

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITÉ MOULOU D MAMMERRI DE TIZI-OUZOU



FACULTÉ DE GÉNIE ÉLECTRIQUE ET D'INFORMATIQUE
DÉPARTEMENT AUTOMATIQUE

Mémoire de Fin d'Études de Master Académique

Spécialité : AUTOMATIQUE ET INFORMATIQUE INDUSTRIELLE

Présenté par :

Mohamed CHERADI

Cherif CHALLAL

Mémoire proposé et dirigé par M. Ahmed MAIDI

THÈME

Commande multiboucle robuste d'un système incertain par l'approche des scénarios

Mémoire soutenu publiquement le 24/06/2024 devant le jury composé de :

M. Ahcene TRIKI	Président	UMMTO
M. Ahmed MAIDI	Encadreur	UMMTO
Mme. Fatiha LOUADJ	Examinatrice	UMMTO
Mme. Saida BEDOUHENE	Examinatrice	UMMTO

Dédicaces

Ce travail est dédié à nos chers parents et à tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail.

Mohamed CHERADI

Cherif CHALLAL

Remerciements

Avant tous, nous remercions DIEU le tout puissant de nous avoir donné la santé, le courage et la patience durant toutes ces années d'études et grâce à qui ce travail a pu être réalisé.

Tout d'abord, nous tenons à exprimer notre gratitude, notre plus grande reconnaissance et notre profond respect à notre promoteur, Monsieur **Ahmed MAIDI**, Professeur à l'université Mouloud Mammeri de Tizi-Ouzou, qui nous a apporté son savoir scientifique indéniable, sa confiance, ses judicieux conseils et sa disponibilité, tout au long de l'élaboration de ce travail. Qu'il trouve ici l'expression de notre profonde reconnaissance.

Nous tenons particulièrement à remercier Monsieur **Ahcene TRIKI**, Maitre de conférences de classe B à l'université Mouloud Mammeri de Tizi-Ouzou, pour sa disponibilité, ses conseils judicieux, et son aide précieuse pour l'élaboration de notre travail. Nous lui en suis très reconnaissant.

Nos remerciements s'adressent à tous les membres du Laboratoire de Conception et Conduite des Systèmes de Production (L2CSP) (UMMTO), pour leur sympathie et l'excellente ambiance de travail qu'ils ont créés.

Nos remerciements vont s'adressé également au président et aux membres du jury qui nous ont fait l'honneur d'évaluer notre travail.

Notre reconnaissance va également s'adressé à tous les enseignants qui ont contribué à notre formation tout au long de notre cursus scolaire et universitaire.

Nos remerciements les plus chaleureux vont à nos chers parents et toute notre famille pour leurs encouragements, leur patience, et leur grand soutien durant toutes ces années d'études.

Nos remercions tous ceux qui ont contribué, de près ou de loin, à la réalisation de ce travail de fin d'étude.

Table des matières

Introduction générale	1
1 Commande multiboucle d'un système multivariable	3
1.1 Introduction	3
1.2 Systèmes multivariables	3
1.2.1 Définition	3
1.2.2 Représentation par matrice de fonctions de transfert	3
1.3 Problème d'interaction dans un système multivariable	5
1.4 Commande multiboucle	6
1.4.1 Choix des couples	7
1.4.2 Synthèse des correcteurs	8
1.5 Exemple d'application	10
1.6 Conclusion	14
2 Optimisation des fonctions en présence d'incertitudes	16
2.1 Introduction	16
2.2 Définition de l'optimisation	16
2.3 Nature de l'optimum	16
2.4 Forme mathématique d'un problème d'optimisation	18
2.5 Classification des problèmes d'optimisation	18
2.5.1 Problème d'optimisation linéaire	19
2.5.2 Problème d'optimisation non linéaire	19
2.5.3 Problème d'optimisation quadratique	20
2.5.4 Problème d'optimisation convexe	20
2.6 Classification des méthodes d'optimisation	20
2.6.1 Méthodes exactes	21
2.6.2 Méthodes heuristiques	22
2.6.3 Méthodes hybrides	22
2.7 Optimisation en présence des incertitudes	23
2.7.1 Méthode de Monte Carlo	23

2.7.2	Optimisation par intervalles	23
2.7.3	Optimisation multi-objectif	23
2.7.4	Méthode du pire cas	23
2.8	Optimisation robuste	24
2.8.1	Méthode des scénarios	24
2.8.2	Exemple d'application	25
2.9	Conclusion	26
3	Conception d'un correcteur PID basée sur l'optimisation	28
3.1	Introduction	28
3.2	Critères de performance	28
3.3	Optimisation d'un correcteur PID	29
3.3.1	Choix du correcteur	30
3.3.2	Directives pour le choix du type du correcteur	30
3.3.3	Optimisation d'un correcteur PID basée sur la méthode d'Åström	32
3.4	Exemple d'application	35
3.5	Conclusion	36
4	Synthèse d'une commande multiboucle pour un système incertain	39
4.1	Introduction	39
4.2	Formulation du problème de commande	39
4.3	Conception du correcteur multiboucle	40
4.4	Exemple d'application	43
4.5	Conclusion	47
	Conclusion générale	50
	Bibliographie	52

