k_2 + k_1k_3 + k_2k_3)(s + k_1)\varepsilon_1\end{aligned}$$

Si l'on introduit, à présent, les paramètres

$$T_d = c(k_1 + k_2 + k_3 - b) \quad (6.6)$$

$$K_c = c(2 + k_1k_2 + k_1k_3 + k_2k_3 - a) \quad (6.7)$$

$$T_i = c(k_1 + k_3 + k_1k_2k_3) \quad (6.8)$$

la dérivée  $\dot{u}$  se met sous la forme

$$\begin{aligned}\dot{u} &= c(ax_2 + bx_3 + y_r^{(3)}) - (T_d + bc)\ddot{\varepsilon}_1 - (K_c + ac)\dot{\varepsilon}_1 - T_i\varepsilon_1 \\ &= c(ax_2 + bx_3 + y_r^{(3)}) - \left[ (T_d + bc)s^2 + (K_c + ac)s + T_i \right] \varepsilon_1 \\ &= c(s^3 + bs^2 + as)y_r - (T_d s^2 + K_c s + T_i)\varepsilon_1\end{aligned}$$

Une simple intégration permet d'obtenir comme commande finale

$$u = c(s^2 + bs + a)y_r - \left( T_d s + K_c + \frac{T_i}{s} \right) \varepsilon_1 \quad (6.9)$$

où l'on reconnaît la structure d'un contrôleur PID parallèle.

### Systèmes non linéaires

Le design, qui vient d'être présenté pour un système linéaire, s'étend facilement au cas non linéaire. Afin d'illustrer cette extension, on considère le système

$$\begin{aligned}\dot{x}_1 &= \varphi_1(x_1) + \psi_1(x_1)x_2 \\ \dot{x}_2 &= \varphi_2(x_1, x_2) + \psi_2(x_1, x_2)u \\ &= \varphi_2(x_1, x_2) + v\end{aligned}$$

où  $\varphi_1$ ,  $\varphi_2$ ,  $\psi_1$  et  $\psi_2$  sont des fonctions non linéaires connues. En dérivant une fois la dernière équation, et en introduisant la nouvelle variable d'état  $x_3$ , on obtient

$$\begin{aligned}\dot{x}_1 &= \varphi_1(x_1) + \psi_1(x_1)x_2 \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= \dot{\varphi}_2(x_1, x_2) + \dot{v} \\ &= \dot{\varphi}_2(x_1, x_2) + w\end{aligned}$$

L'application du **backstepping** à ce nouveau système permet le calcul de la commande intermédiaire  $w$ . Cette dernière est donnée par<sup>3</sup>

$$w = \dot{\alpha}_2 - \varepsilon_2 - k_3\varepsilon_3 - \dot{\varphi}_2(x_1, x_2)$$

dont une simple intégration permet de revenir à  $v$ , qui s'écrit alors

$$\begin{aligned}v &= \frac{w}{s} \\ &= \alpha_2 - \varphi_2(x_1, x_2) - \frac{\varepsilon_2 + k_3\varepsilon_3}{s}\end{aligned}$$

La vraie commande  $u$  peut, maintenant, être calculée

$$\begin{aligned}u &= \frac{v}{\psi_2(x_1, x_2)} \\ &= \frac{1}{\psi_2(x_1, x_2)} \left[ \alpha_2 - \varphi_2(x_1, x_2) \right] - \frac{1}{\psi_2(x_1, x_2)} \left[ \frac{\varepsilon_2 + k_3\varepsilon_3}{s} \right] \\ &= u_0 - \frac{1}{\psi_2(x_1, x_2)} \left[ \frac{\varepsilon_2 + k_3\varepsilon_3}{s} \right]\end{aligned}$$

où l'on voit apparaître la composante  $u_0$ , qui correspond à version classique du **backstepping**, augmentée d'un terme qui contient l'intégrateur introduit par la modification.

### Simulation

Les résultats de l'application du contrôleur obtenu (6.9) au procédé de l'équation (5.6), sont donnés par les **figures 6.1 (a) & (b)**. Ces dernières présentent une comparaison entre la version avec action intégrale (**PID**) et la version obtenue par le **backstepping** direct

<sup>3</sup>Afin d'éviter la redondance, les détails du design sont, encore une fois, volontairement omis.

(qui se réduit à un PD pour les systèmes linéaires). Les comportements en poursuite sont très similaires, alors qu'en régulation le PID assure, contrairement au PD et comme on s'y attendait, une erreur nulle en régime permanent. La variable manipulée du PID, quant à elle, présente un comportement peu désirable. Cela s'explique par l'absence de filtrage sur le signal de sortie dans la boucle de retour. Comme pour le PD (section 5.3), l'introduction d'un tel filtre pose certains problèmes théoriques. Son utilisation n'est pas directe et nécessite une justification par une fonction de Lyapunov. Cette solution est présentée dans la section 6.2.

## 6.2 Filtrage par Lyapunov

Au lieu d'appliquer le filtrage sur  $x_2$ , comme dans le cas du PD, celui-ci est appliqué sur  $x_3$ . Une simple intégration de l'estimé  $\hat{x}_3$  permettra, par la suite, d'obtenir l'estimé de  $x_2$  recherchée. L'introduction d'un filtre sur  $x_3 = \ddot{y}$  consiste à remplacer ce dernier par

$$\hat{x}_3 = x_3 + \tilde{x}_3$$

À cause de l'erreur d'estimation non nulle  $\tilde{x}_3$ , la stabilité ne peut être garantie. L'expression de la commande  $v^*$ , qui utilise  $\hat{x}_3$  au lieu de  $x_3$ , est donnée par

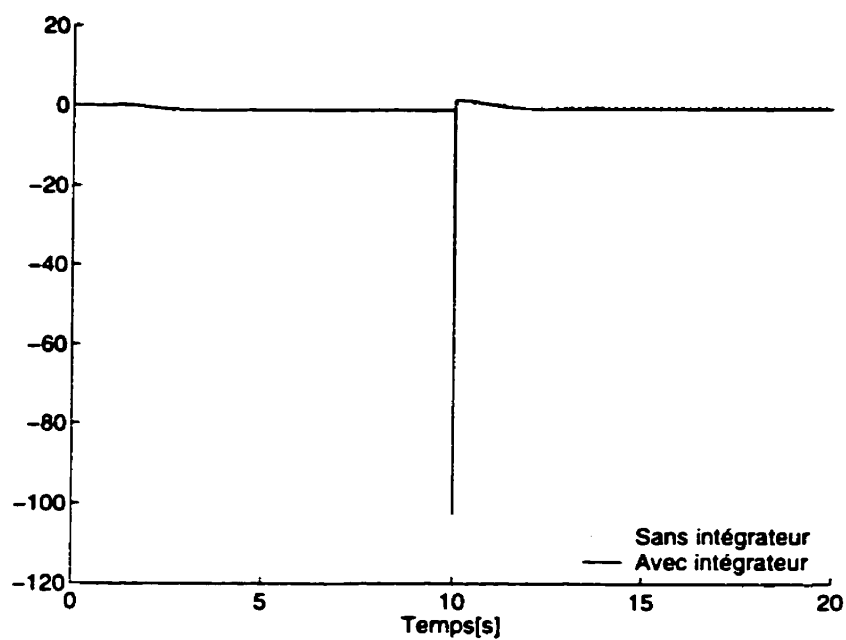
$$\begin{aligned} cv^* &= c(s^3 + bs^2 + as)y_r - (K_c s + T_i)\varepsilon_1 - T_d(\hat{x}_3 - \ddot{y}_r) \\ &= cv - T_d\tilde{x}_3 \end{aligned}$$

où  $v$  est la commande qui utilise la version non filtrée de  $x_3$ . Une modification, identique au cas du PD (section 5.3), donne comme dynamique de filtrage

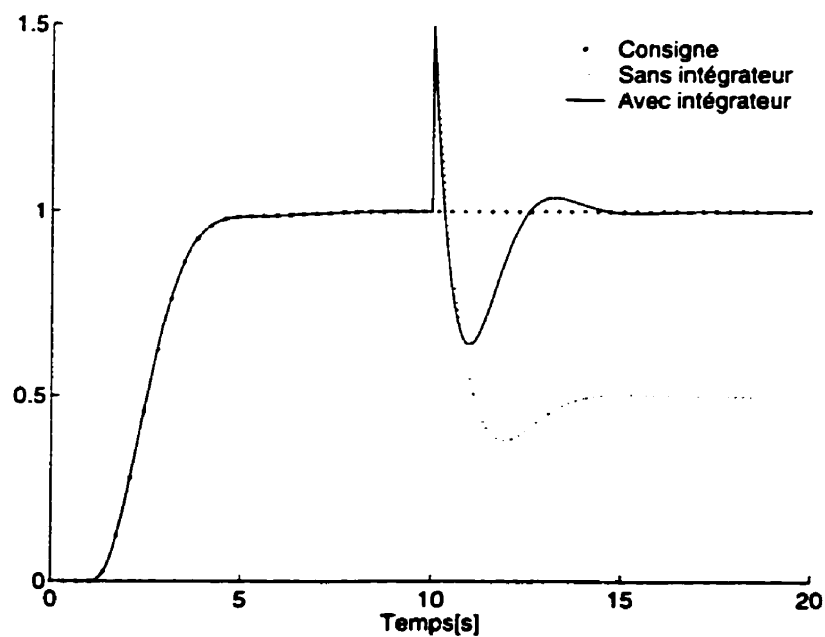
$$\begin{aligned} \dot{\hat{x}}_3 &= \dot{x}_3 + \dot{\tilde{x}}_3 \\ &= v^* - ax_2 - bx_3 + \frac{\gamma T_d}{c}\varepsilon_3 \end{aligned}$$

où  $\gamma$  est un paramètre positif de design. L'erreur  $\varepsilon_3$  étant donnée par

$$\varepsilon_3 = x_3 - \ddot{y}_r + (1 - k_1^2)\varepsilon_1 + (k_1 + k_2)\varepsilon_2$$



(a) Commandes



(b) Réponses

FIG. 6.1: Perturbation en échelon : backstepping avec vs sans intégrateur

la substituer dans l'expression de  $\dot{\hat{x}}_3$  donne

$$\dot{\hat{x}}_3 = v^* - ax_2 + \left(\frac{\gamma T_d}{c} - b\right)x_3 + \frac{\gamma T_d}{c} \left[ (1 - k_1^2)\varepsilon_1 + (k_1 + k_2)\varepsilon_2 - \ddot{y}_r \right]$$

et

$$v^* = ax_2 - \frac{T_d}{c}\dot{\hat{x}}_3 + y_r^{(3)} + \left(\frac{T_d}{c} + b\right)\ddot{y}_r - \left[ \left(\frac{K_c}{c} + a\right)s + \frac{T_i}{c} \right] \varepsilon_1$$

La substitution de  $v^*$  dans l'expression de  $\dot{\hat{x}}_3$  permet d'écrire ce dernier sous la forme

$$\begin{aligned} \dot{\hat{x}}_3 = & -\frac{T_d}{c}\dot{\hat{x}}_3 + \left(\frac{\gamma T_d}{c} - b\right)x_3 + \left[ s^3 + \left(\frac{T_d}{c} + b\right)s^2 \right] y_r \\ & - \left[ \left(\frac{K_c}{c} + a\right)s + \frac{T_i}{c} \right] \varepsilon_1 + \frac{\gamma T_d}{c} \left[ (1 - k_1^2)\varepsilon_1 + (k_1 + k_2)\varepsilon_2 - \ddot{y}_r \right] \end{aligned}$$

La variable  $\hat{x}_3$  étant l'estimé  $x_3$ , afin d'avoir un gain unitaire entre les deux, le paramètre  $\gamma$  doit vérifier

$$\frac{T_d}{c} = \frac{\gamma T_d}{c} - b$$

qui peut s'écrire

$$\gamma = \frac{T_d + cb}{T_d} = \frac{k_1 + k_2 + k_3}{k_1 + k_2 + k_3 - b}$$

En plus, pour que  $\gamma$  soit positif, les paramètres  $k_1$ ,  $k_2$  et  $k_3$  doivent satisfaire

$$k_1 + k_2 + k_3 > b$$

Si les conditions données ci-dessus sont satisfaites, on disposera d'un filtre à la fois stable et de gain unitaire. Ce dernier est donné par

$$\dot{\hat{x}}_3 = \frac{T_d}{c}(x_3 - \hat{x}_3) + s^3 y_r - \left[ \left(\frac{K_c}{c} + a\right)s + \frac{T_i}{c} \right] \varepsilon_1 + \frac{\gamma T_d}{c} \left[ (1 - k_1^2)\varepsilon_1 + (k_1 + k_2)\varepsilon_2 \right]$$

qui, après simplification, peut se mettre sous la forme

$$\dot{\hat{x}}_3 = \frac{T_d}{c}(x_3 - \hat{x}_3) + s^3 y_r + (k_1^2 + k_2^2 + k_1 k_2 - 2)\dot{\varepsilon}_1 + (k_3 + k_1^2 k_2 + k_1 k_2^2)\varepsilon_1$$

L'estimé de l'état  $x_3$  est alors donné par

$$\hat{x}_3 = \frac{T_d s^2}{cs + T_d} x_1 + \frac{s^3}{cs + T_d} y_r + \frac{k_1^2 + k_2^2 + k_1 k_2 - 2}{cs + T_d} \dot{\varepsilon}_1 + \frac{k_3 + k_1^2 k_2 + k_1 k_2^2}{cs + T_d} \varepsilon_1 \quad (6.10)$$

La variable  $x_3$  n'étant pas utilisée dans l'expression (6.9) de la commande finale  $u$ , le calcul de son estimé peut paraître superflu. En effet,  $\hat{x}_3$  ne sera pas utilisée, mais servira à fournir une estimé de  $x_2$  qui, elle, apparaît bien dans l'expression de  $u$ . En intégrant les deux membres de l'égalité (6.10), on obtient

$$\hat{x}_2 = \frac{T_d s}{cs + T_d} x_1 + \frac{s^2}{cs + T_d} y_r + \frac{k_1^2 + k_2^2 + k_1 k_2 - 2}{cs + T_d} \varepsilon_1 + \frac{k_3 + k_1^2 k_2 + k_1 k_2^2}{s(cs + T_d)} \varepsilon_1$$

où seule la sortie  $y = x_1$  est utilisée pour l'estimation de l'état interne  $x_2$ .

### Simulation

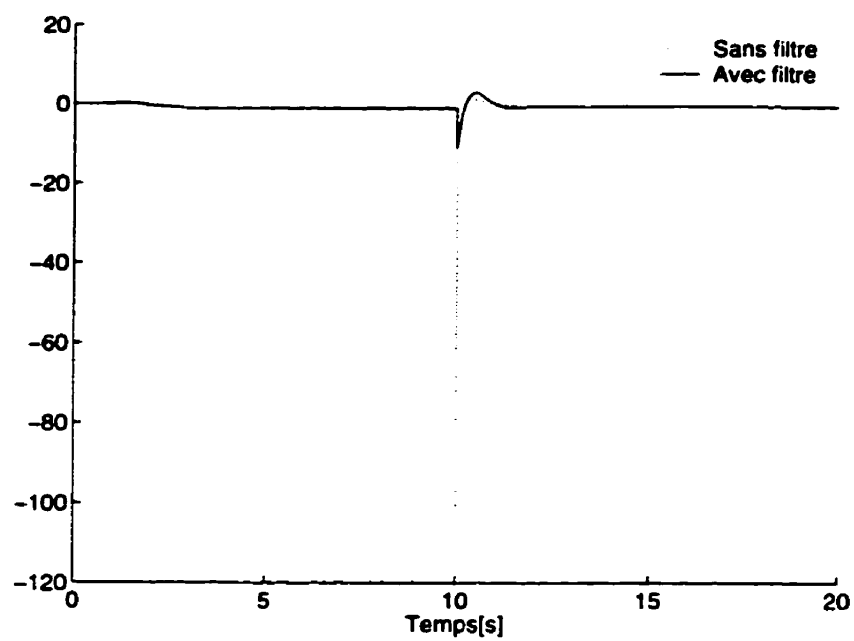
Les résultats de la comparaison des versions avec et sans filtrage du contrôleur (à action intégrale) sont donnés par les figures 6.2 (a) & (b). Ces résultats montrent clairement l'effet du filtre sur le comportement la variable manipulée, dont la variance est considérablement réduite.

## 6.3 Version adaptative

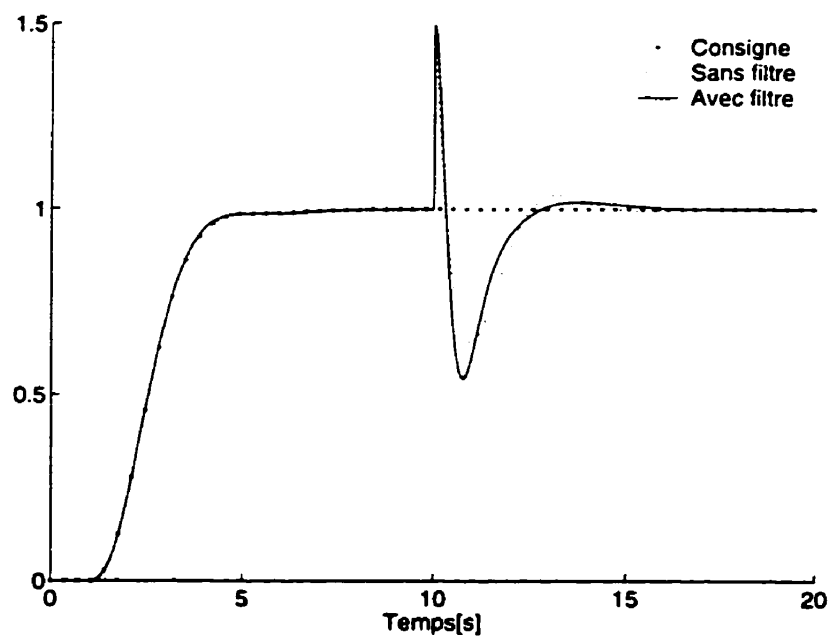
L'introduction de l'action intégrale ne se limite pas aux systèmes à paramètres connus. Dans le contexte adaptatif, outre le rejet des perturbations, cette modification apporte une amélioration substantielle aux performances de l'adaptation. Cette amélioration est plus marquée si l'action intégrale est assortie d'un terme d'amortissement non linéaire.

On considère, à nouveau, le procédé donné par l'équation (5.1). Si les paramètres  $a$ ,  $b$  et  $c$  sont inconnus, on les remplace, dans l'expression du contrôleur obtenu, par leurs estimés respectifs  $\hat{a}$ ,  $\hat{b}$  et  $\hat{c}$ . On suppose que le signe du paramètre  $c$  est connu. On commence par la commande  $v$ , où  $a$  et  $b$  sont remplacés par leur estimés. La version adaptative de  $v$  s'écrit dans ce cas

$$v = \hat{a}x_2 + \hat{b}x_3 + y_a^{(3)} - \left[ (k_1 + k_3 + k_1 k_2 k_3) + (2 + k_1 k_2 + k_1 k_3 + k_2 k_3)s + (k_1 + k_2 + k_3)s^2 \right] \varepsilon_1$$



(a) Commandes



(b) Réponses

FIG. 6.2: Backstepping avec intégrateur : avec vs sans filtre

Ce qui donne pour la dérivée de la vraie commande

$$\begin{aligned}\dot{u} &= \dot{c}v \\ &= cv + \dot{c}v\end{aligned}$$

On considère la fonction de Lyapunov

$$\mathcal{V}_3^\dagger = \frac{1}{2}\varepsilon_1^2 + \frac{1}{2}\varepsilon_2^2 + \frac{1}{2}\varepsilon_3^2 + \frac{\bar{a}^2}{2\gamma_a} + \frac{\bar{b}^2}{2\gamma_b} + \frac{\bar{c}^2}{2\gamma_c|c|} \quad (6.11)$$

où  $\gamma_a$ ,  $\gamma_b$  et  $\gamma_c$  sont des gains d'adaptation positifs. Les erreurs d'estimation sont données par

$$\begin{aligned}\bar{a} &= \hat{a} - a \\ \bar{b} &= \hat{b} - b \\ \bar{c} &= \hat{c} - c\end{aligned}$$

La dérivée de (6.11), le long de la trajectoire du système, s'écrit sous l'action de  $\dot{u}$

$$\dot{\mathcal{V}}_3^\dagger = -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - k_3\varepsilon_3^2 + \frac{\bar{a}}{\gamma_a} \left[ \dot{\bar{a}} + \gamma_a\varepsilon_3x_2 \right] + \frac{\bar{b}}{\gamma_b} \left[ \dot{\bar{b}} + \gamma_b\varepsilon_3x_3 \right] + \frac{\bar{c}}{\gamma_c|c|} \left[ \dot{\bar{c}} + \gamma_c\text{sgn}(c)\varepsilon_3v \right]$$

Un choix évident des dynamiques d'adaptation est donné par

$$\dot{\hat{a}} = \dot{\bar{a}} = -\gamma_a\varepsilon_3x_2 \quad (6.12)$$

$$\dot{\hat{b}} = \dot{\bar{b}} = -\gamma_b\varepsilon_3x_3 \quad (6.13)$$

$$\dot{\hat{c}} = \dot{\bar{c}} = -\gamma_c\varepsilon_3\text{sgn}(c)v \quad (6.14)$$

La dérivée de la fonction de Lyapunov (6.11) vaut, avec ce choix,

$$\dot{\mathcal{V}}_3^\dagger = -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - k_3\varepsilon_3^2 \leq 0$$

**Remarque 6.1 (Intégration)** — Les estimés  $\hat{a}$ ,  $\hat{b}$  et  $\hat{c}$  n'étant pas constants, le passage de  $\dot{u}$  à  $u$  (par intégration) n'est pas aussi simple que dans le cas non adaptatif. Quelques précautions sont à prendre.

□



La commande finale est donnée par

$$u = \int \dot{u} dt = \int \hat{c} v dt = \hat{c} \int v dt - \int \left[ \dot{\hat{c}} \int v dt \right] dt$$

Cette commande peut être réarrangée sous la forme d'une partie PID ( $u_{pid}$ ), correspondant à la version adaptative basée sur équivalence certaine de (6.9), à laquelle vient s'ajouter une composante  $u_{dyn}$ , dont le seul rôle est de compenser les effets instabilisants de l'adaptation. La commande  $u$  s'écrit ces notations

$$u = u_{pid} + u_{dyn}$$

où

$$\begin{aligned} u_{pid} &= \hat{c}(s^2 + \hat{b}s + \hat{a})y_a - \left[ \hat{T}_d s + \hat{K}_c + \frac{\hat{T}_i}{s} \right] \varepsilon_1 \\ u_{dyn} &= -\hat{c} \int x_1 \hat{a} dt - \hat{c} \int x_2 \hat{b} dt - \int \left[ \dot{\hat{c}} \int v dt \right] dt \\ &= \gamma_a \hat{c} \int x_1 x_2 \varepsilon_3 dt + \gamma_b \hat{c} \int x_2 x_3 \varepsilon_3 dt + \gamma_c \text{sgn}(c) \int \left[ v \int v dt \right] \varepsilon_3 dt \end{aligned}$$

Les paramètres (adaptatifs) de réglage sont donnés par

$$\hat{T}_d = \hat{c}(k_1 + k_2 + k_3 - \hat{b}) \quad (6.15)$$

$$\hat{K}_c = \hat{c}(2 + k_1 k_2 + k_1 k_3 + k_2 k_3 - \hat{a}) \quad (6.16)$$

$$\hat{T}_i = \hat{c}(k_1 + k_3 + k_1 k_2 k_3) \quad (6.17)$$

## 6.4 Amortissement non linéaire

Une modification des commandes, par l'introduction d'un terme d'amortissement non linéaire, permet d'améliorer considérablement la qualité des régimes transitoires. Pour ce faire, on définit d'abord la nouvelle commande intermédiaire  $w$  par l'introduction, dans l'expression de  $v$ , de deux termes d'amortissement non linéaire, par rapport aux erreurs d'estimation  $\tilde{a}$  et  $\tilde{b}$ . Cela donne

$$w = v - \varepsilon_3 (m_1 x_2^2 + m_2 x_3^2)$$

La dérivée de la vraie commande  $u$  est, également, augmentée d'un terme d'amortissement non linéaire, mais par rapport à l'erreur d'estimation  $\bar{c}$

$$\dot{u} = \hat{c}w - \text{sgn}(c)m_3\varepsilon_3w^2$$

Avec ces choix et les lois d'adaptation (6.12)–(6.14), la dérivée de la fonction de Lyapunov (6.11) devient

$$\dot{\mathcal{V}}_3^\dagger = -k_1\varepsilon_1^2 - k_2\varepsilon_2^2 - k_3\varepsilon_3^2 - m_1(\varepsilon_3x_2)^2 - m_2(\varepsilon_3x_3)^2 - \frac{m_3}{|c|}(\varepsilon_3w)^2 \leq 0$$

Si l'adaptation est inhibée, i.e.  $\gamma_a = \gamma_b = \gamma_c = 0$ , cette dérivée vérifie<sup>4</sup>

$$\dot{\mathcal{V}}_3^\dagger \leq -k_0|\boldsymbol{\varepsilon}|^2 + \frac{1}{4m_0}|\bar{\boldsymbol{\theta}}|^2$$

où

$$\boldsymbol{\varepsilon} = [\varepsilon_1 \quad \varepsilon_2 \quad \varepsilon_3]^T$$

$$\bar{\boldsymbol{\theta}} = [\bar{a} \quad \bar{b} \quad \bar{c}]^T$$

$$m_0 = \min\{m_1, m_2, m_3|c|\}$$

$$k_0 = \min\{k_1, k_2, k_3\}$$

On constate que  $\dot{\mathcal{V}}_3^\dagger$  est négative pour

$$|\boldsymbol{\varepsilon}| > \frac{|\bar{\boldsymbol{\theta}}|}{2} \sqrt{\frac{1}{k_0m_0}}$$

ce qui garantit la bornétude du vecteur d'erreur  $\boldsymbol{\varepsilon}$ , puisque

$$|\boldsymbol{\varepsilon}| < \frac{|\bar{\boldsymbol{\theta}}|}{2} \sqrt{\frac{1}{k_0m_0}}$$

### Intégration

Le passage de  $\dot{u}$  à  $u$  (par intégration) pose les mêmes difficultés que dans le cas sans amortissement non linéaire. La commande finale est donnée par

$$u = u_{dyn} + u_{ant} + u_{pid} \tag{6.18}$$

---

<sup>4</sup>Voir section 3.3

La partie **équivalence certaine** de la commande ( $u_{pid}$ ) reste identique au cas sans amortissement non linéaire, alors que la partie qui tient compte de la dynamique d'adaptation qui est donnée par

$$\begin{aligned} u_{dyn} &= -\hat{c} \int x_1 \dot{a} dt - \hat{c} \int x_2 \dot{b} dt - \int \left[ \dot{\hat{c}} \int w dt \right] dt \\ &= \gamma_a \hat{c} \int x_1 x_2 \varepsilon_3 dt + \gamma_b \hat{c} \int x_3 x_2 \varepsilon_3 dt + \gamma_c \operatorname{sgn}(c) \int \left[ \varepsilon_3 w \int w dt \right] dt \end{aligned}$$

s'en voit, à travers  $w$ , affectée. La contribution des termes d'amortissement non linéaire, quant à elle, vaut

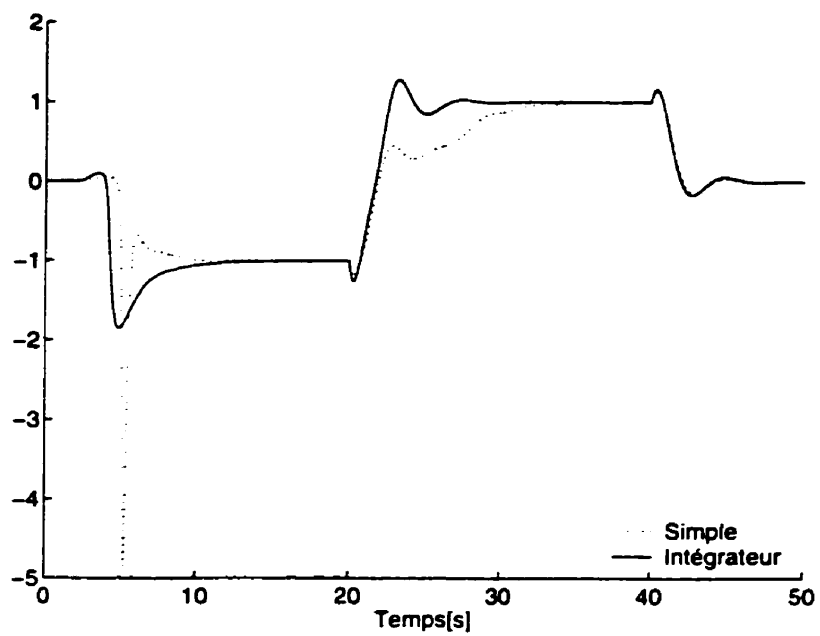
$$u_{anl} = - \int \left[ m_1 \hat{c} x_2^2 + m_2 \hat{c} x_3^2 + m_3 \operatorname{sgn}(c) w^2 \right] \varepsilon_3 dt$$

## 6.5 Simulation

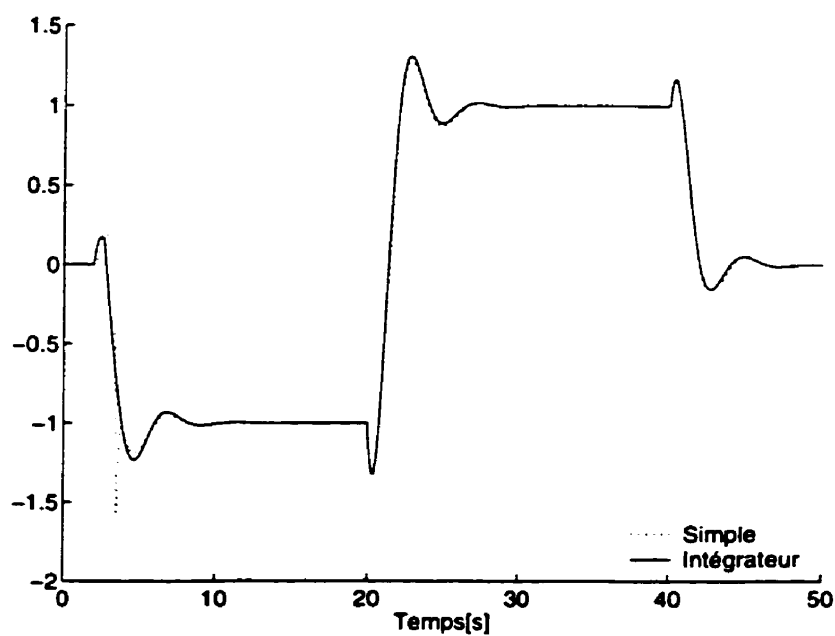
Les résultats comparatifs de l'application des deux contrôleurs PID et PD adaptatifs sont donnés par les figures 6.3 à 6.12. Chacune des figures donne les résultats des versions sans et avec amortissement non linéaire. Les figures 6.3 à 6.7 montrent clairement la supériorité du contrôleur avec intégrateur sur la version de base, notamment en présence d'amortissements non linéaires.

### Perturbation

Cette supériorité est plus marquée, si le procédé est sujet à des perturbations à moyenne non nulle. Dans ce dernier cas, le **backstepping** de base est complètement incapable de garantir des erreurs nulles, qu'il s'agisse d'erreurs de commande ou d'identification. Les figures 6.8 à 6.12 montrent, encore une fois, les bonnes performances de la version avec intégrateur, pour une perturbation en échelon d'amplitude .5 appliquée à la sortie du procédé, à  $t = 12s$ .

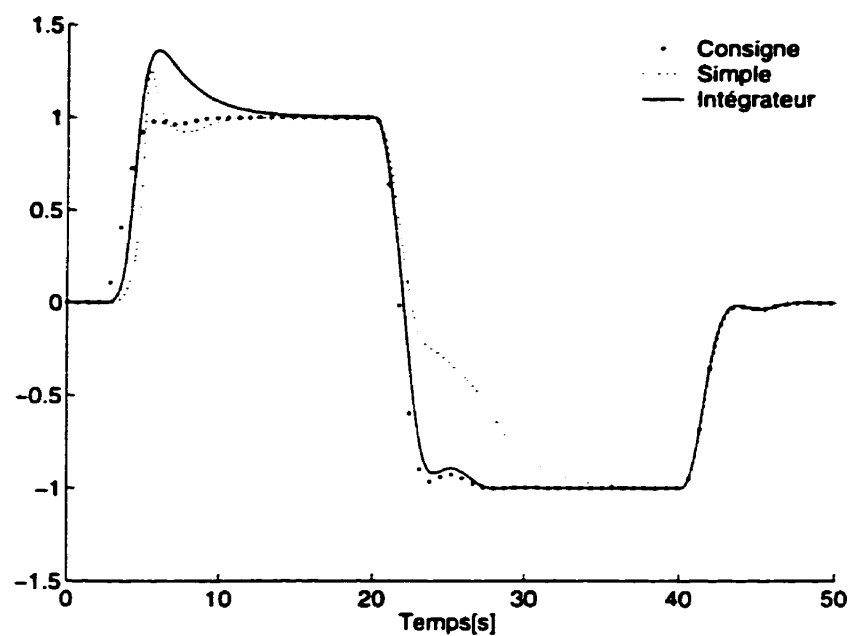


(a) Sans amortissement non linéaire

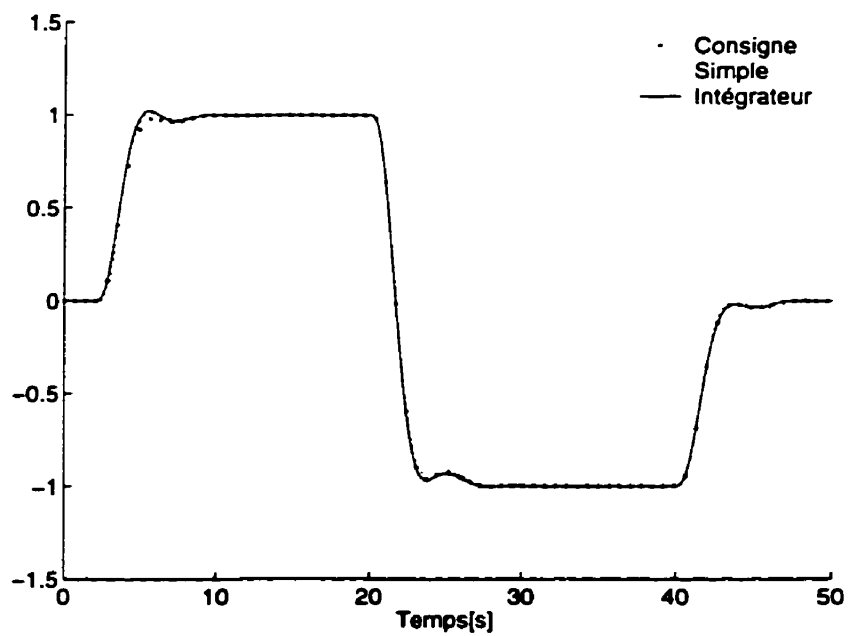


(b) Avec amortissement non linéaire

FIG. 6.3: Backstepping simple vs intégrateur : commandes  $u$

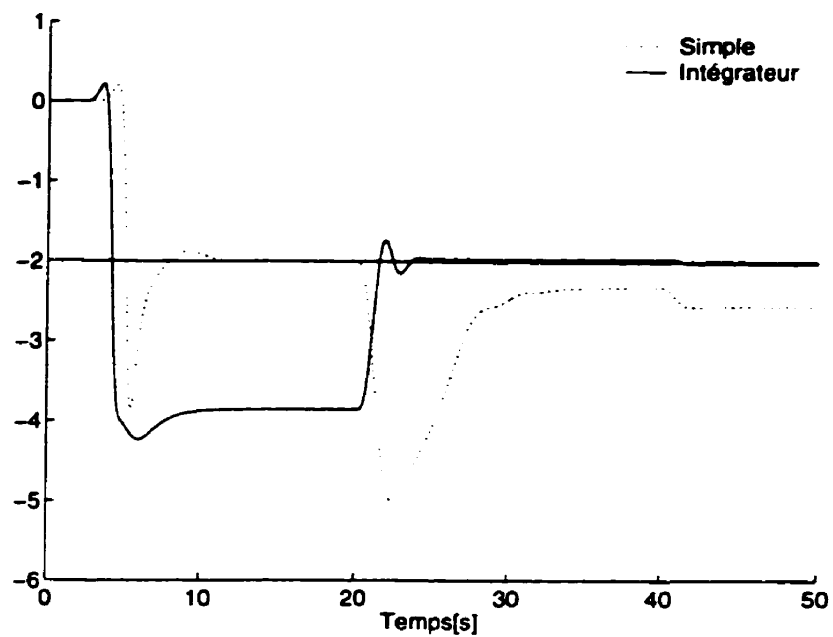


(a) Sans amortissement non linéaire

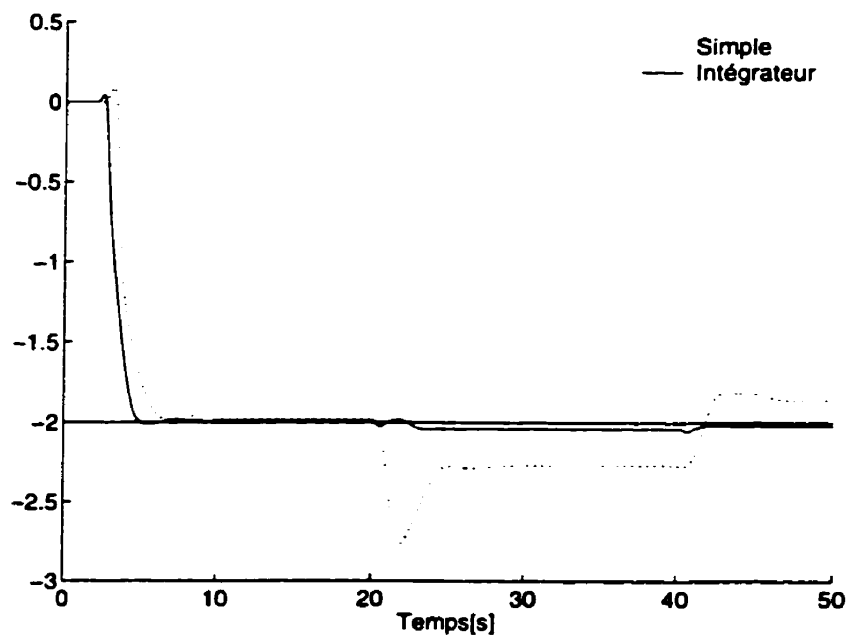


(b) Avec amortissement non linéaire

FIG. 6.4: Backstepping simple vs intégrateur : sorties  $y$

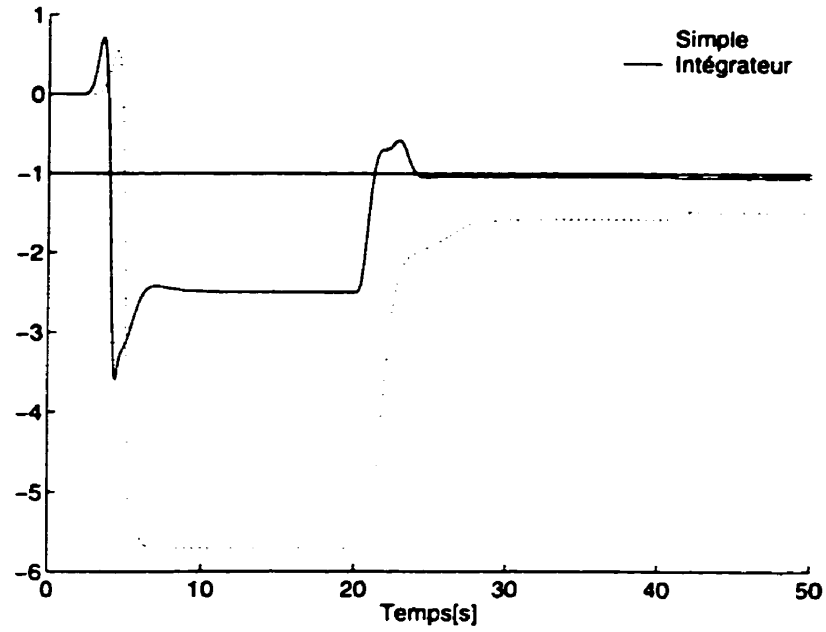


(a) Sans amortissement non linéaire

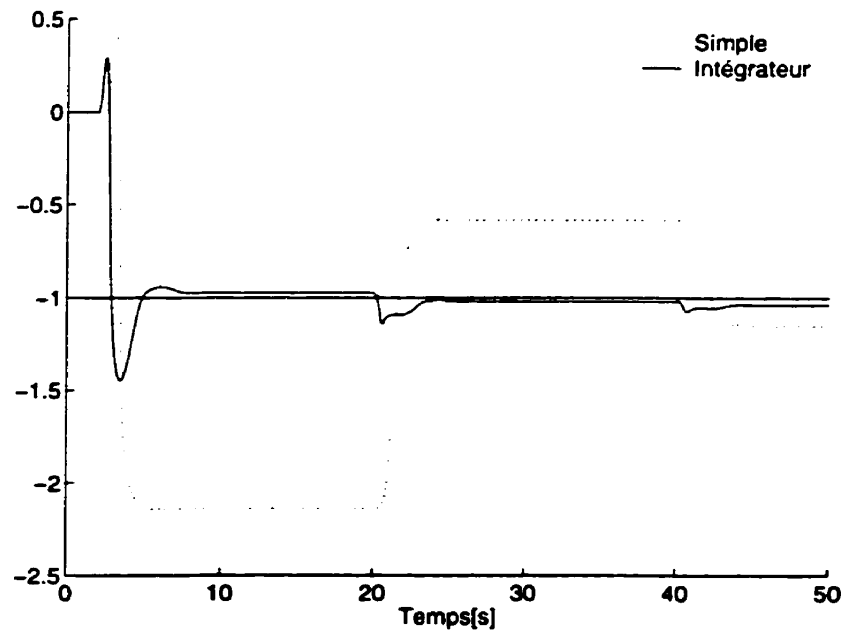


(b) Avec amortissement non linéaire

FIG. 6.5: Backstepping simple vs intégrateur : estimé de  $a$

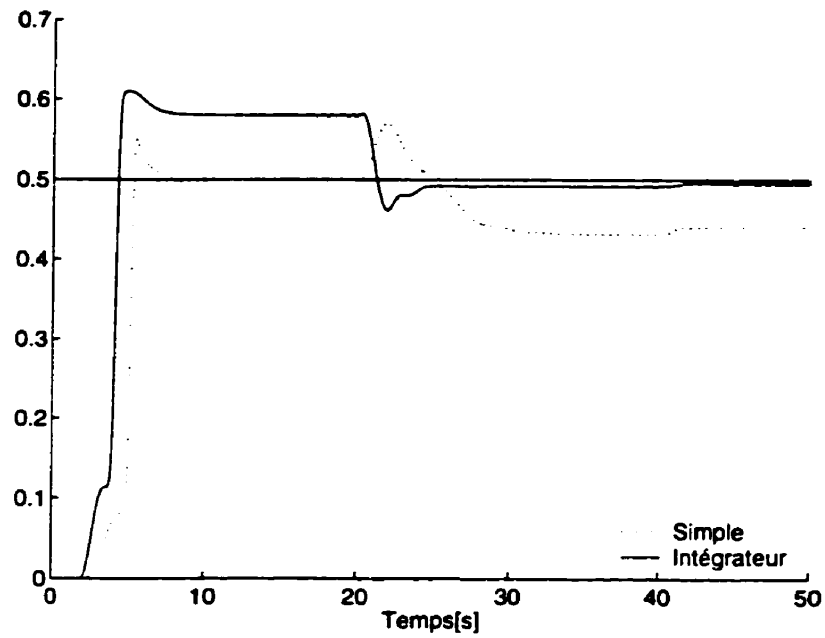


(a) Sans amortissement non linéaire

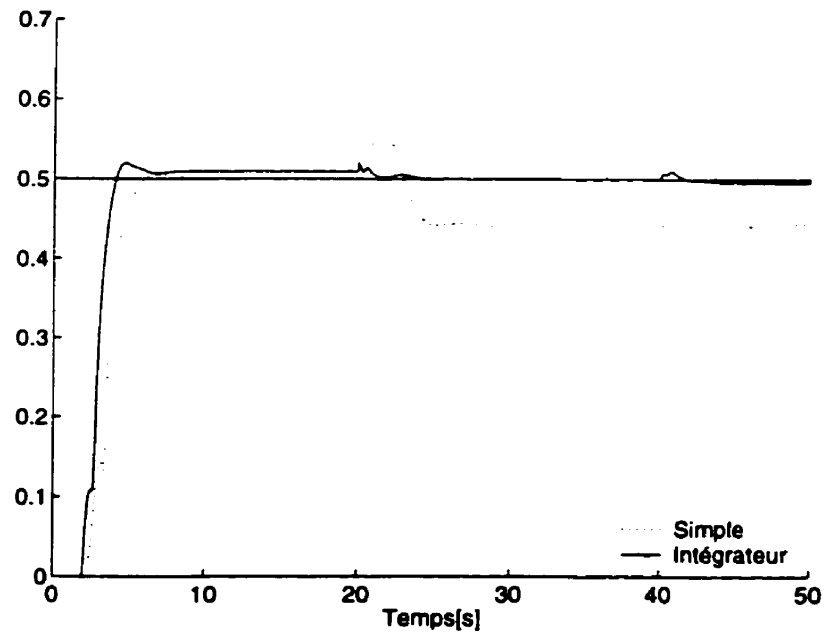


(b) Avec amortissement non linéaire

FIG. 6.6: Backstepping simple vs intégrateur : estimé de  $b$



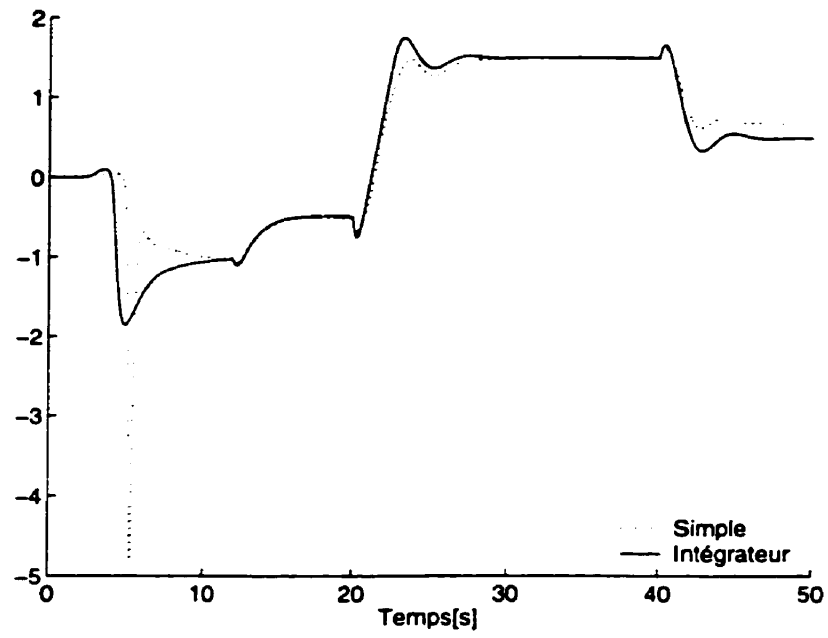
(a) Sans amortissement non linéaire



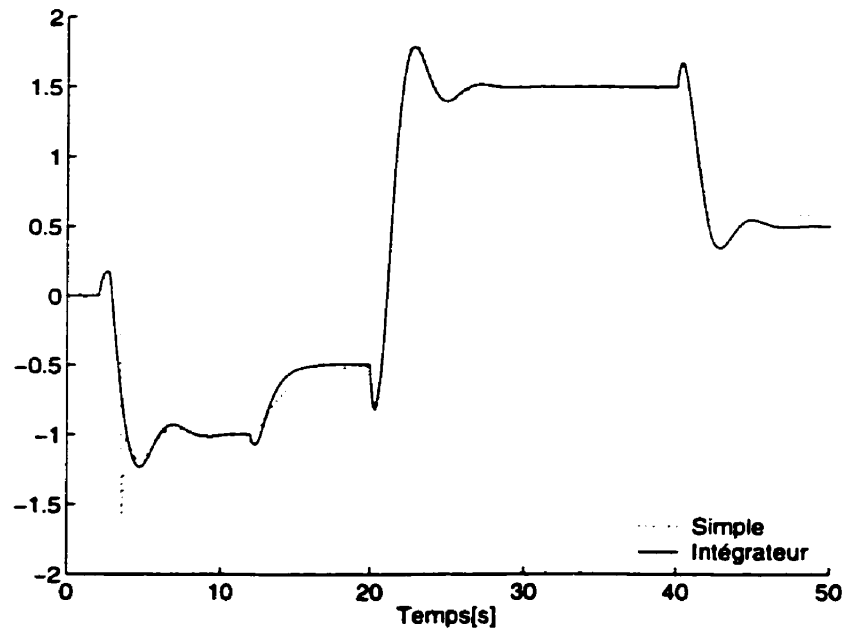
(b) Avec amortissement non linéaire

FIG. 6.7: Backstepping simple vs intégrateur : estimé de  $c$



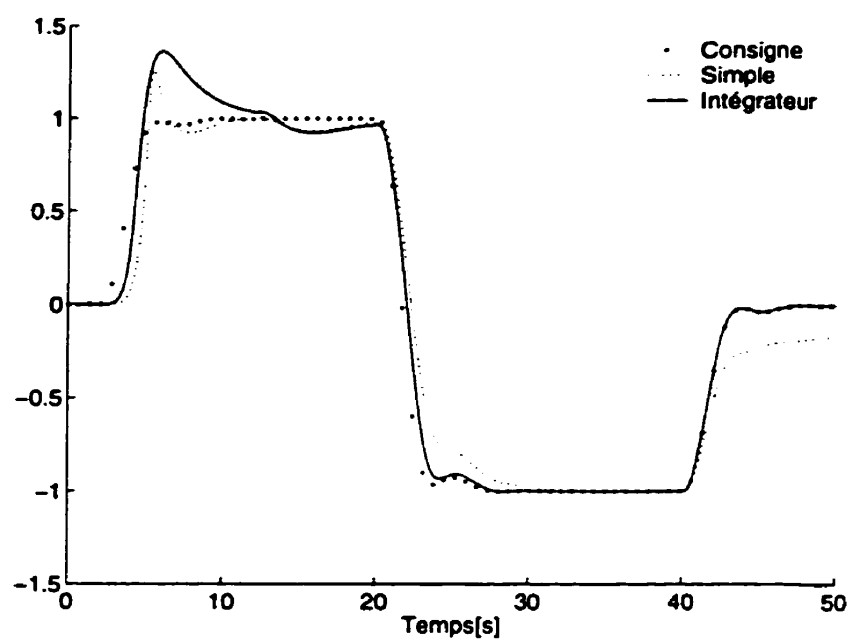


(a) Sans amortissement non linéaire

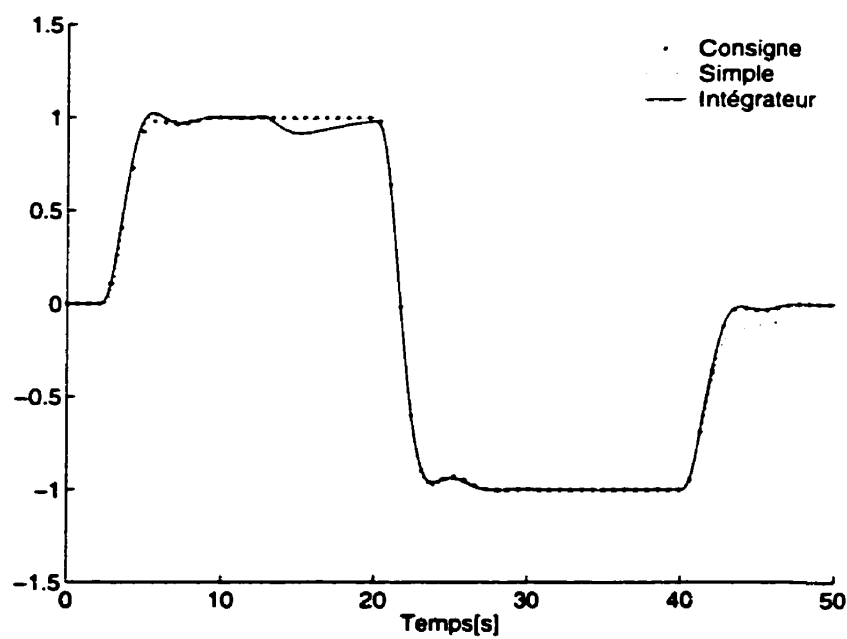


(b) Avec amortissement non linéaire

FIG. 6.8: Perturbation à la sortie (backstepping simple vs intégrateur) :  $u$

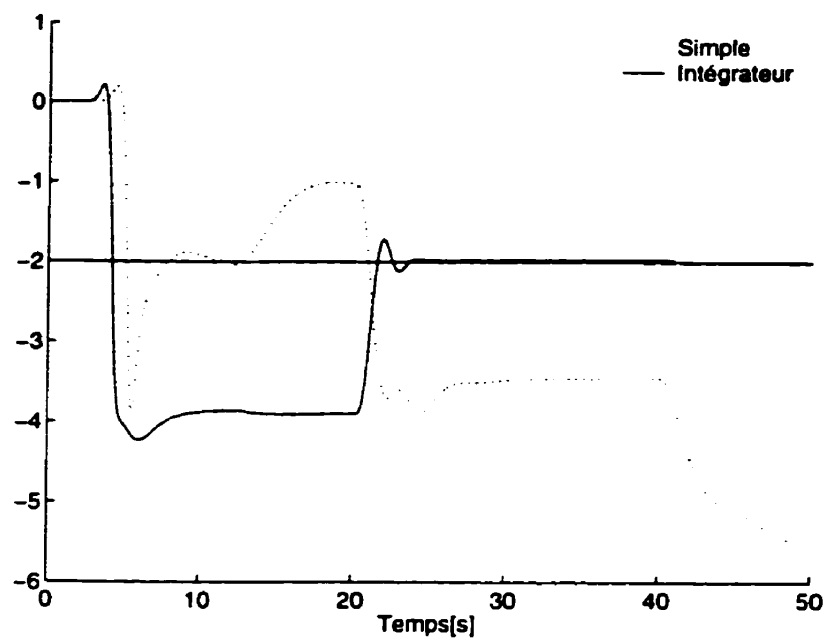


(a) Sans amortissement non linéaire

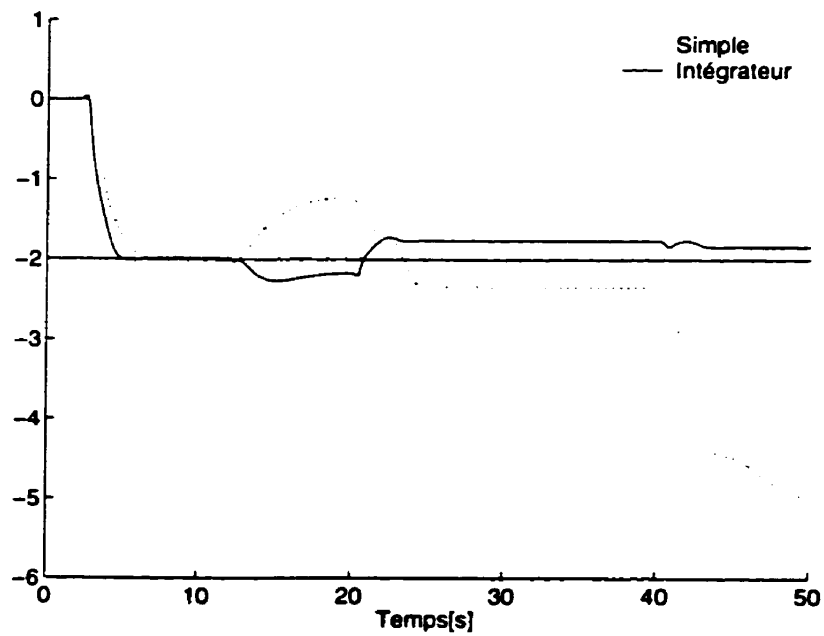


(b) Avec amortissement non linéaire

FIG. 6.9: Perturbation à la sortie (backstepping simple vs intégrateur) :  $y$

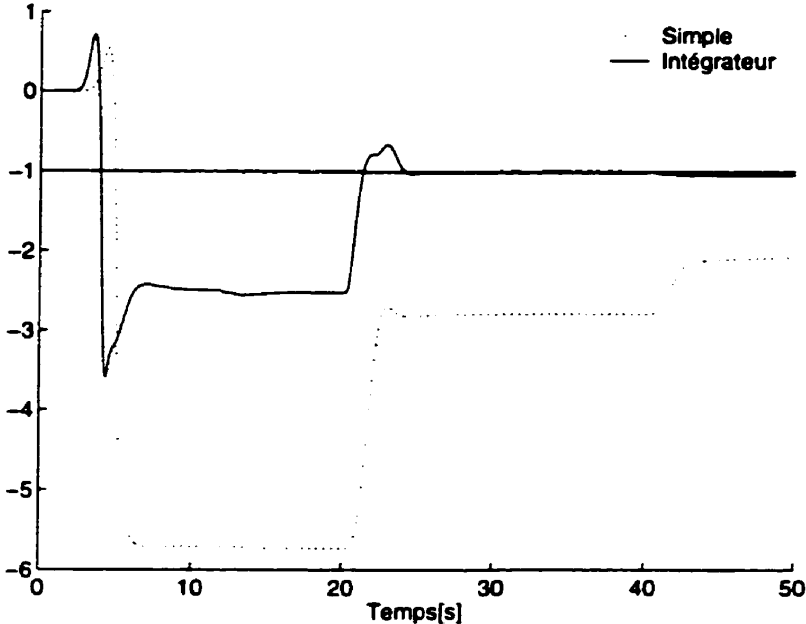


(a) Sans amortissement non linéaire

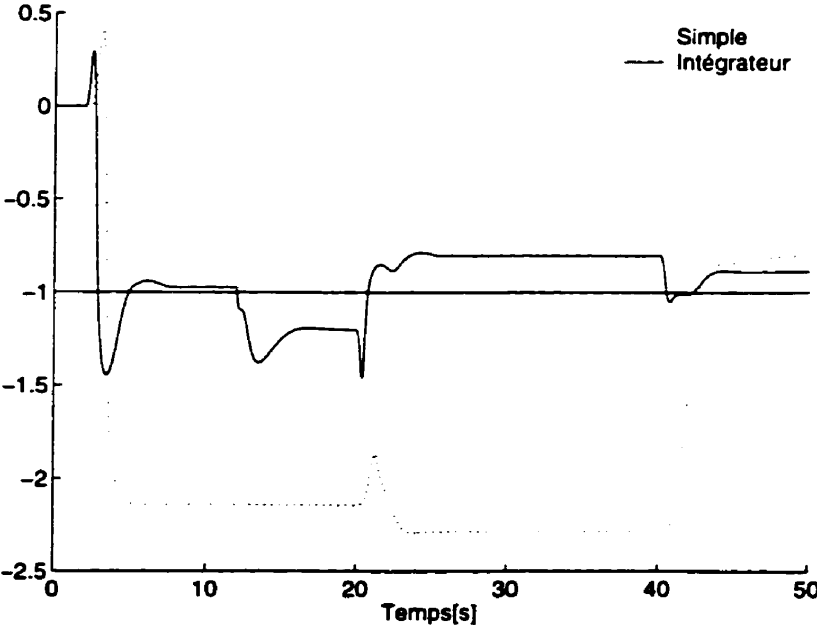


(b) Avec amortissement non linéaire

FIG. 6.10: Perturbation à la sortie (backstepping simple vs intégrateur) : a

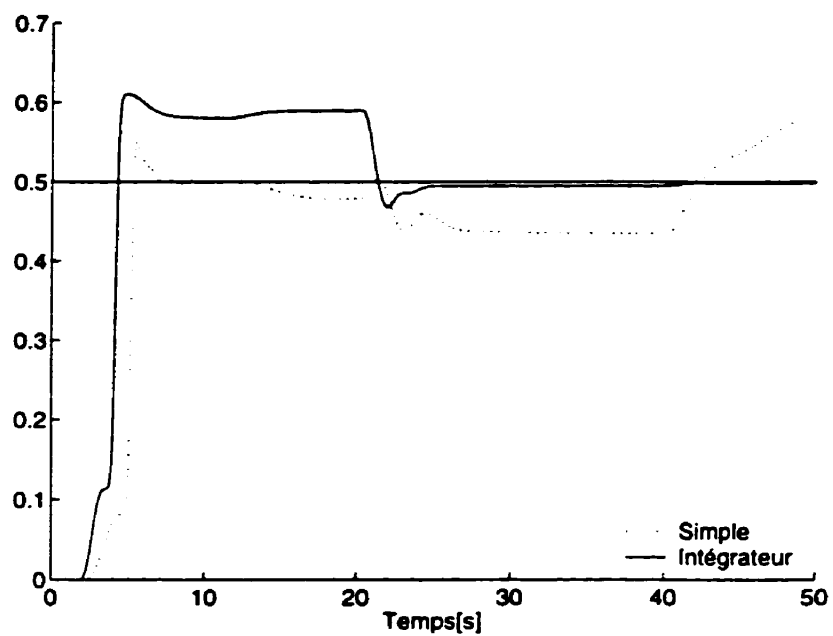


(a) Sans amortissement non linéaire

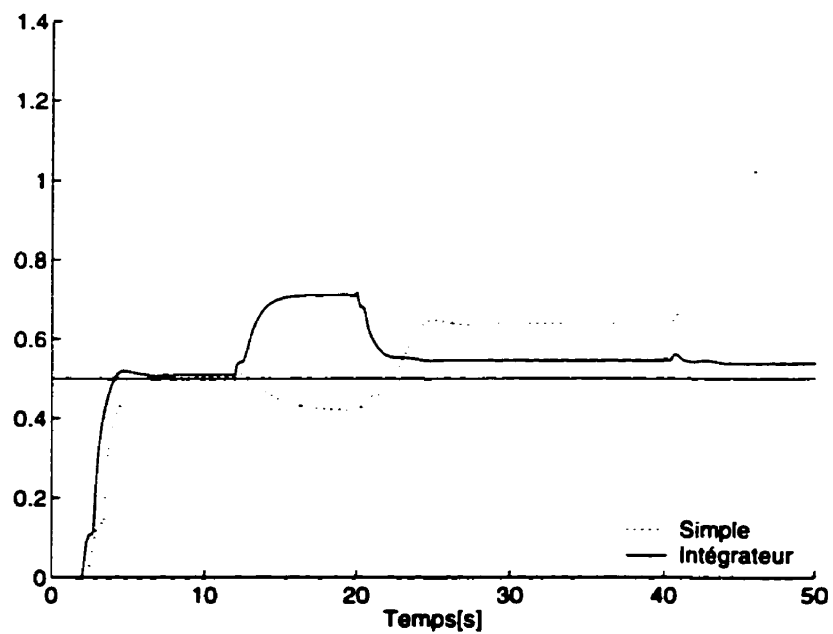


(b) Avec amortissement non linéaire

FIG. 6.11: Perturbation à la sortie (backstepping simple vs intégrateur) : b



(a) Sans amortissement non linéaire



(b) Avec amortissement non linéaire

FIG. 6.12: Perturbation à la sortie (backstepping simple vs intégrateur) : c

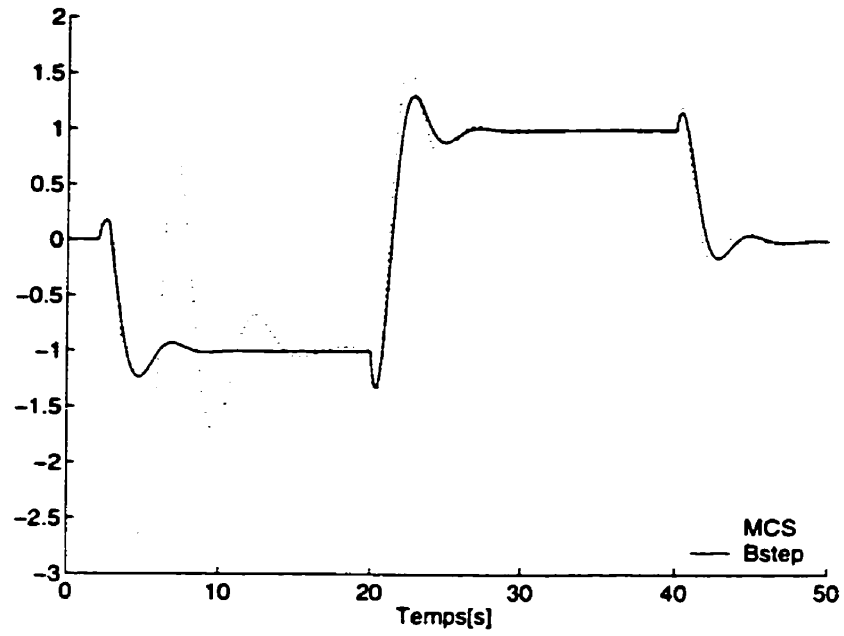
## 6.6 Équivalence certaine

Afin de mieux illustrer les performances du contrôleur PID adaptatif basé sur la méthode **backstepping**, celui-ci est comparé, dans les figures 6.13 et 6.14, à son équivalent basé sur le principe de l'**équivalence certaine**, utilisant les moindres carrés simples comme stratégie de mise à jour des paramètres inconnus. Pour ce dernier, le caractère instable du procédé choisi ici impose un choix prudent du gain d'adaptation. Ce choix se traduit, en toute évidence, par une lente convergence de l'identification, qui explique la contre performance du contrôleur dans la première portion de la réponse donnée par la figure 6.13 (a). La robustesse inhérente au **backstepping**, renforcée par l'introduction d'un amortissement non linéaire, permet d'augmenter les gains d'adaptation sans provoquer l'instabilité de la boucle, assurant ainsi une meilleure convergence des paramètres.

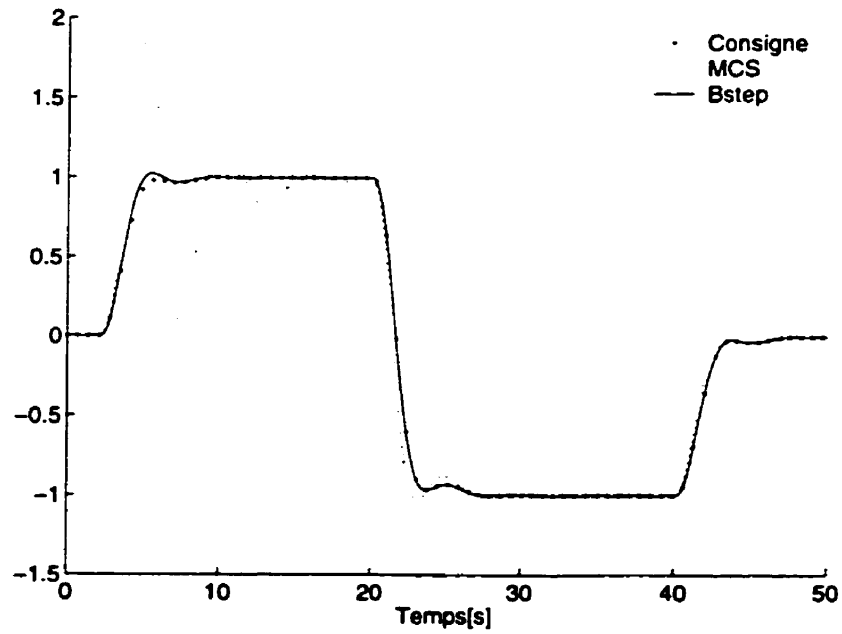
## 6.7 Conclusion

Conçu au départ pour éliminer les erreurs résiduelles, le contrôleur avec action intégrale proposé a permis une nette amélioration des performances des boucles adaptatives basées sur le **backstepping**. Qu'il s'agisse de poursuite/régulation ou d'identification, le contrôleur obtenu, contrairement au **backstepping** de base, garantit des erreurs nulles en régime permanent. Toutefois, dans le contexte non adaptatif, comme pour son équivalent sans intégrateur, le contrôleur obtenu présente, à cause de la présence de l'action dérivée et l'absence de filtrage, un comportement peu désirable de la variable manipulée. Afin d'éliminer cet inconvénient, ce contrôleur est augmenté d'un filtre semblable à celui présenté dans le chapitre 5.

Une autre approche, pour répondre aux problèmes des perturbations à moyenne non nulle, réside dans l'inclusion du **backstepping** dans une prédicteur de Smith. Cette solution offre, également, l'avantage d'étendre l'applicabilité du **backstepping** aux systèmes avec retards et/ou déphasages non minimaux, auxquels, il n'est pas applicable. Le détail de cette solution est donné dans le chapitre 7.



(a) Commande



(b) Sortie

FIG. 6.13: PID adaptatif : backstepping vs équivalence certaine

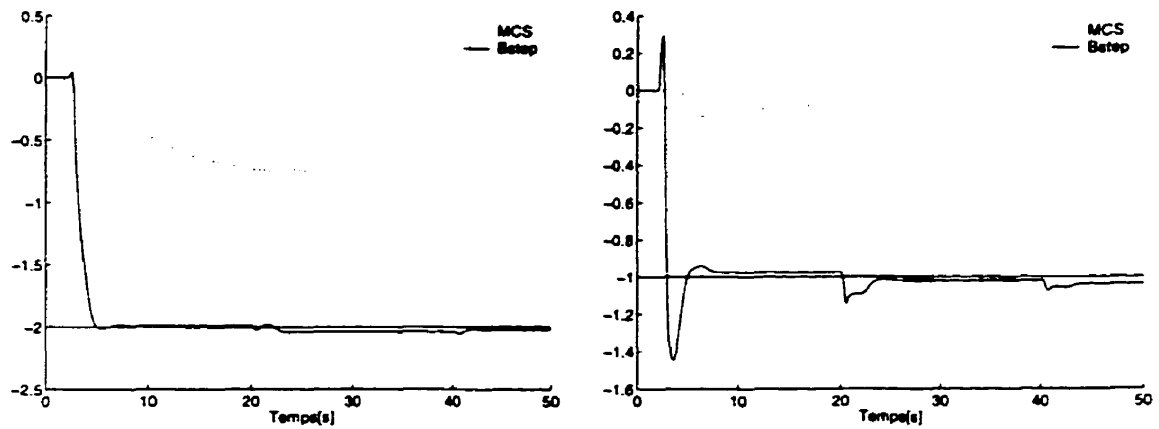
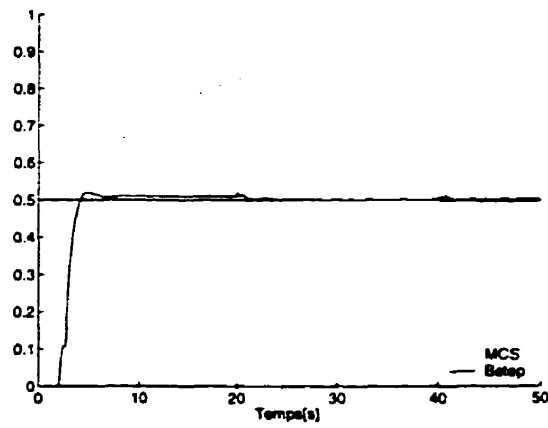
(a) Paramètre  $a$ (b) Paramètre  $b$ (c) Paramètre  $c$ 

FIG. 6.14: PID adaptatif : backstepping vs équivalence certaine



# Chapitre 7

## Déphasage non minimal/retard

Le backstepping étant une méthode de design par retour d'état continu, les retards ne peuvent y être représentés. Leur présence dans les procédés rend, par conséquent, cette méthode complètement inutilisable dans la pratique. Afin de surmonter cette limitation, une solution intéressante est donnée par le prédicteur de Smith (1957) (figure 7.1).

### 7.1 Principe

En effet, en présence de retards, la structure de Smith s'impose comme la solution par excellence, d'où l'idée d'insérer dans cette dernière les contrôleurs obtenus par le backstepping. Étant donné que la procédure récursive de design reste inchangée, l'approche par Smith est applicable là où le backstepping l'est, sans qu'elle ait à souffrir de la limitation des retards. En effet, la procédure de design, jusqu'ici appliquée au procédé<sup>1</sup>, s'effectuera sur le modèle de celui-ci. Si ce dernier est identique au procédé, aucun changement dans la méthode de design n'est nécessaire. La différence réside, alors, uniquement dans la façon dont est réalisée la loi de commande. C'est pourquoi, seuls les résultats de l'application du contrôleur final sont donnés.

Par ailleurs, étant donné que le backstepping réalise une commande par modèle de référence (MRC), il n'est pas applicable aux systèmes à déphasage non minimal. Pour

---

<sup>1</sup>Voir les chapitres 5 et 6.

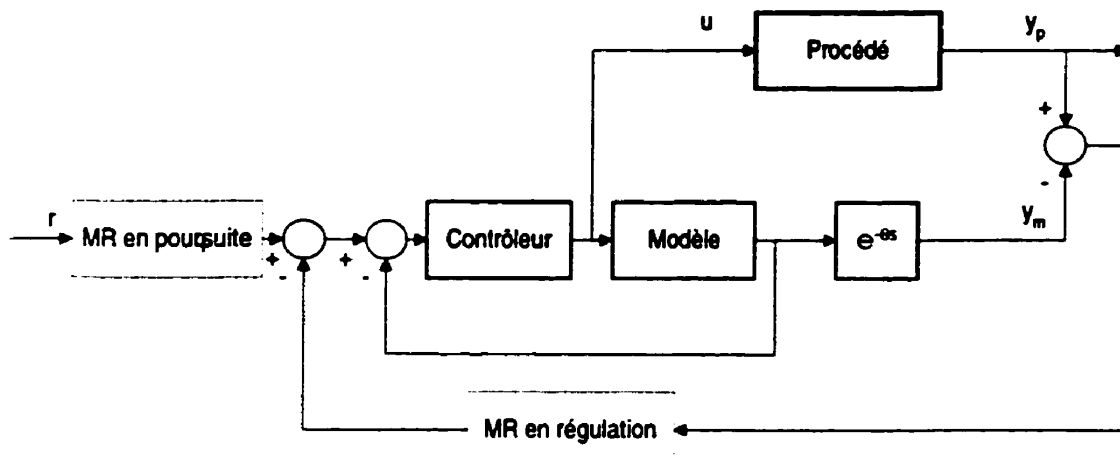


FIG. 7.1: Prédicteur de Smith

pouvoir suivre parfaitement le modèle, la commande MRC rend non observables tous les zéros du procédé. En toute évidence, si ce dernier présente un ou plusieurs zéros dans le demi-plan de droite, cette méthode générera un contrôleur instable. Afin d'éviter cette inversion systématique, le prédicteur de Smith est légèrement modifié. En effet, si ces zéros sont traités au même titre que les retards, i.e. sortis de la boucle interne, la structure de Smith modifiée (figure 7.2) devient une solution fort intéressante. À défaut de pouvoir s'en débarrasser, les zéros situés dans le demi-plan de droite sont imposés dans le modèle de référence. Dans de telles situations<sup>2</sup>, on ne dispose d'aucune marge de manœuvre. Quelle que soit la commande utilisée, la dynamique globale sera inévitablement affectée par la présence des zéros à phase non minimale.

## 7.2 Simulations

Afin d'illustrer l'efficacité de la structure proposée, le procédé dont la fonction de transfert est donnée par

$$G_p(s) = \frac{2(1 - 1.2s)e^{-s}}{(1 + s)(1 - .5s)} = \frac{-4(1 - 1.2s)e^{-s}}{s^2 - s - 2}$$

<sup>2</sup>Tout comme pour les retards.

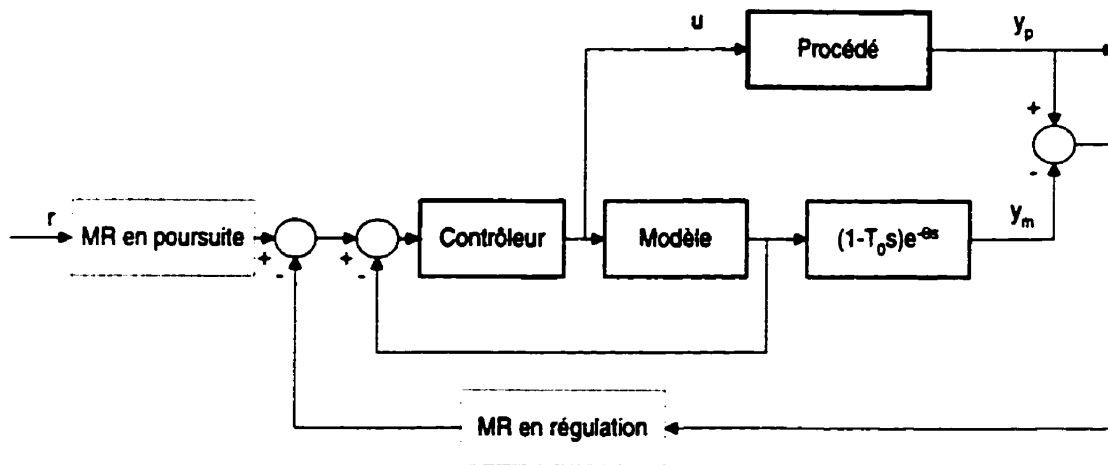


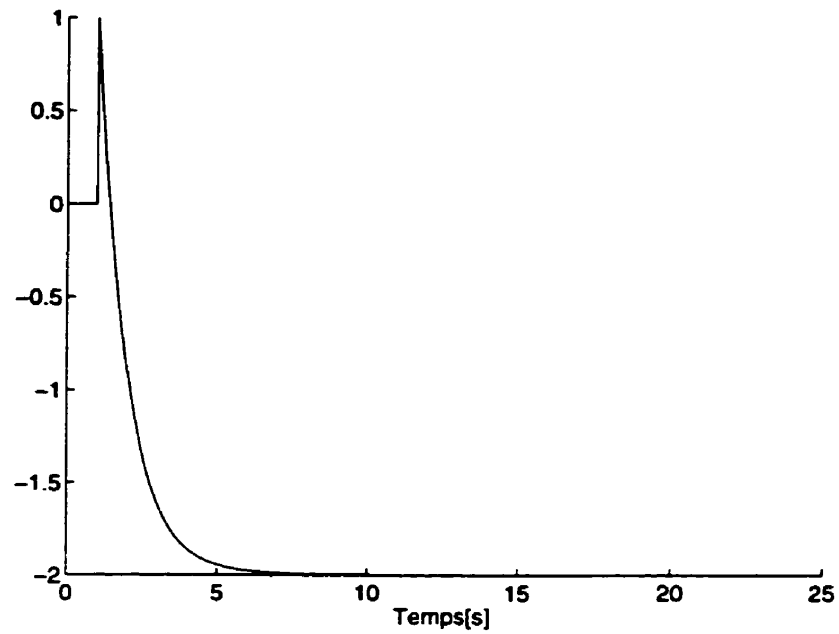
FIG. 7.2: Prédicteur de Smith modifié

est considéré. On y remarque la présence, à la fois, d'un zéro à phase non minimale et d'un retard. Le design par **backstepping** s'effectue, cependant, toujours sur le procédé, dépourvu de son zéro et son retard. Par la suite, ces derniers seront, systématiquement, pris en compte par le prédicteur de Smith.

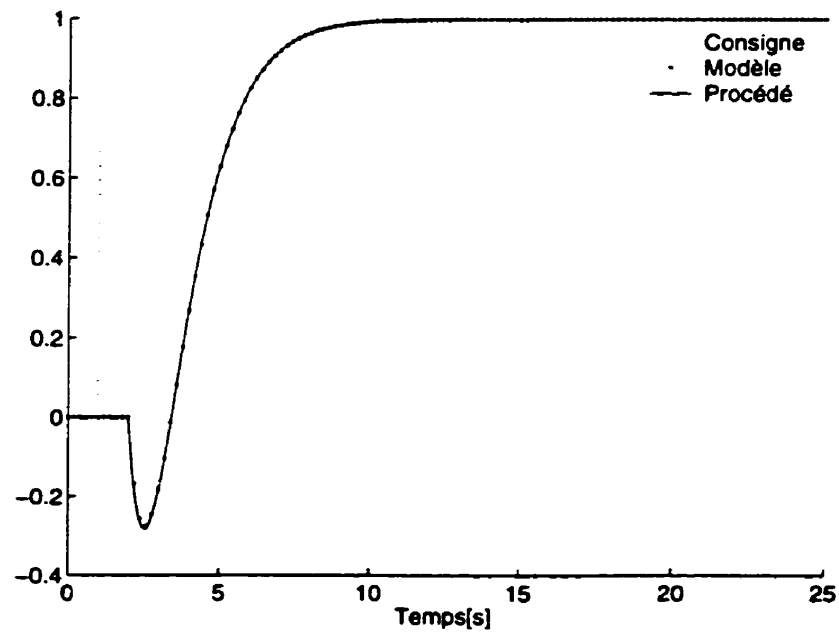
Les résultats de l'application de la combinaison **backstepping/Smith** modifié sont donnés par les figures 7.3 (a) & (b), où l'on applique un échelon de consigne à  $t = 1s$ . Ces résultats se passent de commentaires. On y remarque l'effet apparent sur la sortie du déphasage non minimal et du retard.

### **Perturbation à moyenne non nulle**

Le prédicteur de Smith ne se limite pas à résoudre le problème des retards et des déphasages non minimaux. En effet, la structure même de la boucle garantit des erreurs statiques nulles, en présence de perturbations à moyenne non nulle, et ce sans aucune modification dans la procédure de design. Cette propriété se constate clairement sur les figures 7.4 (a) & (b), où en plus de l'échelon de consigne à  $t = 1s$ , une perturbation est appliquée à la sortie du procédé à  $t = 9s$ . Outre l'erreur nulle en régime permanent, on remarque sur la figure 7.4 (a) un bon comportement de la variable manipulée,



(a) Commande



(b) Réponse

FIG. 7.3: Prédicteur de Smith : retard/phase non minimale

comparé à celui de la figure 6.1 (a), obtenu par **backstepping** avec action intégrale. Ce résultat a été obtenu grâce l'introduction d'un filtre à la sortie du comparateur de la boucle externe. Cette modification trouve sa justification dans la structure de Smith elle-même. L'action de boucle externe étant considérée comme une simple modification de la consigne de la boucle interne, l'utilisation d'un filtrage sur celle-ci n'a aucun effet sur la stabilité<sup>3</sup>. La stabilité globale dépend uniquement de la qualité du modèle utilisé et la capacité du contrôleur, généré par le **backstepping**, à stabiliser la boucle interne.

Cette propriété est également exploitée, afin de répondre aux problèmes que pose, en présence de bruit de mesure, la variable manipulée (chapitre 5). Les résultats des simulations donnés par les figures 7.5 (a) & (b) montrent, clairement, l'importante réduction de la variance qu'engendre l'introduction du filtre. Le choix de ce dernier étant indépendant du reste de la boucle, ses paramètres peuvent, contrairement aux filtres proposés dans les chapitres 5 et 6, être calculés en fonction de variance tolérable de la commande.

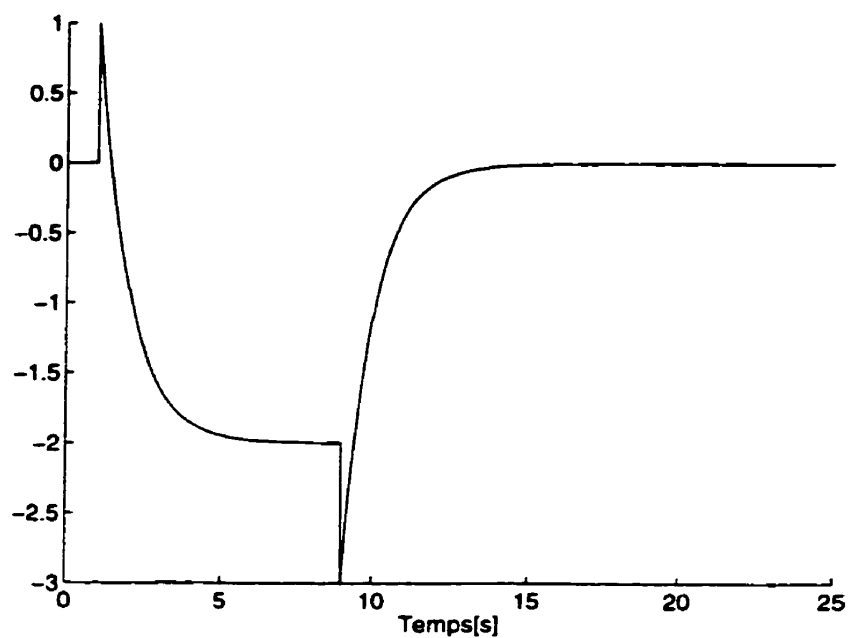
### 7.3 Limitations

Dans la solution qui vient d'être présentée, le **backstepping** est appliqué à un modèle, au lieu du procédé lui-même. Dans ce cas, l'utilisation de la version adaptative de la méthode (chapitre 4) consisterait à identifier les paramètres du modèle et non pas ceux du procédé. Étant donné que ces paramètres (du modèle) sont variables, puisqu'ils sont modifiés par le module d'adaptation, le **backstepping** adaptatif ne peut être utilisé. Ce dernier ne s'applique qu'aux systèmes à paramètres inconnus constants. L'utilisation du prédicteur de Smith<sup>4</sup> dans un contexte adaptatif passe donc, inévitablement, par l'équivalence certaine. Dans une telle approche, le régresseur n'étant pas imposé par la méthode design, son choix peut être effectué de manière à assurer les meilleures propriétés de convergence.

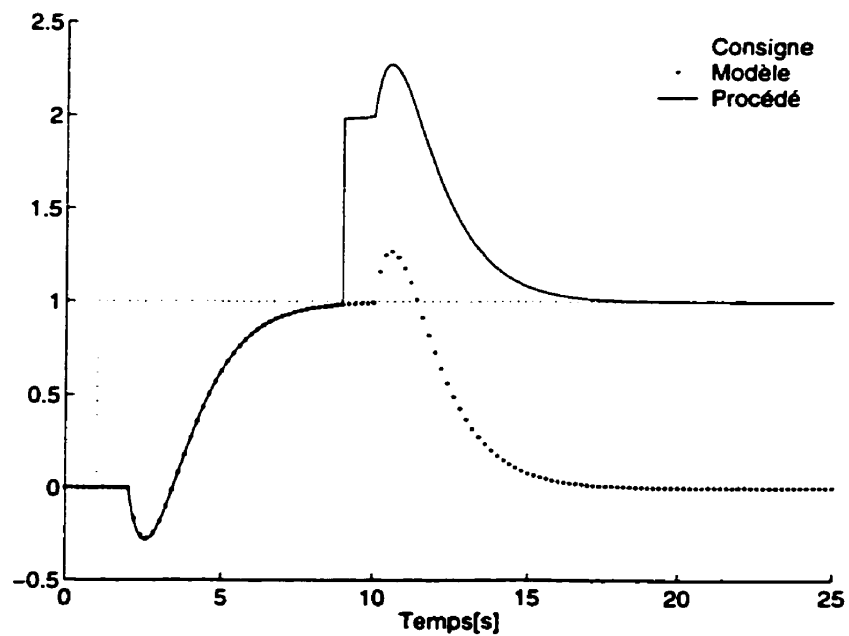
---

<sup>3</sup>C'est une propriété du prédicteur de Smith et de la structure à modèle interne (qui lui est similaire).

<sup>4</sup>Les mêmes commentaires s'appliquent au modèle interne.

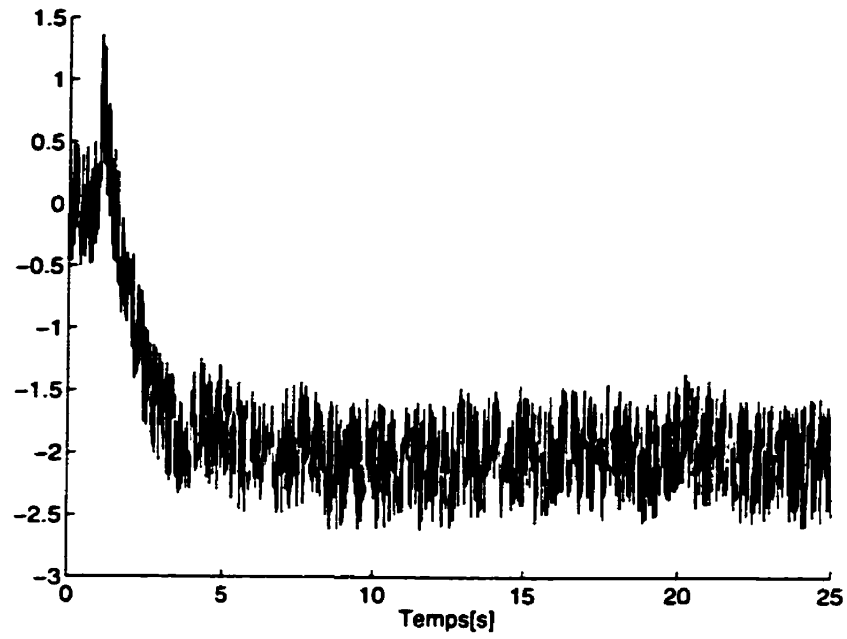


(a) Commande

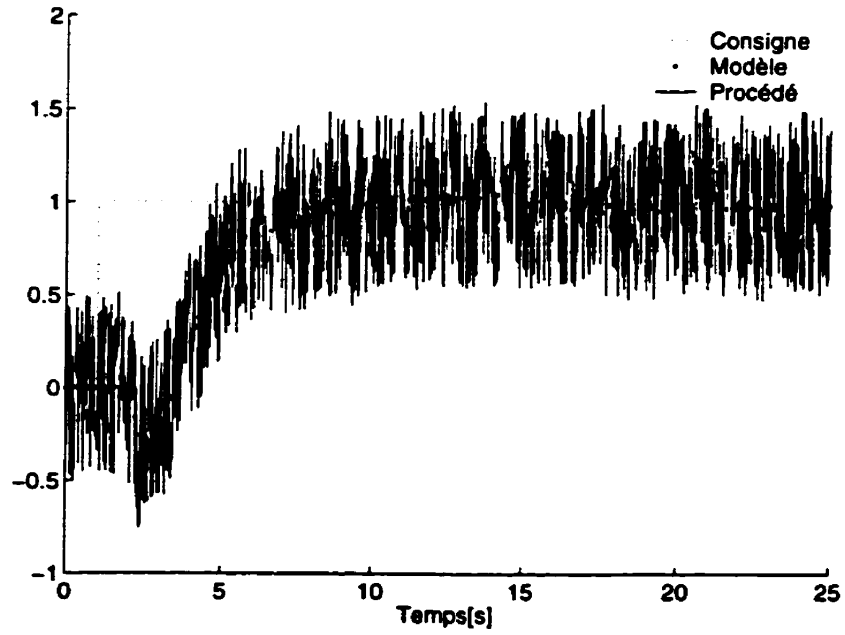


(b) Réponse

FIG. 7.4: Prédicteur de Smith : perturbation en échelon



(a) Commande



(b) Réponse

FIG. 7.5: Prédicteur de Smith : bruit blanc

## 7.4 Conclusion

Initialement proposée pour pallier les limitations du **backstepping** en présence de retards et de déphases non minimaux, la combinaison **backstepping/prédicteur de Smith** s'est avérée d'une grande efficacité pour les questions pratiques soulevées dans les chapitres 5 et 6. La structure même du prédicteur de Smith garantit des erreurs nulles en régime permanent, alors que l'ajout d'un filtre, à l'entrée de la boucle interne, assure un comportement acceptable de la variable manipulée. La solution proposée dans ce chapitre reste toutefois tributaire de la qualité du modèle utilisé. Cette propriété la rend désuète pour une commande par le **backstepping adaptatif**.



# Conclusion

Dans cette troisième partie de la thèse, les aspects théoriques, largement discutés dans la deuxième partie, ont cédé place aux problèmes que pose l'application du **backstepping** aux procédés réels. Les limitations y sont soulignées et des solutions proposées. Ainsi, l'introduction d'un nouveau filtre, dont le design est basé sur la théorie de Lyapunov, a permis d'améliorer le comportement de la variable manipulée, en présence de bruit. Les tests ont montré la supériorité du **backstepping** avec filtrage sur le **backstepping** simple, ainsi que sur le **backstepping** avec observateur.

Les performances en régime permanent ont également été améliorées via l'introduction d'action intégrale dans le **backstepping**. Cette modification garantit, en présence de perturbations, des erreurs résiduelles nulles. Dans le contexte adaptatif, la présence de l'action intégrale s'est également traduite, pour les paramètres inconnus, par de meilleures propriétés de convergence des estimés. Le dernier problème abordé est celui des procédés présentant des retards et/ou des déphasages non minimaux. Afin d'élargir l'applicabilité du **backstepping** à ces derniers, auxquels il n'est pas applicable, le contrôleur résultant est inséré dans une structure de **Smith**. Cette solution offre également avantage de répondre simultanément aux deux problèmes soulevés précédemment, i.e. erreurs résiduelles et comportement de la commande. Toutefois, cette méthode ne peut être généralisée au cas adaptatif.

Le couplage est un autre phénomène très courant dans les procédés réels. Vu son importance, ce problème n'a pas été adressé dans cette partie et sera abordé très en détail dans la quatrième partie, qui lui sera entièrement consacrée.

**Quatrième partie**  
**Application à la commande**  
**décentralisée**

# Introduction

Cette partie traite des problèmes, d'analyse et de conception, associés à la commande des systèmes multivariables. Une nouvelle méthode de design, pour la commande décentralisée des systèmes couplés, est présentée. Cette dernière se base sur la technique du **backstepping**, présentée dans les **chapitres 1 à 4**, et ses fonctions de Lyapunov. L'idée principale de la **méthode de l'erreur modifiée** consiste à modifier, par l'introduction de termes supplémentaires, les variables d'erreur régulées par la procédure du **backstepping**. Le choix adéquat de ces termes permet, dans une seconde étape, d'obtenir différentes lois de commande dont la réalisation est complètement décentralisée. Les conditions de stabilité et les performances en boucle fermée sont dictées par le choix effectué. La méthode permet d'obtenir des commandes par retour d'état, par retour de sortie (avec ou sans observateurs), ainsi que par retour d'état partiel, où seulement quelques variables sont nécessaires à l'implantation. La **méthode de l'erreur modifiée (MEM)** est applicable aussi bien aux systèmes linéaires que non linéaires.

Une revue critique des solutions, proposées dans la littérature pour répondre aux problèmes de la commande adaptative décentralisée, est d'abord présentée dans le chapitre 8. Dans ce dernier, des rappels théoriques sont également donnés, afin de faciliter la lecture des chapitres subséquents. Le **chapitre 9** présente la version originelle, i.e. par retour d'état complet, de la MEM. Des extensions aux cas adaptatifs et non linéaires y sont également proposées. Le problème de la mesure des variables d'état est abordé dans les **chapitres 10 et 11**. Le **chapitre 10** est consacré à la MEM avec observateur, alors que dans le **chapitre 11** sont élaborées les versions par retour d'état partiel et par retour de sortie. Dans ce chapitre l'introduction de l'action intégrale dans la MEM est également présentée.

# Chapitre 8

## Commande décentralisée

Avec la complexité croissante des systèmes commandés, les modèles scalaires sont, très vite, devenus incapables de fournir la précision suffisante garantissant le niveau élevé des performances, souvent exigé. Dans de telles situations, l'effet combiné des différents facteurs ne peut être confiné à une simple fonction de transfert, à une entrée et une sortie. Des modèles multivariables ( $m$  entrées —  $n$  sorties) sont alors à considérer. Ces modèles reflètent mieux la réalité des procédés physiques, où interagissent plusieurs phénomènes simultanément.

Théoriquement, l'approche multivariable représente la solution idéale au problème de la commande des systèmes couplés (Morari & Zafiriou 1989, Skogestad & Postlethwaite 1996). L'élimination des effets nuisibles, causés par les termes d'interconnexion<sup>1</sup>, assure les meilleures performances (possibles) en boucle fermée. La dimension des systèmes commandés reste, cependant, un handicap majeur. Le design et/ou la réalisation d'un contrôleur centralisé, pour des procédés couplés, peut s'avérer de taille, aussi bien en terme de complexité (du design) que de coût (de réalisation) (Datta & Ioannou 1992, Wen 1994). La commande des systèmes couplés (multivariables) s'inspire, souvent, directement des techniques scalaires. L'extension de ces dernières au cas multivariable, a été rendue possible par la généralisation des certains concepts de base, tels que les fonctions de transfert et l'introduction des normes. La représentation d'état a constitué

---

<sup>1</sup>Ce qui revient à réaliser un découplage parfait.

un pas unificateur entre la commande scalaire et la commande multivariable.

**Définition 8.1 (Centralité)** — Le contrôleur d'un procédé multivariable ( $m \times n$ ) est dit centralisé (ou multivariable) si au moins deux de ses  $m$  sorties (commandes) s'échangent de l'information. Dans le cas contraire, i.e. il n'y a aucun échange d'information, on parle de contrôleur décentralisé.

□

Malgré la supériorité des performances (en boucle fermée) fournies par les contrôleurs multivariables, la commande décentralisée a suscité plus d'intérêt ces dernières années. Cette (fausse) contradiction, trouve son explication dans des considérations d'ordre pratique. La fiabilité —notamment en termes de tolérance aux pannes—, la facilité d'implantation, ainsi que les faibles exigences en matière de calculs, sont quelques-unes parmi les nombreuses qualités qui ont contribué à la large popularité de la commande décentralisée dans le contrôle des procédés industriels. En plus, les problèmes d'évolution<sup>2</sup> sont plus facilement gérés dans un environnement décentralisé. Ce dernier offre l'avantage de garder le système fonctionnel, même si un ou plusieurs des contrôleurs locaux sont hors usage. Ainsi, certaines boucles peuvent être contrôlées manuellement (voire même, complètement, arrêtées), pendant que les autres continuent à être compensées automatiquement. Pour toutes ces raisons, seule l'approche décentralisée sera considérée par la suite.

**Remarque 8.1 (Centralité)** — Il est important de rappeler que la notion de centralité fait référence à la manière dont sont réalisés les contrôleurs et non pas à la façon dont ils sont conçus. Cette dernière peut être aussi bien centralisée que décentralisée.

□

Ce chapitre se veut, avant tout, une revue critique des principales méthodes proposées dans la littérature, pour répondre aux problèmes de la commande adaptative décentralisée des systèmes multivariables continus. Mais, afin de faciliter sa lecture,

<sup>2</sup>Les éventuels ajouts et/ou suppressions des sous-systèmes.

certaines notions théoriques seront rappelées, chaque fois que cela s'avère nécessaire. Les détails seront cependant volontairement omis, au profit d'une riche bibliographie qui, pour les lecteurs intéressés, couvre amplement la matière traitée.

## 8.1 Principe

La recherche sur la commande décentralisée a été motivée par les problèmes que rencontrent les méthodes de contrôle conventionnelles, de nature centralisée, face aux systèmes dits à grande échelle (**Large Scale Systems**). De par son principe, qui restreint l'échange d'information entre les différents capteurs et actionneurs, l'approche décentralisée apporte une solution intéressante aux difficultés que pose la commande de tels systèmes. Dans une commande décentralisée, l'unique contrôleur multivariable (centralisé) est remplacé par un ensemble de contrôleurs (locaux) scalaires, dont le design est (relativement) plus facile à réaliser. Chaque contrôleur local se base, dans le calcul de sa commande, sur l'information qui lui est fournie par son (ou ses) capteur(s); aucun échange d'information n'est permis entre les contrôleurs. Chacun de ces derniers possède ses propres critères de performances, indépendamment des autres.

Sans exception, toutes les méthodes proposées (Ioannou & Kokotovic 1985, Ioannou 1986, Hill et al. 1988, Gavel & Siljak 1989, Datta & Ioannou 1991, 1992, Wen 1994, Jain & Khorrami 1995, Wei et al. 1996, Jain & Khorrami 1997*a,b*, Wen & Soh 1997) abordent le problème de la commande décentralisée dans le cadre des systèmes à grande échelle. Le système est composé de  $\mathcal{N}$  sous-systèmes scalaires (essentiels), entre lesquels existent des interactions (beaucoup moins importantes). Les équations du  $i^{\text{ème}}$  sous-système sont données par

$$\mathcal{S}_i : \quad \begin{aligned} \dot{\mathbf{x}}_i &= \boldsymbol{\varphi}_i(\mathbf{x}_i, u_i) + \sum_{j \neq i}^{\mathcal{N}} \boldsymbol{\psi}_{ij}(\mathbf{x}_j, u_j) \\ y_i &= \boldsymbol{\eta}_i(\mathbf{x}_i), \quad i = 1, 2, \dots, \mathcal{N} \end{aligned}$$

où  $u_i$  et  $y_i$  représentent respectivement l'entrée et la sortie du sous-système  $\mathcal{S}_i$ . Le terme  $\boldsymbol{\psi}_{ij}$  modélise l'influence du sous-système  $\mathcal{S}_j$  sur le sous-système  $\mathcal{S}_i$ .

Toutes les solutions présentées dans la littérature considèrent les sous-systèmes  $S_i$  comme des entités indépendantes, dont les modèles nominaux (scalaires) sont affectés par des perturbations (les interconnexions). Dans cet esprit, la conception d'un contrôleur décentralisé est perçue comme l'agencement d'une batterie de contrôleurs scalaires, conçus séparément et modifiés au besoin pour garantir certaines performances. Par la suite, les performances du système global sont analysées et d'éventuelles modifications (robustifications) sont apportées aux contrôleurs afin de satisfaire les exigences globales, et surtout contrecarrer la présence des interactions. Dans ce contexte, le design est effectué suivant des critères de robustesse qui mènent à des conditions de "bon fonctionnement", souvent, très conservatrices. Les fonctions de couplage sont alors contraintes à être stables et faibles en module (Ioannou 1986). L'analyse des structures décentralisées, ainsi obtenues, découlent directement de celle de la commande robuste des systèmes scalaires présentant des incertitudes paramétriques (Ioannou 1986, Ioannou & Sun 1996).

**Remarque 8.2 (Décentralité du design)** — Cette approche, justifiable dans le cadre de la commande des systèmes à grande échelle, paraît inappropriée pour des systèmes de faible dimension, en l'occurrence les systèmes  $2 \times 2$  considérés ici (figure 8.1). On impose une décentralité, non indispensable, à la procédure de design, qui élimine systématiquement toutes les méthodes (de design) centralisées. Ces dernières peuvent, dans certains cas et sous certaines conditions (Benaskeur & Desbiens 1999a,b,c,d,e), fournir des lois de commande, dont la réalisation ne nécessite aucun échange d'information entre les régulateurs.

□

Dans un contexte non adaptatif, la mauvaise identifiabilité des systèmes couplés, causée principalement par leur complexité, explique, en partie, le manque de résultats constaté. C'est pourquoi, l'approche adaptative présente, pour ces systèmes, une solution particulièrement attrayante. Avec une telle méthode, les interactions, mal connues, sont automatiquement prises en compte.

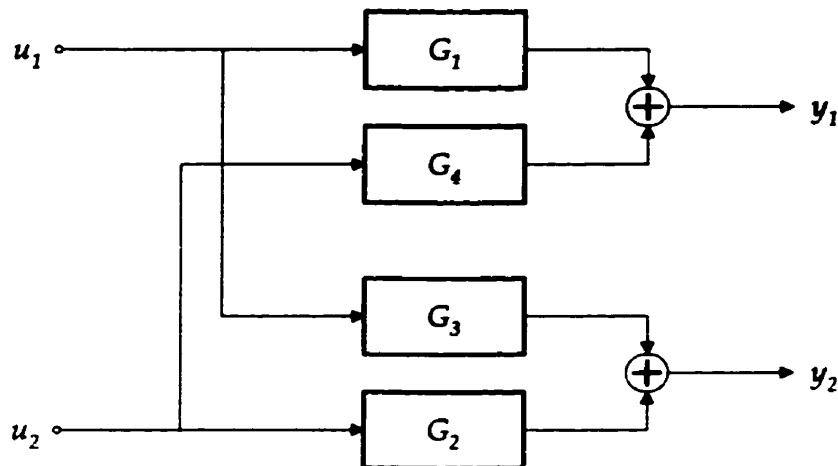


FIG. 8.1: Procédé à deux entrées et deux sorties

## 8.2 Commande adaptative décentralisée

Les premières solutions, proposées (Borisson 1979, Koivo 1980) pour répondre aux problèmes de la commande adaptative des systèmes couplés, furent des tentatives d'extensions quasi directes des méthodes monovariabiles. Depuis, plusieurs études ont montré que cette extension (directe) est loin d'être simple. Ceci s'explique surtout par les difficultés que pose la généralisation, au cas multivariable, de certains concepts de base de la commande monovariabiale.

Jusqu'à très récemment, les méthodes proposées pour résoudre le problème de la commande adaptative décentralisée des systèmes couplés, se basaient toutes (ou presque) sur le principe de l'équivalence certaine. Les approches développées dans cette direction s'inspiraient, principalement et directement, de la commande adaptative scalaire.

Il a été, cependant, montré que la présence des erreurs de modélisation (mauvaise structure et/ou incertitudes sur les paramètres), ou même de faibles perturbations, peut rendre instable la plupart des structures adaptatives "classiques" (Egardt 1979,



Ioannou & Kokotovic 1983, 1984). Étant donné que dans un environnement décentralisé, les effets des autres boucles sont perçus comme des perturbations, l'application directe des techniques adaptatives centralisées ne garantit, en aucun cas, le bon fonctionnement d'une structure décentralisée les utilisant. Dans de telles situations, certaines précautions sont à prendre et des techniques plus élaborées sont à considérer.

Trois approches se distinguent<sup>3</sup>, grosso modo, pour surmonter ce problème de la stabilité : la commande à gain fort, la commande adaptative robuste et la commande par backstepping. Les solutions proposées seront présentées, par ordre chronologique de leur apparition. Les deux premières méthodes seront présentées à titre indicatif et ne seront pas considérées par la suite. Seule l'approche basée sur le backstepping sera utilisée pour l'élaboration d'une méthode de design (chapitres 9 à 11).

### 8.3 Commande à gain fort

Les techniques, utilisant une commande à gain fort (high gain feedback control), issues de la commande robuste (non adaptative) des systèmes scalaires, ont été les premières à être appliquées aux problèmes de la commande décentralisée. Cette approche consiste à utiliser une commande, dont le gain est suffisamment élevé, afin de réduire celui de la fonction de sensibilité. Malheureusement, la forte sensibilité au bruit (à cause du gain élevé), la difficulté à prouver la stabilité, ainsi que la nécessité de connaître, comme dans le cas scalaire, une borne supérieure sur les perturbations et/ou les incertitudes du modèle, le degré relatif et le gain en hautes fréquences du procédé, constituent des handicaps majeurs à l'application de ces méthodes. Diverses lois d'adaptation ont été utilisées, pour le paramètre du gain, dans le seul but de s'affranchir de ces connaissances a priori ; mais sans grands succès.

Dans cette direction, la stabilité a été prouvée par Byrnes & Isidori (1984), mais uniquement pour les systèmes dont le degré relatif est égal à un. Khalil & Saberi (1987) ont réussi, à prouver la stabilité pour des systèmes de degré relatif plus élevé, en utilisant une adaptation par pallier (à des instants discrets) du paramètre du gain. La

---

<sup>3</sup>Dans le contexte algébrique continu.

solution rapportée par Wei et al. (1996), s'applique également aux systèmes de degré relatif arbitraire, mais avec de fortes restrictions sur les termes d'interconnexions, qui sont traités comme des perturbations.

Des lois de commande plus élaborées, souvent du style MRAC (Parks 1966, Butler 1992, Åström & Wittenmark 1995, Ioannou & Sun 1996), ont été proposées afin de pallier aux limitations des méthodes à gain fort. Dans cette approche le problème de la commande adaptative décentralisée est tout simplement réduit à celui d'une commande adaptative scalaire. Les interactions, souvent mal connues, y jouent le rôle des perturbations, qui affectent les modèles nominaux des sous-systèmes qui composent le procédé. Certaines propriétés de la commande adaptative limitent, cependant, son utilisation directe dans ce contexte.

En effet, telle qu'elle a été définie jusqu'ici, la commande adaptative (section 2.4) ne tient pas compte de certains aspects pratiques<sup>4</sup> de son application aux procédés. Les modèles de ces derniers sont considérés parfaits, i.e. pas d'incertitudes dans la structure, et le procédé n'est soumis à aucune perturbation externe. Dans la pratique, certaines précautions sont à prendre afin de s'assurer du bon fonctionnement de la boucle. Cela explique le nombre important de travaux qui ont été effectués, dans le seul but de formuler et analyser la commande adaptative des procédés, en présence d'incertitudes. Cela a donné naissance à une nouvelle discipline, communément appelée commande adaptative robuste (Ioannou & Tsakalis 1986b, Ioannou & Sun 1988, 1996).

Vu son importance pour la commande adaptative décentralisée, quelques aspects théoriques de la commande adaptative robuste sont donnés dans l'annexe A.

## 8.4 Commande adaptative robuste

Une loi de commande adaptative est dite robuste, si elle est dotée d'une loi de mise à jour des paramètres qui, même en présence d'incertitudes, permet de conserver la stabilité de l'ensemble. Ces incertitudes peuvent être dues aussi bien aux perturbations

---

<sup>4</sup>Sauf dans le cas l'utilisation d'amortissements non linéaires.

externes qu'aux dynamiques ignorées du procédé. De ce fait, les algorithmes d'identification classiques, du type gradient, Lyapunov et moindres carrés, largement utilisés en commande adaptative, doivent subir certaines modifications, afin d'améliorer leur robustesse et être en mesure de contrecarrer la présence des incertitudes.

### **Application à la commande décentralisée**

La robustesse apportée par les modifications<sup>5</sup> rend la commande adaptative robuste extrêmement intéressante pour la commande des systèmes couplés, où la stabilité souffrait de l'omniprésence des incertitudes (dynamiques ignorées, couplages mal connus, perturbations externes). Toutefois, l'application de ces méthodes aux systèmes adaptatifs distribués s'est trouvée confrontée aux mêmes restrictions que dans le cas scalaire. Ces dernières portent principalement sur le degré relatif des différents sous-systèmes et/ou la structure des interactions, qui constituent la principale source d'incertitudes pour les systèmes à grande échelle. Dans certains cas, c'est la centralité qui a été sacrifiée, afin de garantir la stabilité.

La commande décentralisée par MRAC constitue l'essentiel des solutions proposées dans ce contexte. La différence réside, souvent, uniquement dans le type d'adaptation utilisée. L'idée consiste à contrôler chaque sous-système par un régulateur MRAC local, utilisant une loi d'adaptation robuste. Ce problème a été adressé par Ioannou et al., qui considèrent le cas des systèmes connectés, en présence de perturbations bornées et de non-linéarités. Les sous-systèmes commandés sont de degré relatif inférieur ou égal à 2. Dans (Ioannou & Kokotovic 1985), toujours par une approche MRAC, ils utilisent un terme  $\sigma$  fixe, comme robustification. Ce dernier est remplacé dans (Ioannou 1986) par le "switching- $\sigma$ ", dont on connaît la supériorité (Voir l'annexe A). Avec cette dernière robustification, les ensembles résiduels pour les erreurs de poursuite et d'identification dépendent directement de l'amplitude des perturbations locales et de l'intensité des interactions. Par conséquent, si ces dernières sont nulles, les ensembles résiduels le seront également. Ce résultat est obtenu moyennant une certaine borne supérieure  $M_0$ , imposée au vecteur des paramètres désirés et une borne sur les interactions entre les

---

<sup>5</sup>Voir l'annexe A.

sous-systèmes, qui sont exigées par la robustification. La limitation du degré relatif admissible est essentiellement due à la loi d'adaptation, du type Lyapunov, utilisée.

Cette même limitation se retrouve dans (Gavel & Siljak 1989). Toujours avec une structure MRAC, une solution qui garantit la stabilité globale, est proposée. Certaines conditions structurelles sont, toutefois, imposées aux fonctions d'interaction (non modélisées). Ce résultat est obtenu sous l'hypothèse que les interactions et la commande locale se trouvent dans la même équation (*matching condition*). Dans le cas d'une commande par retour de sorties<sup>6</sup>, les régulateurs proposés s'appliquent uniquement aux systèmes de degré relatif  $\rho_i$  égal à un. L'idée consiste à assurer la stabilité en utilisant un gain fort pour contrecarrer l'effet des interactions et forcer les sous-systèmes locaux à suivre leurs modèles de référence respectifs. L'inconvénient avec cette méthode, outre la sensibilité au bruit, réside dans l'hypothèse que les incertitudes se situent uniquement au niveau des interconnexions. Les sous-systèmes locaux sont supposés connus avec exactitude. L'efficacité de cette méthode dépend, donc, de la validité de cette hypothèse.

La généralisation de la méthode du terme  $\sigma$  aux systèmes de degré relatif supérieur à 2 nécessite une normalisation. Celle-ci permet, dans un contexte décentralisé, de s'assurer que les interactions non modélisées affectent la loi d'adaptation comme des perturbations bornées. Malheureusement, une telle normalisation n'est possible que s'il y a un échange d'information entre les sous-systèmes. Cette solution a été adoptée par Datta & Ioannou (1991, 1992), avec le *switching- $\sigma$*  comme modification, pour contrôler des procédés de degré relatif arbitraire (mais connu). Chaque loi d'adaptation locale utilise un signal de normalisation  $\mu_i$ , généré par la commande locale et les sorties de tous les sous-systèmes.

$$\dot{\mu}_i = -\delta_{0i}\mu_i + u_i^2 + \frac{1}{\mathcal{N}} \sum_{j=1}^{\mathcal{N}} y_j^2 + q_i^2$$

avec  $q_i > 0$  et  $\mu_i(0) = q_i^2/\delta_{0i}$ . La loi d'adaptation utilisée est du type gradient robustifié

$$\dot{\hat{\theta}} = \frac{\gamma \hat{y} u}{\mu_i} - \gamma \sigma_s \hat{\theta}$$

---

<sup>6</sup>Nécessite l'estimation des états.

Reste à noter que la structure obtenue n'est que partiellement décentralisée. Une version modifiée, et qui utilise une projection des paramètres comme robustification supplémentaire, est proposée dans (Datta 1993). L'idée consiste à utiliser une possible connaissance des gains en hautes fréquences des sous-systèmes pour l'amélioration du régime transitoire. Cette commande, comme la précédente, n'est que partiellement décentralisée. On retrouve cette perte de décentralité également dans (Hill et al. 1988), où cette fois, la robustesse a été obtenue grâce l'introduction d'une zone morte dans les estimateurs locaux. Comme la normalisation, l'implémentation de la zone morte requiert un échange d'information entre les contrôleurs.

### **Limitations**

L'origine des limitations de cette approche n'est pas la robustification, mais plutôt le principe de l'équivalence certaine, sur lequel se base la méthode MRAC. Bien que la robustification apporte une amélioration substantielle, les critiques, portées précédemment (chapitre 2) à toutes les méthodes qui se basent sur l'équivalence certaine, restent valables. Quand MRAC est combiné à des lois de commande du type Lyapunov, le problème du degré relatif vient restreindre l'applicabilité des méthodes. C'est pourquoi, récemment, les recherches se sont orientées vers l'utilisation du backstepping et ses différentes variantes (Kanellakopoulos et al. 1991, Krstić et al. 1992, Kanellakopoulos et al. 1993, Krstić et al. 1994, Krstić & Kokotović 1994, Krstić et al. 1995, Krstić & Kokotović 1995, 1996), dans le but de surmonter ces limitations. Cette approche permet, tout en évitant les inconvénients de l'équivalence certaine, de proposer des méthodes pour des systèmes de degré relatif arbitraire.

## **8.5 Backstepping**

La commande adaptative des systèmes couplés, se basant sur la technique du backstepping, commence à susciter beaucoup d'intérêt ces dernières années. Les recherches dans cette direction ont été motivées par les performances obtenues par cette méthode, dans le domaine de la commande des systèmes monovariabiles. Certains résultats commencent déjà à démontrer la supériorité de cette approche dans de nombreux cas, que ce soit en commande multivariable (Ling & Tao 1997) ou décentralisée (Wen 1994, Jain

& Khorrami 1995, 1997*a,b*, Wen & Soh 1997).

Jain & Khorrami (1995) et Wen (1994) utilisent le **backstepping**, sans majeures modifications, pour la conception d'un régulateur adaptatif direct décentralisé, pour des systèmes de degré relatif arbitraire. Ces deux méthodes se contentent, dans un environnement décentralisé, de la robustesse propre au **backstepping** pour établir des conditions suffisantes de stabilité. Il faut toutefois rappeler que leurs résultats ont été obtenus au prix d'une sévère contrainte sur les interactions qui existent entre les différents sous-systèmes. En présence de perturbations paramétriques, les deux solutions garantissent des erreurs de poursuite nulles. Cependant, dans les deux cas, l'effet des perturbations externes n'est pas pris en compte dans le design. Afin d'alléger les contraintes imposées aux sous-systèmes, sans perdre la stabilité, Wen & Soh (1997) combinent le **backstepping** avec une loi d'adaptation utilisant la projection des paramètres et une normalisation locale. Cela a pour avantage d'éviter le phénomène de la dérive des paramètres (**parameter drift**), tout en profitant des qualités de stabilité que garantit la commande par Lyapunov. Le problème majeur avec cette méthode réside dans l'incompatibilité de la nature récursive du **backstepping** avec la projection et la non bornétude des effets des dynamiques ignorées, à cause de l'aspect local de la normalisation utilisée. Le problème de la projection n'a été soulevé et résolu que beaucoup plus tard par Ikhoulane & Krstic (1998*a*), dans le cas monovariable, alors que la normalisation pourrait être remplacée par un amortissement non linéaire, dont le rôle est très similaire.

Dans toutes les solutions rapportées ici, les fonctions de couplage sont traitées comme des perturbations qui affectent les modèles nominaux des sous-systèmes. Les modèles considérés y sont tous linéaires. Seules les interactions peuvent être non linéaires. Ceci est sans intérêt, étant donné qu'elles sont considérées comme des perturbations, et utilisées sans modèle précis. Dans (Wen 1994), comme dans beaucoup d'autres solutions moins intéressantes, ces non-linéarités doivent être limitées par une droite.

La commande adaptative décentralisée des systèmes non linéaires par le **backstepping** a été adressée pour la première fois par Jain & Khorrami (1997*a,b*). Dans (Jain & Khorrami 1997*a*) c'est une commande décentralisée par retour d'état qui est

présentée, alors qu'une variante à retour de sortie est proposée dans (Jain & Khorrami 1997b). L'élément clé dans ces deux solutions réside dans l'utilisation d'un nouveau type d'amortissements non linéaires, afin de contrer la présence des interactions. Ces dernières, bien qu'encore une fois traitées comme des perturbations, sont étendues au cas où elles sont d'amplitude inconnue et bornée par un polynôme d'ordre  $p$ . Elles doivent, cependant, être fonction uniquement des sorties des sous-systèmes, ce qui ne reflète guère la réalité. Dans la pratique, le couplage par les entrées est le plus souvent rencontré. Ceci s'explique par la facilité "relative" d'identification et d'analyse d'une telle configuration. Les paramètres inconnus doivent tous se trouver au niveau des interactions, ce qui suppose que les modèles des sous-systèmes sont connus.

Dans les solutions de Jain & Khorrami (1997a,b), le seul but de l'adaptation est donc de compenser l'effet instabilisant des interactions inconnues. Ce qui ne peut avoir de l'intérêt que pour les systèmes à grande échelle. En plus, le design n'est pas effectué directement sur la représentation du système, mais plutôt sur une transformée de celle-ci. La transformation utilisée, qui n'est pas toujours facile à trouver, doit être décentralisée (diagonale par bloc) afin de préserver la décentralité de la représentation d'état du système. Les solutions proposées ont, au moins, le mérite d'adresser, pour la première fois, le problème des perturbations externes (non mesurables).

## 8.6 Conclusion

Qu'elle soit basée sur des méthodes de design conventionnelles ou de nouvelles méthodes, telles que le *backstepping*, la commande adaptative décentralisée n'est qu'une extension robuste des méthodes monovariables. La robustesse inhérente au contrôleur, quand il en a, ne suffit pas à contrecarrer la présence du couplage, qui agit comme une perturbation non bornée sur les lois d'adaptation. Ces dernières doivent, dans un environnement décentralisé, subir certaines robustifications, afin d'être en mesure de préserver l'intégrité de la boucle de contrôle. On parle, dans ce cas, de lois d'adaptation robustes.

Une nouvelle approche de design pour la commande décentralisée des systèmes

couplés, basée principalement sur la technique du **backstepping**, est présentée dans les chapitres 9 à 11. La **méthode de l'erreur modifiée** s'applique aussi bien aux systèmes linéaires que non linéaires, en commande à retour d'état ou à retour de sortie et, sous certaines conditions, dans le contexte adaptatif.



# Chapitre 9

## Méthode de l'erreur modifiée

La commande décentralisée présente, pour beaucoup de procédés et dans beaucoup de situations, une solution plus intéressante que la commande multivariable. Toutefois, toutes les méthodes, proposées jusqu'ici dans la littérature, abordent le problème de la commande décentralisée dans le cadre des systèmes à grande échelle. Dans un tel contexte, les interactions sont négligées et traitées comme des perturbations qui affectent un modèle nominal du procédé. Malheureusement, les interactions sont, souvent, d'une importance capitale. Leur dynamique impose celle du système global. Négliger le couplage peut poser d'énormes problèmes, notamment de stabilité.

La notion de décentralité fait référence à la façon dont est réalisé un contrôleur et non à la méthode utilisée pour sa conception. Pourquoi devrait-on, dans ce cas, se priver de la puissance d'une méthode centralisée, aussi efficace que le *backstepping*, si on est en mesure d'en obtenir des contrôleurs dont la réalisation soit complètement décentralisée ? Partant de ces constatations, nous nous sommes intéressés à la procédure récursive du *backstepping* (chapitres 3 et 4), afin d'y apporter certaines modifications. Ces dernières nous ont permis, sous certaines conditions, d'obtenir une réalisation décentralisée du contrôleur multivariable originel. L'avantage majeur avec cette approche réside dans la prise en compte des fonctions de couplage, comme telles dans le design, et non plus comme des perturbations. La méthode de l'erreur modifiée que nous présentons en détail dans ce chapitre et les chapitres 10 et 11, s'inscrit dans cet esprit et vient confirmer le

bien fondé de cette approche.

Ce chapitre présente la version de base de la méthode de l'erreur modifiée. L'approche récursive du **backstepping** est d'abord utilisée, dans la section 9.1, pour la conception d'un régulateur multivariable. Par la suite, dans la section 9.3, l'algorithme de base est modifié, tel que suggéré dans (Benaskeur & Desbiens 1999c), afin de rendre possible l'obtention d'une structure de commande décentralisée. Le procédé considéré est du type linéaire à deux entrées et deux sorties (figure 8.1). Une extension de la méthode aux cas non linéaires, ainsi qu'adaptatifs, est donnée dans la section 9.5.

La matrice de transfert du procédé linéaire à commander est donnée par

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} G_1 & G_4 \\ G_3 & G_2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (9.1)$$

Les fonctions  $G_j(s)$  sont définies<sup>1</sup> par

$$G_j(s) = \frac{B_j(s)}{A_j(s)} = \frac{s^m + b_{j_{m-1}}s^{m-1} + \dots + b_{j_0}}{s^n + a_{j_{n-1}}s^{n-1} + \dots + a_{j_0}}$$

Chacune de ces fonctions  $G_j(s)$  est réécrite sous la représentation d'état minimale contrôlable suivante

$$\dot{x}_{j1} = x_{j2} \quad (9.2)$$

$$\dot{x}_{j2} = x_{j3} \quad (9.3)$$

$$\vdots$$

$$\dot{x}_{ji} = x_{j_{i+1}} \quad (9.4)$$

$$\vdots$$

$$\dot{x}_{jn} = -a_{j_0}x_{j1} - a_{j_1}x_{j2} - \dots - a_{j_{n-1}}x_{jn} + v_j \quad (9.5)$$

Sous forme matricielle, ces équations s'écrivent

$$\dot{\mathbf{x}}_j = \mathbf{A}_j \mathbf{x}_j + \mathbf{B}_j v_j$$

<sup>1</sup>Pour simplifier les calculs, les quatre fonctions de transfert sont prises du même ordre. Le passage au cas général, i.e. ordres différents, est direct et ne nécessite aucun traitement supplémentaire.

avec

$$\mathbf{A}_j = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \ddots \\ 0 & 0 & \cdots & 0 & 1 \\ -a_{j_0} & -a_{j_1} & \cdots & -a_{j_{n-1}} & -a_{j_n} \end{bmatrix}, \quad \mathbf{B}_j = \begin{bmatrix} [0]_{(n-1) \times 1} \\ 1 \end{bmatrix}$$

et

$$\mathbf{x}_j = [x_{j_1} \quad x_{j_2} \quad \cdots \quad x_{j_n}]^T$$

Les commandes filtrées  $v_j$  sont définies par

$$v_j = \begin{cases} B_j(s)u_j & \text{si } j = 1, 2, \\ B_j(s)u_{j-2} & \text{si } j = 3, 4. \end{cases}$$

## 9.1 Design récursif

L'objectif principal du design est de faire suivre la sortie  $y_1$  à la trajectoire de référence définie par  $y_{r_1}$ , pour se rendre à la consigne  $r_1$ . La stabilité est un autre facteur important, dont il faut tenir compte tout au long de la procédure de design. Cette dernière comporte  $n$  étapes, dont seules les deux premières seront abordées avec détails. Le développement sera effectué sur le premier sous-système. Le passage au deuxième est direct et seuls les résultats finals seront donnés. L'équation de la première sortie est donnée par

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{v} \quad (9.6)$$

$$y_1 = \mathbf{C}^T \mathbf{x} \quad (9.7)$$

avec

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 & [0]_{n \times n} \\ [0]_{n \times n} & \mathbf{A}_4 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{B}_1 & [0]_{n \times 1} \\ [0]_{n \times 1} & \mathbf{B}_4 \end{bmatrix},$$

$$\mathbf{C} = \begin{bmatrix} [1 \quad [0]_{1 \times (n-1)}]^T & [0]_{n \times 1} \\ [0]_{n \times 1} & [1 \quad [0]_{1 \times (n-1)}]^T \end{bmatrix},$$

$$\mathbf{x} = [\mathbf{x}_1^T \quad \mathbf{x}_4^T]^T \quad \text{et} \quad \mathbf{v} = [v_1 \quad v_4]^T$$

**Étape 1** — Selon l'algorithme du **backstepping**, la première erreur se définit par

$$\varepsilon_{1_1} = y_1 - y_{r_1}$$

L'idée de l'approche de l'erreur modifiée consiste à remplacer cette dernière par

$$\begin{aligned} \varepsilon_{1_1} &= y_1 - y_{r_1} + \chi_1 \\ &= x_{1_1} + x_{4_1} - y_{r_1} + \chi_1 \end{aligned}$$

où la composante  $\chi_1$  reste à déterminer. Afin de garantir une erreur de poursuite  $(y_{r_1} - y_1)$  nulle quand  $\varepsilon_{1_1} \rightarrow 0$ , la condition

$$\lim_{t \rightarrow \infty} \chi_1 = 0 \quad (9.8)$$

est imposée au choix de  $\chi_1$ . La première fonction de Lyapunov est alors donnée par

$$\mathcal{V}_{1_1} = \frac{1}{2} \varepsilon_{1_1}^2$$

dont la dérivée vaut, le long de l'équation (9.2),

$$\begin{aligned} \dot{\mathcal{V}}_{1_1} &= \varepsilon_{1_1} \dot{\varepsilon}_{1_1} \\ &= \varepsilon_{1_1} \left[ \dot{x}_{1_1} + \dot{x}_{4_1} - \dot{y}_{r_1} + \dot{\chi}_1 \right] \\ &= \varepsilon_{1_1} \left[ x_{1_2} + x_{4_2} - \dot{y}_{r_1} + \dot{\chi}_1 \right] \end{aligned} \quad (9.9)$$

Afin de rendre cette dernière négative,  $x_{1_2} + x_{4_2}$  peut être choisi comme première commande virtuelle, et sa valeur désirée serait alors

$$\begin{aligned} (x_{1_2} + x_{4_2})_d &\triangleq \alpha_{1_1} \\ &= -k_1 \varepsilon_{1_1} + \dot{y}_{r_1} - \dot{\chi}_1 \end{aligned}$$

où  $k_1$  est un paramètre de design positif. Avec ce choix, la dérivée (9.9) devient

$$\dot{\mathcal{V}}_{1_1} = -k_1 \varepsilon_{1_1}^2 \leq 0$$

Ce qui garantit la stabilité du sous-système (9.2).

**Remarque 9.1 (Paramètres de design)** — Dans toute cette partie de la thèse, les paramètres de design sont pris égaux pour les deux sous-systèmes. Ce choix n'affecte en rien la procédure de design, et permet uniquement de simplifier les notations.

□

**Étape 2** — La partie (9.2)–(9.3) du sous-système est, à présent, considérée. On augmente la fonction de Lyapunov précédente d'un nouveau terme d'erreur, ce qui donne

$$\mathcal{V}_{12} = \frac{1}{2}\varepsilon_{11}^2 + \frac{1}{2}\varepsilon_{12}^2 \quad (9.10)$$

où la seconde l'erreur  $\varepsilon_{12}$  est définie par

$$\begin{aligned} \varepsilon_{12} &= x_{12} + x_{42} - \alpha_{11} \\ &= x_{12} + x_{42} + k_1\varepsilon_{11} - \dot{y}_{r1} + \dot{\chi}_1 \\ &= (s + k_1)\varepsilon_{11} \end{aligned}$$

La dérivée de la fonction (9.10), le long des trajectoires possibles du sous-système (9.2)–(9.3) considéré, est alors donnée par

$$\begin{aligned} \dot{\mathcal{V}}_{12} &= -k_1\varepsilon_{11}^2 + \varepsilon_{12} \left[ \varepsilon_{11} + \dot{\varepsilon}_{12} \right] \\ &= -k_1\varepsilon_{11}^2 + \varepsilon_{12} \left[ (1 - k_1^2)\varepsilon_{11} + k_1\varepsilon_{12} + x_{13} + x_{43} - \ddot{y}_{r1} + \ddot{\chi}_1 \right] \end{aligned} \quad (9.11)$$

Choisir  $x_{13} + x_{43}$  comme deuxième commande virtuelle et sélectionner sa valeur désirée qui rend (9.11) définie négative, donne

$$\begin{aligned} (x_{13} + x_{43})_d &\triangleq \alpha_{12} \\ &= (k_1^2 - 1)\varepsilon_{11} - (k_1 + k_2)\varepsilon_{12} + \ddot{y}_{r1} - \ddot{\chi}_1 \end{aligned}$$

où  $k_2 > 0$ . Ceci permet d'obtenir

$$\dot{\mathcal{V}}_{12} = -k_1\varepsilon_{11}^2 - k_2\varepsilon_{12}^2 \leq 0$$

dont la négativité assure la stabilité de (9.2)–(9.3).

**Étape  $i$**  — La  $i^{\text{ème}}$  variable d'erreur est définie par

$$\begin{aligned}\varepsilon_{1_i} &= x_{1_i} + x_{4_i} - \alpha_{1_{i-1}} \\ &= \left| s\mathbf{I}_{i-1} - \mathbf{K}_{i-1} \right| \varepsilon_{1_1} \\ &\triangleq \mathcal{F}_{i-1}(s)\varepsilon_{1_1}\end{aligned}$$

où les matrices de paramètres  $\mathbf{K}_j$  sont définies par

$$\mathbf{K}_j = \begin{bmatrix} -k_1 & 1 & 0 & \cdots & 0 \\ -1 & -k_2 & 1 & 0 & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & -1 & -k_{j-1} & 1 \\ 0 & 0 & \cdots & -1 & -k_j \end{bmatrix}$$

et

$$\begin{aligned}\left| s\mathbf{I}_j - \mathbf{K}_j \right| &= \mathcal{F}_j(s) \\ &\triangleq \sum_{k=0}^j \varrho_{j_k} s^k\end{aligned}$$

Les coefficients  $\varrho_{j_k}$ , qui déterminent la dynamique désirée en boucle fermée, sont calculés de la manière récursive suivante

$$\varrho_{j_k} = k_j \varrho_{(j-1)_k} + \varrho_{(j-2)_k} + \varrho_{(j-1)_{(k-1)}} \quad (9.12)$$

avec  $k = 0, \dots, j$ , et

$$\varrho_{x_x} = 1 \quad (9.13)$$

$$\varrho_{x_y} = 0 \quad \text{si } x < y, \quad y \leq 1 \quad \text{ou } x < 0 \quad (9.14)$$

La nouvelle fonction de Lyapunov est alors donnée par

$$\mathcal{V}_{1_i} = \frac{1}{2} \sum_{j=1}^i \varepsilon_{1_j}^2$$

et sa dérivée s'écrit

$$\begin{aligned}\dot{V}_{1_i} &= - \sum_{j=1}^{i-1} k_j \varepsilon_{1_j}^2 + \varepsilon_{1_i} \left[ \varepsilon_{1_{i-1}} + \dot{\varepsilon}_{1_i} \right] \\ &= - \sum_{j=1}^{i-1} k_j \varepsilon_{1_j}^2 + \varepsilon_{1_i} \left[ \varepsilon_{1_{i-1}} + x_{1_{i+1}} + x_{4_{i+1}} - \dot{\alpha}_{1_{i-1}} \right]\end{aligned}$$

Afin de la rendre négative,  $x_{1_{i+1}} + x_{4_{i+1}}$  est choisi comme  $i^{\text{ieme}}$  commande virtuelle. Sa valeur désirée est donnée par

$$\begin{aligned}(x_{1_{i+1}} + x_{4_{i+1}})_d &\triangleq \alpha_i \\ &= \dot{\alpha}_{1_{i-1}} - \varepsilon_{1_{i-1}} - k_i \varepsilon_{1_i}\end{aligned}$$

Avec un tel choix, la stabilité du sous-système (9.2)–(9.4) est garantie, puisque la dérivée se réduit à

$$\dot{V}_{1_i} = - \sum_{j=1}^i k_j \varepsilon_{1_j}^2 \leq 0$$

**Étape  $n$**  — La dernière erreur est donnée par

$$\varepsilon_{1_n} = x_{1_n} + x_{4_n} - \alpha_{1_{n-1}}$$

et la fonction de Lyapunov associée est

$$V_{1_n} = \frac{1}{2} \sum_{j=1}^n \varepsilon_{1_j}^2 \quad (9.15)$$

Sa dérivée le long des trajectoires possibles du sous-système (9.2)–(9.5) est

$$\begin{aligned}\dot{V}_{1_n} &= - \sum_{j=1}^{n-1} k_j \varepsilon_{1_j}^2 + \varepsilon_{1_n} \left[ \varepsilon_{1_{n-1}} + \dot{\varepsilon}_{1_n} \right] \\ &= - \sum_{j=1}^{n-1} k_j \varepsilon_{1_j}^2 + \varepsilon_{1_n} \left[ \varepsilon_{1_{n-1}} + \dot{x}_{1_n} + \dot{x}_{4_n} - \dot{\alpha}_{1_{n-1}} \right] \\ &= - \sum_{j=1}^{n-1} k_j \varepsilon_{1_j}^2 + \varepsilon_{1_n} \left[ v_1 + \left( \sum_{j=0}^{n-1} a_{1_j} s^j \right) x_{1_1} + \varepsilon_{1_{n-1}} + \dot{x}_{4_n} - \dot{\alpha}_{1_{n-1}} \right]\end{aligned}$$

À présent, un choix judicieux de la commande filtrée  $v_1$  permet de rendre la dérivée de (9.15) négative. Un choix évident est donné par

$$v_1 = \left( \sum_{j=0}^{n-1} a_{1,j} s^j \right) x_{1_1} - \dot{x}_{4_n} + \dot{\alpha}_{1_{n-1}} - \varepsilon_{1_{n-1}} - k_n \varepsilon_{1_n}$$

Avec une telle commande, la dérivée s'écrit

$$\dot{V}_{1_n} = - \sum_{j=1}^n k_j \varepsilon_{1_j}^2 \leq 0$$

assurant, ainsi, la stabilité du sous-système (9.2)–(9.5). Dans l'espace d'état originel (des  $x_i$ ), la commande  $v_1$  prend la forme

$$v_1 = \sum_{j=0}^{n-1} a_{1,j} s^j x_{1_1} - s^n x_{4_1} + \mathcal{F}(s)(y_{r_1} - \chi_1) + \left[ s^n - \mathcal{F}(s) \right] y_1 \quad (9.16)$$

où, pour alléger l'écriture, on note  $\mathcal{F}$  le dernier filtre  $\mathcal{F}_n$ . La vraie commande  $u_1$  est alors donnée par

$$\begin{aligned} u_1 &= \frac{v_1}{B_1(s)} \\ &= \frac{1}{B_1} \left[ \sum_{j=0}^{n-1} a_{1,j} s^j x_{1_1} - s^n x_{4_1} + \mathcal{F}(s)(y_{r_1} - \chi_1) + s^n y_1 - \mathcal{F}(s)y_1 \right] \end{aligned} \quad (9.17)$$

Et de la même façon,

$$\begin{aligned} u_2 &= \frac{v_2}{B_2(s)} \\ &= \frac{1}{B_2} \left[ \sum_{j=0}^{n-1} a_{2,j} s^j x_{2_1} - s^n x_{3_1} + \mathcal{F}(s)(y_{r_2} - \chi_2) + s^n y_2 - \mathcal{F}(s)y_2 \right] \end{aligned} \quad (9.18)$$

Différentes structures de commande sont rendues possibles, par des choix adéquats des termes  $\chi_1$  et  $\chi_2$ . Ces derniers permettent, sans changements majeurs, de passer d'une structure multivariable à une structure décentralisée, et d'une commande par retour d'état à une commande par retour de sortie.



## 9.2 Commande multivariable

Le cas multivariable correspond à la version non modifiée du **backstepping**, où l'erreur régulée est l'erreur de poursuite elle-même. Ceci revient à choisir  $\chi_1$  et  $\chi_2$  tels que

$$\chi_1 = \chi_2 = 0$$

Avec un tel choix, l'expression de la commande  $v_1$  se réduit à

$$v_1 = \sum_{j=0}^{n-1} a_{1j} s^j x_{11} - s^n x_{41} + \mathcal{F}(s) y_{r1} + s^n y_1 - \mathcal{F}(s) y_1$$

que l'on peut écrire, en introduisant l'équation de  $G_4(s)$ ,

$$v_1 + v_4 = \left[ s^n + \sum_{j=0}^{n-1} a_{1j} s^j \right] x_{11} + \left[ s^n + \sum_{j=0}^{n-1} a_{4j} s^j \right] x_{41} + \mathcal{F}(s) (y_{r1} - y_1)$$

et de la même façon, on obtient pour le deuxième sous-système,

$$v_2 + v_3 = \left[ s^n + \sum_{j=0}^{n-1} a_{2j} s^j \right] x_{21} + \left[ s^n + \sum_{j=0}^{n-1} a_{3j} s^j \right] x_{31} + \mathcal{F}(s) (y_{r2} - y_2)$$

Ce qui donne pour les vraies commandes  $u_1$  et  $u_2$

$$B_1(s)u_1 + B_4(s)u_2 = A_1(s)x_{11} + A_4(s)x_{41} + \mathcal{F}(s)(y_{r1} - y_1)$$

$$B_3(s)u_1 + B_2(s)u_2 = A_2(s)x_{21} + A_3(s)x_{31} + \mathcal{F}(s)(y_{r2} - y_2)$$

Ces équations montrent clairement l'aspect multivariable du contrôleur obtenu.

### Conditions de stabilité

La fonction de Lyapunov (9.15) assure la convergence vers zéro de toutes les variables d'erreur  $\varepsilon_{1j}$ . Sous l'action du contrôleur multivariable obtenu, cette propriété garantit la stabilité entrée/sortie du système en boucle fermée, indépendamment de la structure du procédé. En effet, les équations du système en boucle fermée sont données par

$$\mathcal{F}(s)(y_{r1} - y_1) = 0 \quad (9.19)$$

$$\mathcal{F}(s)(y_{r2} - y_2) = 0 \quad (9.20)$$

où  $\mathcal{F}$  est un polynôme hurwitzien<sup>2</sup>. La fonction de Lyapunov, telle qu'elle est définie, ne donne cependant aucune indication quant à la stabilité interne (entrée/état). Afin d'analyser cette dernière, l'équation (9.19) est réécrite sous la forme

$$\begin{aligned}\mathcal{F}(s)x_{1_1} &= \mathcal{F}(s)y_{r_1} - \mathcal{F}(s)x_{4_1} \\ &= \mathcal{F}(s)y_{r_1} - \mathcal{F}(s)\frac{G_4(s)}{G_2(s)}x_{2_1}\end{aligned}$$

et, de la même façon,

$$\mathcal{F}(s)x_{2_1} = \mathcal{F}(s)y_{r_2} - \mathcal{F}(s)\frac{G_3(s)}{G_1(s)}x_{1_1}$$

Ces équations donnent, après la disparition de l'effet des conditions initiales,

$$\begin{cases} [1 - \mathcal{Q}_I(s)]x_{1_1} = y_{r_1} - \frac{G_4(s)}{G_2(s)}y_{r_2} \\ [1 - \mathcal{Q}_I(s)]x_{2_1} = y_{r_2} - \frac{G_3(s)}{G_1(s)}y_{r_1} \end{cases}$$

Avec

$$\mathcal{Q}_I(s) = \frac{G_3(s)G_4(s)}{G_1(s)G_2(s)}$$

**Remarque 9.2 (Quotient d'interaction)** — La fonction  $\mathcal{Q}_I(s)$  est connue sous le nom de quotient d'interaction (Rijnsdorp 1965). C'est une mesure quantitative du niveau d'interaction entre les différents sous-systèmes, à l'intérieur d'un procédé couplé.

□

Une condition suffisante, pour garantir la stabilité interne de la boucle, est donnée par le théorème du petit gain (Desoer & Vidyasagar 1974).

### **Théorème 9.1 (Petit gain)**

On considère le système donné par la figure 9.1. On suppose  $H_1, H_2 : \mathcal{L}_e \rightarrow \mathcal{L}_e$ ;  $z_1, z_2 \in \mathcal{L}_e$ . S'il existe des constantes  $c_1, c_2 \geq 0$ , et  $b_1, b_2$ , telles que

$$\begin{aligned}\|(H_1 z_1)_t\| &\leq c_1 \|z_{1t}\| + b_1 \\ \|(H_2 z_2)_t\| &\leq c_2 \|z_{2t}\| + b_2\end{aligned}$$

<sup>2</sup>Ses racines sont à partie réelle négative.

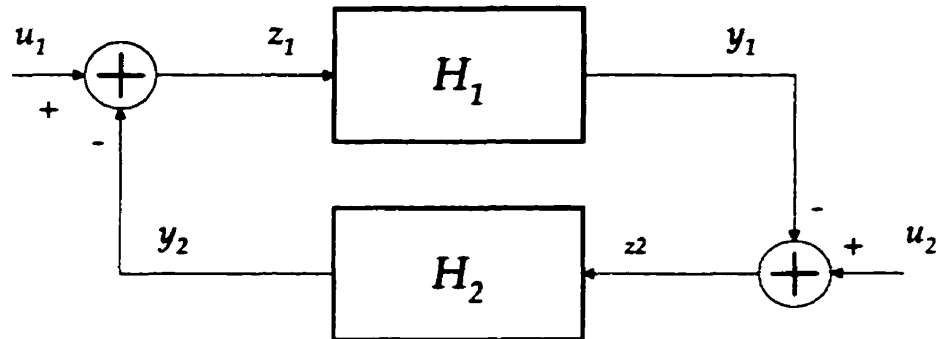


FIG. 9.1: Systèmes connectés

$\forall t \in \mathcal{R}^+$ , et si

$$c_1 c_2 < 1$$

alors

$$\|z_{1t}\| \leq \frac{\|u_{1t}\| + c_2 \|u_{2t}\| + b_2 + b_1 c_2}{1 - c_1 c_2}$$

$$\|z_{2t}\| \leq \frac{\|u_{2t}\| + c_1 \|u_{1t}\| + b_1 + b_2 c_1}{1 - c_1 c_2}$$

$\forall t \geq 0$ . Si, en plus,  $\|u_1\|, \|u_2\| < \infty$ , alors  $z_1, z_2, y_1$  et  $y_2$  ont tous une norme finie et

$$\|z_1\| \leq \frac{\|u_1\| + c_2 \|u_2\| + b_2 + b_1 c_2}{1 - c_1 c_2}$$

$$\|z_2\| \leq \frac{\|u_2\| + c_1 \|u_1\| + b_1 + b_2 c_1}{1 - c_1 c_2}$$

■

D'après ce théorème, sous l'action du contrôleur multivariable obtenu, tous les signaux à l'intérieur de la boucle sont bornés, si  $Q_I(s)$  est stable et vérifie la condition

$$\|Q_I(s)\|_\infty < 1$$

Ce résultat a été obtenu, en posant :  $z_1 = x_{11}$ ,  $z_2 = x_{21}$ ,  $u_1 = y_{r1}$ ,  $u_2 = y_{r2}$ ,  $H_1 = G_3(s)/G_1(s)$  et  $H_2 = G_4(s)/G_2(s)$ . Pour garantir la stabilité de  $Q_I(s)$ , deux conditions sont imposées aux équations en boucle ouverte du procédé.

**Condition 1** — Les fonctions de transfert  $G_1(s)$  et  $G_2(s)$  sont à phase minimale, i.e.  $B_1(s)$  and  $B_2(s)$  sont des polynômes de Hurwitz.

**Condition 2** —  $G_3(s)$  et  $G_4(s)$  sont des fonctions de transfert stables, avec le nombre d'intégrateurs dans  $G_3(s)$  inférieur ou égal à  $G_2(s)$ , et dans  $G_4(s)$  inférieur ou égal à  $G_1(s)$ .

### Performances

À cause des termes introduits dans la version modifiée, l'erreur régulée  $\varepsilon_{1_1}$  ne représente pas la vraie erreur de poursuite (ou de régulation si  $y_{r_1} = 0$ ). Elle lui est cependant égale au terme  $\chi_1$  près. En effet, l'erreur de poursuite réelle  $e_1$  est donnée par

$$\begin{aligned} e_1 &= y_1 - y_{r_1} \\ &= \varepsilon_{1_1} - \chi_1 \end{aligned}$$

La convergence vers zéro de  $\varepsilon_{1_1}$  est garantie par le **backstepping** et sa fonction de Lyapunov indépendamment de la valeur de  $\chi_1$ . En régime permanent, le comportement de  $e_1$  dépend alors uniquement de celui de  $\chi_1$

$$\lim_{t \rightarrow \infty} e_1 = \lim_{t \rightarrow \infty} \chi_1$$

Étant donné que, dans le cas multivariable,  $\chi_1$  est nul, l'erreur de poursuite l'est également.

**Remarque 9.3 (Dépendance)** — La présence du terme  $s^n x_{4_1}$ , qui invoque explicitement la commande  $u_2$ , dans l'expression (9.17) de  $u_1$ , rend les deux contrôleurs  $u_1$  et  $u_2$  interdépendants. Une dépendance similaire est engendrée par la présence du terme  $s^n x_{3_1}$  dans l'expression (9.18) de  $u_2$ . Dans ce qui suit, les termes  $\chi_1$  et  $\chi_2$  sont choisis non nuls, afin de briser cette dépendance.

□

### 9.3 Décentralisation

Un choix approprié du terme  $\chi_1$  permet d'éliminer le couplage entre les deux contrôleurs (9.17) et (9.18). En effet, un contrôleur décentralisé à retour d'état peut être obtenu, si  $v_1$  est réécrite, comme il est suggéré dans (Benaskeur & Desbiens 1999c), sous la forme

$$\begin{aligned} v_1 &= \mathcal{F}(s)y_{r_1} + \left[ s^n + \sum_{j=0}^{n-1} a_{1,j}s^j - \mathcal{F}(s) \right] x_{1_1} - \mathcal{F}(s)(\chi_1 + x_{4_1}) \\ &= \mathcal{F}(s)y_{r_1} - \varrho_0 y_1 + \varrho_0 y_1 + \left[ s^n + \sum_{j=0}^{n-1} a_{1,j}s^j - \mathcal{F}(s) \right] x_{1_1} - \mathcal{F}(s)(\chi_1 + x_{4_1}) \\ &= \mathcal{F}(s)y_{r_1} - \varrho_0 y_1 + \left[ s^n + \sum_{j=0}^{n-1} a_{1,j}s^j - \mathcal{F}(s) + \varrho_0 \right] x_{1_1} - \mathcal{F}(s) \left[ \chi_1 - (\mathcal{M} - 1)x_{4_1} \right] \end{aligned}$$

où le filtre à gain unitaire  $\mathcal{M}(s)$  est défini par

$$\begin{aligned} \mathcal{M}(s) &= \frac{\varrho_0}{\mathcal{F}(s)} \\ &= \frac{\varrho_0}{s^n + \varrho_{n-1}s^{n-1} + \dots + \varrho_0} \end{aligned}$$

Si  $\chi_1$  est choisi tel que

$$\begin{aligned} \chi_1 &= \left[ \mathcal{M}(s) - 1 \right] x_{4_1} \\ &= - \frac{s^n + \varrho_{n-1}s^{n-1} + \dots + \varrho_1 s}{s^n + \varrho_{n-1}s^{n-1} + \dots + \varrho_1 s + \varrho_0} x_{4_1} \end{aligned}$$

la commande  $u_1$  sera donnée par

$$\begin{aligned} u_1 &= \frac{v_1}{B_1(s)} \\ &= \frac{1}{B_1} \left[ \mathcal{F}(s)y_{r_1} - \varrho_0 y_1 + \left( s^n + \sum_{j=0}^{n-1} a_{1,j}s^j - \mathcal{F}(s) + \varrho_0 \right) x_{1_1} \right] \\ &= \frac{1}{B_1} \left[ \mathcal{F}(s)y_{r_1} - \varrho_0 y_1 + a_{1_0} x_{1_1} + \sum_{j=1}^{n-1} (a_{1_j} - \varrho_j) x_{1_{j+1}} \right] \end{aligned} \quad (9.21)$$

De la même façon, le design pour le deuxième sous-système donne

$$\begin{aligned}
 u_2 &= \frac{v_2}{B_2(s)} \\
 &= \frac{1}{B_2} \left[ \mathcal{F}(s)y_{r_2} - \varrho_0 y_2 + \left( s^n + \sum_{j=0}^{n-1} a_{2j} s^j - \mathcal{F}(s) + \varrho_0 \right) x_{2_1} \right] \\
 &= \frac{1}{B_2} \left[ \mathcal{F}(s)y_{r_2} - \varrho_0 y_2 + a_{2_0} x_{2_1} + \sum_{j=1}^{n-1} (a_{2j} - \varrho_j) x_{2_{j+1}} \right] \quad (9.22)
 \end{aligned}$$

Chacun de ces contrôleurs dépend uniquement de l'information qui lui est disponible localement, ce qui assure une décentralité complète.

### Conditions de stabilité

Si les contrôleurs (9.21) et (9.22) sont appliqués à l'équation du procédé, la dynamique, en boucle fermée du premier sous-système, sera donnée par

$$s^n x_{1_1} = \mathcal{F}(s)y_{r_1} - \varrho_0 y_1 + \left[ \varrho_0 - \mathcal{F}(s) + s^n \right] x_{1_1}$$

qui peut être réécrite sous la forme

$$0 = \mathcal{F}(s)y_{r_1} - \varrho_0 x_{1_1} - \mathcal{F}(s)x_{1_1}$$

Le même développement, appliqué au deuxième sous-système, donne

$$\begin{aligned}
 x_{1_1} &= y_{r_1} - \frac{G_4(s)}{G_2(s)} \mathcal{M}(s) x_{2_1} \\
 x_{2_1} &= y_{r_2} - \frac{G_3(s)}{G_1(s)} \mathcal{M}(s) x_{1_1}
 \end{aligned}$$

En utilisant le quotient d'interaction  $\mathcal{Q}_I$ , ces équations se mettent sous la forme

$$\begin{cases}
 \left[ 1 - \mathcal{G}(s) \right] x_{1_1} = y_{r_1} - \frac{G_4(s)}{G_2(s)} \mathcal{M}(s) y_{r_2} \\
 \left[ 1 - \mathcal{G}(s) \right] x_{2_1} = y_{r_2} - \frac{G_3(s)}{G_1(s)} \mathcal{M}(s) y_{r_1}
 \end{cases}$$

où

$$\mathcal{G}(s) = \mathcal{M}^2(s) \mathcal{Q}_I(s)$$

À partir du théorème du petit gain<sup>3</sup>, tous les signaux à l'intérieur de la boucle sont bornés, si  $\mathcal{G}(s)$  est stable et vérifie

$$\|\mathcal{G}(s)\|_{\infty} < 1 \quad (9.23)$$

Étant donné que  $\mathcal{M}(s)$  est stable et de gain unitaire, si en plus il est choisi sans résonance, on aura

$$\|\mathcal{M}(s)\|_{\infty} = 1$$

À partir des conditions 1 et 2, le quotient  $\mathcal{Q}_I(s)$  est également stable. La condition (9.23) se réduit, dans ce cas, à

$$\|\mathcal{Q}_I(s)\|_{\infty} < 1 \quad (9.24)$$

Ainsi, si (9.24) est vérifiée, le procédé peut être stabilisé par les deux contrôleurs locaux, donnés par

$$\begin{aligned} u_1 &= \frac{v_1}{B_1(s)} \\ &= \frac{1}{B_1(s)} \left[ \mathcal{F}(s)y_{r1} - \varrho_0 y_1 + \left( a_{10} + \sum_{i=1}^{n-1} (a_{1i} - \varrho_i) s^i \right) x_{11} \right] \end{aligned} \quad (9.25)$$

et

$$\begin{aligned} u_2 &= \frac{v_2}{B_2(s)} \\ &= \frac{1}{B_2(s)} \left[ \mathcal{F}(s)y_{r2} - \varrho_0 y_2 + \left( a_{20} + \sum_{i=1}^{n-1} (a_{2i} - \varrho_i) s^i \right) x_{21} \right] \end{aligned} \quad (9.26)$$

### Performances

La convergence vers zéro de la variable d'erreur  $\varepsilon_1$ , étant garantie par la fonction de Lyapunov, la vraie erreur de poursuite pour le premier sous-système sera donnée, en régime permanent, par

$$\begin{aligned} e_1 &= y_1 - y_{r1} \\ &= (1 - \mathcal{M})x_{41} \\ &= -\chi_1 \end{aligned}$$

---

<sup>3</sup>Voir énoncé à la page 184.

Il est clair, à partir de la définition de  $\chi_1$ , que si  $x_{4_1}$  est borné<sup>4</sup>, alors

$$\begin{aligned}\lim_{t \rightarrow \infty} \chi_1 &= \lim_{t \rightarrow \infty} \left[ (\mathcal{M} - 1)x_{4_1} \right] \\ &= 0\end{aligned}$$

puisque  $\mathcal{M}(s)$  est de gain unitaire. Ainsi, si la condition de stabilité (9.24) est vérifiée, la contrainte (9.8) imposée au terme  $\chi_1$  sera respectée. Ceci donne pour l'erreur de poursuite

$$\begin{aligned}\lim_{t \rightarrow \infty} e_1 &= \lim_{t \rightarrow \infty} (y_1 - y_{r_1}) \\ &= 0\end{aligned}$$

## 9.4 Application

Afin d'illustrer l'efficacité de la loi de commande proposée, cette dernière est appliquée au contrôle d'un simulateur phénoménologique. Le procédé simulé représente une unité de flottation composée de cellules de  $8m^3$  en série (Makni 1996, Desbiens et al. 1998). La flottation est un procédé de séparation solide-solide, couramment utilisé dans l'industrie minière. Son objectif principal consiste à séparer le débit d'alimentation (en minerai brut) en débit de concentré (minerai de valeur) et débit de rejet (gangue). Au voisinage du point d'opération

$$\begin{aligned}(u_{a_0}, u_{c_0}) &= (38.0, 33.0) \\ (y_{c_0}, y_{t_0}) &= (27.51, 6.85)\end{aligned}$$

une représentation linéaire (Mailloux 1997) de la dynamique du procédé est donnée par

$$\begin{bmatrix} y_c - y_{c_0} \\ y_t - y_{t_0} \end{bmatrix} = \begin{bmatrix} \frac{-0.2149(1+.6706s)}{(1+1.136s)(1+.003921s)} & \frac{-0.1697(1-.9537s)}{(1+.8371s)^2} \\ \frac{-0.01692}{(1+1.246s)(1+.2644s)} & \frac{-0.05283}{(1+1.111s)^2} \end{bmatrix} \begin{bmatrix} u_a - u_{a_0} \\ u_c - u_{c_0} \end{bmatrix} \quad (9.27)$$

où les sorties sont la teneur du concentré  $y_c$  et la teneur du rejet  $y_t$ , alors que  $u_a$  et  $u_c$  sont les variables manipulées, qui représentent respectivement le débit d'air et le débit

<sup>4</sup>Ce qui a été prouvé dans la section précédente.



du collecteur. Les essais consistent à observer le comportement global du système, suite à des changements des consignes : teneur du concentré à  $t = 2 \text{ min}$  et teneur du rejet à  $t = 15 \text{ min}$ . Ce procédé servira, tout au long de cette partie de la thèse, pour l'illustration des lois de commande développées.

## Simulation

Les figures 9.2 (a) & (b) donnent les réponses du procédé aux deux changements de consigne. On constate un bon comportement aussi bien en poursuite qu'en régulation. Le comportement des deux variables manipulées sont données par les figures 9.3 (a) & (b), où en régulation, celles-ci présentent des variations brusques, à cause de la fonction dérivée présente dans les deux régulateurs.

## 9.5 Extensions

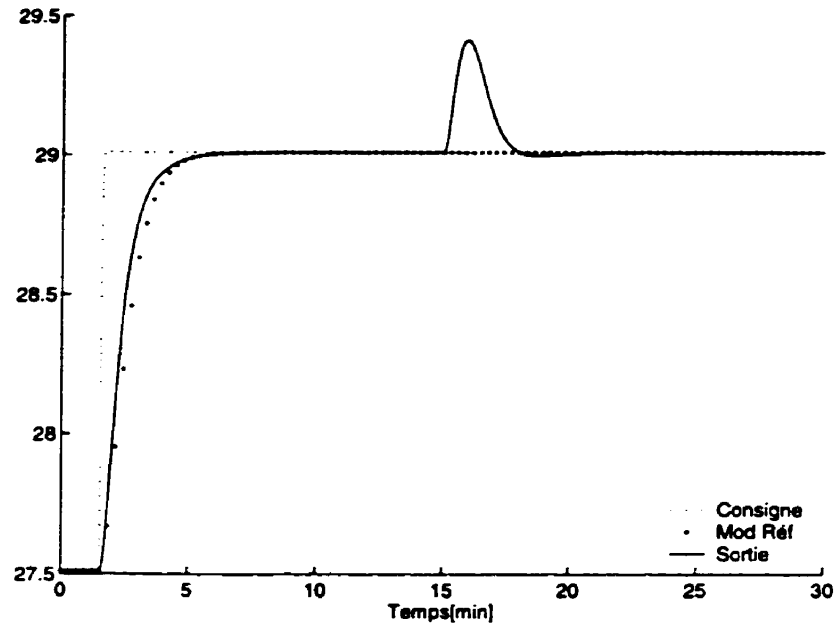
Sous certaines conditions, la méthode de l'erreur modifiée, développée jusqu'ici sur un système linéaire à paramètres connus, s'étend facilement aux systèmes non linéaires et/ou à paramètres inconnus. La version adaptative de la MEM est présentée dans la section 9.5.1, alors que son extension aux systèmes non linéaires est donnée dans la section 9.5.2. Les deux extensions proposées se basent sur le **backstepping** de base, ainsi que sa version adaptative (chapitre 4).

### 9.5.1 Approche adaptative

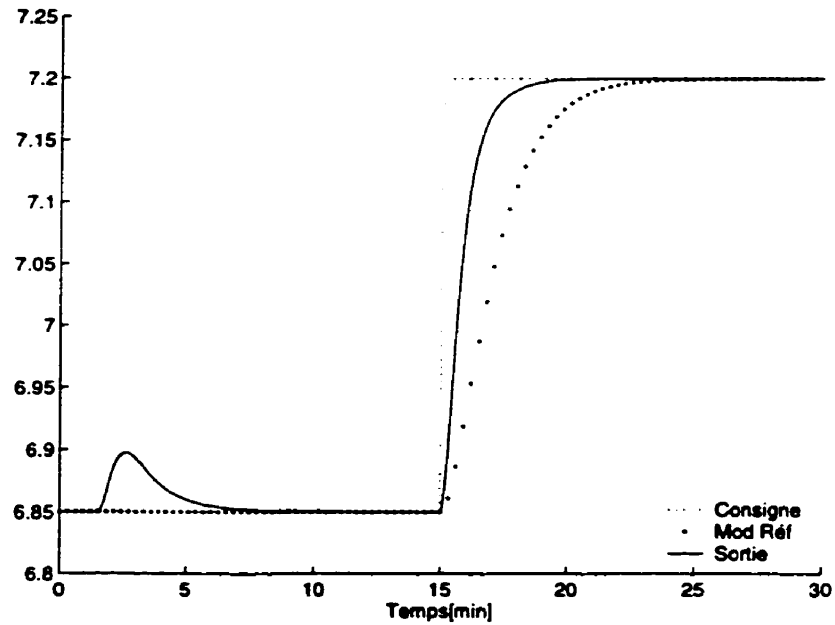
Si les paramètres  $a_{1i}$  du procédé ne sont pas connus, la commande adaptative consiste à les remplacer par leurs estimés<sup>5</sup>  $\hat{a}_{1i}$ , dans toutes les expressions des commandes obtenues dans les sections précédentes. La commande adaptative<sup>6</sup> est, dans ce

<sup>5</sup>Le même développement s'applique aux paramètres  $a_{2i}$ .

<sup>6</sup>Où, pour simplifier le raisonnement, on suppose les polynômes  $B_i(s)$  connus

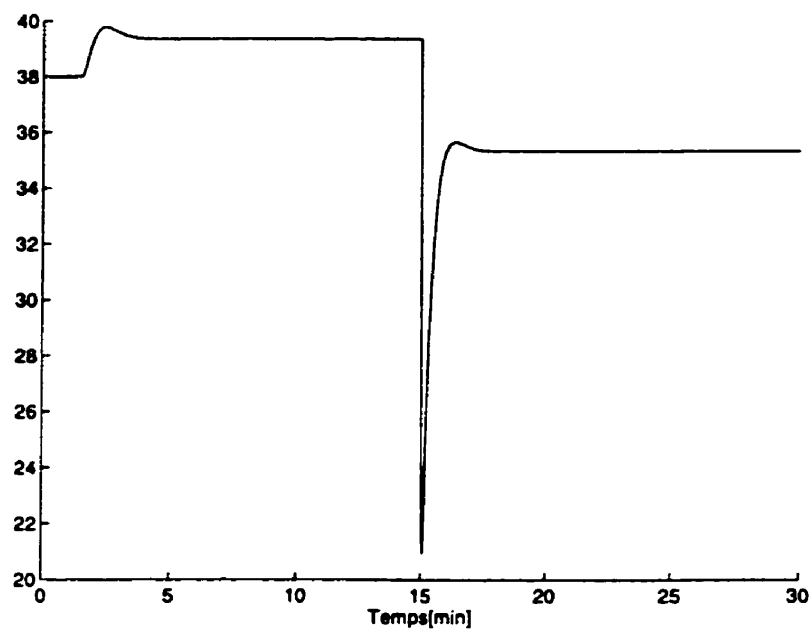


(a) Teneur du concentré [%]

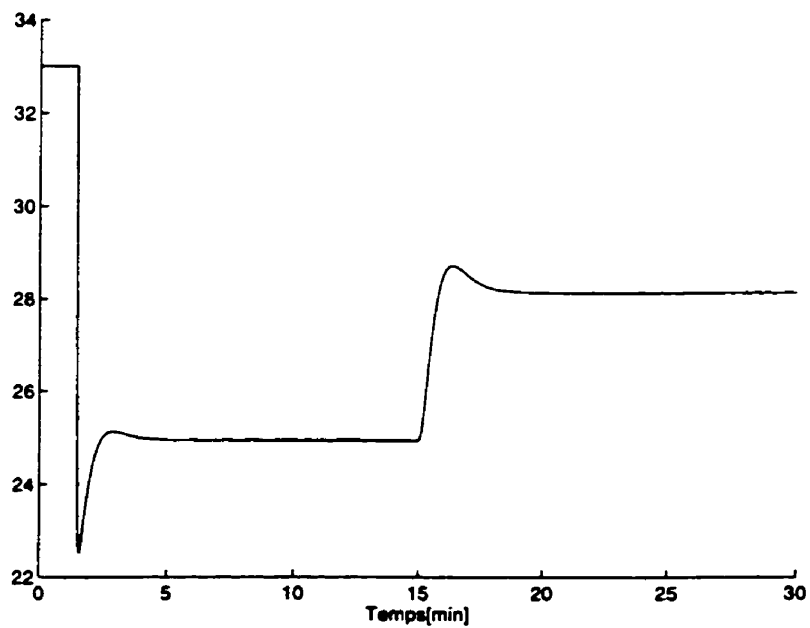


(b) Teneur du rejet [%]

FIG. 9.2: Retour d'état complet — Sorties



(a) Débit d'air [%]



(b) Débit du collecteur [%]

FIG. 9.3: Retour d'état complet — Commandes

cas, donnée par

$$u_1^\dagger = \frac{1}{B_1(s)} \left[ \left( s^n + \sum_{i=0}^{n-1} \rho_i s^i \right) y_{r_1} + \left( \sum_{i=0}^{n-1} (\hat{a}_{1,i} - \rho_i) s^i \right) x_{1,1} - \rho_0 x_{4,1} \right] \quad (9.28)$$

$$u_2^\dagger = \frac{1}{B_2(s)} \left[ \left( s^n + \sum_{i=0}^{n-1} \rho_i s^i \right) y_{r_2} + \left( \sum_{i=0}^{n-1} (\hat{a}_{2,i} - \rho_i) s^i \right) x_{2,1} - \rho_0 x_{3,1} \right] \quad (9.29)$$

Avec (9.28) comme commande, la dérivée de la fonction de Lyapunov (9.15) s'écrit

$$\begin{aligned} \dot{V}_{1,n} &= - \sum_{i=1}^n k_{1,i} \varepsilon_{1,i}^2 + \varepsilon_{1,n} \sum_{i=0}^{n-1} \bar{a}_{1,i} s^i x_{1,1} \\ &= - \sum_{i=1}^n k_{1,i} \varepsilon_{1,i}^2 + \varepsilon_{1,n} \sum_{i=0}^{n-1} \bar{a}_{1,i} x_{1,i+1} \end{aligned}$$

où les erreurs d'identification sont données par

$$\bar{a}_{1,i} = \hat{a}_{1,i} - a_{1,i}$$

**Remarque 9.4 (Signe)** — La présence des termes inconnus  $\bar{a}_{1,i}$  rend la dérivée de la fonction de Lyapunov de signe indéfini. Dans un tel cas, la stabilité du système ne peut être garantie. □

Afin de surmonter ce handicap, une nouvelle fonction de Lyapunov est introduite

$$\mathcal{V}_{1,n}^\dagger = \frac{1}{2} \sum_{i=1}^n \varepsilon_{1,i}^2 + \frac{1}{2\gamma} \sum_{i=1}^n \bar{a}_{1,i}^2$$

La dérivée de  $\mathcal{V}_{1,n}^\dagger$ , le long des trajectoires du système, est donnée par

$$\begin{aligned} \dot{\mathcal{V}}_n^\dagger &= - \sum_{i=1}^n k_{1,i} \varepsilon_{1,i}^2 + \varepsilon_{1,n} \sum_{i=0}^{n-1} \bar{a}_{1,i} x_{1,i+1} + \frac{1}{\gamma} \sum_{i=0}^{n-1} \bar{a}_{1,i} \dot{\bar{a}}_{1,i} \\ &= - \sum_{i=1}^n k_{1,i} \varepsilon_{1,i}^2 + \frac{1}{\gamma} \sum_{i=0}^{n-1} \bar{a}_{1,i} (\dot{\bar{a}}_{1,i} + \gamma x_{1,i+1} \varepsilon_{1,n}) \end{aligned}$$

Afin de la rendre définie négative, les lois de mise à jour des estimés  $\hat{a}_{1,i}$  peuvent être choisies telles que

$$\dot{\hat{a}}_{1,i} = \dot{\bar{a}}_{1,i} = -\gamma x_{1,i+1} \varepsilon_{1,n} \quad (9.30)$$

où

$$\begin{aligned}\varepsilon_{1_i} &= |I_{i-1} - sK_{1_{i-1}}| \varepsilon_{1_i} \\ &= |I_{i-1} - sK_{1_{i-1}}| (x_{1_i} + x_{4_i} - y_{r_i} + \chi_{1_i}) \\ &= |I_{i-1} - sK_{1_{i-1}}| (x_{1_i} - y_{r_i} + \mathcal{F}(s)x_{4_i})\end{aligned}$$

avec

$$\begin{aligned}\mathcal{F}(s) &= \varrho_0 |sI_n - K_{1_n}|^{-1} \\ &= \frac{\varrho_0}{s^n + \varrho_{n-1}s^{n-1} + \dots + \varrho_0}\end{aligned}$$

La dérivée de la fonction de Lyapunov devient, avec ce choix,

$$\dot{V}_{1_n}^{\dagger} = - \sum_{i=1}^n k_{1_i} \varepsilon_{1_i}^2 \leq 0$$

**Remarque 9.5 (Convergence)** La négativité de la dérivée de la fonction de Lyapunov garantit la convergence vers zéro des erreurs  $\varepsilon_{1_i}$  et  $\bar{a}_{1_i}$ . Toutefois, de par sa définition, l'erreur modifiée  $\varepsilon_{1_i}$  est différente de l'erreur de poursuite. La preuve et les conditions de convergence de cette dernière doivent être données séparément.

□

L'application de la commande (9.28) à l'équation du procédé donne

$$s^n x_{1_1} = (s^n + \sum_{i=0}^{n-1} \varrho_i s^i) y_{r_1} + \sum_{i=0}^{n-1} (\bar{a}_{1_i} - \varrho_i) s^i x_{1_1} - \varrho_0 x_{4_1}$$

que l'on peut réécrire

$$\left[ s^n + \sum_{i=0}^{n-1} (\varrho_{1_i} - \bar{a}_{1_i}) s^i \right] x_{1_1} = (s^n + \sum_{i=0}^{n-1} \varrho_i s^i) y_{r_1} - \varrho_0 x_{4_1} \quad (9.31)$$

Si l'on définit les filtres

$$\begin{aligned}\tilde{\mathcal{F}}_1(s) &= \frac{\varrho_0}{s^n + (\varrho_{n-1} - \bar{a}_{1_{n-1}})s^{n-1} + \dots + (\varrho_0 - \bar{a}_{1_0})} \\ \tilde{\mathcal{F}}_2(s) &= \frac{\varrho_0}{s^n + (\varrho_{n-1} - \bar{a}_{2_{n-1}})s^{n-1} + \dots + (\varrho_0 - \bar{a}_{2_0})}\end{aligned}$$

l'équation (9.31) se met sous la forme

$$x_{1i} = \tilde{\mathcal{F}}_1(s) \left[ \frac{y_{r1}}{\mathcal{F}(s)} - x_{4i} \right]$$

Avec les deux commandes  $(u_1^\dagger, u_2^\dagger)$  données par les équations (9.28) et (9.29), la dynamique en boucle fermée du procédé s'écrit

$$\begin{aligned} x_{1i} &= \frac{\tilde{\mathcal{F}}_1(s)}{\mathcal{F}(s)} y_{r1} - \tilde{\mathcal{F}}_1(s) \frac{G_4(s)}{G_2(s)} x_{2i} \\ x_{2i} &= \frac{\tilde{\mathcal{F}}_2(s)}{\mathcal{F}(s)} y_{r2} - \tilde{\mathcal{F}}_2(s) \frac{G_3(s)}{G_1(s)} x_{1i} \end{aligned}$$

Ces équations peuvent être réécrites sous la forme compacte

$$\left[ 1 - Q_I(s) \tilde{\mathcal{F}}_1(s) \tilde{\mathcal{F}}_2(s) \right] x_{1i} = \frac{\tilde{\mathcal{F}}_1(s)}{\mathcal{F}(s)} \left[ y_{r1} - \tilde{\mathcal{F}}_2(s) \frac{G_4(s)}{G_2(s)} y_{r2} \right] \quad (9.32)$$

$$\left[ 1 - Q_I(s) \tilde{\mathcal{F}}_1(s) \tilde{\mathcal{F}}_2(s) \right] x_{2i} = \frac{\tilde{\mathcal{F}}_2(s)}{\mathcal{F}(s)} \left[ y_{r2} - \tilde{\mathcal{F}}_1(s) \frac{G_3(s)}{G_1(s)} y_{r1} \right] \quad (9.33)$$

### Condition de stabilité

À partir des équations en boucle fermée (9.32)–(9.33) et du théorème du petit gain, tous les signaux à l'intérieur de la boucle sont bornés<sup>7</sup>, si

- $G_3(s)$  et  $G_4(s)$  sont stables,
- $G_1(s)$  et  $G_2(s)$  sont à phase minimale,
- $\tilde{\mathcal{F}}_1(s)$  et  $\tilde{\mathcal{F}}_2(s)$  sont stables et vérifient

$$\left\| Q_I(s) \tilde{\mathcal{F}}_1(s) \tilde{\mathcal{F}}_2(s) \right\|_{\infty} < 1$$

Comme dans le cas non adaptatif (section 9.3), les deux premières conditions sont imposées par la nature du procédé, alors que les deux dernières doivent être vérifiées.

### Stabilité de $\tilde{\mathcal{F}}_1$ et $\tilde{\mathcal{F}}_2$

Étant donné que  $\tilde{\mathcal{F}}_1$  et  $\tilde{\mathcal{F}}_2$  dépendent des erreurs d'estimation  $\bar{a}_{1i}$ , et  $\bar{a}_{2i}$ , leur stabilité doit être prouvée. Pour ce faire, on considère le cas scalaire, où

$$G_3(s) = G_4(s) = 0 \quad \text{et} \quad \chi_i = 0$$

<sup>7</sup>Donc la stabilité interne est garantie

Dans ce cas, le contrôleur (9.28) donne comme dynamique en boucle fermée du premier sous-système

$$x_{11} = y_1 = \frac{\bar{\mathcal{F}}_1(s)}{\mathcal{F}(s)} y_{r1}$$

La fonction de Lyapunov garantit, pour le cas monovariante, la stabilité de la boucle fermée  $(G_1(s), u_1^\dagger)$ , si la loi d'adaptation est choisie telle donnée par l'équation (9.30). Cela prouve que le filtre  $\bar{\mathcal{F}}_1$  est stable, et ce malgré la présence des erreurs d'identification  $\bar{a}_{1i}$ . Le même raisonnement s'applique à  $\bar{\mathcal{F}}_2$ ,  $G_2(s)$  et  $u_2^\dagger$ .

On a, d'autre part,

$$\bar{\mathcal{F}}_1^{-1}(s) = \mathcal{F}^{-1}(s) + \frac{A_1(s) - \hat{A}_1(s)}{\varrho_0}$$

où  $A_1(s)$  est le dénominateur de  $G_1(s)$  et  $\hat{A}_1(s)$  son estimé. Ceci donne comme condition de stabilité

$$\begin{aligned} \|\mathcal{Q}_I\|_\infty &< \left\| \frac{1}{\bar{\mathcal{F}}_1(s)\bar{\mathcal{F}}_2(s)} \right\| \\ &= \left\| \frac{1}{\mathcal{F}(s)} + \frac{A_1(s) - \hat{A}_1(s)}{\varrho_0} \right\| \left\| \frac{1}{\mathcal{F}(s)} + \frac{A_2(s) - \hat{A}_2(s)}{\varrho_0} \right\| \\ &= \left\| \frac{1}{\mathcal{F}(s)} - \frac{\bar{A}_1(s)}{\varrho_0} \right\| \left\| \frac{1}{\mathcal{F}(s)} - \frac{\bar{A}_2(s)}{\varrho_0} \right\| \end{aligned} \quad (9.34)$$

**Remarque 9.6 (Comparaison)** — La condition de stabilité est identique, aux erreurs d'identifications  $\bar{A}_1(s)$  et  $\bar{A}_2(s)$  près, à celle obtenue dans le cas non adaptatif. Ces erreurs étant garanties de converger vers zéro, leur effet sur la stabilité reste limité. Toutefois, contrairement au cas non adaptatif, dans le cas adaptatif, la condition de stabilité (9.34), se basant sur le quotient d'interaction et les polynômes  $A_1(s)$  et  $A_2(s)$  qui sont inconnus, ne peut être testée a priori. Alléger ces conditions, voire s'en affranchir complètement, constitue l'un des axes majeurs de recherche que nous continuons d'explorer avec beaucoup d'intérêt. D'un point de vue qualitatif, tout comme pour le cas non adaptatif, cette condition exige tout simplement que les branches directes soient plus rapides que les branches croisées. Une autre différence, entre le cas adaptatif

et non adaptatif, réside dans la nécessité d'implanter, pour le premier, les termes  $\chi_i$ , dont dépendent les lois d'adaptation.

□

### Simulation

Le contrôleur adaptatif obtenu est appliqué à la commande du procédé de flottation donné par l'équation (9.27), où les paramètres sont supposés inconnus. Les résultats des simulations sont donnés par les figures 9.4 à 9.8. On y constate que la stabilité de la boucle, élément critique pour toute commande adaptative décentralisée, est préservée en dépit de la présence des erreurs d'identification. Le contrôle assure un bon comportement en poursuite, sauf qu'en régulation, une erreur statique persiste sur la sortie du deuxième sous-système figure 9.4 (b). Ce problème sera soulevé et résolu, dans la section 11.3.2, par l'introduction d'une action intégrale dans les contrôleurs décentralisés obtenus.

## 9.5.2 Systèmes non linéaires

La méthode, qui vient d'être présentée dans le cadre linéaire, s'étend facilement aux systèmes non linéaires, si ces derniers se présentent sous la forme donnée par la figure 9.9. Si les fonctions non linéaires  $N_j$  peuvent être transformées sous la forme triangulaire

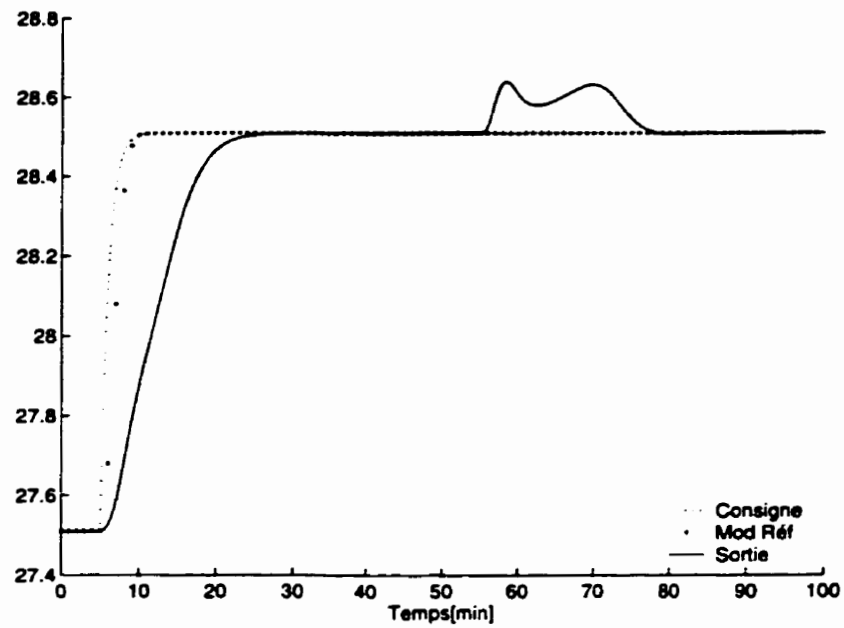
$$\begin{aligned}\dot{x}_{j_1} &= x_{j_2} \\ \dot{x}_{j_2} &= x_{j_3} \\ &\vdots \\ \dot{x}_{j_n} &= \varphi_j(x_{j_1}, \dots, x_{j_n}) + v_j\end{aligned}$$

où les commandes filtrées  $v_j$  sont définies par

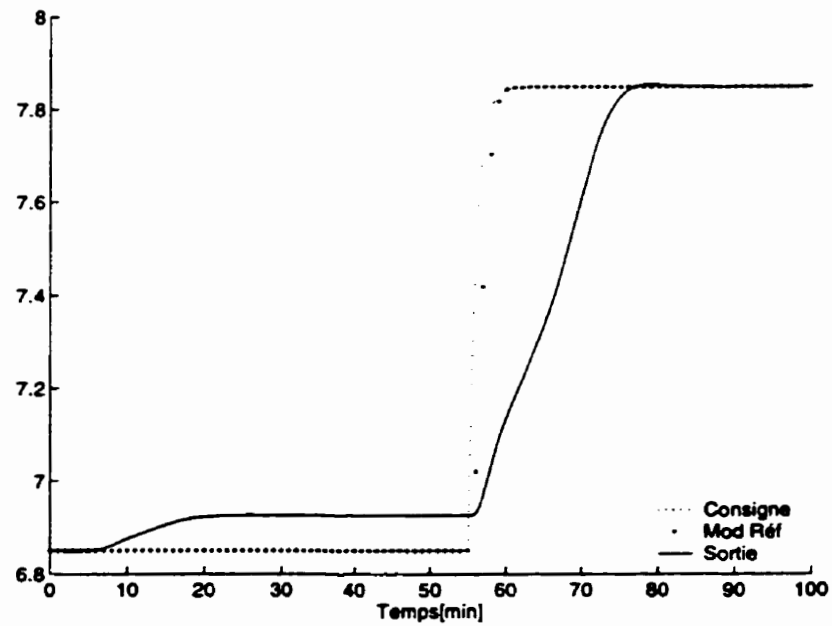
$$v_j = \begin{cases} \psi_j(x_{j_1}, \dots, x_{j_n})u_j & \text{si } j = 1, 2, \\ \psi_j(x_{j_1}, \dots, x_{j_n})u_{j-2} & \text{si } j = 3, 4. \end{cases}$$

alors la procédure de design reste identique au cas linéaire pour le  $n - 1$  premières étapes, et seule la dernière diffère. En effet, le même développement donne pour la



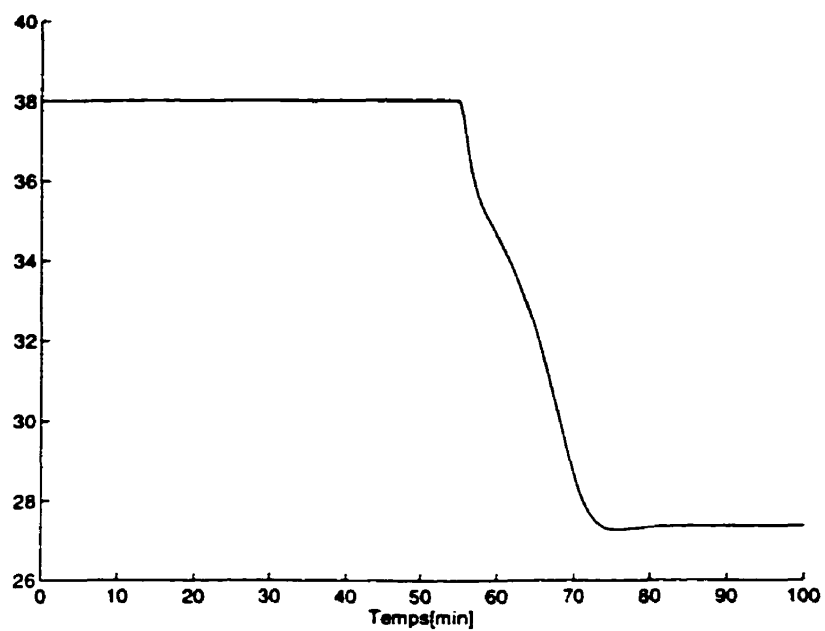


(a) Teneur du concentré [%]

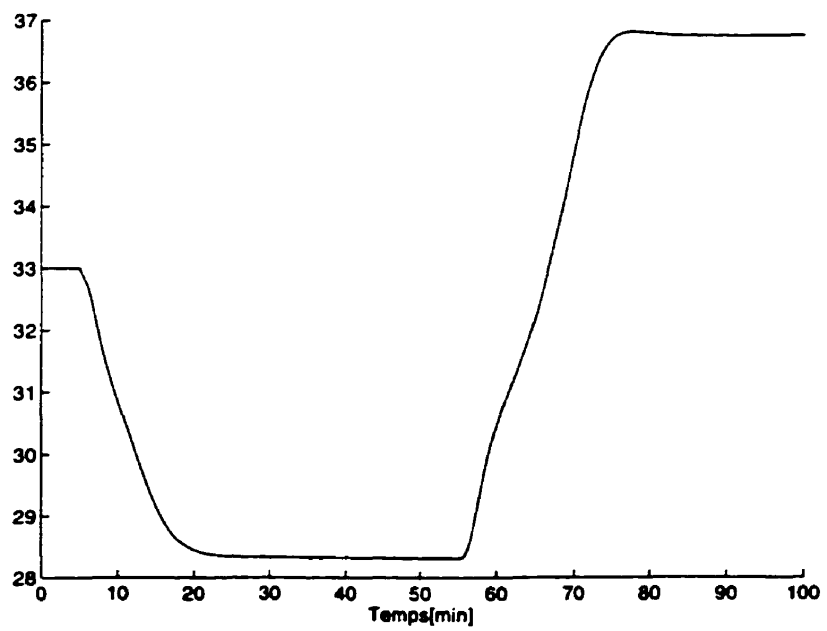


(b) Teneur du rejet [%]

FIG. 9.4: Approche adaptative — Sorties

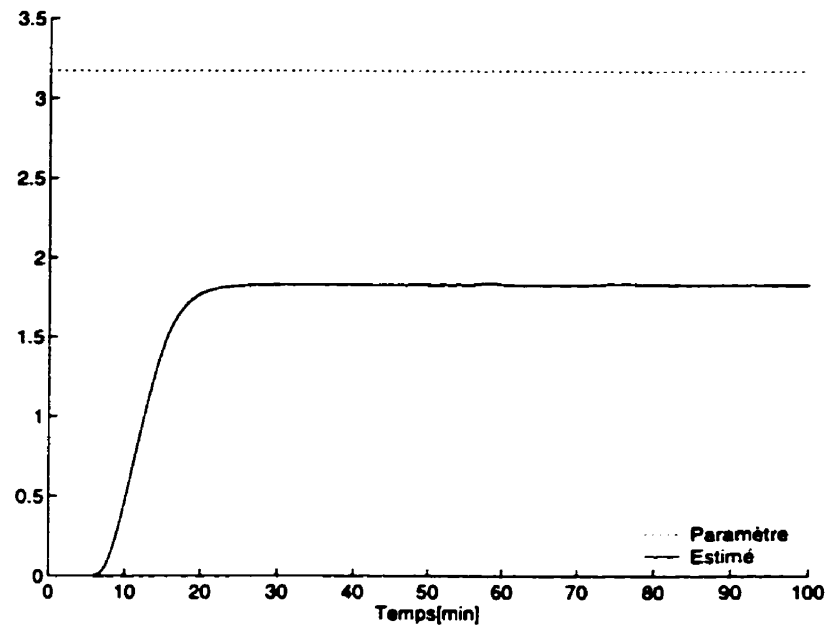
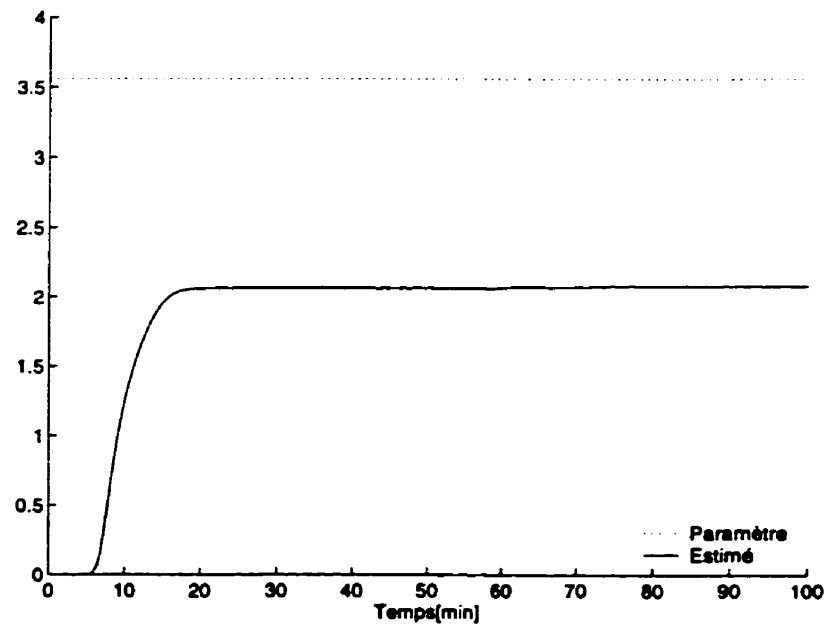


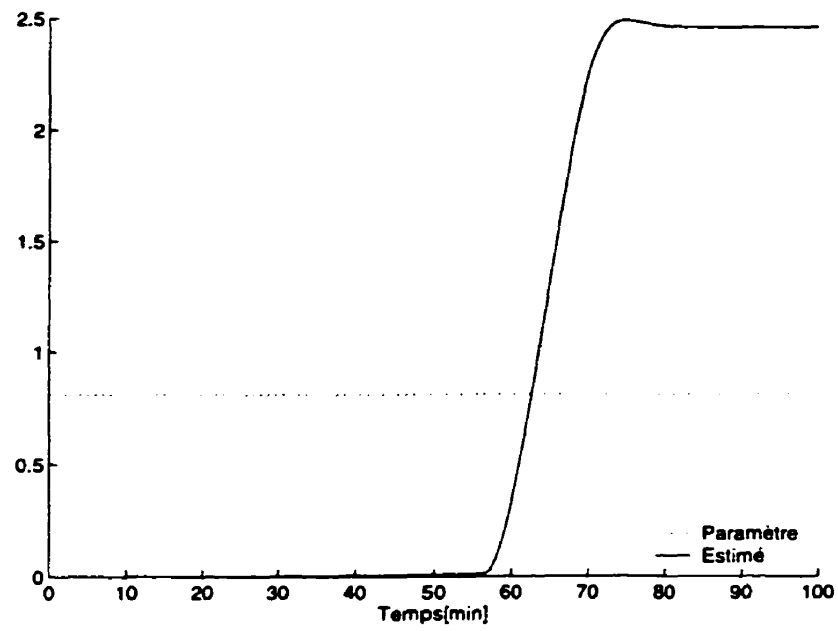
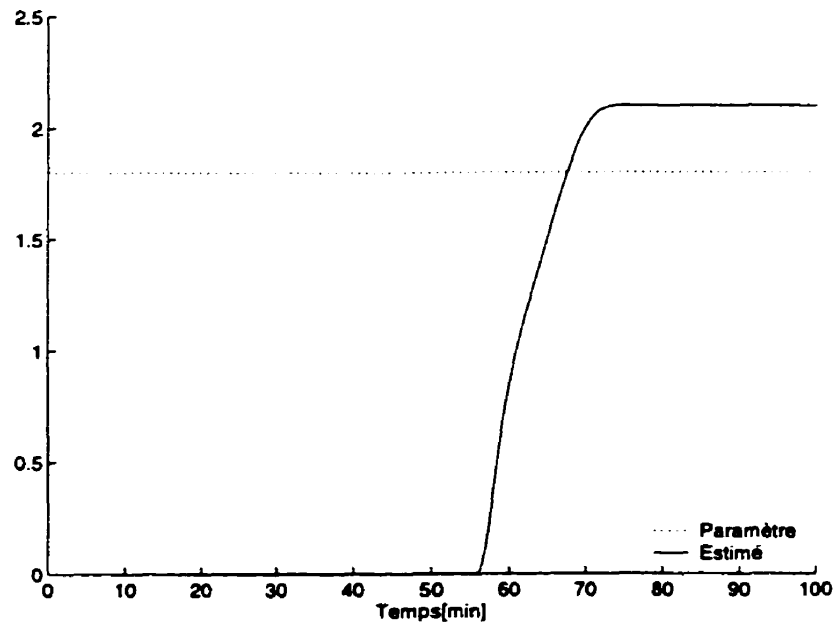
(a) Débit d'air [%]

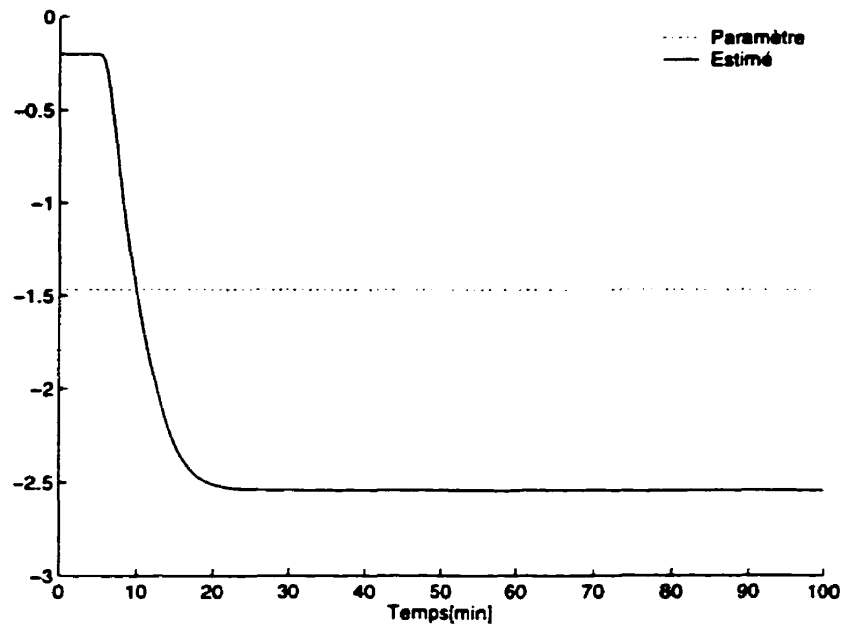
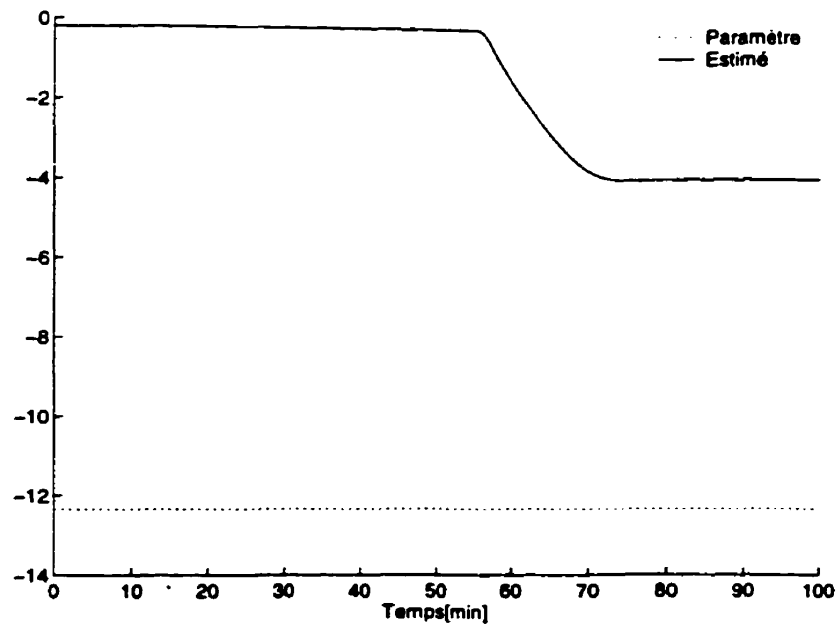


(b) Débit du collecteur [%]

FIG. 9.5: Approche adaptative — Commandes

(a)  $a_{1_0}$ (b)  $a_{1_1}$ FIG. 9.6: Approche adaptative — Paramètres de  $G_1$

(a)  $a_{2_0}$ (b)  $a_{2_1}$ FIG. 9.7: Approche adaptative — Paramètres de  $G_2$

(a)  $c_1$ (b)  $c_2$ FIG. 9.8: Approche adaptative — Paramètres  $c_1$  &  $c_2$

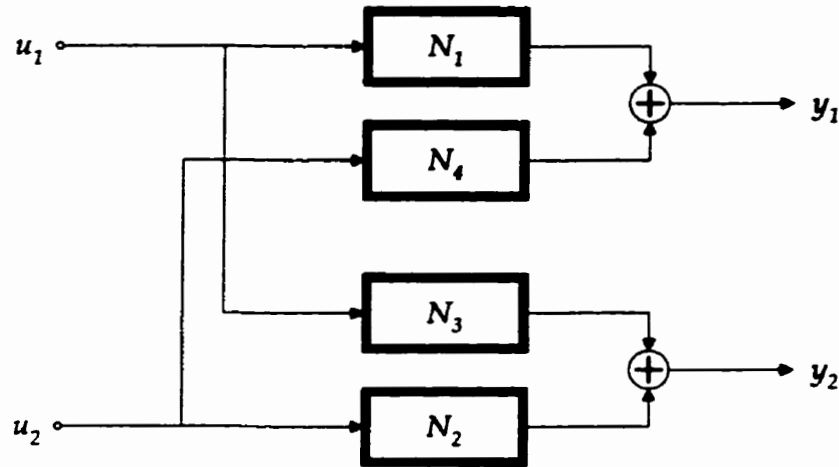


FIG. 9.9: Procédé non linéaire deux-par-deux

première commande intermédiaire

$$v_1 = \dot{\alpha}_{1_{n-1}} - \varphi_1(x_{1_1}, \dots, x_{i_n}) - \dot{x}_{4_n} - \varepsilon_{1_{n-1}} - k_{1_n} \varepsilon_{1_n}$$

Le contrôleur multivariable  $(u_1, u_2)$  qui en résulte est donné par

$$u_1 = \frac{v_1}{\psi_1(x_{1_1}, \dots, s^n x_{1_1})} = \frac{1}{\psi_1} \left[ s^n y_{r_1} + (\mathcal{F}(s) - s^n)(y_{r_1} - y_1) - \dot{x}_{4_n} - \varphi_1 \right]$$

$$u_2 = \frac{v_2}{\psi_2(x_{2_1}, \dots, s^n x_{2_1})} = \frac{1}{\psi_2} \left[ s^n y_{r_2} + (\mathcal{F}(s) - s^n)(y_{r_2} - y_2) - \dot{x}_{3_n} - \varphi_2 \right]$$

### Erreur modifiée

Une modification de l'erreur, identique au cas linéaire, donne comme commande intermédiaire  $v_1$

$$v_1 = \mathcal{F}(s)(y_{r_1} - y_1 - \chi_1) + \dot{x}_{1_n} - \varphi_1$$

que l'on peut réécrire sous la forme

$$\begin{aligned} v_1 &= \mathcal{F}(s)(y_{r_1} - x_{1_1}) - \mathcal{F}(s)(\chi_1 + x_{4_1}) + (\dot{x}_{1_n} - \varphi_1) \\ &= \mathcal{F}(s)y_{r_1} - \mathcal{F}(s)(\chi_1 + x_{4_1}) + (s^n - \mathcal{F}(s))x_{1_1} - \varphi_1 \\ &= \mathcal{F}(s)y_{r_1} - \varrho_0 y_1 - \mathcal{F}(s) \left[ \chi_1 - \left( \frac{\varrho_0}{\mathcal{F}(s)} - 1 \right) x_{4_1} \right] + \left[ s^n - \mathcal{F}(s) + \varrho_0 \right] x_{1_1} - \varphi_1 \end{aligned}$$

Il suffit à présent de choisir, comme dans le cas linéaire,

$$\begin{aligned}\chi_1 &= \left[ \frac{\varrho_0}{\mathcal{F}(s)} - 1 \right] x_{4_1} \\ &= - \frac{s^n + \varrho_{n-1}s^{n-1} + \dots + \varrho_1s}{s^n + \varrho_{n-1}s^{n-1} + \dots + \varrho_1s + \varrho_0} x_{4_1}\end{aligned}$$

pour obtenir

$$\begin{aligned}v_1 &= \mathcal{F}(s)y_{r_1} - \varrho_0y_1 + \left[ s^n - \mathcal{F}(s) + \varrho_0 \right] x_{1_1} - \varphi_1(x_{1_1}, \dots, s^n x_{1_1}) \\ &= \mathcal{F}(s)y_{r_1} - \varrho_0y_1 - \sum_{j=1}^{n-1} \varrho_j s^j x_{1_1} - \varphi_1(x_{1_1}, \dots, s^n x_{1_1})\end{aligned}$$

De la même façon, on obtient pour le deuxième sous-système

$$\begin{aligned}v_2 &= \mathcal{F}(s)y_{r_2} - \varrho_0y_2 + \left[ s^n - \mathcal{F} + \varrho_0 \right] x_{2_1} - \varphi_2(x_{2_1}, \dots, s^n x_{2_1}) \\ &= \mathcal{F}(s)y_{r_2} - \varrho_0y_2 - \sum_{j=1}^{n-1} \varrho_j s^j x_{2_1} - \varphi_2(x_{2_1}, \dots, s^n x_{2_1})\end{aligned}$$

### Conditions de stabilité

Les équations du système en boucle fermée sont données par

$$\begin{aligned}x_{1_1} &= y_{r_1} - \frac{\varrho_0}{\mathcal{F}(s)} \left[ N_4(N_2^{-1}(x_{2_1})) \right] = y_{r_1} - \frac{\varrho_0}{\mathcal{F}(s)} \left[ N_2^{-1} \circ N_4(x_{2_1}) \right] \\ x_{2_1} &= y_{r_2} - \frac{\varrho_0}{\mathcal{F}(s)} \left[ N_3(N_1^{-1}(x_{1_1})) \right] = y_{r_2} - \frac{\varrho_0}{\mathcal{F}(s)} \left[ N_1^{-1} \circ N_3(x_{1_1}) \right]\end{aligned}$$

À partir du théorème du **petit gain**, on peut conclure que si  $N_1^{-1}$ ,  $N_2^{-1}$ ,  $N_3$  et  $N_4$  sont tous stables<sup>8</sup>, et vérifient

$$\left\| N_2^{-1} \circ N_4(x) \right\| \left\| N_1^{-1} \circ N_3(x) \right\| < \|x\|^2 \left\| \frac{\mathcal{F}(s)}{\varrho_0} \right\|^2$$

alors tous les signaux à l'intérieur de la boucle sont bornés; ce qui implique la stabilité. Or, le polynôme  $\mathcal{F}(s)$ , qui fixe la dynamique en poursuite/régulation, est choisi tel que

$$\left\| \frac{\varrho_0}{\mathcal{F}(s)} \right\| \leq 1$$

<sup>8</sup>Étant donné que  $\mathcal{F}(s)$  est déjà choisi hurwitzien.

Cette propriété permet de réduire la condition de stabilité à

$$\left\| \left[ N_2^{-1} \circ N_4(x) \right] \left[ N_1^{-1} \circ N_3(x) \right] \right\| < \|x\|^2 \quad (9.35)$$

**Remarque 9.7** — Une condition suffisante, pour que cette condition soit vérifiée, est que la non-linéarité  $N_1$  croît plus rapidement que  $N_3$  et  $N_2$  plus rapidement que  $N_4$ . Dans de telles conditions, on a

$$\|N_1^{-1} \circ N_3(x)\| < \|x\|$$

et

$$\|N_2^{-1} \circ N_4(x)\| < \|x\|$$

ce qui permet de satisfaire (9.35).

□

## 9.6 Conclusion

À partir de la méthode du **backstepping**, un seul contrôleur multivariable, garantissant les performances requises du système global, est conçu. Par la suite, des conditions sous lesquelles ce contrôleur centralisé peut être fragmenté en un ensemble de contrôleurs locaux, sans altérer la stabilité et les performances en boucle fermée, sont dérivées. La modification des erreurs régulées dans la procédure du **backstepping**, via l'introduction de termes supplémentaires appropriés, a rendu possible l'élimination des composantes croisées de la loi de commande multivariable. Chacun des contrôleurs locaux obtenus dépend uniquement de l'information qui lui est localement disponible et possède son propre critère de performances. Sous certaines conditions, des versions non linéaires et/ou adaptatives de la MEM peuvent être obtenues.

Les versions de la MEM présentées dans ce chapitre ont été élaborées sous l'hypothèse de la disponibilité à la mesure du vecteur d'état du procédé. Cette hypothèse,



quand elle est réalisable, peut se traduire par des coûts exorbitants, limitant l'application du contrôleur proposé. Pour palier à cette situation, des versions à retour de sortie, qui réduisent le nombre des variables mesurées, sont présentées dans les chapitres 10 et 11.

# Chapitre 10

## Design avec observateur

Dans ce chapitre une version à retour de sortie de la **méthode de l'erreur modifiée** est présentée (Benaskeur & Desbiens 1999e). Cette dernière combine la technique récursive du **backstepping** avec observateur introduite dans le chapitre 3 avec la méthode de l'erreur modifiée par retour d'état présentée dans le chapitre 9.

Afin d'être en mesure d'estimer l'état du procédé, un observateur de Luenberger (Luenberger 1964, 1971) est utilisé. Son équation est donnée par (les détails du design sont volontairement omis)

$$\begin{aligned}\dot{\hat{\mathbf{x}}} &= \mathbf{A}\hat{\mathbf{x}} + \mathbf{H}\left[y_1 - \mathbf{C}^T\hat{\mathbf{x}}\right] + \mathbf{B}v \\ &= \mathbf{A}_0\hat{\mathbf{x}} + \mathbf{H}y_1 + \mathbf{B}v\end{aligned}$$

où le vecteur des gains

$$\mathbf{H} = [h_{1_1} \quad h_{1_2} \quad \cdots \quad h_{1_n} \quad h_{4_1} \quad h_{4_2} \quad \cdots \quad h_{4_n}]^T$$

est choisi afin de rendre la matrice  $\mathbf{A}_0$  hurwitzienne. L'équation de l'erreur d'estimation résultante est donnée par

$$\dot{\tilde{\mathbf{x}}} = \mathbf{A}_0\tilde{\mathbf{x}} \tag{10.1}$$

avec  $\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$  est assuré de converger vers zéro.

## 10.1 MEM avec observateur

La disponibilité d'une estimation de l'état du procédé permet, à présent, de combiner la MEM et la technique du **backstepping** avec observateur, afin d'obtenir un contrôleur décentralisé à retour de sortie. Étant donné que cette approche diffère en plusieurs points de celle basée sur la mesure des variables d'état, la procédure récursive de design sera abordée avec plus de détails que dans le chapitre 9.

**Étape 1** — La fonction de contrôle de Lyapunov (fcl) suivante est considérée

$$V_{1_1} = \frac{1}{2}\varepsilon_{1_1}^2 + \frac{1}{m_1}\tilde{\mathbf{x}}^T \mathbf{P}\tilde{\mathbf{x}} \quad (10.2)$$

où  $\mathbf{P}$  est une matrice symétrique définie positive, solution de l'équation de Lyapunov

$$\mathbf{P}\mathbf{A}_0 + \mathbf{A}_0^T \mathbf{P} = -\mathbf{I}$$

L'existence et la positivité de  $\mathbf{P}$  sont garanties par la stabilité du système d'erreurs (10.1). Pour obtenir une structure de commande décentralisée, l'erreur modifiée est toujours définie par

$$\varepsilon_{1_1} = y_1 - y_{r_1} + \chi_1$$

Toutefois, à cause de la présence de l'observateur (et surtout des erreurs d'estimation), un choix différent du terme  $\chi_1$  s'impose. Ce dernier doit cependant vérifier la condition

$$\lim_{t \rightarrow \infty} \chi_1 = 0$$

afin d'assurer des erreurs nulles en régime permanent. La dérivée de la fcl (10.2) est

$$\begin{aligned} \dot{V}_{1_1} &= \varepsilon_{1_1} \dot{\varepsilon}_{1_1} + \frac{1}{m_1} \tilde{\mathbf{x}}^T \left[ \mathbf{P}\mathbf{A}_0 + \mathbf{A}_0^T \mathbf{P} \right] \tilde{\mathbf{x}} \\ &= \varepsilon_{1_1} \left[ x_{1_2} + x_{4_2} - \xi_0 + \dot{\chi}_1 \right] - \frac{1}{m_1} \tilde{\mathbf{x}}^T \tilde{\mathbf{x}} \end{aligned}$$

où  $\xi_0 = \dot{y}_{r_1}$ . Étant donné que  $x_{1_2} + x_{4_2}$  n'est pas disponible pour la mesure, il ne peut être choisi comme commande virtuelle. L'idée du **backstepping** avec observateur

consiste à le remplacer par son estimé, fourni par l'observateur. L'équation de la dérivée est réécrite sous la forme

$$\dot{\mathcal{V}}_{1_1} = \varepsilon_{1_1} \left[ \hat{x}_{1_2} + \hat{x}_{4_2} + \bar{x}_{1_2} + \bar{x}_{4_2} - \xi_0 + \dot{\chi}_1 \right] - \frac{1}{m_1} \bar{\mathbf{x}}^T \bar{\mathbf{x}}$$

Il est à présent possible de choisir comme commande virtuelle

$$(\hat{x}_{1_2} + \hat{x}_{4_2})_d \triangleq \alpha_{1_1} + \varsigma_{1_1} = -k_1 \varepsilon_{1_1} + \xi_0 - \dot{\chi}_1 + \varsigma_{1_1}$$

où  $\varsigma_{1_1}$  est un terme d'amortissement non linéaire, dont la valeur reste à déterminer.

L'expression de la dérivée devient avec ce choix

$$\begin{aligned} \dot{\mathcal{V}}_{1_1} &= -k_1 \varepsilon_{1_1}^2 + \varepsilon_{1_1} \varsigma_{1_1} + \varepsilon_{1_1} \left[ \bar{x}_{1_2} + \bar{x}_{4_2} \right] - \frac{1}{m_1} \bar{\mathbf{x}}^T \bar{\mathbf{x}} \\ &\leq -k_1 \varepsilon_{1_1}^2 + \varepsilon_{1_1} \varsigma_{1_1} + \varepsilon_{1_1} \left[ \bar{x}_{1_2} + \bar{x}_{4_2} \right] - \frac{1}{2m_1} \left[ \bar{x}_{1_2}^2 + \bar{x}_{4_2}^2 \right] - \frac{1}{2m_1} \bar{\mathbf{x}}^T \bar{\mathbf{x}} \\ &= -k_1 \varepsilon_{1_1}^2 + \varepsilon_{1_1} \varsigma_{1_1} + m_1 \varepsilon_{1_1}^2 - \frac{m_1}{2} \left[ \varepsilon_{1_1} - \frac{\bar{x}_{1_2}}{m_1} \right]^2 - \frac{m_1}{2} \left[ \varepsilon_{1_1} - \frac{\bar{x}_{4_2}}{m_1} \right]^2 - \frac{1}{2m_1} \bar{\mathbf{x}}^T \bar{\mathbf{x}} \\ &\leq -k_1 \varepsilon_{1_1}^2 + \left[ \varsigma_{1_1} + m_1 \varepsilon_{1_1} \right] \varepsilon_{1_1} \end{aligned}$$

Il suffit, à présent, de prendre

$$\varsigma_{1_1} = -m_1 \varepsilon_{1_1}$$

pour assurer la négativité de la dérivée  $\dot{\mathcal{V}}_{1_1}$ . Ceci aura comme conséquence de garantir la stabilité de la première équation d'erreur qui se réduit, alors, à

$$\dot{\varepsilon}_{1_1} = -d_1 \varepsilon_{1_1} + \bar{x}_{1_2} + \bar{x}_{4_2}$$

où  $d_1 = k_1 + m_1$ . La commande virtuelle résultante est donnée par

$$\begin{aligned} \alpha_{1_1} + \varsigma_{1_1} &= -d_1 \varepsilon_{1_1} + \xi_0 - \dot{\chi}_1 \\ &= (\varrho_{1_0} + \varrho_{1_1} s)(y_{r_1} - \chi_1) - \varrho_{1_0} y_1 \end{aligned}$$

avec

$$\varrho_{1_0} = d_1$$

$$\varrho_{1_1} = 1$$

**Étape 2** — La deuxième variable d'erreur est définie par

$$\varepsilon_{12} = \hat{x}_{12} + \hat{x}_{42} - (\alpha_{11} + \varsigma_{11})$$

et la nouvelle fonction de Lyapunov est une version augmentée de la précédente

$$\mathcal{V}_{12} = \mathcal{V}_{11} + \frac{1}{2}\varepsilon_{12}^2 + \frac{1}{m_2}\bar{\mathbf{x}}^T \mathbf{P} \bar{\mathbf{x}}$$

Sa dérivée est donnée par

$$\begin{aligned} \dot{\mathcal{V}}_2 &= \dot{\mathcal{V}}_1 + \varepsilon_{12}\dot{\varepsilon}_{12} - \frac{1}{m_2}\bar{\mathbf{x}}^T \bar{\mathbf{x}} \\ &\leq -k_1\varepsilon_{11}^2 + \varepsilon_{12}(\varepsilon_{11} + \dot{\varepsilon}_{12}) - \frac{1}{m_2}\bar{\mathbf{x}}^T \bar{\mathbf{x}} \end{aligned}$$

où

$$\dot{\varepsilon}_{12} = \hat{x}_{13} + \hat{x}_{43} - (\dot{\alpha}_{11} + \dot{\varsigma}_{11}) + (h_{12} + h_{42}) \left[ y_1 - (\hat{x}_{11} + \hat{x}_{41}) \right]$$

et

$$\begin{aligned} \dot{\alpha}_{11} + \dot{\varsigma}_{11} &= (\varrho_{10}s + \varrho_{11}s^2)(y_{r1} - \chi_1) - \varrho_{10}(\hat{x}_{12} + \hat{x}_{42}) + \varrho_{10}(\bar{x}_{12} + \bar{x}_{42}) \\ &= \boldsymbol{\xi}_1 + \varrho_{10}(\bar{x}_{12} + \bar{x}_{42}) \end{aligned}$$

La partie connue de  $\dot{\alpha}_{11} + \dot{\varsigma}_{11}$  est donnée par

$$\boldsymbol{\xi}_1 = (\varrho_{10}s + \varrho_{11}s^2)(y_{r1} - \chi_1) - \varrho_{10}(\hat{x}_{12} + \hat{x}_{42})$$

La seconde commande virtuelle peut maintenant être choisie comme

$$(\hat{x}_{13} + \hat{x}_{43})_d \triangleq \alpha_{12} + \varsigma_{12} = -k_2\varepsilon_{12} - \varepsilon_{11} - (h_{12} + h_{42}) \left[ y_1 - (\hat{x}_{11} + \hat{x}_{41}) \right] + \boldsymbol{\xi}_1 + \varsigma_{12}$$

Avec ce choix, la dérivée de la fcl devient

$$\begin{aligned} \dot{\mathcal{V}}_{12} &\leq -k_1\varepsilon_{11}^2 - k_2\varepsilon_{12}^2 - \frac{1}{m_2}\bar{\mathbf{x}}^T \bar{\mathbf{x}} + \varsigma_{12}\varepsilon_{12} + \left[ \varrho_{10}(\bar{x}_{12} + \bar{x}_{42}) \right] \varepsilon_{12} \\ &\leq -k_1\varepsilon_{11}^2 - k_2\varepsilon_{12}^2 - \frac{1}{2m_2}\bar{\mathbf{x}}^T \bar{\mathbf{x}} + \varsigma_{12}\varepsilon_{12} + \left[ \varrho_{10}(\bar{x}_{12} + \bar{x}_{42}) \right] \varepsilon_{12} - \frac{1}{2m_2} \left[ \bar{x}_{12}^2 + \bar{x}_{42}^2 \right] \\ &= -k_1\varepsilon_{11}^2 - k_2\varepsilon_{12}^2 + \varsigma_{12}\varepsilon_{12} + m_2 \left[ \varrho_{10}\varepsilon_{12} \right]^2 \\ &\quad - \frac{1}{2m_2}\bar{\mathbf{x}}^T \bar{\mathbf{x}} - \frac{m_2}{2} \left[ \varrho_{10}\varepsilon_{12} - \frac{\bar{x}_{12}}{m_2} \right]^2 - \frac{m_2}{2} \left[ \varrho_{10}\varepsilon_{12} - \frac{\bar{x}_{42}}{m_2} \right]^2 \\ &\leq -k_1\varepsilon_{11}^2 - k_2\varepsilon_{12}^2 + \left[ \varsigma_{12} + m_2\varrho_{10}^2\varepsilon_{12} \right] \varepsilon_{12} \end{aligned}$$

où, une fois encore, un choix approprié de l'amortissement non linéaire va permettre de garantir la négativité de la dérivée. En effet, avec

$$\varsigma_{1_2} = -m_2 \varrho_{1_0}^2 \varepsilon_{1_2}$$

la stabilité du système d'erreur résultant est assurée. Ce dernier est donné par

$$\begin{aligned}\dot{\varepsilon}_{1_1} &= -d_1 \varepsilon_{1_1} + \varepsilon_{1_2} + \tilde{x}_{1_2} + \tilde{x}_{4_2} \\ \dot{\varepsilon}_{1_2} &= -\varepsilon_{1_1} - d_2 \varepsilon_{1_2} + \varrho_{1_0} (\tilde{x}_{1_2} + \tilde{x}_{4_2})\end{aligned}$$

où

$$d_2 = k_2 + m_2 \varrho_{1_0}^2$$

La commande virtuelle devient avec ce choix

$$\begin{aligned}\alpha_{1_2} &= -d_2 \varepsilon_{1_2} - \varepsilon_{1_1} - (h_{1_2} + h_{4_2}) \left[ y_1 - (\hat{x}_{1_1} + \hat{x}_{4_1}) \right] + \xi_1 \\ &= \left[ \sum_{j=0}^2 \varrho_{1_{2j}} s^j \right] (y_{r_1} - \chi_1) - \sum_{j=1}^2 p_{2j} (\hat{x}_{1_j} + \hat{x}_{4_j}) + (p_{2_1} - \varrho_{2_0}) y_1\end{aligned}$$

avec

$$\begin{aligned}\varrho_{2_0} &= 1 + d_2 \varrho_{1_0} \\ \varrho_{2_1} &= \varrho_{1_0} + d_2 \varrho_{1_1} \\ \varrho_{2_2} &= \varrho_{1_1}\end{aligned}$$

et

$$\begin{aligned}p_{2_1} &= -h_{1_2} - h_{4_2} \\ p_{2_2} &= d_2 + \varrho_{1_0}\end{aligned}$$

**Étape  $i$**  — La  $i^{\text{ème}}$  fonction de Lyapunov est définie par

$$V_{1_i} = \sum_{j=1}^i \left[ \frac{1}{2} \varepsilon_{1_j}^2 + \frac{1}{m_j} \tilde{\mathbf{x}}^T \mathbf{P} \tilde{\mathbf{x}} \right]$$

où

$$\varepsilon_{1j} = \begin{cases} x_{1j} + x_{4j} - y_{r1} + \chi_1 & \text{si } j = 1, \\ \hat{x}_{1j} + \hat{x}_{4j} - (\alpha_{1j-1} + \varsigma_{1j-1}) & \text{sinon} \end{cases}$$

La  $i^{\text{ieme}}$  commande virtuelle résultante est donnée

$$\begin{aligned} (\hat{x}_{1i+1} + \hat{x}_{4i+1})_d &\triangleq \alpha_{1i} + \varsigma_{1i} \\ &= \left[ \sum_{j=0}^i \varrho_{ij} s^j \right] (y_{r1} - \chi_1) - \sum_{j=1}^i p_{ij} \left[ \hat{x}_{1j} + \hat{x}_{4j} \right] + (p_{i1} - \varrho_{i0}) y_1 \end{aligned} \quad (10.3)$$

où  $\varrho_{ij}$  et  $p_{ij}$  sont calculés récursivement par

$$\varrho_{ij} = d_i \varrho_{(i-1)j} + \varrho_{(i-2)j} + \varrho_{(i-1)(j-1)}, \quad j = 0, \dots, i \quad (10.4)$$

$$p_{ij} = d_i p_{(i-1)j} + p_{(i-2)j} + p_{(i-1)(j-1)} + \beta_j + \pi_j, \quad j = 1, \dots, i \quad (10.5)$$

avec

$$\varrho_{xx} = 1 \quad (10.6)$$

$$\varrho_{xy} = p_{xy} = 0 \quad \text{si } x < y, \quad x \leq 1 \quad \text{ou} \quad y < 0 \quad (10.7)$$

et

$$d_i = k_i + m_i \left[ \frac{\partial \alpha_{1i-1}}{\partial y_1} \right]^2 \quad (10.8)$$

$$= k_i + m_i \left[ p_{(i-1)1} - \varrho_{(i-1)0} \right]^2 \quad (10.9)$$

$k_i$  et  $m_i$  sont des paramètres de design positifs, qui fixent la dynamique en boucle fermée désirée. Les termes  $\pi_j$  et  $\beta_j$  sont donnés par

$$\pi_j = \begin{cases} -(h_{1j} + h_{4j}) - \sum_{k=1}^{j-1} p_{(i-1)k} (h_{1k} + h_{4k}), & \text{si } j = 1, \\ \varrho_{(i-1)0} - p_{(i-1)1}, & \text{si } j = 2, \\ 0 & \text{sinon} \end{cases} \quad (10.10)$$

et

$$\beta_j = \begin{cases} d_i & \text{si } j = i, \\ 1 & \text{si } j = i - 1 \quad \& \quad i \neq 2, \\ 0 & \text{sinon} \end{cases} \quad (10.11)$$

### Démonstration

La  $i^{\text{ieme}}$  commande virtuelle, telle qu'elle est donnée par la méthode du backstepping avec observateur (Krstić et al. 1995), s'écrit

$$\begin{aligned}\alpha_{1_i} &= -d_i \varepsilon_{1_i} - \varepsilon_{1_{i-1}} - (h_{1_i} + h_{4_i}) \left[ y_1 - (\hat{x}_{1_i} + \hat{x}_{4_i}) \right] + \xi_{i-1} \\ &= d_i \alpha_{1_{i-1}} - d_i (\hat{x}_{1_i} + \hat{x}_{4_i}) \\ &\quad - (\hat{x}_{1_{i-1}} + \hat{x}_{4_{i-1}}) + \alpha_{i-2} - (h_{1_i} + h_{4_i}) \left[ y_1 - (\hat{x}_{1_i} + \hat{x}_{4_i}) \right] + \xi_{i-1}\end{aligned}$$

Avec la notation de l'équation (10.3), la partie connue de  $\alpha_{1_{i-1}}$  se met sous la forme

$$\begin{aligned}\xi_{i-1} &= \left[ \sum_{j=0}^{i-1} \varrho^{(i-1)_j} s^{j+1} \right] (y_{r_1} - \chi_1) \\ &\quad - \sum_{j=1}^{i-1} p^{(i-1)_j} \left[ \hat{x}_{1_{j+1}} + \hat{x}_{4_{j+1}} - (h_{1_j} + h_{4_j})(\hat{x}_{1_j} + \hat{x}_{4_j}) \right] \\ &\quad + (p^{(i-1)_1} - \varrho^{(i-1)_0})(\hat{x}_{1_2} + \hat{x}_{4_2}) - \left[ \sum_{j=1}^{i-1} p^{(i-1)_j} (h_{1_j} + h_{4_j}) \right] y_1\end{aligned}$$

Remplacée dans l'expression de  $\alpha_{1_i}$ , donne

$$\begin{aligned}\alpha_{1_i} &= -d_i (\hat{x}_{1_i} + \hat{x}_{4_i}) - (\hat{x}_{1_{i-1}} + \hat{x}_{4_{i-1}}) - d_i \sum_{j=1}^{i-1} p^{(i-1)_j} (\hat{x}_{1_j} + \hat{x}_{4_j}) \\ &\quad + \left[ \sum_{j=0}^{i-1} \varrho^{(i-1)_j} s^{j+1} + \sum_{j=0}^{i-2} \varrho^{(i-2)_j} s^j + d_i \sum_{j=0}^{i-1} \varrho^{(i-1)_j} s^j \right] (y_{r_1} - \chi_1) \\ &\quad - \sum_{j=1}^{i-2} p^{(i-2)_j} (\hat{x}_{1_j} + \hat{x}_{4_j}) + \left[ \sum_{j=1}^{i-1} p^{(i-1)_j} (h_{1_j} + h_{4_j}) + h_{1_i} + h_{4_i} \right] (\hat{x}_{1_i} + \hat{x}_{4_i}) \\ &\quad + (p^{(i-1)_1} - \varrho^{(i-1)_0})(\hat{x}_{1_2} + \hat{x}_{4_2}) - \sum_{j=1}^{i-1} p^{(i-1)_j} (\hat{x}_{1_{j+1}} + \hat{x}_{4_{j+1}}) - \left[ h_{1_i} + h_{4_i} \right. \\ &\quad \left. + \sum_{j=1}^{i-1} p^{(i-1)_j} (h_{1_j} + h_{4_j}) + d_i (\varrho^{(i-1)_0} - p^{(i-1)_1}) + (\varrho^{(i-2)_0} - p^{(i-2)_1}) \right] y_1\end{aligned}$$

qui, une fois arrangée sous la forme (10.3), donne les équations (10.4) à (10.11). ■



### Commande finale

La dernière commande virtuelle est donnée par

$$\alpha_{1_n} = \left[ \sum_{j=0}^n \varrho_{1_n, j} s^j \right] (y_{r_1} - \chi_1) - \sum_{j=1}^n p_{n_j} (\hat{x}_{1_j} + \hat{x}_{4_j}) + (p_{n_1} - \varrho_{n_0}) y_1$$

qui, en utilisant les équations du procédé, donne comme expression de  $v_1 + v_4$

$$v_1 + v_4 = \alpha_{1_n} + \varsigma_{1_n} + \sum_{j=1}^n \left[ a_{1_j} \hat{x}_{1_j} + a_{4_j} \hat{x}_{4_j} \right] \quad (10.12)$$

Avec comme fonction de Lyapunov,

$$V_{1_n} = \sum_{j=1}^n \left[ \frac{1}{2} \varepsilon_{1_j}^2 + \frac{1}{m_j} \bar{\mathbf{x}}^T \mathbf{P} \bar{\mathbf{x}} \right]$$

la commande composée (10.12) garantit la convergence vers zéro de toutes les variables d'erreur  $\varepsilon_{1_j}$ . L'application de (10.12) aux équations du procédé donne

$$\begin{aligned} \dot{x}_{1_n} + \dot{x}_{4_n} &= v_1 + v_4 - \sum_{j=1}^n (a_{1_j} x_{1_j} + a_{4_j} x_{4_j}) \\ &= \mathcal{F}(s)(y_{r_1} - \chi_1) + (p_{n_1} - \varrho_{n_0}) y_1 - \sum_{j=1}^n p_{1_n, j} (\hat{x}_{1_j} + \hat{x}_{4_j}) - \sum_{j=1}^n (a_{1_j} \bar{x}_{1_j} + a_{4_j} \bar{x}_{4_j}) \end{aligned}$$

que l'on peut réécrire

$$\mathcal{J}(s) y_1 = \mathcal{F}(s)(y_{r_1} - \chi_1) + \sum_{j=1}^n \left[ (p_{n_j} - a_{1_j}) \bar{x}_{1_j} + (p_{n_j} - a_{4_j}) \bar{x}_{4_j} \right]$$

où

$$\begin{aligned} \mathcal{J}(s) &= s^n + \sum_{j=1}^{n-1} p_{n_{(j+1)}} s^j + \varrho_{n_0} \\ \mathcal{F}(s) &= \sum_{j=0}^n \varrho_{n_j} s^j \end{aligned}$$

De la même manière, le second sous-système donne

$$v_2 + v_3 = \mathcal{F}(s)(y_{r_2} - \chi_2) + (p_{n_1} - \varrho_{n_0}) y_2 - \sum_{j=1}^n \left[ (p_{n_j} - a_{2_j}) \hat{x}_{2_j} + (p_{n_j} - a_{3_j}) \hat{x}_{3_j} \right]$$

et la deuxième sortie vérifie

$$\mathcal{J}(s)y_2 = \mathcal{F}(s)(y_{r_2} - \chi_2) + \sum_{j=1}^n \left[ (p_{n_j} - a_{2_j})\tilde{x}_{2_j} + (p_{n_j} - a_{3_j})\tilde{x}_{3_j} \right]$$

**Remarque 10.1 (Stabilité)** — Un choix adéquat de la matrice des gains  $\mathbf{H}$  assure la convergence vers zéro de toutes les erreurs d'estimation des variables d'état du procédé  $\tilde{x}_{1_j}$ ,  $\tilde{x}_{2_j}$ ,  $\tilde{x}_{3_j}$  et  $\tilde{x}_{4_j}$ . En plus, le design récursif par Lyapunov garantit la stabilité entrée/sortie du système en boucle fermée, en assurant la convergence des erreurs régulées  $\varepsilon_{1_1}$  et  $\varepsilon_{2_1}$ . Ces deux propriétés prouvent que la fonction  $\mathcal{J}(s)$ , qui est indépendante des termes introduits  $\chi_1$  et  $\chi_2$ , est hurwitzienne. □

Afin de permettre un retour aux commandes réelles ( $u_1$  et  $u_2$ ), les variables intermédiaires  $v_1$ ,  $v_2$ ,  $v_3$  et  $v_4$  sont remplacées par leurs expressions respectives. Ceci donne lieu au contrôleur multivariable

$$\begin{aligned} B_1(s)u_1 + B_4(s)u_2 &= \alpha_{1_n} + \varsigma_{1_n} + \sum_{j=1}^n \left[ a_{1_j}\hat{x}_{1_j} + a_{4_j}\hat{x}_{4_j} \right] \\ B_3(s)u_1 + B_2(s)u_2 &= \alpha_{2_n} + \varsigma_{2_n} + \sum_{j=1}^n \left[ a_{2_j}\hat{x}_{2_j} + a_{3_j}\hat{x}_{3_j} \right] \end{aligned}$$

Jusqu'ici les termes additionnels  $\chi_1$  et  $\chi_2$  n'ont eu aucun rôle dans le design. À ce stade de la procédure, il est possible de choisir leurs valeurs afin de briser la dépendance entre  $u_1$  et  $u_2$ .

## 10.2 Décentralisation

Afin de rendre possible la séparation des deux commandes, l'expression (10.12) est réécrite sous la forme

$$\begin{aligned} v_1 &= -v_4 + \alpha_{1_n} + \varsigma_{1_n} + \sum_{j=1}^n (a_{1_j}\hat{x}_{1_j} + a_{4_j}\hat{x}_{4_j}) \\ &= -v_4 + (p_{n_1} - \varrho_{n_0})y_1 + \sum_{j=1}^n a_{4_j}\hat{x}_{4_j} + \mathcal{F}(s)(y_{r_1} - \chi_1) - \sum_{j=1}^n \left[ (p_{n_j} - a_{1_j})\hat{x}_{1_j} + p_{n_j}\hat{x}_{4_j} \right] \end{aligned}$$

Un choix intéressant de  $\chi_1$  est mis en évidence, si l'expression de  $v$  est réécrite sous la forme

$$v_1 = \mathcal{F}(s)y_{r_1} - \sum_{j=1}^n (p_{n_j} - a_{1_j})\hat{x}_{1_j} + (p_{n_1} - \varrho_{n_0})y_1 \\ + h_{4_n} \left[ y_1 - (\hat{x}_{1_1} + \hat{x}_{4_1}) \right] - \dot{\hat{x}}_{4_n} - \mathcal{F}(s)\chi_1 - \sum_{j=2}^n p_{n_j}\hat{x}_{4_j} - p_{n_1}\hat{x}_{4_1}$$

En effet, il suffit à présent de prendre

$$\chi_1 = \frac{1}{\mathcal{F}(s)} \left[ h_{4_n}(\hat{x}_{1_1} + \hat{x}_{4_1} - y_1) - \dot{\hat{x}}_{4_n} - \sum_{j=2}^n p_{n_j}\hat{x}_{4_j} \right]$$

pour obtenir comme la commande filtrée

$$v_1 = \mathcal{F}(s)y_{r_1} + (p_{n_1} - \varrho_{n_0})y_1 - \sum_{j=1}^n (p_{n_j} - a_{1_j})\hat{x}_{1_j} - p_{n_1}\hat{x}_{4_1}$$

Un traitement similaire appliqué au second sous-système donne, pour les vraies commandes,

$$u_1 = \frac{1}{B_1(s)} \left[ \mathcal{F}(s)y_{r_1} + (p_{n_1} - \varrho_{n_0})y_1 - \sum_{j=1}^n (p_{n_j} - a_{1_j})\hat{x}_{1_j} - p_{n_1}\hat{x}_{4_1} \right] \quad (10.13)$$

$$u_2 = \frac{1}{B_2(s)} \left[ \mathcal{F}(s)y_{r_2} + (p_{n_1} - \varrho_{n_0})y_2 - \sum_{j=1}^n (p_{n_j} - a_{2_j})\hat{x}_{2_j} - p_{n_1}\hat{x}_{3_1} \right] \quad (10.14)$$

qui sont deux contrôleurs scalaires complètement indépendants, où chacun dépend uniquement de l'information locale.

### Conditions de stabilité

L'application de (10.13) et (10.14) aux équations du procédé donne comme équations, en boucle fermée,

$$\dot{\hat{x}}_{1_n} = - \sum_{j=1}^n a_{1_j}x_{1_j} + \mathcal{F}(s)y_{r_1} - \sum_{j=1}^n (p_{n_j} - a_{1_j})\hat{x}_{1_j} + (p_{n_1} - \varrho_{n_0})y_1 - p_{n_1}\hat{x}_{4_1} \\ = - \sum_{j=1}^n p_{n_j}x_{1_j} + \mathcal{F}(s)y_{r_1} - \sum_{j=1}^n (p_{n_j} - a_{1_j})\hat{x}_{1_j} + (p_{n_1} - \varrho_{n_0})y_1 - p_{n_1}\hat{x}_{4_1}$$

que l'on peut mettre sous la forme

$$\begin{aligned}\mathcal{J}(s)x_{1_1} &= \mathcal{F}(s)y_{r_1} - \sum_{j=1}^n (p_{n_j} - a_{1_j})\tilde{x}_{1_j} + (p_{n_1} - \varrho_{n_0})x_{4_1} - p_{n_1}\tilde{x}_{4_1} \\ &= \mathcal{F}(s)y_{r_1} - \varrho_{n_0}x_{4_1} - \sum_{j=1}^n (p_{n_j} - a_{1_j})\tilde{x}_{1_j} + p_{n_1}\tilde{x}_{4_1}\end{aligned}$$

Si les termes d'erreur sont groupés ensemble, on obtient

$$x_{1_1} = \frac{\mathcal{F}(s)}{\mathcal{J}(s)}y_{r_1} - \frac{\varrho_{n_0}}{\mathcal{J}(s)}x_{4_1} + \frac{1}{\mathcal{J}(s)} \left[ p_{n_1}\tilde{x}_{4_1} - \sum_{j=1}^n (p_{n_j} - a_{1_j})\tilde{x}_{1_j} \right] \quad (10.15)$$

De la même manière,

$$x_{2_1} = \frac{\mathcal{F}(s)}{\mathcal{J}(s)}y_{r_2} - \frac{\varrho_{n_0}}{\mathcal{J}(s)}x_{3_1} + \frac{1}{\mathcal{J}(s)} \left[ p_{n_1}\tilde{x}_{3_1} - \sum_{j=1}^n (p_{n_j} - a_{2_j})\tilde{x}_{2_j} \right] \quad (10.16)$$

La combinaison de (10.15), (10.16) et des équations du procédé donne

$$\begin{aligned}\left[ 1 - \mathcal{G}(s) \right] x_{1_1} &= \frac{\mathcal{F}(s)}{\mathcal{J}(s)}y_{r_1} - \frac{\varrho_{n_0}\mathcal{F}G_4}{\mathcal{J}^2G_2}y_{r_2} + \tilde{\Xi}_1 \\ \left[ 1 - \mathcal{G}(s) \right] x_{2_1} &= \frac{\mathcal{F}(s)}{\mathcal{J}(s)}y_{r_2} - \frac{\varrho_{n_0}\mathcal{F}G_3}{\mathcal{J}^2G_1}y_{r_1} + \tilde{\Xi}_2\end{aligned}$$

où

$$\mathcal{G} = \frac{\varrho_{n_0}^2}{\mathcal{J}^2(s)}\mathcal{Q}_I$$

et

$$\begin{aligned}\tilde{\Xi}_1 &= \frac{1}{\mathcal{J}(s)} \left[ p_{n_1}\tilde{x}_{4_1} - \sum_{j=1}^n (p_{n_j} - a_{1_j})\tilde{x}_{1_j} - \frac{\varrho_{n_0}}{\mathcal{J}(s)} \frac{G_4(s)}{G_2(s)} \left( p_{n_1}\tilde{x}_{3_1} - \sum_{j=1}^n (p_{n_j} - a_{2_j})\tilde{x}_{2_j} \right) \right] \\ \tilde{\Xi}_2 &= \frac{1}{\mathcal{J}(s)} \left[ p_{n_1}\tilde{x}_{3_1} - \sum_{j=1}^n (p_{n_j} - a_{2_j})\tilde{x}_{2_j} - \frac{\varrho_{n_0}}{\mathcal{J}(s)} \frac{G_3(s)}{G_1(s)} \left( p_{n_1}\tilde{x}_{4_1} - \sum_{j=1}^n (p_{n_j} - a_{1_j})\tilde{x}_{1_j} \right) \right]\end{aligned}$$

dépendent uniquement des erreurs d'estimation et convergent par conséquent vers zéro. Étant donné que  $\mathcal{J}(s)$  est hurwitzien, le théorème du **petit gain**, donne comme condition de stabilité

$$\|\mathcal{G}(s)\|_{\infty} < 1$$

Comme en plus,  $\mathcal{J}(s)$  a pour gain  $\varrho_{n_0}$ , il suffit de le choisir sans résonance pour réduire la condition de stabilité précédente à

$$\|\mathcal{Q}_I(s)\|_{\infty} < 1$$

### Performances en poursuite

Afin d'être en mesure de prouver que la vraie erreur de poursuite  $e_1$  est nulle en régime permanent, il suffit, comme dans le cas sans observateur, de montrer que le choix de  $\chi_1$  garantit

$$\lim_{t \rightarrow \infty} \chi_1 = 0$$

En effet, à partir de la définition de  $\chi_1$ , on a

$$\begin{aligned} \lim_{t \rightarrow \infty} \chi_1 &= -\frac{1}{\mathcal{F}(s)} \lim_{t \rightarrow \infty} \left[ h_{4n}(\hat{x}_{11} + \hat{x}_{41} - y_1) - \dot{\hat{x}}_{4n} - \sum_{j=2}^n p_n \hat{x}_{4j} \right] \\ &= 0 \end{aligned}$$

ce qui montre que  $e_1(\infty)$  est également nulle, puisque

$$\begin{aligned} \lim_{t \rightarrow \infty} e_1 &= \lim_{t \rightarrow \infty} \varepsilon_{11} - \lim_{t \rightarrow \infty} \chi_1 \\ &= -\lim_{t \rightarrow \infty} \chi_1 \end{aligned}$$

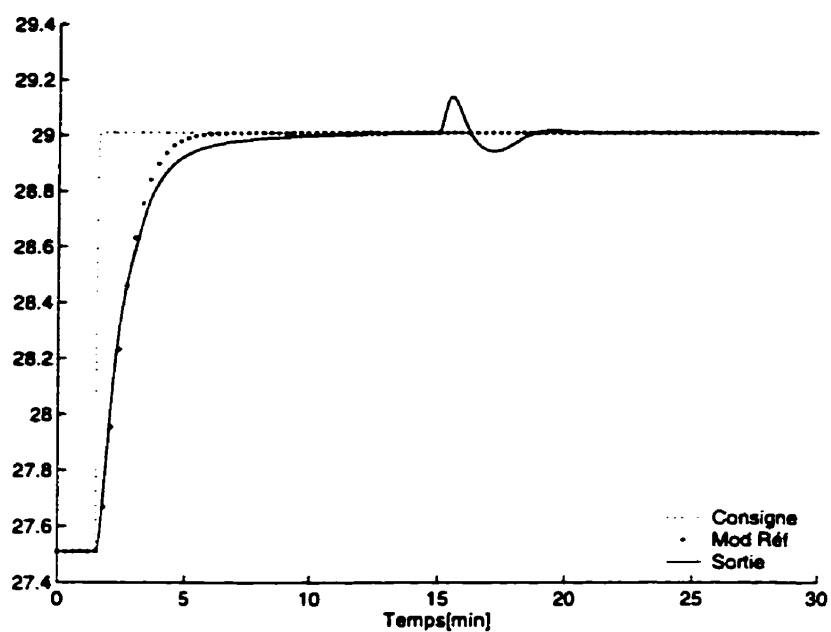
étant donnée que la convergence de  $\varepsilon_{11}$  est garantie par la fonction de Lyapunov.

### 10.3 Simulation

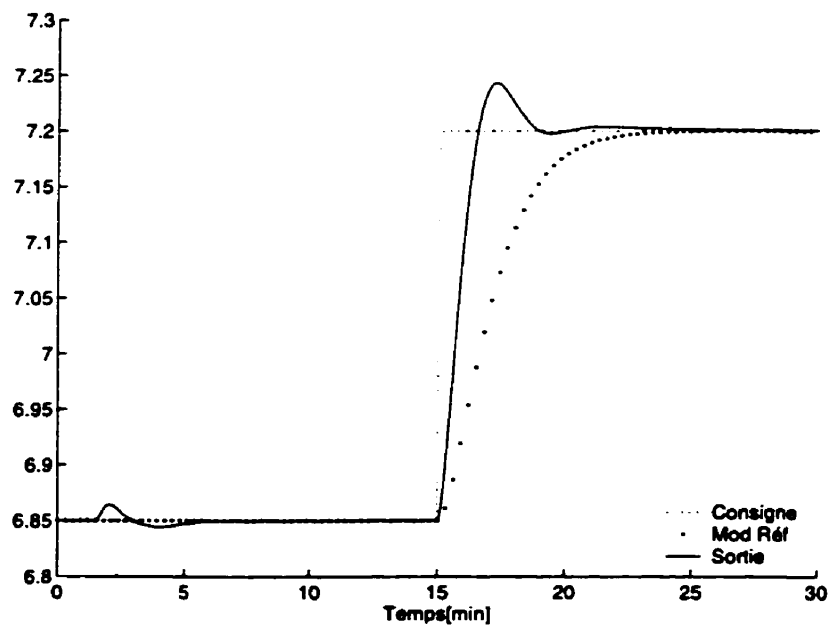
Les deux contrôleurs scalaires obtenus sont utilisés pour la commande décentralisée du simulateur du banc de flottation présenté dans le chapitre 9. Les réponses aux deux changements de consigne sont données par les figures 10.1 & 10.2. À cause de l'effet filtre de l'observateur, les commandes (figures 10.2 a & b) sont plus douces, surtout en régulation, que celles obtenues par le retour d'état complet (figures 9.2 a & b), pour des performances comparables.

### 10.4 Conclusion

L'approche avec observateur a été élaborée dans le seul but d'éviter la mesure des variables d'état du procédé. L'introduction de l'observateur dans la méthode de l'erreur modifiée s'est, cependant, traduite par une augmentation de la complexité de la

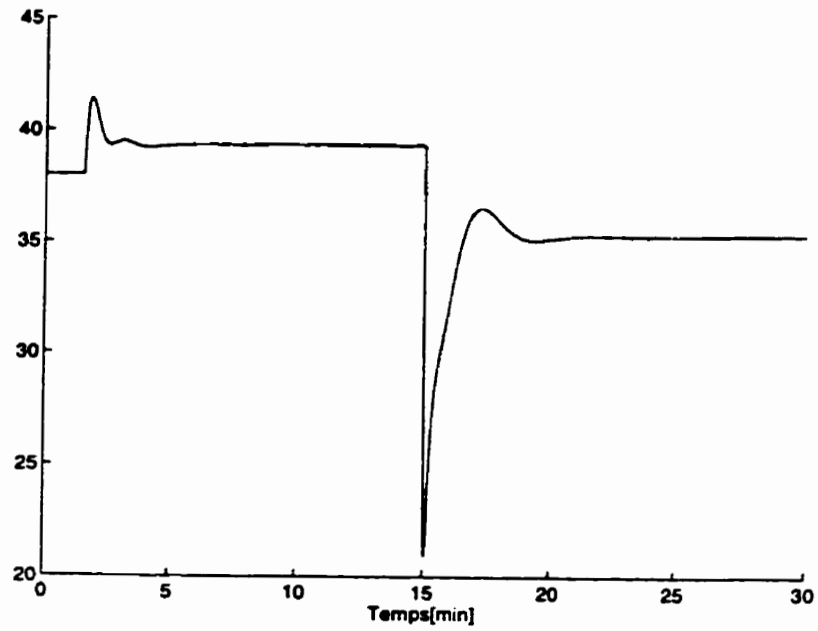


(a) Teneur du concentré [%]

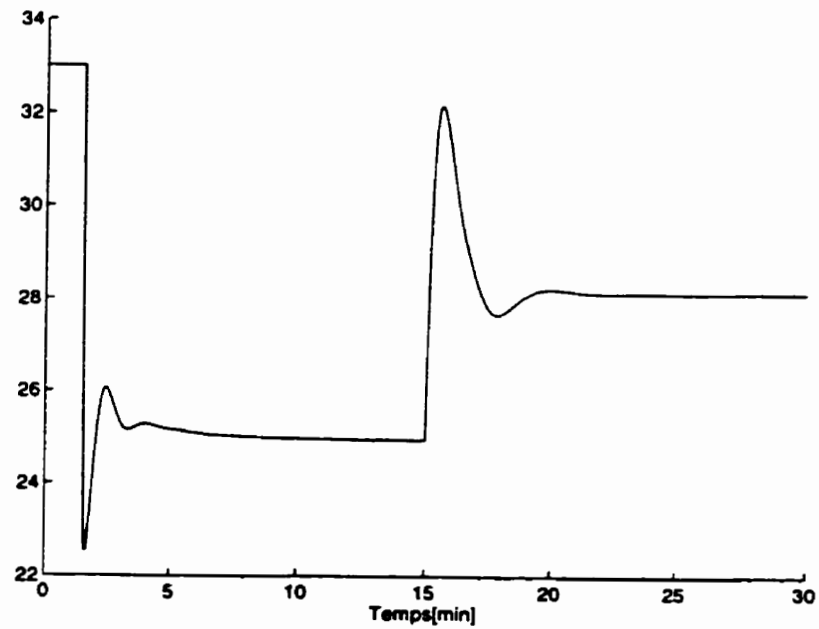


(b) Teneur du rejet [%]

FIG. 10.1: Retour d'état avec observateur — Sorties



(a) Débit d'air [%]



(b) Débit du collecteur [%]

FIG. 10.2: Retour d'état avec observateur — Commandes

procédure design. À cause la présence des erreurs d'estimation des variables d'état, des preuves de stabilité plus élaborées sont devenues nécessaires. En plus, étant donné que l'observateur est inévitablement multivariable, la décentralité n'est que partielle, puisqu'elle ne concerne que l'aspect contrôle.

Des nouvelles méthodes, qui offrent des compromis intéressants entre la complexité du design et les exigences en termes de mesure, sont présentées dans le chapitre 11. Le but ultime, des modifications apportées, demeure l'obtention d'une loi de commande décentralisée par retour de sortie.



# Chapitre 11

## Commande par retour de sortie

Ce chapitre apporte des réponses aux problèmes que pose, pour la méthode de l'erreur modifiée, la mesure des variables d'état du procédé. L'idée de base consiste à réduire, aux seules sorties, le nombre des variables d'état nécessaires à l'implantation du contrôleur obtenu. Cette réduction passe évidemment par l'élimination, de l'équation du contrôleur, des composantes qui dépendent des variables non disponibles. Toutefois, comme il est montré dans la section 11.1, une telle élimination peut engendrer des erreurs non nulles en régime permanent. Deux solutions sont alors proposées pour surmonter cette limitation. La première, présentée dans la section 11.2, assure des erreurs nulles au prix de mesures supplémentaires. Elle ne permet, par conséquent, qu'un retour d'état partiel. Une solution plus intéressante, au problème des erreurs résiduelles, consiste à profiter des performances du **backstepping** avec action intégrale<sup>1</sup>. Doter chaque contrôleur local d'un intégrateur permet d'éliminer les erreurs en régime permanent, en se contentant, comme mesures, uniquement des sorties. Cette solution par retour de sortie sera abordée dans la section 11.3.

### 11.1 Modification directe

Afin d'être en mesure de réduire le nombre des variables utilisées dans l'expression de  $v_1$ , donnée par (9.16), les parties qui dépendent des états internes du procédé doivent

---

<sup>1</sup>Voir le chapitre 6 pour plus de détail sur **backstepping** avec action intégrale.

être éliminées. Pour ce faire,  $v_1$  est réécrite sous la forme

$$\begin{aligned}
 v_1 &= \left[ s^n + \sum_{j=0}^{n-1} a_{1j} s^j \right] x_{11} + \mathcal{F}(s)(y_{r1} - \chi_1 - y_1) \\
 &= \mathcal{F}(s)y_{r1} + \left[ s^n + \sum_{j=0}^{n-1} a_{1j} s^j - \mathcal{F}(s) \right] x_{11} - \mathcal{F}(s)(\chi_1 + x_{41}) \\
 &= \mathcal{F}(s)y_{r1} + \left[ s^n + \sum_{j=0}^{n-1} a_{1j} s^j - \mathcal{F}(s) \right] x_{11} \\
 &\quad + \left[ s^n + \sum_{j=0}^{n-1} a_{1j} s^j - \mathcal{F}(s) \right] y_1 - \left[ s^n + \sum_{j=0}^{n-1} a_{1j} s^j - \mathcal{F}(s) \right] y_1 - \mathcal{F}(s)(\chi_1 + x_{41})
 \end{aligned}$$

ou encore

$$v_1 = \mathcal{F}(s)(y_{r1} - y_1) + \left[ s^n + \sum_{j=0}^{n-1} a_{1j} s^j \right] y_1 - \mathcal{F}(s) \left[ \chi_1 + \frac{1}{\mathcal{F}(s)} \left( s^n + \sum_{j=0}^{n-1} a_{1j} s^j \right) x_{41} \right]$$

Le terme  $\chi_1$  peut, maintenant, être choisi tel que

$$\begin{aligned}
 \chi_1 &= -\frac{1}{\mathcal{F}(s)} \left[ s^n + \sum_{j=0}^{n-1} a_{1j} s^j \right] x_{41} \\
 &= -\frac{s^n + a_{1n-1} s^{n-1} + \dots + a_{11} s + a_{10}}{s^n + \rho_{n-1} s^{n-1} + \dots + \rho_1 s + \rho_0} x_{41}
 \end{aligned} \tag{11.1}$$

Ce choix permet de réduire la commande filtrée  $v_1$  à

$$v_1 = \mathcal{F}(s)(y_{r1} - y_1) + \left[ s^n + \sum_{j=0}^{n-1} a_{1j} s^j \right] y_1$$

Les deux contrôleurs finals s'écrivent, dans ces conditions,

$$u_1 = \frac{1}{B_1(s)} \left[ \mathcal{F}(s)(y_{r1} - y_1) + \left( s^n + \sum_{j=0}^{n-1} a_{1j} s^j \right) y_1 \right] \tag{11.2}$$

$$u_2 = \frac{1}{B_2(s)} \left[ \mathcal{F}(s)(y_{r2} - y_2) + \left( s^n + \sum_{j=0}^{n-1} a_{2j} s^j \right) y_2 \right] \tag{11.3}$$

où seules les variables de sortie  $y_1$  et  $y_2$  sont nécessaires à la réalisation des lois de commande (11.2) et (11.3). L'application de ces dernières au procédé donne, en boucle

fermée,

$$\begin{aligned} s^n x_{1_1} &= \mathcal{F}(s)(y_{r_1} - y_1) + \left[ s^n + \sum_{j=0}^{n-1} a_{1_j} s^j \right] y_1 - \sum_{j=0}^{n-1} a_{1_j} s^j x_{1_1} \\ &= \mathcal{F}(s)(y_{r_1} - y_1) + \left[ s^n + \sum_{j=0}^{n-1} a_{1_j} s^j \right] x_{4_1} + s^n x_{1_1} \end{aligned}$$

et

$$\begin{aligned} s^n x_{2_1} &= \mathcal{F}(s)(y_{r_2} - y_2) + \left[ s^n + \sum_{j=0}^{n-1} a_{2_j} s^j \right] y_2 - \sum_{j=0}^{n-1} a_{2_j} s^j x_{2_1} \\ &= \mathcal{F}(s)(y_{r_2} - y_2) + \left[ s^n + \sum_{j=0}^{n-1} a_{2_j} s^j \right] x_{3_1} + s^n x_{2_1} \end{aligned}$$

Après quelques simplifications, ces deux équations peuvent être mises sous la forme

$$\begin{aligned} \mathcal{F}(s)(y_{r_1} - y_1) + \left[ s^n + \sum_{j=0}^{n-1} a_{1_j} s^j \right] x_{4_1} &= 0 \\ \mathcal{F}(s)(y_{r_2} - y_2) + \left[ s^n + \sum_{j=0}^{n-1} a_{2_j} s^j \right] x_{3_1} &= 0 \end{aligned}$$

Un développement, similaire à celui de la commande par retour d'état, donne comme résultat final

$$\begin{aligned} \left[ 1 - \mathcal{G}(s) \right] x_{1_1} &= y_{r_1} - \frac{G_4(s)}{G_2(s)} \left[ 1 - \frac{1}{\mathcal{F}(s)} \left( s^n + \sum_{j=0}^{n-1} a_{1_j} s^j \right) \right] y_{r_2} \\ \left[ 1 - \mathcal{G}(s) \right] x_{2_1} &= y_{r_2} - \frac{G_3(s)}{G_1(s)} \left[ 1 - \frac{1}{\mathcal{F}(s)} \left( s^n + \sum_{j=0}^{n-1} a_{2_j} s^j \right) \right] y_{r_1} \end{aligned}$$

avec

$$\mathcal{G}(s) = \mathcal{Q}_f(s) \left[ 1 - \frac{1}{\mathcal{F}(s)} \left( s^n + \sum_{j=0}^{n-1} a_{1_j} s^j \right) \right] \left[ 1 - \frac{1}{\mathcal{F}(s)} \left( s^n + \sum_{j=0}^{n-1} a_{2_j} s^j \right) \right]$$

Une condition suffisante, pour la stabilité interne, est encore une fois donnée par le théorème du petit gain, i.e.

$$\|\mathcal{G}(s)\|_{\infty} < 1$$

**Remarque 11.1 (Comparaison)** — Jusqu'ici, la commande par retour de sortie présente de fortes similitudes avec celle par retour d'état, tout en présentant l'avantage de ne nécessiter que très peu de variables à mesurer<sup>2</sup>. On peut s'interroger, pourquoi un tel choix n'a été utilisé directement dans la solution présentée dans le chapitre 9. Afin d'être en mesure de répondre à cette question, les performances en régime permanent des contrôleurs (11.2) et (11.3) sont examinées.

□

## Performances

Pour le premier sous-système, l'erreur de poursuite est toujours donnée par

$$\begin{aligned} e_1 &= y_1 - y_{r_1} \\ &= \varepsilon_{1_1} - \chi_1 \end{aligned}$$

qui, évidemment, diffère de l'erreur modifiée  $\varepsilon_{1_1}$ , dont la procédure du **backstepping** assure la régulation. Même si la convergence vers zéro de cette dernière est garantie par la fonction de Lyapunov, la vraie erreur de poursuite  $e_1$  reste non nulle, à moins que  $\chi_1$  s'annule. Sous les conditions de stabilité précédemment données, le comportement en régime permanent de  $e_1$  serait

$$\begin{aligned} \lim_{t \rightarrow \infty} e_1 &= \lim_{t \rightarrow \infty} \chi_1 \\ &= - \lim_{t \rightarrow \infty} \left[ \frac{1}{\mathcal{F}(s)} \left( s^n + \sum_{j=0}^{n-1} a_{1_j} s^j \right) x_{4_1} \right] \\ &= - \frac{a_{1_0}}{\varrho_0} x_{4_1} \Big|_{t \rightarrow \infty} \\ &= - \frac{a_{1_0} b_{4_0}}{a_{4_0} \varrho_0} u_2 \Big|_{t \rightarrow \infty} \\ &\neq 0 \end{aligned}$$

## Simulation

Le résultats de l'application, au procédé de flottation, du contrôleur à retour de sortie obtenu sont donnés par les figures 11.1 (a) & (b) et 11.2 (a) & (b). On y remarque,

<sup>2</sup>Seules les variables de sorties sont mesurées

effectivement, la présence d'erreurs résiduelles en régime permanent. Le fait que ces erreurs soient non nulles est principalement dû à la présence du terme  $a_{1_0}x_{4_1}$ , dans l'expression de la commande  $u_1$ . Si la fonction de transfert  $G_1(s)$  contient au moins une intégration, i.e.  $a_{1_0} = 0$ , ou  $G_4(s)$  contient au moins une dérivation, i.e.  $b_{4_0} = 0$ , le contrôleur à retour de sortie (11.2) assurera une erreur de poursuite nulle<sup>3</sup>.

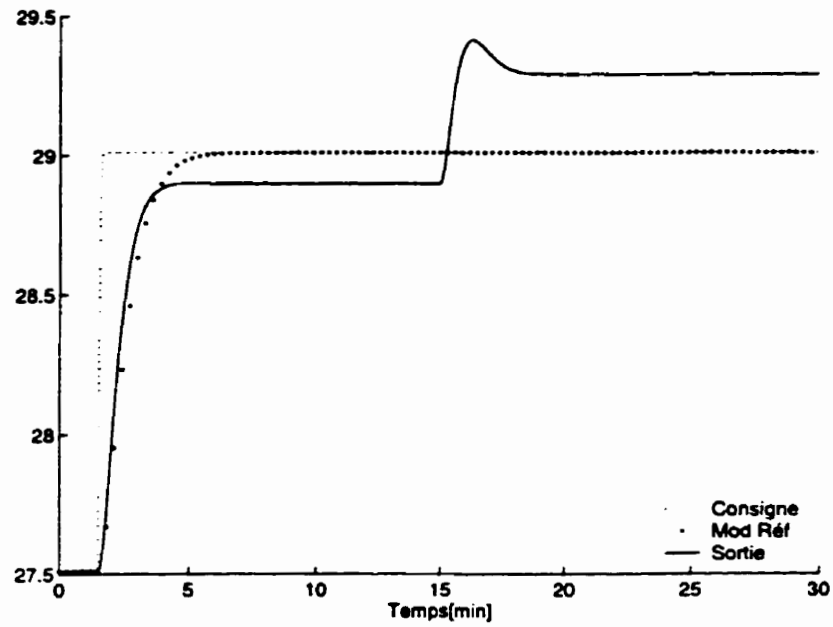
Afin d'éviter de telles conditions, jugées trop restrictives, sans perte de performance, la méthode de design est encore une fois modifiée. L'amélioration apportée passe par l'élimination des termes  $a_{1_0}x_{4_1}$  et  $a_{2_0}x_{3_1}$  des expressions des deux contrôleurs. Deux solutions sont proposées dans ce qui suit.

La première (Benaskeur & Desbiens 1999a) consiste à choisir de nouvelles valeurs pour les termes  $\chi_1$  et  $\chi_2$  qui vont, sous certaines conditions, rendre cette élimination possible. Étant donné que certaines variables d'état restent indispensables à l'implantation de la commande, on parle de commande par retour d'état partiel. La seconde solution (Benaskeur & Desbiens 1999b) consiste à éliminer les termes indésirables d'une manière indirecte, via l'introduction d'actions intégrales dans les contrôleurs scalaires obtenus. L'avantage, avec cette dernière solution, réside dans le fait que la commande obtenue est complètement à retour de sortie.

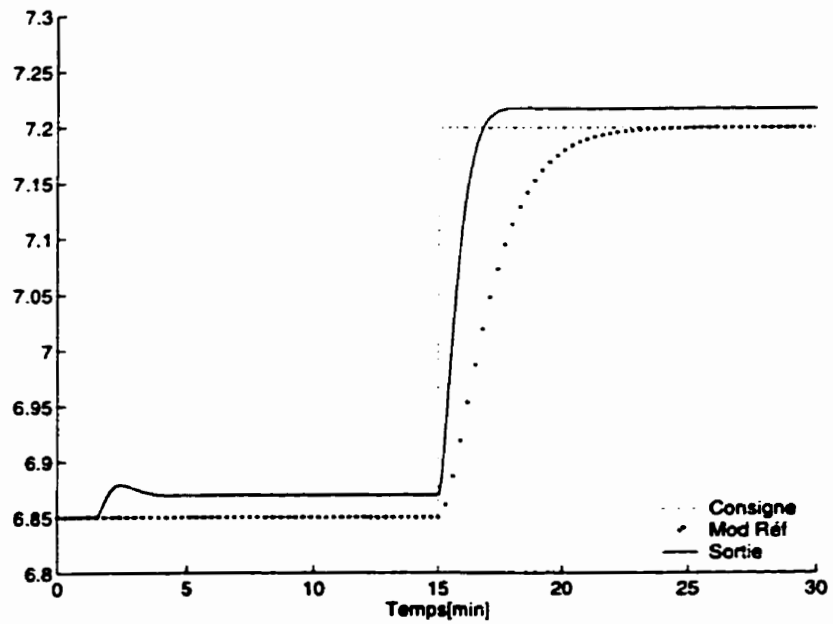
## 11.2 Commande par retour d'état partiel

Cette approche offre une solution intéressante qui garantit des erreurs de poursuite nulles, tout en réduisant le nombre de variables d'état nécessaires à l'implantation des contrôleurs. Cette amélioration est tout simplement rendue possible par un nouveau choix (Benaskeur & Desbiens 1999a) des termes  $\chi_1$  et  $\chi_2$ . Ces derniers deviennent plus évidents, si les expressions des commandes filtrées  $v_1$  et  $v_2$  sont réarrangées. En effet,

<sup>3</sup>Les mêmes commentaires s'appliquent à  $G_2(s)$ ,  $G_3(s)$ ,  $e_{2_1}$ ,  $a_{2_0}$  et  $b_{3_0}$

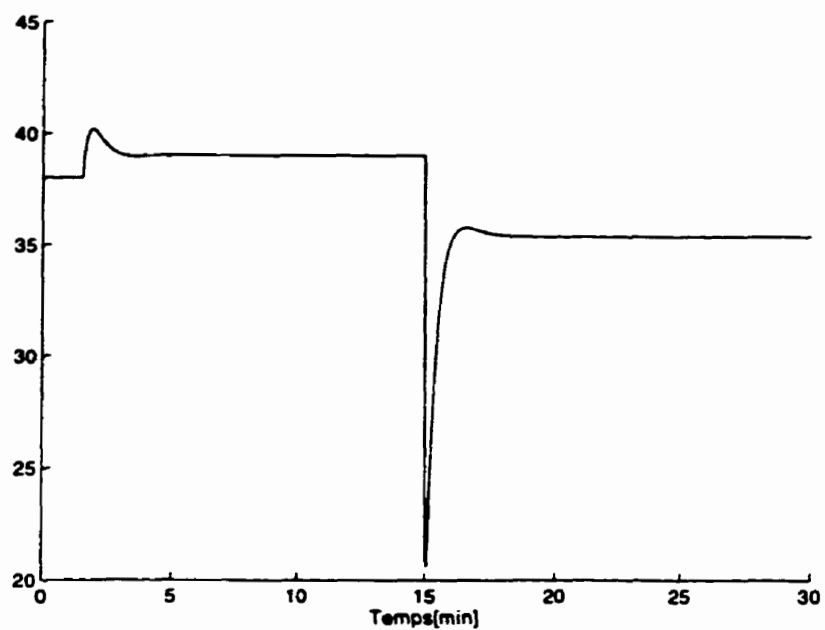


(a) Teneur du concentré [%]

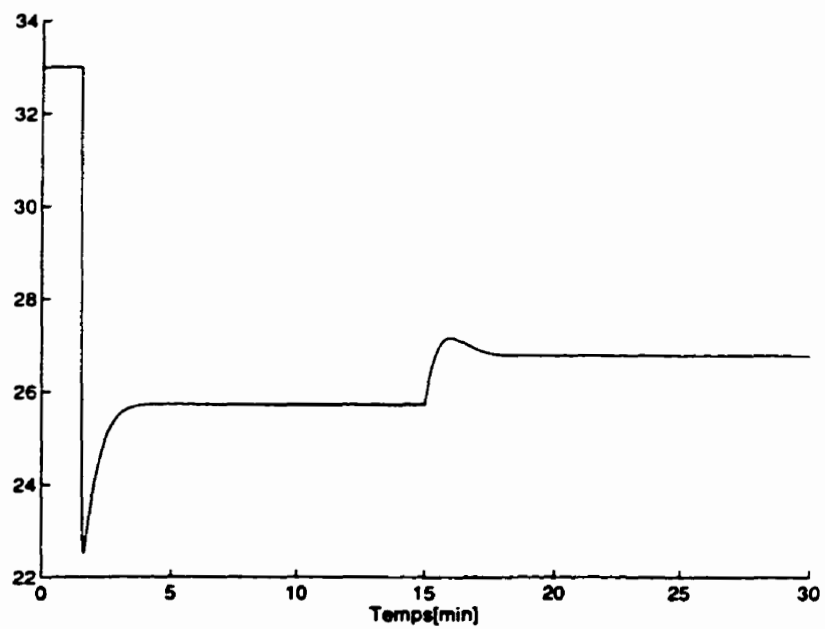


(b) Teneur du rejet [%]

FIG. 11.1: Retour de sortie sans intégrateur — Sorties



(a) Débit d'air [%]



(b) Débit du collecteur [%]

FIG. 11.2: Retour de sortie sans intégrateur — Commandes

si l'expression (9.16) de  $v_1$  est réécrite sous la forme

$$\begin{aligned}
v_1 &= \mathcal{F}(s)y_{r_1} + \left[ s^n + \sum_{j=0}^{n-1} a_{1,j}s^j - \mathcal{F}(s) \right] x_{1_1} - \mathcal{F}(s)(\chi_1 + x_{4_1}) \\
&= \mathcal{F}(s)y_{r_1} + \left[ s^n + \sum_{j=0}^{n-1} a_{1,j}s^j - \mathcal{F}(s) \right] x_{1_1} \\
&\quad + \left[ s^n + \sum_{j=1}^{n-1} a_{1,j}s^j - \mathcal{F}(s) \right] y_1 - \left[ s^n + \sum_{j=1}^{n-1} a_{1,j}s^j - \mathcal{F}(s) \right] y_1 - \mathcal{F}(s)(\chi_1 + x_{4_1}) \\
&= \mathcal{F}(s) \left[ y_{r_1} - y_1 \right] + a_{1_0}x_{1_1} \\
&\quad + \left[ s^n + \sum_{j=1}^{n-1} a_{1,j}s^j \right] y_1 - \left[ s^n + \sum_{j=1}^{n-1} a_{1,j}s^j - \mathcal{F}(s) \right] x_{4_1} - \mathcal{F}(s)(\chi_1 + x_{4_1}) \\
&= \mathcal{F}(s)(y_{r_1} - y_1) + a_{1_0}x_{1_1} \\
&\quad + \left[ s^n + \sum_{j=1}^{n-1} a_{1,j}s^j \right] y_1 - \mathcal{F}(s) \left[ \chi_1 + \frac{1}{\mathcal{F}(s)} \left( s^n + \sum_{j=1}^{n-1} a_{1,j}s^j \right) x_{4_1} \right]
\end{aligned}$$

Il suffit à présent de choisir

$$\chi_1 = -\frac{1}{\mathcal{F}(s)} \left[ s^n + \sum_{j=1}^{n-1} a_{1,j}s^j \right] x_{4_1}$$

pour obtenir la commande

$$\begin{aligned}
u_1 &= \frac{v_1}{B_1(s)} \\
&= \frac{1}{B_1(s)} \left[ \mathcal{F}(s)(y_{r_1} - y_1) + \left( s^n + \sum_{j=1}^{n-1} a_{1,j}s^j \right) y_1 + a_{1_0}x_{1_1} \right] \quad (11.4)
\end{aligned}$$

**Remarque 11.2 (Mesure)** — En plus de la sortie  $y_1$ , seule la variable d'état  $x_{1_1}$  est requise pour l'implantation de ce contrôleur. Cette mesure est le prix à payer pour assurer une erreur de poursuite nulle. Il faut, cependant, noter que si  $G_1(s)$  comporte au moins une intégration, i.e.  $a_{1_0} = 0$ , cette mesure ne sera plus nécessaire, et la commande se réduit à un retour de sortie.

□



Un calcul identique donne, pour le second contrôleur

$$\begin{aligned} u_2 &= \frac{v_2}{B_2(s)} \\ &= \frac{1}{B_2(s)} \left[ \mathcal{F}(s)(y_{r_2} - y_2) + \left( s^n + \sum_{j=1}^{n-1} a_{2,j} s^j \right) y_2 + a_{2_0} x_{2_1} \right] \end{aligned} \quad (11.5)$$

### Conditions de stabilité

L'application du contrôleur (11.4) au premier sous-système (9.2)–(9.5) donne comme dynamique en boucle fermée

$$\begin{aligned} s^n x_{1_1} &= \mathcal{F}(s)(y_{r_1} - y_1) + \left[ s^n + \sum_{j=1}^{n-1} a_{1,j} s^j \right] y_1 + a_{1_0} x_{1_1} - \sum_{j=0}^{n-1} a_{1,j} s^j x_{1_1} \\ &= \mathcal{F}(s)(y_{r_1} - y_1) + \left[ s^n + \sum_{j=1}^{n-1} a_{1,j} s^j \right] x_{4_1} + s^n x_{1_1} \end{aligned}$$

que l'on peut réécrire

$$\mathcal{F}(s)(y_{r_1} - y_1) + \left[ s^n + \sum_{j=1}^{n-1} a_{1,j} s^j \right] x_{4_1} = 0$$

ou encore

$$\begin{aligned} x_{1_1} &= y_{r_1} - \left[ 1 - \frac{1}{\mathcal{F}(s)} \left( s^n + \sum_{j=1}^{n-1} a_{1,j} s^j \right) \right] x_{4_1} \\ &= y_{r_1} - \left[ 1 - \frac{1}{\mathcal{F}(s)} \left( s^n + \sum_{j=1}^{n-1} a_{1,j} s^j \right) \right] \frac{G_4(s)}{G_2(s)} x_{2_1} \end{aligned}$$

Pour le second sous-système, un développement similaire donne

$$\begin{aligned} x_{2_1} &= y_{r_2} - \left[ 1 - \frac{1}{\mathcal{F}(s)} \left( s^n + \sum_{j=1}^{n-1} a_{2,j} s^j \right) \right] x_{3_1} \\ &= y_{r_2} - \left[ 1 - \frac{1}{\mathcal{F}(s)} \left( s^n + \sum_{j=1}^{n-1} a_{2,j} s^j \right) \right] \frac{G_3(s)}{G_1(s)} x_{1_1} \end{aligned}$$

L'équation en boucle fermée de système global devient alors

$$\begin{aligned} \left[1 - \mathcal{G}(s)\right] x_{1_1} &= y_{r_1} - \frac{G_4(s)}{G_2(s)} \left[1 - \frac{1}{\mathcal{F}(s)} \left(s^n + \sum_{j=1}^{n-1} a_{1_j} s^j\right)\right] y_{r_2} \\ \left[1 - \mathcal{G}(s)\right] x_{2_1} &= y_{r_2} - \frac{G_3(s)}{G_1(s)} \left[1 - \frac{1}{\mathcal{F}(s)} \left(s^n + \sum_{j=1}^{n-1} a_{2_j} s^j\right)\right] y_{r_1} \end{aligned}$$

où, cette fois,  $\mathcal{G}(s)$  est donné par

$$\mathcal{G}(s) = Q_I(s) \left[1 - \frac{1}{\mathcal{F}(s)} \left(s^n + \sum_{j=1}^{n-1} a_{1_j} s^j\right)\right] \left[1 - \frac{1}{\mathcal{F}(s)} \left(s^n + \sum_{j=1}^{n-1} a_{2_j} s^j\right)\right]$$

Ceci mène à la condition de stabilité

$$\|\mathcal{G}(s)\|_{\infty} < 1$$

### Performances

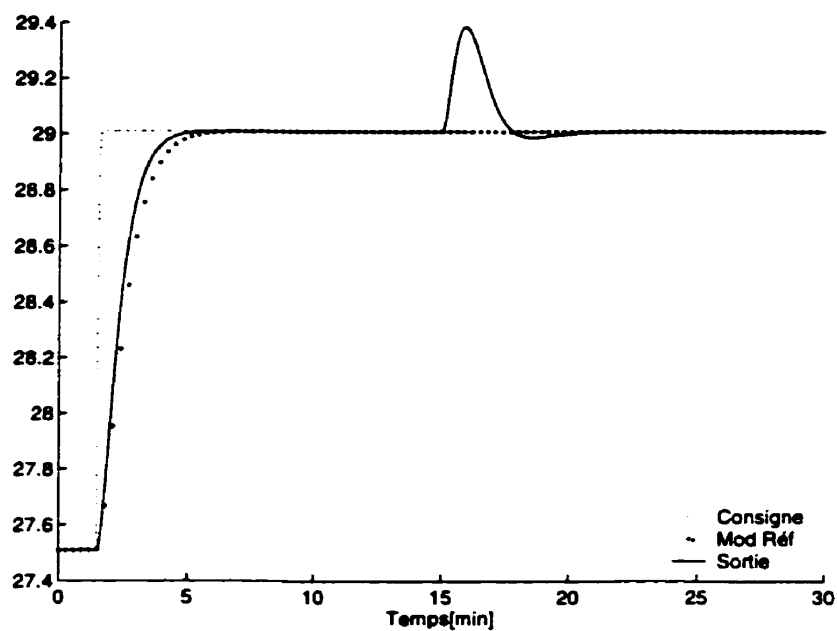
L'erreur de poursuite est toujours donnée par

$$\begin{aligned} e_1 &= y_1 - y_{r_1} \\ &= \varepsilon_{1_1} - \chi_1 \\ &= \varepsilon_{1_1} + \frac{1}{\mathcal{F}(s)} \left[s^n + \sum_{j=1}^{n-1} a_{1_j} s^j\right] x_{4_1} \end{aligned}$$

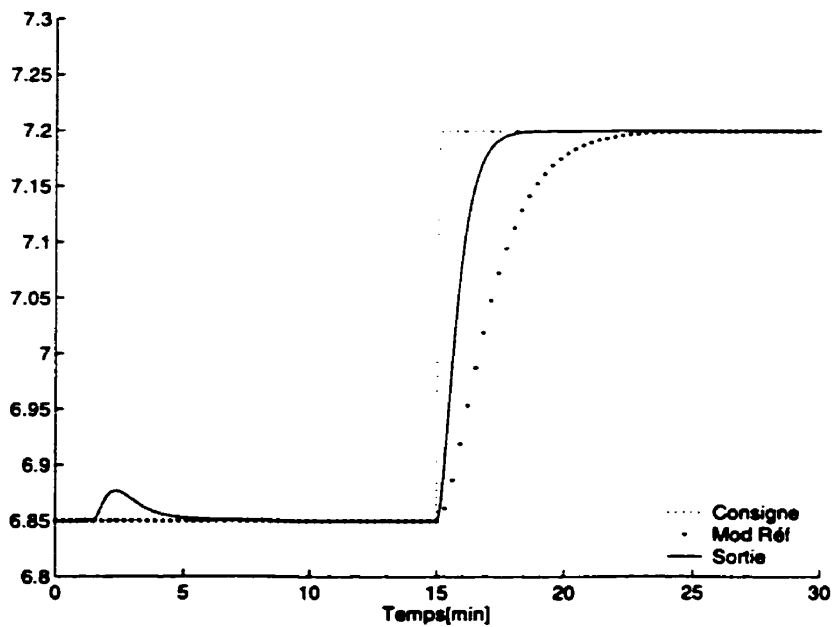
ce qui, sous les conditions de stabilité données, se traduit en régime permanent par

$$\begin{aligned} \lim_{t \rightarrow \infty} \chi_1 &= - \lim_{t \rightarrow \infty} \left[ \frac{1}{\mathcal{F}(s)} \left(s^n + \sum_{j=1}^{n-1} a_{1_j} s^j\right) x_{4_1} \right] \\ &= 0 \end{aligned}$$

Ce résultat, intéressant, se constate sur les réponses données par les figures 11.3 (a) & (b). Le nombre des variables mesurées peut, cependant, être davantage réduit, pour des performances comparables. Ceci passe par l'introduction d'actions intégrales dans les expressions des contrôleurs locaux  $u_1$  et  $u_2$ .

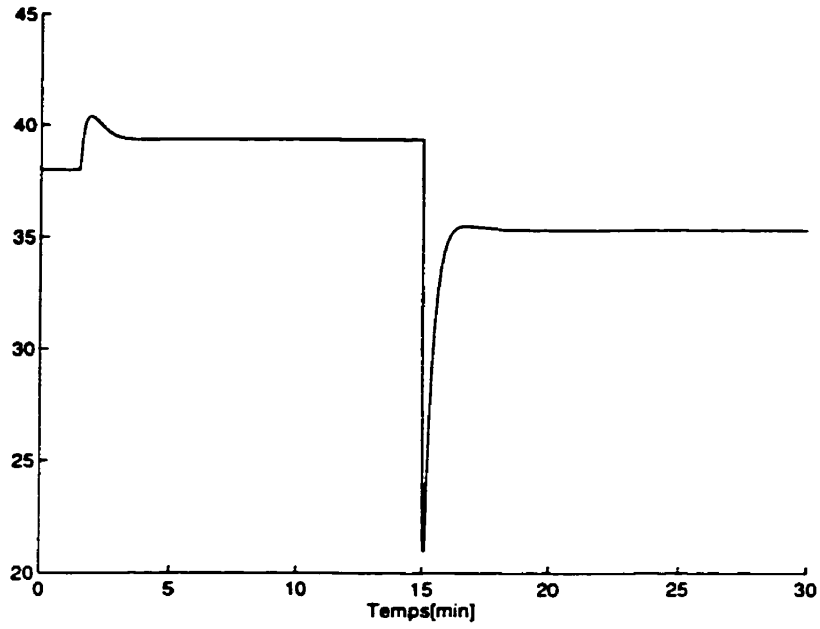


(a) Teneur du concentré [%]

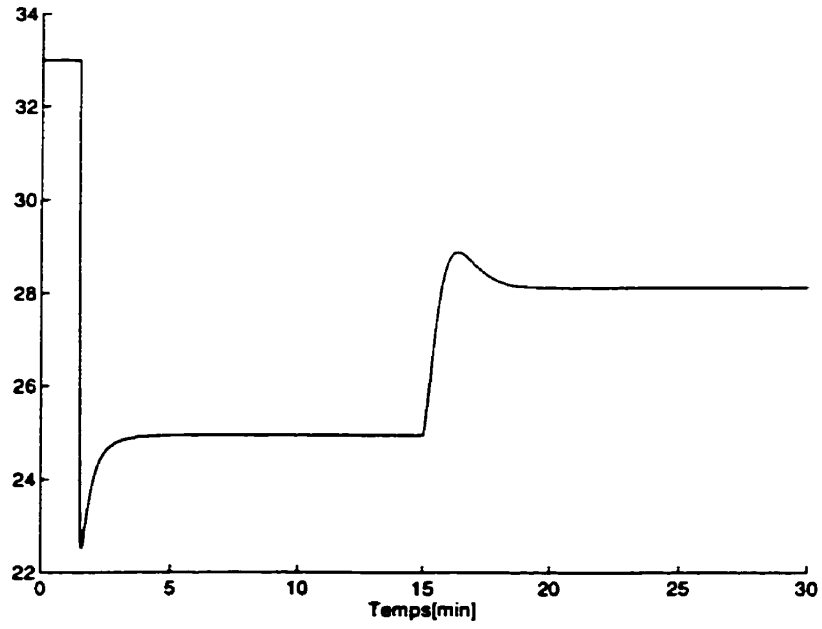


(b) Teneur du rejet [%]

FIG. 11.3: Retour d'état partiel — Sorties



(a) Débit d'air [%]



(b) Débit du collecteur [%]

FIG. 11.4: Retour d'état partiel — Commandes

## 11.3 Design avec action intégrale

Bien que cette modification ait été introduite pour la première fois (Benaskeur & Desbiens 1999b) dans le but de palier aux limitations de la commande par retour de sortie (erreur de poursuite non nulle), elle sera d'abord présentée dans le cadre général de la commande par retour d'état. Par la suite les changements appropriés seront apportés, afin de la ramener au cas, plus intéressant, de la commande par retour de sortie avec action intégrale.

### 11.3.1 Commande par retour d'état

L'idée principale de la méthode consiste à introduire, d'une manière virtuelle, des intégrateurs dans les fonctions de transfert des branches directes  $G_1(s)$  et  $G_2(s)$ . Par la suite, ces intégrateurs seront transférés dans les régulateurs, afin de garantir des erreurs nulles en régime permanent. Pour rendre possible cette manipulation, une représentation d'état quasi minimale ( $n+1$  états) est utilisée pour chacune des fonctions de transfert  $G_j(s)$

$$\begin{aligned}\dot{x}_{j_1} &= x_{j_2} \\ \dot{x}_{j_2} &= x_{j_3} \\ &\vdots \\ \dot{x}_{j_n} &= x_{j_{n+1}} \\ \dot{x}_{j_{n+1}} &= -a_{j_0}x_{j_2} - \dots - a_{j_{n-1}}x_{j_{n+1}} + w_j\end{aligned}$$

où les  $w_j$  sont définies par

$$w_j = \begin{cases} B_j(s)\dot{u}_j & \text{si } j = 1, 2, \\ B_j(s)\dot{u}_{j-2} & \text{si } j = 3, 4 \end{cases}$$

La suite du design est identique à ce qui a été présenté auparavant. Il faut toutefois noter que, dans la procédure récursive, une étape supplémentaire sera nécessaire, à cause de la représentation non minimale utilisée. Le design s'effectuera alors en  $n+1$  étapes, au lieu de  $n$ . La définition des erreurs modifiées, les fonctions Lyapunov ainsi

que les commandes virtuelles restent inchangées. La seule différence réside la dernière commande virtuelle, dont la valeur ( $\alpha_{1_{n+1}}$ ) est donnée par

$$\begin{aligned}
 \alpha_{1_{n+1}} &= w_1 \\
 &= \sum_{j=0}^{n-1} a_{1_j} s^{j+1} x_{1_1} - \dot{x}_{4_{n+1}} + \dot{\alpha}_{1_n} - \varepsilon_{1_n} - k_{1_{n+1}} \varepsilon_{1_{n+1}} \\
 &= \mathcal{F}(s) y_{r_1} + \left[ s^{n+1} + \sum_{j=0}^{n-1} a_{1_j} s^{j+1} - \mathcal{F}(s) \right] x_{1_1} - \mathcal{F}(s) (\chi_1 + x_{4_1}) \\
 &\quad + \left[ s^{n+1} + \sum_{j=0}^{n-1} a_{1_j} s^{j+1} - \mathcal{F}(s) \right] y_1 - \left[ s^{n+1} + \sum_{j=0}^{n-1} a_{1_j} s^{j+1} - \mathcal{F}(s) \right] y_1
 \end{aligned}$$

que l'on peut réécrire

$$\begin{aligned}
 \alpha_{1_{n+1}} &= \mathcal{F}(s) y_{r_1} + \\
 &\quad \left[ s^{n+1} + \sum_{j=0}^{n-1} a_{1_j} s^{j+1} - \mathcal{F}(s) \right] y_1 - \mathcal{F}(s) \left[ \chi_1 + \frac{s}{\mathcal{F}(s)} \left( s^n + \sum_{j=0}^{n-1} a_{1_j} s^j \right) \right] \quad (11.6)
 \end{aligned}$$

où cette fois le polynôme  $\mathcal{F}(s)$  est d'ordre  $n + 1$  et non pas  $n$ .

### Décentralisation

Le choix, des termes  $\chi_1$  et  $\chi_2$ , qui permet de décentraliser la loi de commande est très similaire à celui obtenu en l'absence de l'action intégrale. La seule différence réside dans l'ordre du filtre  $\mathcal{M}(s)$  utilisé. En effet, avec le choix

$$\begin{aligned}
 \chi_1 &= (\mathcal{M} - 1) x_{4_1} \\
 &= - \frac{s^{n+1} + \varrho_n s^n + \dots + \varrho_1 s}{s^{n+1} + \varrho_n s^n + \dots + \varrho_1 s + \varrho_0} x_{4_1}
 \end{aligned}$$

la commande filtrée  $w_1$  devient

$$\begin{aligned}
 w_1 &= \mathcal{F}(s) y_{r_1} - \varrho_0 y_1 + \left[ s^{n+1} + \sum_{j=0}^{n-1} a_{1_j} s^{j+1} - \mathcal{F}(s) \right] x_{1_1} \\
 &= \varrho_0 (y_{r_1} - y_1) + \left[ s^{n+1} + \sum_{i=1}^n \varrho_i s^i \right] y_{r_1} - \sum_{i=1}^n (\varrho_i - a_{1_{i-1}}) s^i x_{1_1}
 \end{aligned}$$

et

$$w_2 = \varrho_0 (y_{r_2} - y_2) + \left[ s^{n+1} + \sum_{i=1}^n \varrho_i s^i \right] y_{r_2} - \sum_{i=1}^n (\varrho_i - a_{2_{i-1}}) s^i x_{2_1}$$

### Conditions de stabilité

Une démarche identique à celle effectuée dans le cas sans intégrateur donne comme équations en boucle fermée

$$\begin{cases} \left[1 - \mathcal{G}(s)\right] x_{1i} = y_{r1} - \frac{G_3(s)}{G_2(s)} \mathcal{M}(s) y_{r2} \\ \left[1 - \mathcal{G}(s)\right] x_{2i} = y_{r2} - \frac{G_4(s)}{G_1(s)} \mathcal{M}(s) y_{r1} \end{cases}$$

avec

$$\mathcal{G}(s) = \mathcal{M}^2(s) \mathcal{Q}_I(s)$$

Or,  $\mathcal{M}(s)$  est choisi de gain unitaire et sans résonance, i.e.

$$\|\mathcal{M}(s)\|_{\infty} = 1$$

La condition de stabilité se réduit, avec ce choix, à

$$\|\mathcal{Q}_I(s)\|_{\infty} < 1$$

La loi de commande décentralisée finale résultante est composée de deux contrôleurs scalaires

$$\begin{aligned} u_1 &= \frac{w_1}{sB_1(s)} \\ &= \frac{1}{B_1(s)} \left[ \left( s^n + \sum_{i=1}^n \varrho_i s^{i-1} \right) y_{r1} - \sum_{i=0}^{n-1} (\varrho_{i+1} - a_{1i}) s^i x_{1i} + \varrho_0 \frac{y_{r1} - y_1}{s} \right] \end{aligned}$$

et

$$\begin{aligned} u_2 &= \frac{w_2}{sB_2(s)} \\ &= \frac{1}{B_2(s)} \left[ \left( s^n + \sum_{i=1}^n \varrho_i s^{i-1} \right) y_{r2} - \sum_{i=0}^{n-1} (\varrho_{i+1} - a_{2i}) s^i x_{2i} + \varrho_0 \frac{y_{r2} - y_2}{s} \right] \end{aligned}$$

où l'on remarque, comme prévu, la présence d'intégrateurs.

## Performances

Un calcul, similaire au cas sans actions intégrales (chapitre 9), permet d'obtenir pour l'erreur de poursuite, en régime permanent,

$$\begin{aligned}\lim_{t \rightarrow \infty} e_1 &= \lim_{t \rightarrow \infty} (y_1 - y_{r_1}) \\ &= \lim_{t \rightarrow \infty} \chi_1 \\ &= 0\end{aligned}$$

### 11.3.2 Version adaptative

Comme pour la version de base de la MEM, une version adaptative de la MEM avec intégrateur peut être obtenue en substituant les paramètres inconnus par leurs estimés respectifs. La commande intermédiaire  $w_1$  s'écrit dans ce cas

$$w_1 = -\varrho_0 x_{4_1} + \left[ s^{n+1} + \sum_{i=0}^n \varrho_i s^i \right] (y_{r_1} - x_{1_1}) + \left[ s^{n+1} + \sum_{i=0}^{n-1} \hat{a}_{1_i} s^{i+1} \right] x_{1_1}$$

qui, une fois remplacée dans les équations du procédé, donne

$$x^{n+1} = -\varrho_0 x_{4_1} + \left[ s^{n+1} + \sum_{i=0}^n \varrho_i s^i \right] (y_{r_1} - x_{1_1}) + \left[ s^{n+1} + \sum_{i=0}^{n-1} \bar{a}_{1_i} s^{i+1} \right] x_{1_1}$$

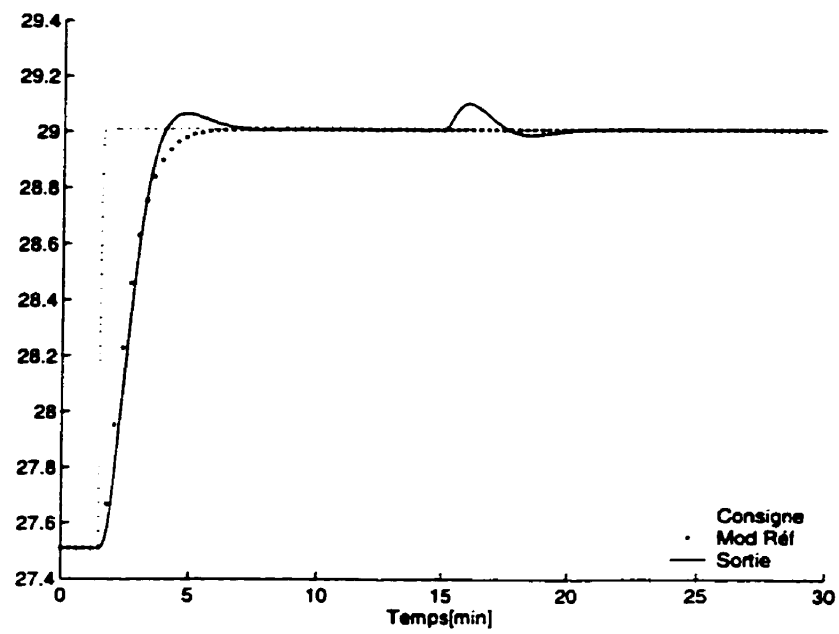
Cette équation peut se réécrire sous la forme

$$\begin{aligned}0 &= \left[ s^{n+1} + \sum_{i=0}^n \varrho_i s^i \right] (y_{r_1} - x_{1_1}) + \left[ \sum_{i=0}^{n-1} \bar{a}_{1_i} s^{i+1} \right] x_{1_1} - \varrho_0 x_{4_1} \\ &= \left[ s^{n+1} + \sum_{i=0}^n \varrho_i s^i \right] y_{r_1} - \left[ s^{n+1} + \sum_{i=1}^n (\varrho_i - \bar{a}_{1_{i-1}}) s^i + \varrho_0 \right] x_{1_1} - \varrho_0 x_{4_1}\end{aligned}$$

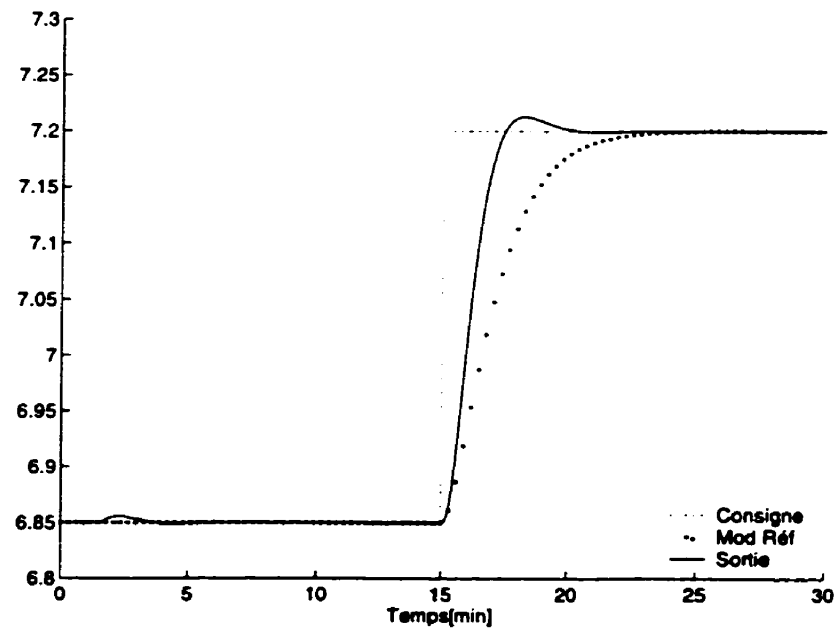
ou encore

$$\begin{aligned}x_{1_1} &= \frac{\varrho_0}{s^{n+1} + (\varrho_n - \bar{a}_{1_{n-1}}) s^n + \dots + (\varrho_1 - \bar{a}_{1_0}) s + \varrho_0} \left[ \frac{y_{r_1}}{\mathcal{F}} - x_{4_1} \right] \\ &\triangleq \tilde{\mathcal{F}}_1 \left[ \frac{y_{r_1}}{\mathcal{F}} - x_{4_1} \right]\end{aligned}$$



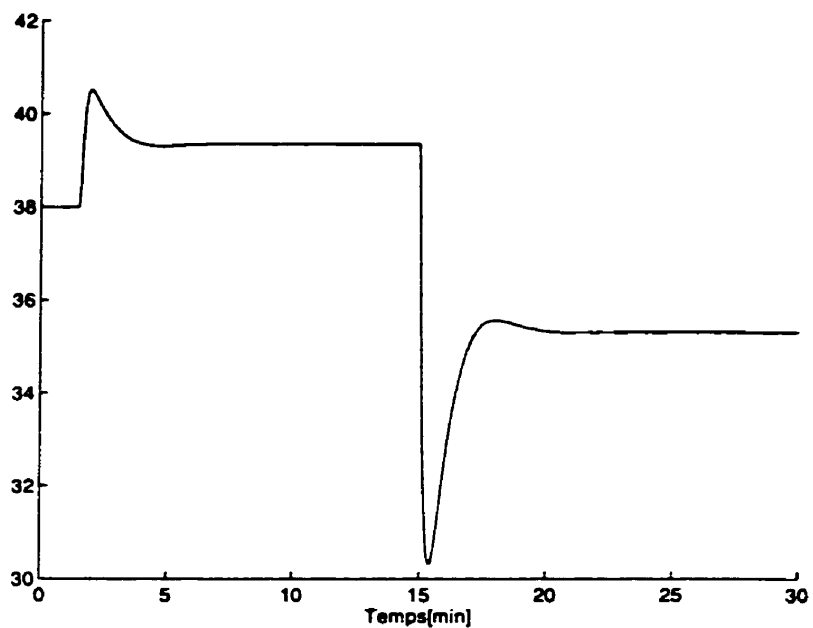


(a) Teneur du concentré [%]

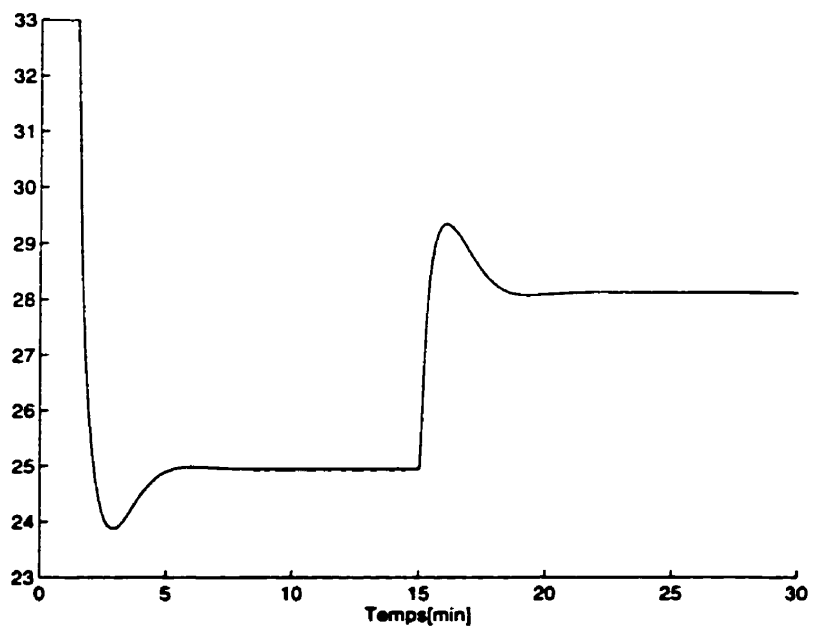


(b) Teneur du rejet [%]

FIG. 11.5: Retour d'état avec intégrateur — Sorties



(a) Débit d'air [%]



(b) Débit du collecteur [%]

FIG. 11.6: Retour d'état avec intégrateur — Commandes

et

$$x_{2_1} = \frac{\varrho_0}{s^{n+1} + (\varrho_n - \bar{a}_{2_{n-1}})s^n + \dots + (\varrho_1 - \bar{a}_{2_0})s + \varrho_0} \left[ \frac{y_{r_2}}{\mathcal{F}} - x_{3_1} \right]$$

$$\triangleq \bar{\mathcal{F}}_2 \left[ \frac{y_{r_2}}{\mathcal{F}} - x_{3_1} \right]$$

Les équations finales résultantes sont données par

$$\left[ 1 - Q_I \bar{\mathcal{F}}_1 \bar{\mathcal{F}}_2 \right] x_{1_1} = \bar{\mathcal{F}}_1 [r_1] - \bar{\mathcal{F}}_1 \bar{\mathcal{F}}_2 \frac{G_4}{G_2} [r_2] \quad (11.7)$$

$$\left[ 1 - Q_I \bar{\mathcal{F}}_1 \bar{\mathcal{F}}_2 \right] x_{2_1} = \bar{\mathcal{F}}_2 [r_2] - \bar{\mathcal{F}}_1 \bar{\mathcal{F}}_1 \frac{G_3}{G_1} [r_1] \quad (11.8)$$

### Condition de stabilité

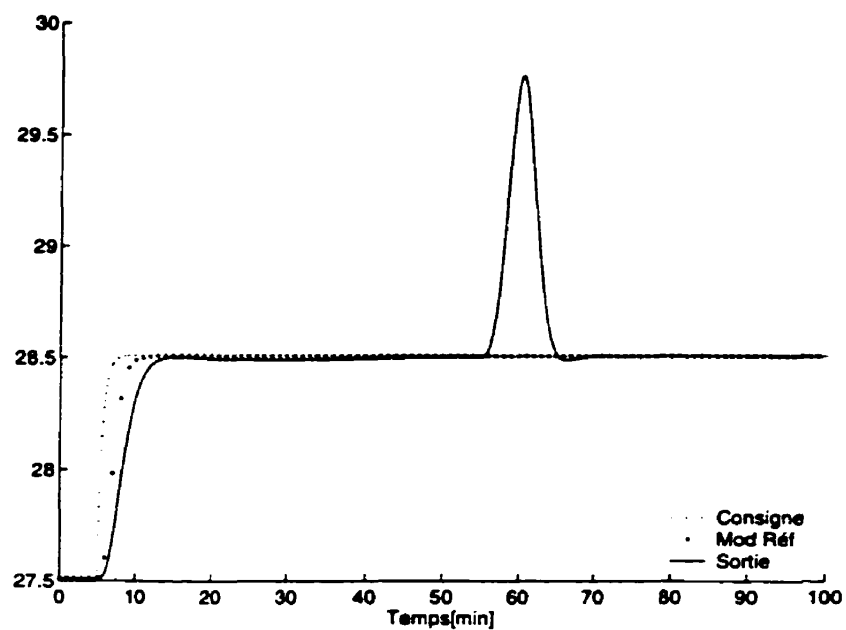
Les équations en boucle fermée (11.7)–(11.8) permettent de déduire la condition de stabilité, en utilisant, comme dans le cas sans action intégrale (section 9.5.1), le théorème du petit gain. Ce dernier donne comme condition

$$\| Q_I(s) \bar{\mathcal{F}}_1(s) \bar{\mathcal{F}}_2(s) \|_{\infty} < 1$$

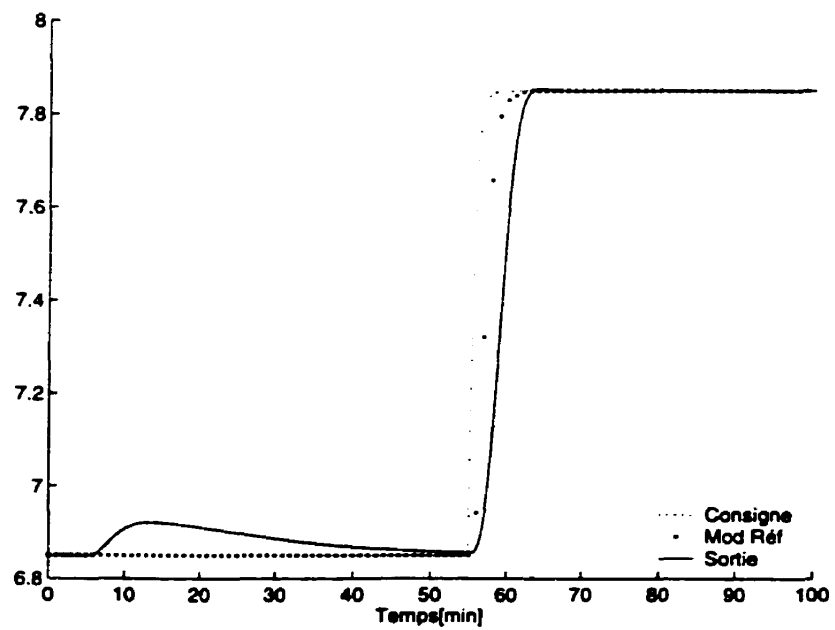
La suite des démonstrations est identique au cas sans action intégrale, présentée dans la section 9.5.1, et ne sera pas donc repris ici. La seule différence réside dans la définition des fonctions  $\mathcal{F}(s)$ ,  $\bar{\mathcal{F}}_1(s)$  et  $\bar{\mathcal{F}}_2(s)$ , qui sont ici d'ordre  $n + 1$ .

### Simulation

Les résultats de l'application de la commande adaptative décentralisée obtenu au procédé (9.27) sont donnés par les figures 11.7 à 11.11. Contrairement à ce qui a été constaté dans le cas monovariante (chapitre 6), l'introduction de l'action intégrale dans le contexte décentralisé n'apporte aucune amélioration notable à la qualité de la convergence de l'estimation. Elle permet cependant de l'éliminer les erreurs résiduelles (figure 9.4 (b)) dont souffrait la version sans action intégrale (section 9.5.1).

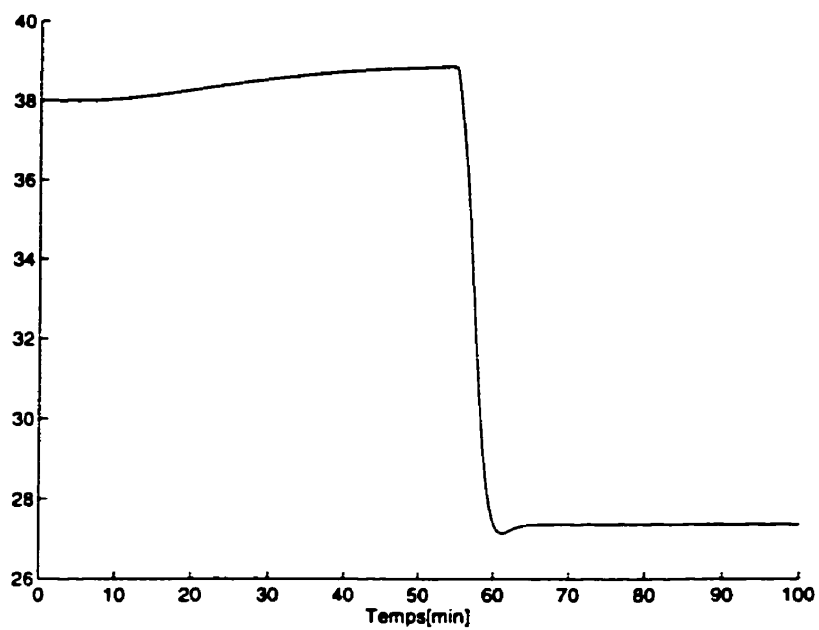


(a) Teneur du concentré [%]

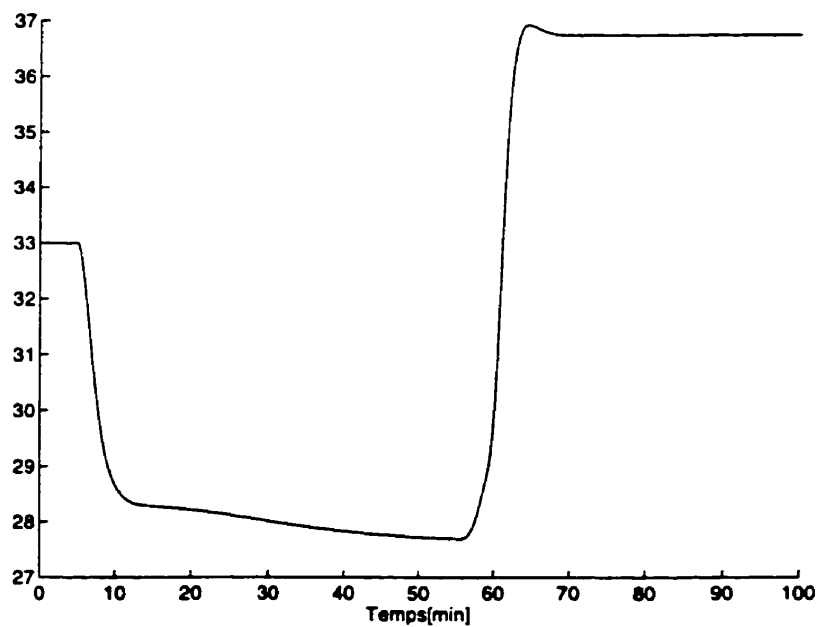


(b) Teneur du rejet [%]

FIG. 11.7: Approche adaptative avec intégrateur — Sorties

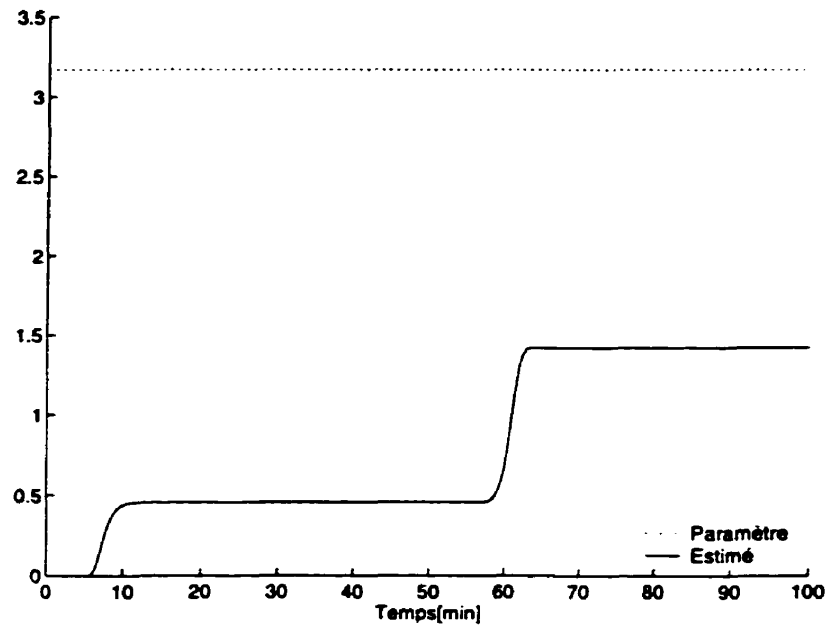
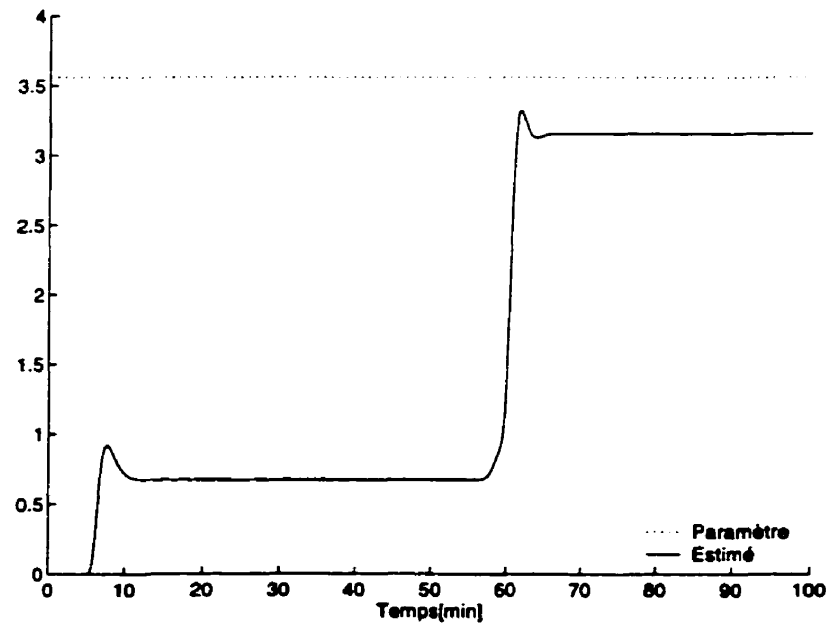


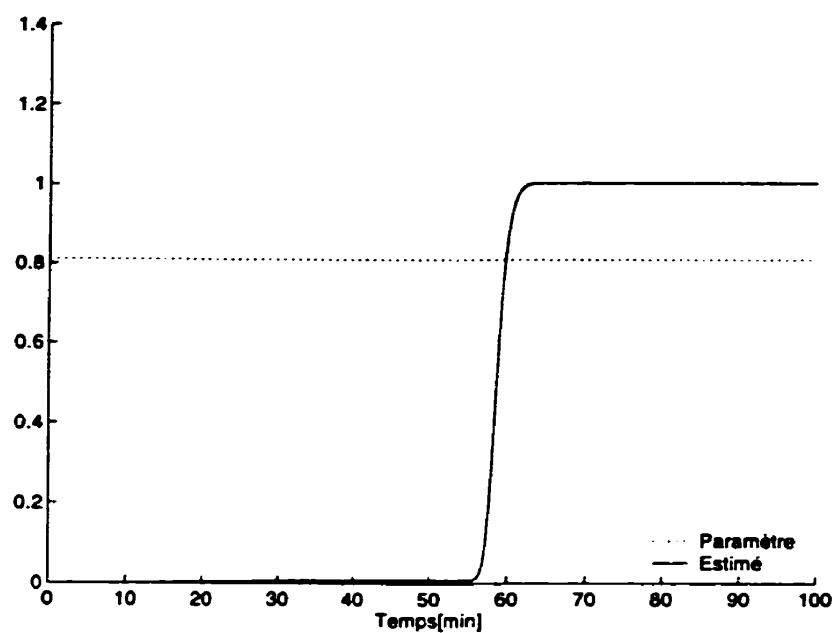
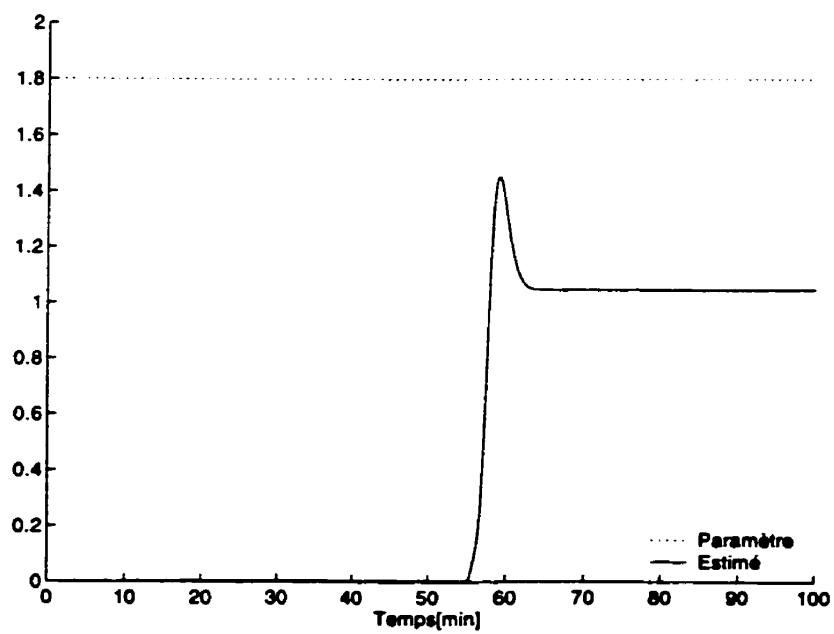
(a) Débit d'air [%]

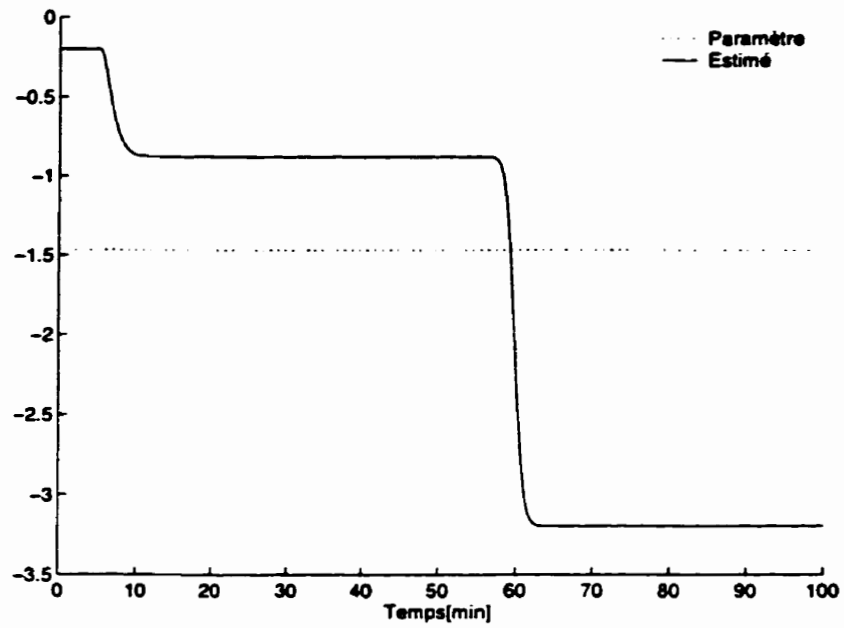
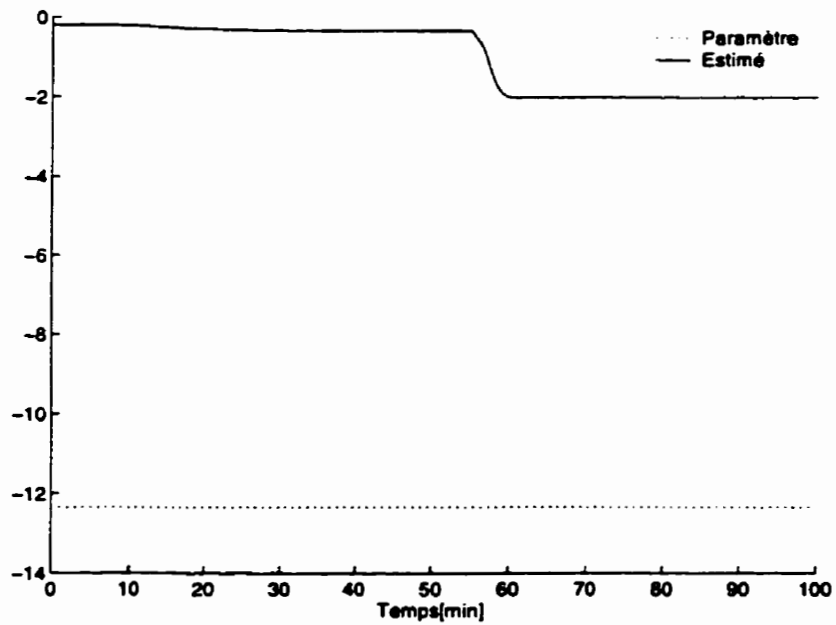


(b) Débit du collecteur [%]

FIG. 11.8: Approche adaptative avec intégrateur — Commandes

(a)  $a_{10}$ (b)  $a_{11}$ FIG. 11.9: Approche adaptative avec intégrateur — Paramètres de  $G_1$

(a)  $a_{20}$ (b)  $a_{21}$ FIG. 11.10: Approche adaptative avec intégrateur — Paramètres de  $G_2$

(a)  $c_1$ (b)  $c_2$ FIG. 11.11: Approche adaptative avec intégrateur — Paramètres  $c_1$  &  $c_2$



### 11.3.3 Commande par retour de sortie

Afin d'être en mesure de réduire la commande de la section précédente au cas d'une commande par retour de sortie, l'expression de la dernière commande virtuelle  $\alpha_{1_{n+1}}$  donnée par l'équation (11.6) est réécrite sous la forme

$$\begin{aligned}
 \alpha_{1_{n+1}} &= \sum_{j=0}^{n-1} a_{1j} s^{j+1} x_{11} - \dot{x}_{4_{n+1}} + \dot{\alpha}_{1_n} - \varepsilon_{1_n} - k_{1_{n+1}} \varepsilon_{1_{n+1}} \\
 &= \mathcal{F}(s) y_{r_1} + \left[ s^{n+1} + \sum_{j=0}^{n-1} a_{1j} s^{j+1} - \mathcal{F} \right] x_{11} - \mathcal{F}(s) (\chi_1 + x_{4_1}) \\
 &\quad + \left[ s^{n+1} + \sum_{j=0}^{n-1} a_{1j} s^{j+1} - \mathcal{F}(s) \right] y_1 - \left[ s^{n+1} + \sum_{j=0}^{n-1} a_{1j} s^{j+1} - \mathcal{F}(s) \right] y_1 \\
 &= \mathcal{F}(s) y_{r_1} + \left[ s^{n+1} + \sum_{j=0}^{n-1} a_{1j} s^{j+1} - \mathcal{F}(s) \right] y_1 - \mathcal{F}(s) \left[ \chi_1 + \frac{s}{\mathcal{F}(s)} \left( s^n + \sum_{j=0}^{n-1} a_{1j} s^j \right) \right]
 \end{aligned}$$

Avec cette notation, le choix suivant de  $\chi_1$

$$\begin{aligned}
 \chi_1 &= -\frac{s}{\mathcal{F}(s)} \left[ s^n + \sum_{j=0}^{n-1} a_{1j} s^j \right] x_{4_1} \\
 &= -\frac{s^{n+1} + a_{1_{n-1}} s^n + \dots + a_{1_0} s}{s^{n+1} + \rho_n s^n + \dots + \rho_1 s + \rho_0} x_{4_1}
 \end{aligned}$$

s'impose. Ce choix réduit l'expression de  $w_1$  à

$$w_1 = \mathcal{F}(s) (y_{r_1} - y_1) + s \left[ s^n + \sum_{j=0}^{n-1} a_{1j} s^j \right] y_1$$

et de la même manière, le deuxième sous-système donne

$$w_2 = \mathcal{F}(s) (y_{r_2} - y_2) + s \left[ s^n + \sum_{j=0}^{n-1} a_{2j} s^j \right] y_2$$

Une simple intégration de ces deux expressions, donne pour les commandes réelles

$$u_1 = \frac{1}{B_1(s)} \left[ \frac{\mathcal{F}(s)}{s} (y_{r_1} - y_1) + \left( s^n + \sum_{j=0}^{n-1} a_{1j} s^j \right) y_1 \right] \quad (11.9)$$

$$u_2 = \frac{1}{B_2(s)} \left[ \frac{\mathcal{F}(s)}{s} (y_{r_2} - y_2) + \left( s^n + \sum_{j=0}^{n-1} a_{2j} s^j \right) y_2 \right] \quad (11.10)$$

que l'on peut réécrire sous la forme

$$u_1 = \frac{1}{B_1} \left[ \varrho_0 \frac{y_{r_1} - y_1}{s} + \left( s^n + \sum_{j=0}^{n-1} \varrho_{j+1} s^j \right) y_{r_1} + \left( s^n + \sum_{j=0}^{n-1} (a_{1j} - \varrho_{j+1}) s^j \right) y_1 \right]$$

$$u_2 = \frac{1}{B_2} \left[ \varrho_0 \frac{y_{r_2} - y_2}{s} + \left( s^n + \sum_{j=0}^{n-1} \varrho_{j+1} s^j \right) y_{r_2} + \left( s^n + \sum_{j=0}^{n-1} (a_{2j} - \varrho_{j+1}) s^j \right) y_2 \right]$$

**Remarque 11.3 (Erreurs)** — La présence d'un intégrateur, dans chacun des deux contrôleurs scalaires obtenus améliore le comportement en régime permanent de la boucle de contrôle. Comme conséquence directe de cette présence, les erreurs  $y_{r_1} - y_1$  et  $y_{r_2} - y_2$  sont garanties de s'annuler. □

### Conditions de stabilité

L'application des lois de commande (11.9) et (11.10) aux équations du procédé, donne comme dynamiques en boucle fermée (pour le premier sous-système)

$$s^n x_{1_1} = - \sum_{j=0}^{n-1} a_{1j} s^j x_{1_1} + \frac{\mathcal{F}(s)}{s} (y_{r_1} - y_1) + \left[ s^n + \sum_{j=0}^{n-1} a_{1j} s^j \right] y_1$$

$$= s^n x_{1_1} + \frac{\mathcal{F}(s)}{s} (y_{r_1} - y_1) + \left[ s^n + \sum_{j=0}^{n-1} a_{1j} s^j \right] x_{4_1}$$

que l'on peut écrire sous la forme

$$\frac{\mathcal{F}(s)}{s} (y_{r_1} - y_1) + \left[ s^n + \sum_{j=0}^{n-1} a_{1j} s^j \right] x_{4_1} = 0$$

Le deuxième sous-système, quant à lui, donne

$$\frac{\mathcal{F}(s)}{s} (y_{r_2} - y_2) + \left[ s^n + \sum_{j=0}^{n-1} a_{2j} s^j \right] x_{3_1} = 0$$

Ces deux équations ensemble aboutissent à

$$\left[ 1 - \mathcal{G}(s) \right] x_{1_1} = y_{r_1} - \frac{G_4(s)}{G_2(s)} \left[ 1 - \frac{1}{\mathcal{F}(s)} \left( s^n + \sum_{j=1}^{n-1} a_{1j} s^j \right) \right] y_{r_2}$$

$$\left[ 1 - \mathcal{G}(s) \right] x_{2_1} = y_{r_2} - \frac{G_3(s)}{G_1(s)} \left[ 1 - \frac{1}{\mathcal{F}(s)} \left( s^n + \sum_{j=1}^{n-1} a_{2j} s^j \right) \right] y_{r_1}$$

avec

$$\mathcal{G}(s) = \mathcal{Q}_I(s) \left[ 1 - \frac{s}{\mathcal{F}(s)} \left( s^n + \sum_{j=1}^{n-1} a_{1j} s^j \right) \right] \left[ 1 - \frac{s}{\mathcal{F}(s)} \left( s^n + \sum_{j=1}^{n-1} a_{2j} s^j \right) \right]$$

Encore une fois, la condition de stabilité est tirée du théorème du petit gain

$$\|\mathcal{G}(s)\|_{\infty} < 1$$

### Performances

L'erreur de poursuite est toujours donnée par

$$\begin{aligned} e_1 &= y_1 - y_{r_1} \\ &= \varepsilon_{1_1} - \chi_1 \\ &= \varepsilon_{1_1} + \frac{s}{\mathcal{F}(s)} \left[ s^n + \sum_{j=0}^{n-1} a_{1j} s^j \right] x_{4_1} \end{aligned}$$

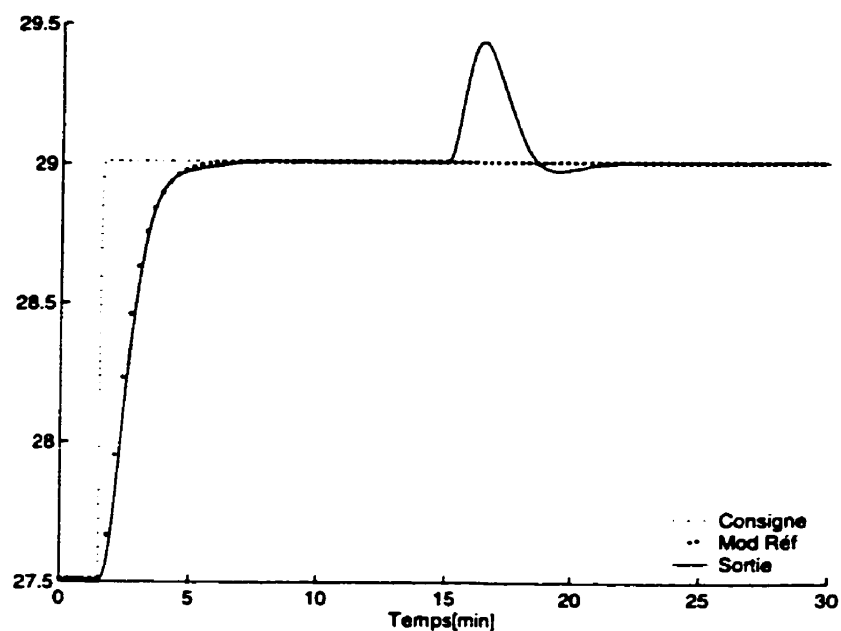
Ce qui montre clairement que

$$\begin{aligned} \lim_{t \rightarrow \infty} e_1 &= - \lim_{t \rightarrow \infty} \chi_1 \\ &= 0 \end{aligned}$$

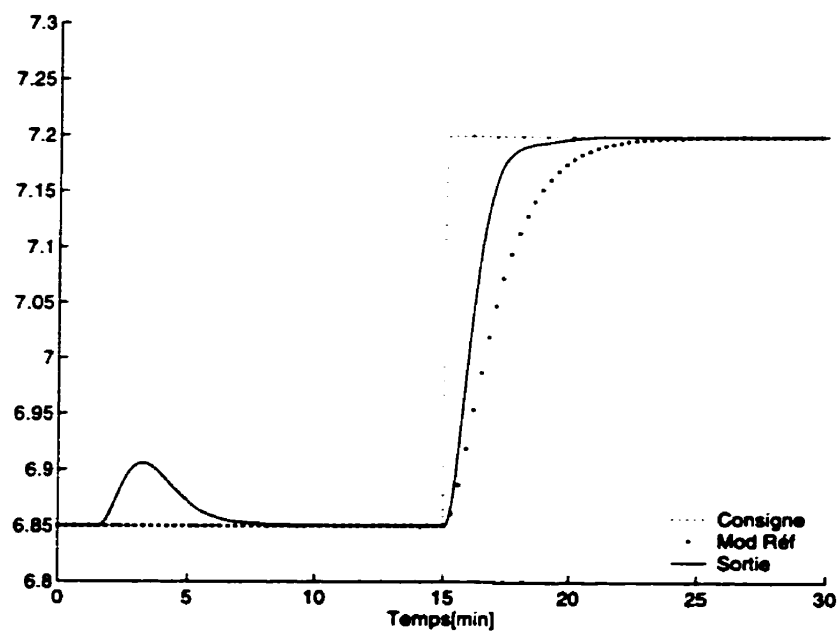
Ce résultat est confirmé par les simulations, dont les résultats sont donnés par les figures 11.12 (a) & (b) et 11.13 (a) & (b).

## 11.4 Conclusion

Le problème de la mesure des variables d'état a été adressé, et différentes solutions ont été proposées. Il a été montré que le passage direct, de la version par retour d'état à une version par de retour de sortie, se traduit par des erreurs de poursuite et/ou régulation non nulles en régime permanent. La MEM par retour d'état partiel offre un bon compromis entre les performances en régime permanent et le nombre de variables mesurées. Ces dernières se limitent aux sorties du procédé dans le cas de la version avec action intégrale. Cette modification (action intégrale) est introduite afin de garantir des erreurs de poursuite nulles.

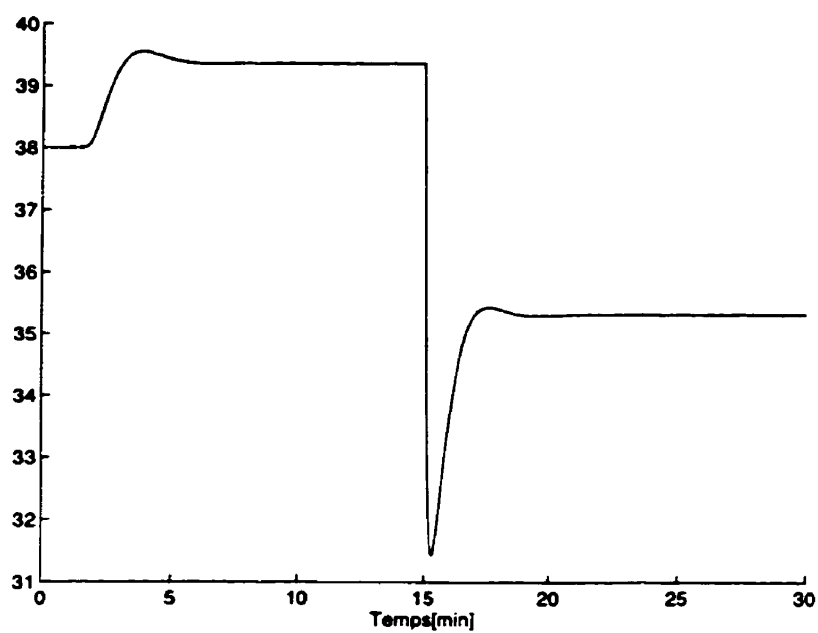


(a) Teneur du concentré [%]

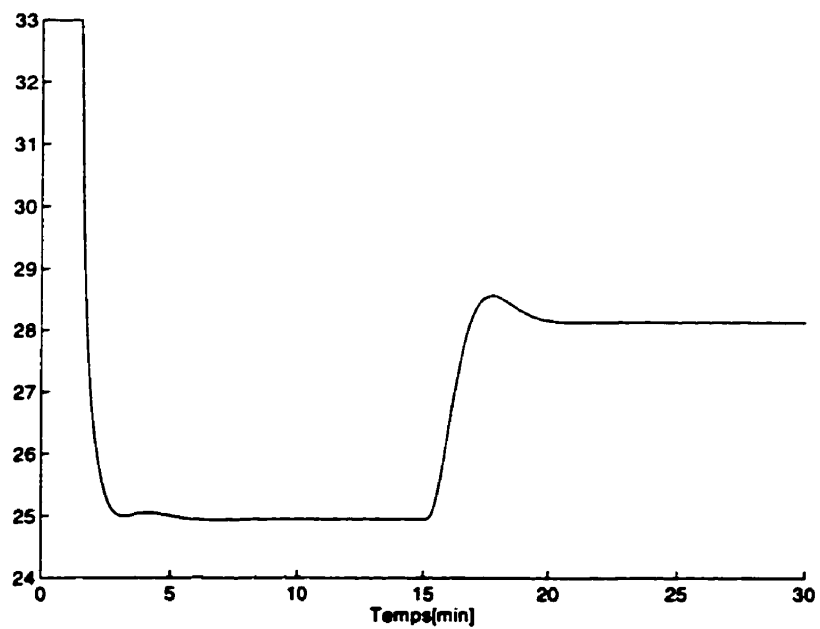


(b) Teneur du rejet [%]

FIG. 11.12: Retour de sortie avec intégrateur — Sorties



(a) Débit d'air [%]



(b) Débit du collecteur [%]

FIG. 11.13: Retour de sortie avec intégrateur — Commandes

# Conclusion

Une structure de commande, basée sur des versions modifiées des fonctions de Lyapunov, est proposée pour les procédés à deux entrées et deux sorties. Les termes croisés, présents dans le contrôleur multivariable obtenu par la méthode du **backstepping**, sont éliminés par l'introduction de termes supplémentaires adéquats dans la fonction de contrôle de Lyapunov associée. Ce subterfuge se traduit par une modification des erreurs régulées par le **backstepping**, permettant l'obtention de contrôleurs scalaires totalement indépendants. Si la norme  $\mathcal{L}_\infty$  du quotient d'interaction du procédé est inférieur à un, cette élimination peut s'effectuer sans altérer la stabilité et les performances en poursuite du système en boucle fermée.

La mesure de plusieurs variables d'état du procédé reste toutefois nécessaire à l'implémentation des contrôleurs obtenus. C'est pourquoi, plusieurs modifications sont apportées à la méthode originelle, dans le but de réduire le nombre des variables mesurées. La première solution consiste à fusionner la version par retour d'état et la technique du **backstepping** avec observateur, pour obtenir une structure à retour de sortie décentralisée. Un nouveau choix des variables régulées assure la stabilité interne du système en boucle fermée, avec des conditions similaires à celles obtenues par la version par retour d'état. Cette méthode nécessite, cependant, une profonde modification de la procédure récursive de design et des preuves de stabilité beaucoup plus élaborées. L'approche par retour d'état partiel présente, pour cette raison, une solution fort intéressante.

En effet, sans aucune modification dans la procédure de design, uniquement par un choix différent des termes supplémentaires (i.e. erreurs régulées) et sous des conditions

de stabilité similaires, cette solution permet une réduction considérable du nombre de variables mesurées. Une version par retour de sortie de cette approche est également proposée. Ceci est, encore une fois, rendu possible par un nouveau choix des erreurs régulées. Toutefois, à moins d'imposer de sévères conditions structurelles au procédé, le passage direct du retour d'état partiel au retour de sortie engendre des erreurs de poursuite non nulles en régime permanent. Ce handicap est éliminé par l'introduction d'une action intégrale dans chacun des contrôleurs scalaires résultants. La stratégie de commande développée est appliquée, pour les versions par retour de sortie, au contrôle d'un simulateur phénoménologique d'un banc de flottation. Pour les versions nécessitant la mesure des variables d'état, autres que les sorties, un modèle deux par deux, obtenu par identification du simulateur, est utilisé.

Dans le cas du retour d'état, la méthode de l'erreur modifiée, s'étend, sans majeure difficulté, aux systèmes non linéaires et/ou à paramètres inconnus. L'extension, moins directe, des versions par retour d'état partiel et retour de sortie est actuellement à l'étude.

**Cinquième partie**  
**Conclusion générale**



# Conclusion générale

Malgré les progrès réalisés, la commande adaptative continue de souffrir du manque de résultats, notamment en ce qui concerne les procédés industriels. Cette contre-performance s'explique, surtout, par la faiblesse, en termes de stabilité, du principe de l'équivalence certaine qui sépare le design du contrôleur de celui de l'adaptation. Pour contrer ces problèmes de stabilité, des méthodes issues de la commande non linéaire sont utilisées. Celles-ci apportent une amélioration substantielle aux performances des contrôleurs adaptatifs basés sur l'équivalence certaine. Pour les méthodes non linéaires, la stabilité passe au premier plan pour devenir l'élément clé du design. La loi de commande tient compte de la dynamique d'adaptation. Ces deux dernières, ainsi que la fonction de Lyapunov qui garantit la stabilité et les performances du système global, sont conçues simultanément, grâce à la technique récursive du **backstepping** et ses variantes.

Toutefois, malgré ses bonnes performances sur le plan théorique, le **backstepping** souffre de certaines limitations qui freinent son application aux procédés réels. Partant de ces constatations, et conscient de l'efficacité et de la souplesse de la méthode, on s'est fixé comme objectif d'apporter certaines modifications au **backstepping**, afin de le rendre exploitable dans la pratique, en particulier pour les procédés industriels. Les principales réalisations, issues de ce projet de recherche, sont données dans ce qui suit.

## Réalisations

- L'exploration d'un nouveau domaine de recherche très actif, i.e. la commande adaptative non linéaire par la méthode du **backstepping**.

- La conception d'un nouveau filtre pour améliorer le comportement de la variable manipulée en milieux bruités. Le design de ce filtre, très particulier, se base sur la théorie de Lyapunov, garantissant ainsi la stabilité de la boucle.
- L'introduction de l'action intégrale dans les contrôleurs générés par le backstepping. Cette modification, tout en préservant la stabilité, élimine les erreurs résiduelles, en présence de perturbations à moyenne non nulle. Pour les procédés linéaires, l'introduction de cette action intégrale a permis la conception d'une nouvelle classe de contrôleurs adaptatifs robustes du type PID.
- L'extension, par l'utilisation du prédicteur de Smith, de l'applicabilité du backstepping aux procédés avec retards et/ou déphasages non minimaux.
- Le développement d'une nouvelle méthode de design pour la commande décentralisée. L'idée principale de cette méthode consiste à modifier, par l'introduction de termes adéquats, les variables d'erreur régulées par la procédure du backstepping. Cela permet d'obtenir différentes lois de commande dont la réalisation est complètement décentralisée. Suivant le choix des termes introduits, le design donne lieu à des commandes par retour d'état, par retour de sortie (avec ou sans observateurs), ainsi que par retour d'état partiel, où seulement quelques variables sont nécessaires à l'implantation. La méthode proposée est applicable aussi bien aux systèmes linéaires que non linéaires.

### **Perspectives**

Bien qu'elles répondent à de nombreux problèmes, souvent d'une importance capitale pour le contrôle industriel, les solutions proposées restent perfectibles et ouvrent la voie à de nouveaux axes de recherche. Parmi les directions, jugées prometteuses et qui pourraient améliorer les performances et la stabilité des boucles, on retient principalement :

- L'assouplissement des conditions (suffisantes) de stabilité obtenues par la méthode l'erreur modifiée. Toutes ces conditions découlent du théorème du petit gain. Or, ce

dernier se contente de l'information contenue dans le module des fonctions de transfert, et n'utilise guère la phase. Cela conduit, en toute évidence, à des conditions trop restrictives. Remplacer le théorème du petit gain par une approche qui exploiterait l'information de la phase, conduirait certainement à des conditions de stabilité beaucoup plus souples. Cela aurait comme conséquence directe d'élargir l'applicabilité de la méthode.

- La robustification, dans la version adaptative de la MEM, des lois de mise à jour des paramètres. Cela passe par l'introduction de l'une (ou plusieurs) des modifications présentées dans l'annexe A.
- L'évaluation des effets de la discrétisation. Le backstepping, donnant lieu à des lois de commandes continues, son application aux procédés nécessite une discrétisation. Contrairement aux méthodes "classiques", où l'effet de la discrétisation est facilement quantifié en terme de perte de phase, et éventuellement compensé, dans le backstepping cette quantification et/ou compensation est loin d'être évident. Cette difficulté s'explique par la nature "retour d'état continu" de la méthode du backstepping.
- Application de la méthode de l'erreur modifiée à la commande décentralisée d'une colonne de flottation. Les essais commenceront, dans un proche avenir, sur une colonne pilote, considérée comme un procédé  $2 \times 2$ .

Il faut noter que certains de ces points sont, déjà, en cours d'exploration, dans le cadre de projets de maîtrise.

## **Sixième partie**

### **Annexes**

# Annexe A

## Commande adaptative robuste

L'idée de base des techniques de robustification, qui ont pour but de prévenir l'instabilité, consiste à éliminer l'action intégrale pure des lois adaptation couramment utilisées. Ce changement permet de garantir la bornétude de tous les signaux à l'intérieur de la boucle, en évitant que les paramètres du modèle (du procédé) prennent des valeurs extrêmement élevées. On parle, dans ce cas, de dérive des paramètres (parameter drift). Les techniques les plus rencontrées dans la littérature, pour modifier cet effet intégrateur des lois, sont données dans ce qui suit. Ces méthodes seront illustrées sur l'exemple simple d'une régression linéaire, donné par le système

$$y = \theta u + \delta$$

où  $y$  et  $u$  sont respectivement la sortie et l'entrée du procédé,  $\theta$  et  $\delta$  sont le paramètre inconnu du modèle du procédé et la perturbation affectant celui-ci. On utilise comme loi d'adaptation (gradient sans modification)

$$\begin{aligned}\dot{\hat{\theta}} &= \gamma \bar{y} u \\ \bar{y} &= y - \hat{\theta} u\end{aligned}$$

où  $\hat{\theta}$  est l'estimé du paramètre  $\theta$  et  $\gamma$  est le gain d'adaptation.

**Remarque A.1 (Erreurs)** — Par la suite, les modifications utilisées seront jugées en fonction (entre autres) des erreurs de commande (poursuite et/ou régulation) et d'identification qu'elles engendrent. Il faut toutefois rappeler que la présence d'une erreur de

commande résiduelle dépend du type de contrôleur utilisé et de la présence des perturbations à moyenne non nulle, à la sortie et/ou entrée du procédé. Afin d'être en mesure de juger de la qualité des lois d'adaptation, une structure MRC est considérée. Pour un procédé à paramètres connus et en absence de perturbations, cette commande garantit une erreur nulle en régime permanent. Ainsi, avec une commande MRAC, si erreur il y a, celle-ci ne peut être due qu'à la loi d'adaptation (toujours en absence de perturbations).

□

### Zone morte

Introduite pour la première fois par Egardt (1979), la zone morte constitue l'une des premières modifications à être proposées. L'idée consiste à arrêter l'adaptation quand la valeur absolue d'un certain signal d'erreur<sup>1</sup> est en deçà d'un seuil  $g_0$ , préalablement établi. La loi d'adaptation obtenue est donnée par

$$\dot{\hat{\theta}} = \gamma(\bar{y} + g)u$$

avec

$$g = \begin{cases} 0 & \text{si } |\bar{y}| > g_0 \\ -\bar{y} & \text{si } |\bar{y}| \leq g_0 \end{cases}$$

L'inconvénient majeur de cette méthode réside dans le fait que plusieurs termes qui dépendent des paramètres inconnus du procédé, ainsi que des bornes sur les perturbations, sont nécessaires à l'implantation de la zone morte. Des bornes trop sévères réduisent la classe des procédés auxquels le contrôleur est applicable, alors que des bornes trop molles conduisent à des fortes erreurs de poursuite et d'identification. Ces erreurs persistent, même en l'absence des perturbations.

### Terme $\sigma$ fixe

Cette méthode a été proposée par Ioannou & Kokotovic (1983, 1984). Elle consiste à introduire un terme  $-\sigma_0\hat{\theta}$  (avec  $\sigma_0 > 0$ ) dans la loi d'adaptation, afin de modifier l'ac-

<sup>1</sup>On choisit souvent l'erreur de poursuite.

tion intégrale pure de celle-ci et éviter toute possibilité de divergence des paramètres. Avec cette modification, on obtient comme loi de mise à jour du paramètre

$$\dot{\hat{\theta}} = \gamma \bar{y}u - \gamma \sigma_0 \hat{\theta}$$

Il est clair qu'avec une telle loi, si l'entrée  $u$  est bornée,  $\hat{\theta}$  ne peut diverger. Bien qu'aucune connaissance a priori ne soit nécessaire, cette méthode présente le même inconvénient que la zone morte, à savoir des erreurs de poursuite et d'identification non nulles, même en l'absence des perturbations.

### Projection des paramètres

C'est une méthode très efficace (et largement utilisée) pour éviter le phénomène du "drift" et permettre de garder le vecteur des paramètres à l'intérieur d'une certaine bande définie au préalable. Elle a été introduite par Kreisselmeier & Narendra (1982), afin de forcer les estimés à rester à l'intérieur d'un certain ensemble convexe, défini sur l'espace des paramètres, et qui contient les vraies valeurs de ceux-ci. La connaissance d'une borne supérieure  $M_0$  sur la norme du vecteur des paramètres est utilisée dans la loi d'adaptation, pour définir la contrainte

$$g(\hat{\theta}) \triangleq \{ \hat{\theta} \mid |\hat{\theta}| - M_0 \leq 0 \}$$

avec  $M_0$  choisi tel que  $M_0 \geq |\theta|$ . Ceci donne comme loi modifiée

$$\dot{\hat{\theta}} = \begin{cases} \gamma \bar{y}u & \text{si } |\hat{\theta}| < M_0 & \text{ou} \\ & \text{si } |\hat{\theta}| = M_0 & \text{et } \hat{\theta} \bar{y}u \leq 0 \\ 0 & \text{si } |\hat{\theta}| = M_0 & \text{et } \hat{\theta} \bar{y}u > 0 \end{cases}$$

### Terme $\epsilon_1$

Narendra & Annaswamy (1986) le proposent pour pallier l'inconvénient majeur de la méthode du terme  $\sigma$  fixe, i.e. erreurs résiduelles non nulles en absence de perturbations. Un terme  $-|\epsilon_1|s_0\hat{\theta}$  est introduit dans la loi d'adaptation, où  $\epsilon_1 = \bar{y}$  et  $s_0 > 0$  est une constante de design. La loi d'adaptation devient dans ce cas

$$\dot{\hat{\theta}} = \gamma \bar{y}u - \gamma |\epsilon_1| s_0 \hat{\theta}$$

La logique derrière ce choix réside dans le fait que, étant donné que dans le cas idéal  $\epsilon_1$  est assuré de converger vers zéro (sous certaines conditions), le terme introduit s'annule également en absence de perturbations ( $\delta = 0$ ). Dans ce cas, contrairement au terme  $\sigma$ , les propriétés idéales de la loi d'adaptation ne sont pas affectées par la présence du terme  $\epsilon_1$ .

**Remarque A.2 (Limitations)** — Les méthodes basées sur la zone morte, le terme  $\sigma$  fixe et le terme  $\epsilon_1$  nécessitent, toutes, une certaine connaissance a priori, concernant les perturbations. En plus, la robustesse des deux premières est établie au prix de possibles (grandes) erreurs de poursuite et d'identification, même dans le cas idéal (pas de perturbations externes). Dans une telle situation (cas idéal), la méthode du "switching"  $\sigma$  apporte une solution particulièrement intéressante, car elle garantit des erreurs résiduelles (de poursuite et d'identification) nulles, avec moins d'information a priori que celle exigée par le terme  $\epsilon_1$ .

□

### Switching $\sigma$

Cette méthode est une sorte de combinaison de la projection et du terme  $\sigma$  fixe, auxquelles, elle apporte une amélioration substantielle. Étant donné que le rôle du terme  $\sigma$  est d'éviter le phénomène du "drift", sa présence est sans intérêt (voire même nuisible), tant que le vecteur conserve des valeurs finies. La connaissance d'une borne supérieure  $M_0$  pour la norme du vecteur des paramètres est, encore une fois, exploitée par Ioannou & Tsakalis (1986b) pour n'activer ce terme que lorsque la norme du vecteur estimé excède  $M_0$ . La loi d'adaptation obtenue est identique à celle du terme  $\sigma$  fixe,

$$\dot{\hat{\theta}} = \gamma \bar{y} u - \gamma \sigma_s \hat{\theta}$$

sauf que cette fois  $\sigma_s$ , qui remplace  $\sigma_0$ , est donné par

$$\sigma_s = \begin{cases} 0 & \text{si } |\hat{\theta}| < M_0 \\ \sigma_0 & \text{si } |\hat{\theta}| \geq M_0 \end{cases}$$



Cette modification a l'avantage de garantir des erreurs nulles, si la perturbation est supprimée. Dans le cas où le vecteur des paramètres excède la borne  $M_0$ , l'adaptation n'est pas complètement inhibée, comme c'est le cas avec la projection, mais seul le terme  $\sigma_s$ .

**Remarque A.3 (Erreurs de modélisation)** — Toutes les propriétés de stabilité et de convergence des méthodes, présentées jusqu'ici, ont été établies sous l'hypothèse de la bornétude des perturbations. En présence d'erreurs de modélisation (mauvaise structure), ces performances ne peuvent être garanties. Ceci est dû au simple fait que les dynamiques ignorées agissent sur la loi d'adaptation, certes comme une perturbation externe, mais celle-ci ne peut, en aucun cas, être considérée comme bornée. Ce qui, en toute logique, pose des difficultés supplémentaires. Afin de surmonter cette limitation et ramener le problème des erreurs de modélisation à un problème de perturbations bornées<sup>2</sup>, on a introduit la notion de normalisation.

□

### Normalisation

Un signal  $\mu > 0$  bien choisi (souvent, dépendant de l'entrée) sert à normaliser les différents signaux utilisés dans l'algorithme d'adaptation

$$\mu^2 = 1 + m_s^2$$

où  $m_s$  est une fonction choisie de façon à avoir  $y/\mu$  et  $u/\mu \in \mathcal{L}_\infty$ . En fonction de l'information dont on dispose au sujet des perturbations qui affectent le système,  $m_s$  peut être généré soit d'une manière statique, soit par un système dynamique,

$$\begin{aligned}\mu^2 &= 1 + m_s^2 \\ \dot{m}_s &= -\delta_0 m_s + u^2 + y^2\end{aligned}$$

où  $m_s(0) = 0$  et  $\delta_0$  est une constante qui dépend du terme incertain. Cette modification a pour effet de garantir que tous les signaux qui entrent la module d'adaptation, aussi bien le régresseur que les perturbations, soient  $\mathcal{L}_\infty$ . Une fois la normalisation effectuée,

<sup>2</sup>Pour lequel plusieurs solutions existent.

on retombe dans le cas des perturbations bornées. Il suffit alors de choisir, parmi les méthodes présentées précédemment, celle qui répond le mieux aux besoins du problème.

Cette technique a été introduite et utilisée avec succès par Praly (1983, 1984), où la normalisation a été combinée avec la projection, pour établir la stabilité globale des systèmes discrets dont les parties modélisées peuvent avoir un degré relatif arbitraire (mais connu), alors que les parties non modélisées doivent être stables. La normalisation a été également utilisée, conjointement avec la zone morte et la projection, par Kreisselmeier & Anderson (1986), toujours pour des systèmes discrets. Ioannou l'a utilisée, avec sa méthode du  $\sigma$ -switching, dans une structure MRAC pour prouver la stabilité globale des systèmes continus dans (Ioannou 1984) et les systèmes discrets dans (Ioannou & Tsakalis 1986a).

### **Approche fréquentielle (PE)**

Ces modifications ne représentent pas la seule méthode utilisée pour prouver (et assurer) la stabilité des structures adaptatives. Des solutions, basées sur la richesse fréquentielle du signal de référence (**Persistent Excitation Condition**), ont été proposées (Anderson 1982, Åström 1983). L'idée consiste à s'assurer que le signal d'entrée est suffisamment riche (en fréquences) pour que le système en boucle fermée, sous l'action d'un régulateur adaptatif, conserve une stabilité locale. Les perturbations doivent être relativement faibles par rapport au signal d'excitation, afin d'éviter une instabilité. À cause, vraisemblablement, de la difficulté à garantir cette condition, cette technique n'a pas été exploitée dans la commande décentralisée, au même titre que la robustification. Les travaux dans cette direction ont toutefois le mérite d'avoir permis de comprendre les phénomènes d'instabilité dans les boucles de contrôle adaptatif, en présence d'erreurs de modélisation et perturbations bornées.

### **Conclusion**

Une loi adaptative robuste se construit d'une manière directe, en apportant deux modifications essentielles aux lois d'identification classiques. La première de ces modifications consiste à normaliser les signaux utilisés dans l'algorithme de mise à jour, ce

qui a pour effet de borner (par le haut) toutes les variables entrant le module d'adaptation. Par la suite, on choisit parmi les méthodes disponibles, i.e. la zone morte, la projection, le terme  $\epsilon_1$  et les deux  $\sigma$  (fixe ou variable), celle(s) qui correspond(ent) le mieux à la situation.

# Annexe B

## Définitions

**Définition B.1 (Normes)** — Les différentes normes, définies sur un espace euclidien à  $n$  dimensions, sont données par

$$\|x\|_p = \left( |x_1|^p + |x_2|^p + \cdots + |x_n|^p \right)^{\frac{1}{p}}, \quad 1 \leq p < \infty$$

et

$$\|x\|_\infty = \max_i (|x_i|)$$

**Définition B.2 (Fonction définie en signe)** — Soit  $\mathcal{V}$ , une fonction définie dans un domaine qui contient l'origine.

- $\mathcal{V}$  est positive définie si  $\mathcal{V}(0) = 0$  et  $\mathcal{V}(x) > 0$  pour tout  $x \neq 0$ .
- $\mathcal{V}$  est positive semi-définie si  $\mathcal{V}(0) = 0$  et  $\mathcal{V}(x) \geq 0$  pour tout  $x \neq 0$ .
- $\mathcal{V}$  est négative définie si  $-\mathcal{V}$  est positive définie.
- $\mathcal{V}$  est négative semi-définie si  $-\mathcal{V}$  est semi-positive définie.

**Définition B.3 (Classe  $\mathcal{K}$ )** — Une fonction continue  $\kappa : [0, a) \rightarrow [0, \infty)$  appartient à la classe  $\mathcal{K}$  si elle est strictement croissante et  $\kappa(0) = 0$ . Si en plus  $a = \infty$  et  $\kappa(p) \rightarrow \infty$  quand  $p \rightarrow \infty$ , on dit que  $\kappa$  est de la classe  $\mathcal{K}_\infty$ .

**Définition B.4 (Classe  $\mathcal{KL}$ )** — Une fonction continue  $\kappa : [0, a) \times [0, \infty) \rightarrow [0, \infty)$  appartient à la classe  $\mathcal{KL}$ , si pour chaque  $\tau$  fixe la fonction  $\kappa(p, \tau)$  est de classe  $\mathcal{K}$  par rapport à  $p$ , et pour chaque  $p$  fixe  $\kappa(p, \tau)$  est décroissante par rapport à  $\tau$ , avec  $\kappa(p, \tau) \rightarrow \infty$  quand  $\tau \rightarrow \infty$ .

**Définition B.5 (Difféomorphisme)** — Un difféomorphisme est une transformation  $\mathcal{T}$  différentiable (continue), dont l'inverse  $\mathcal{T}^{-1}$  est aussi différentiable (continue).

# Bibliographie

- Anderson, B. D. O. (1982), 'Exponential convergence and persistent excitation', in *Proc. 21st IEEE Conf. on Decision and Control*, pp. 12–17, Orlando, Florida.
- Åström, K. J. (1983), 'Analysis of Rohrs counterexamples to adaptive control', in *Proc. 22nd IEEE Conf. on Decision and Control*, pp. 982–987, San Antonio, Texas.
- Åström, K. J. & Wittenmark, B. (1995), *Adaptive Control, Second Edition*, Addison-Wesley Publishing Company.
- Barmish, B. R. & Leitmann, G. (1982), 'On ultimate boundedness control of uncertain systems in absence of matching assumptions', *IEEE Transactions On Automatic Control*, **27**, 153–158.
- Benaskeur, A. (1997a), *Commande non linéaire*, Technical Report No. 97–13, GRAIIM, Université Laval.
- Benaskeur, A. (1997b), *Systèmes non linéaires*, Technical Report No. 97–11, GRAIIM, Université Laval.
- Benaskeur, A. (1998), *Backstepping adaptatif*, Technical Report No. 98–03, GRAIIM, Université Laval.
- Benaskeur, A. & Desbiens, A. (1999a), 'Decentralized backstepping-based control, Part I : Partial state feedback', *Soumis à Automatica*,.
- Benaskeur, A. & Desbiens, A. (1999b), 'Decentralized backstepping-based control, Part II : Output feedback with integral action', *Soumis à Automatica*,.
- Benaskeur, A. & Desbiens, A. (1999c), 'Decentralized control : A modified Lyapunov function scheme', *American Control Conference*, pp. 2087–2091, San Diego, California.

- Benaskeur, A. & Desbiens, A. (1999d), 'Lyapunov-based decentralized control of a rougher flotation phenomenological simulator', In *D. Hodouin, C. Bazin and A. Desbiens (Ed.)*, *Control and Optimization in Minerals, Metals and Materials Processing*, Canadian Institute on Mining, Metallurgy and Petroleum, Montréal, pp. 311–321.
- Benaskeur, A. & Desbiens, A. (1999e), 'Modified error decentralized control with observer backstepping', *Soumis à International Journal of Control*,.
- Benaskeur, A. & Desbiens, A. (2000), 'Robust adaptive backstepping-based PID', *Soumis à American Control Conference*,.
- Borisson, U. (1979), 'Self-tuning regulators for a class of multivariable systems', *Automatica*, **15**, 209–215.
- Butchart, R. L. & Shackloth, B. (1965), 'Synthesis of model reference adaptive control systems by Lyapunov's second method', *Proceedings of the 1965 IFAC Symposium on Adaptive Control*, pp. 145–152, Teddington, U.K.
- Butler, H. (1992), *Model Reference Adaptive Control : From Theory to Practice*, Prentice Hall.
- Byrnes, C. I. & Isidori, A. (1984), 'A frequency domain philosophy for nonlinear systems with application to stabilization and to adaptive control', *Proceedings of the 23th Conference on Decision and Control*, pp. 1569–1573, Las Vegas, Nevada.
- Corless, M. J. & Leitmann, G. (1981), 'Continuous-state feedback guaranting uniform ultimate boundedness for uncertain dynamic systems', *IEEE Transactions On Automatic Control*, **26**, 1139–1144.
- Datta, A. (1993), 'Performance improvement in decentralized adaptive control : A modified model reference scheme', *IEEE Transactions On Automatic Control*, **38**(11), 1717–1722.
- Datta, A. & Ioannou, P. (1991), 'Decentralized indirect adaptive control of interconnected systems', *International Journal of Control*, **5**, 259–281.
- Datta, A. & Ioannou, P. (1992), 'Decentralized adaptive control', In *C.T. Leondes(Ed.)*, *Advances in Control and Dynamic Systems* pp. 125–171.
- Desbiens, A., Hodouin, D. & Mailloux, M. (1998), 'Nonlinear predictive control of a rougher flotation unit using local models.', *International Federation of Automatic*

- Control (IFAC) Symposium on Automation in Mining, Mineral and Metal Processing*, pp. 297–302, Cologne, Germany.
- Desoer, C. A. & Vidyasagar, M. (1974), 'Feedback systems : Input–output properties', *Academic Press Inc., New York*.
- Egardt, B. (1979), 'Stability of adaptive controller', *Springer–Verlag, Berlin*.
- Feurer, A. & Morse, A. S. (1978), 'Adaptive control of single–input, single–output linear systems', *IEEE Transactions On Automatic Control*, **23**(4), 557–569.
- Fossard, A. (1993), *Systèmes non linéaires, 3 : Commande*, Masson.
- Friedland, B. (1996), *Advanced Control System Design*, Prentice Hall.
- Gavel, D. T. & Siljak, D. D. (1989), 'Decentralized adaptive control : Structural conditions for stability', *IEEE Transactions On Automatic Control*, **34**(4), 413–426.
- Gille, J.-C. (1988), *Systèmes asservis non linéaires*, Dunod.
- Green, M. & Limebeer, D. J. N. (1996), *Linear Robust Control*, Prentice Hall.
- Hill, D. J., Wen, C. & Goodwin, G. C. (1988), 'Stability analysis of decentralized robust adaptive control', *Systems & Control Letters*, **11**, 277–284.
- Ikhouane, F. & Krstic, M. (1998a), 'Adaptive backstepping with parameters projection : Robustness and asymptotic performance', *Automatica*, **34**(4), 429–435.
- Ikhouane, F. & Krstic, M. (1998b), 'Robustness of the tuning function adaptive backstepping design for linear systems', *IEEE Transactions On Automatic Control*, **43**(3), 431–437.
- Ioannou, P. A. (1984), 'Robust direct adaptive control', in *Proc. 23rd IEEE Conf. on Decision and Control*, pp. 1015–1020, Las Vegas, Nevada.
- Ioannou, P. A. (1986), 'Decentralized adaptive control of interconnected systems', *IEEE Transactions On Automatic Control*, **31**(4), 291–298.
- Ioannou, P. A. & Kokotovic, P. (1983), 'Adaptive systems with reduced models', *Springer–Verlag, New York*.
- Ioannou, P. A. & Kokotovic, P. (1984), 'Instability analysis and improvement of robustness of adaptive control', *Automatica*, **20**(5), 583–594.



- Ioannou, P. A. & Kokotovic, P. (1985), 'Decentralized adaptive control of interconnected systems with reduced-order models', *Automatica*, **21**, 401–412.
- Ioannou, P. A. & Sun, J. (1988), 'Theory and design of robust direct and indirect adaptive control schemes', *International Journal of Control*, **47**(3), 775–813.
- Ioannou, P. A. & Sun, J. (1996), *Robuste Adaptive Control*, Prentice Hall, New Jersey.
- Ioannou, P. A. & Tsakalis, K. (1986a), *Adaptive and Learning Systems : Theory and Applications*, Plenum, New York.
- Ioannou, P. A. & Tsakalis, K. S. (1986b), 'A robust direct adaptive controller', *IEEE Transactions On Automatic Control*, **31**, 1033–1043.
- Isidori, A. (1981), 'Nonlinear decoupling via feedback : Differential geometric approach', *IEEE Transactions On Automatic Control*, **26**, 331–345.
- Jain, S. & Khorrami, F. (1995), 'Global decentralized adaptive control of large scale nonlinear systems without strict matching', in *Proc. American Control Conf.*, pp. 2938–2942.
- Jain, S. & Khorrami, F. (1997a), 'Decentralized adaptive control of a class of large-scale interconnected nonlinear systems', *IEEE Transactions On Automatic Control*, **42**(2), 136–154.
- Jain, S. & Khorrami, F. (1997b), 'Decentralized adaptive output feedback design for large-scale nonlinear systems', *IEEE Transactions On Automatic Control*, **42**(2), 729–735.
- Jiang, Z. P. & Praly, L. (1991), 'Iterative designs of adaptative controllers for systems with nonlinear integrators', in *Proceedings of the 30th IEEE Conference on Decision and Control*, pp. 2482–2487, Brighton, UK.
- Johansen, T. A. & Foss, B. A. (1993), 'Constructing narmax models using armax models', *International Journal of Control*, **58**(5), 1125–1153.
- Kanellakopoulos, I., Kokotović, P. V. & Morse, A. S. (1991), 'Systematic design of adaptive controllers for feedback linearizable systems', *IEEE Transactions On Automatic Control*, **36**(11), 1241–1253.
- Kanellakopoulos, I., Krstić, M. & Kokotović, P. V. (1993), 'Trajectory initialization in

- adaptive nonlinear systems', in *Proceedings of the 1993 IEEE Mediterranean Symposium on New Directions in Control Theory*, pp. Chania, Greece.
- Khalil, H. (1996), *Nonlinear Systems*, Prentice Hall.
- Khalil, H. K. & Saberi, A. (1987), 'Adaptive stabilization of a class of nonlinear systems using high-gain feedback', *IEEE Transactions On Automatic Control*, **32**, 1031–1035.
- Koivo, H. N. (1980), 'A multivariable self-tuning controller', *Automatica*, **16**, 351–366.
- Kokotović, P. V. & Sussmann, H. J. (1989), 'A positive real condition for global stabilization of nonlinear systems', *Systems & Control Letters*, **13**, 125–133.
- Kreisselmeier, G. & Anderson, B. D. O. (1986), 'Robust model reference adaptive control', *IEEE Transactions On Automatic Control*, **31**(2), 127–134.
- Kreisselmeier, G. & Narendra, K. S. (1982), 'Stable model reference adaptive control in presence of bounded disturbances', *IEEE Transactions On Automatic Control*, **27**, 1169–1176.
- Krstić, M. & Kokotović, P. (1994), 'Observer-based schemes for adaptive nonlinear state-feedback control', *International Journal of Control*, **59**, 1373–1381.
- Krstić, M. & Kokotović, P. (1995), 'Adaptive nonlinear design with controller-identifier separation and swapping', *IEEE Transactions On Automatic Control*, **40**(3), 426–440.
- Krstić, M. & Kokotović, P. (1996), 'Modular approach to adaptive nonlinear stabilization', *Automatica*, **32**(4), 625–629.
- Krstić, M., Kanellakopoulos, I. & Kokotović, P. (1992), 'Adaptive nonlinear control without overparametrization', *Systems & Control Letters*, **19**, 177–185.
- Krstić, M., Kanellakopoulos, I. & Kokotović, P. (1994), 'Nonlinear design of adaptive controllers for linear systems', *IEEE Transactions On Automatic Control*, **39**(4), 738–752.
- Krstić, M., Kanellakopoulos, I. & Kokotović, P. (1995), *Nonlinear and Adaptive Control Design*, Wiley-Interscience Publication.
- Landau, I. D. (1979), *Adaptive Control: The Model Reference Approach*, Marcel-Dekker, New York.

- Lasalle, J. & Lefschetz, S. (1961), *Stability by Liapunov's Direct Method*, Academic Press.
- Ling, Y. & Tao, G. (1997), 'Adaptive backstepping design for linear multivariable plants', *International Journal of Control*, **68**(6), 1289–304.
- Luenberger, D. G. (1964), 'Observing the state of a linear system', *IEEE Transactions on Military Electronics* **8**, 74–80.
- Luenberger, D. G. (1971), 'An introduction to observers', *IEEE Transactions On Automatic Control*, **16**(6), 596–602.
- Lyapunov, A. M. (1966), *Stability of Motion*, Academic Press.
- Mailloux, M. (1997), Procédé de flottation, Technical report, GRAIIM, Université Laval.
- Makni, S. (1996), Reconciliation des Données des Systèmes Dynamiques : Application aux procédé flottation, PhD thesis, Université Laval, Dept. of Mining and Metallurgy.
- Morari, M. & Zafiriou, E. (1989), *Robust Process Control*, Prentice Hall.
- Narendra, K. S. & Annaswamy, A. M. (1986), 'A new adaptive law for robust adaptation without persistent excitation', *Report No. 8510, Elec. Eng. Dept., Yale University*.
- Olas, A. (1990), 'Recursive liapunov functions : Properties, linear systems', *Control and Dynamic Systems*, **35**, 1–30.
- Parks, P. C. (1966), 'Lyapunov redesign of model reference adaptive control systems', *IEEE Transactions On Automatic Control*, **11**, 362–365.
- Praly, L. (1983), 'Robustness of model reference adaptive control', in *Proc. 3rd Yale Workshop on Application on Adaptive Systems Theory*.
- Praly, L. (1984), 'Robust model reference adaptive controllers—part 1 : Stability analysis', in *Proc. 23rd IEEE Conference on Decision and Control*.
- Qu, Z. (1993), 'Robust control of nonlinear uncertain systems under generalized matching conditions', *Automatica*, **29**, 985–998.
- Rijnsdorp, J. E. (1965), 'Interaction on two-variable control systems for distillation columns', *Automatica*, **1**, 15–28.

- Sastry, S. S. & Isidori, A. (1989), 'Adaptive control of linearizable systems', *IEEE Transactions On Automatic Control*, **34**(11), 1123–1131.
- Singh, S. N. & Rugh, W. J. (1972), 'Decoupling in a class of nonlinear systems by state variable feedback', *Trans. ASME J. Dyn. Syst. Measur. Contr.* **94** pp. 323–324.
- Skogestad, S. & Postlethwaite, I. (1996), *Multivariable Feedback Control : Analysis and Design*, John Wiley & Sons Ltd, Chichester, England.
- Smith, O. J. M. (1957), 'Close control of loops with dead time', *Chemical Engineering Progress*, **53**(5), 217–219.
- Sontag, E. D. & Sussmann, H. J. (1990), 'Further facts about input to state stabilisation', *IEEE Transactions On Automatic Control*, **35**, 473–476.
- Tsinias, J. (1989), 'Sufficient Lyapunov-like conditions for stabilization', *Math. Contr. Signal. Syst. vol.2*, pp. 343–357.
- Wei, R., Sezer, M. E. & Ocali, O. (1996), 'Robust adaptive stabilization of a class of systems under structured nonlinear perturbations with application to interconnected systems', *International Journal of Control*, **63**(1), 183–194.
- Wen, C. (1994), 'Decentralized adaptive regulation', *IEEE Transactions On Automatic Control*, **39**(10), 2163–2166.
- Wen, C. & Soh, Y. C. (1997), 'Decentralized adaptive control using integrator backstepping', *Automatica*, **33**(9), 1719–1724.