Table des figures

1.1	Système multivariable.	4
1.2	Commande multiboucle.	6
1.3	Évolutions temporelles des sorties y_1 et y_2	14
2.1	Minima et maxima d'une fonction $f(x)$	17
3.1	Boucle de régulation avec un régulateur PID.	31
3.2	Évolution temporelle de la sortie y	37
3.3	Évolution temporelle de la commande u	37
4.1	Évolution temporelle des sorties y_1 et y_2	48
4.2	Évolution temporelle des commandes u_1 et u_2	48
4.3	Évolution temporelle des sorties y_1 et y_2 pour des consignes variables.	49
4.4	Évolution temporelle des commandes u_1 et u_2 pour des consignes variables.	49

Liste des tableaux

1.1	Réglages d'un PID recommandés par Ziegler-Nichols	9
2.1	Classification des problèmes d'optimisation.	19
3.1	Tableau d'Åström généralisé pour le numérateur de $e(s)$	34
3.2	Tableau d'Åström généralisé pour le dénominateur de $e(s)$	34
3.3	Tableau d'Åström pour le numérateur de l'erreur (3.19)	35
3.4	Tableau d'Åström pour le dénominateur de l'erreur (3.19)	36

Liste des abréviations

BLT	Méthode du plus grand module (Biggest Log Modulus Tuning)
PID	Proportionnel Intégral Dérivé
RGA	Matrice des gains relatifs (Relative Gain Array)

Liste des symboles

a, b	Paramètres incertains
D	Ensemble de définition
d_i	Contrainte du type égalité
e	Erreur de poursuite
E	Vecteur de l'erreur de poursuite
f	Fonction objectif
F	Facteur de dérèglement
G	Matrice de fonctions de transfert du système
G_c	Matrice de fonctions de transfert du correcteur multiboucle
$g_{c_{ij}}$	Fonction de transfert du correcteur
g_{ij}	Fonction de transfert d'un système monovarié
h_j	Contrainte du type inégalité
I	Matrice identité
I_n	Ensemble des incertitudes
J	Critère de performance
K	Gain statique du système
K_c	Gain du correcteur
K_{BLT}, T_{BLT}	Paramètres de la BLT
K_{cr}	Gain critique
K_{ZN}, T_{ZN}	Paramètres de Ziegler-Nichols
K_s	Gain statique du système
m	Nombre d'entrées
n	Nombre de variables de décision
N	Nombre d'échantillons
r	Nombre de sorties
\mathbb{R}	Ensemble des nombres réels
\mathbb{R}^n	Espace Euclidien
s	Variable de Laplace

t	Variable de temps
T_{cr}	Période des oscillations entretenues
T_d	Constante dérivée
T_i	Constante intégrale
u_j	Entrée du système
U	Vecteur d'entrée
V_ε	Voisinage de l'optimum
x	Vecteur de variables de décision
\hat{x}	Optimum local
\tilde{x}	Optimum local strict
x^*	Optimum global
x_i	Variable de décision
y_i	Sortie du système
y_i^d	Consigne désirée
Y	Vecteur de sortie
Y^d	Vecteur de consignes désirées
z	Variable de décision
z_{ij}	Perturbation
α, β	Intervalles positifs
γ	Paramètre de confiance
$\delta, \delta_a, \delta_b$	Vecteurs des paramètres incertains
ε	Paramètre de risque de violation des contraintes
θ	Retard
λ_{ij}	Gain relatif
ξ	Coefficient d'amortissement
τ	Constante de temps
ω	Pulsation propre non amortie
ω_{cr}	Pulsation critique

Introduction générale

La commande des systèmes multivariables linéaires est un domaine crucial en ingénierie des systèmes, particulièrement dans les applications industrielles où les systèmes complexes nécessitent des méthodes de contrôle avancées pour assurer leur performance et leur stabilité. Ces systèmes se caractérisent par le phénomène d'interaction ou de couplage entre leurs différentes variables d'entrée et de sortie, rendant la conception de stratégies de commande efficaces et robustes plus difficile [4]. Pour ce type de systèmes, une action sur une entrée peut affecter plusieurs sorties, compliquant ainsi le processus de commande [8].

L'une des méthodes couramment utilisée pour la commande de ces systèmes est la commande multiboucle. Cette approche consiste à décomposer le système en plusieurs boucles de commande plus simples (monovariabiles) [3][4]. Chaque commande est utilisée pour commander une seule sortie. Cependant, la présence des interactions entre les variables du système et des incertitudes paramétriques, rend la conception d'une commande multiboucle difficile [5][12]. Les correcteurs PID (Proportionnel-Intégral-Dérivé) [7][8] sont souvent employés en raison de leur simplicité et de leur efficacité, mais en présence de forts couplages et des incertitudes sur les paramètres, un réglage optimal de ces correcteurs est une nécessité pour garantir la stabilité et des performances acceptables.

L'objectif principal de ce travail est de développer une approche de synthèse d'une commande multiboucle robuste permettant de maintenir un certain niveau de performances et la stabilité en boucle fermée malgré les incertitudes et le couplage. L'idée de notre travail consiste à utiliser l'approche des scénarios [5][12] pour la conception d'une commande multiboucle, à base des PID, optimale pour des systèmes multivariables incertains. Ainsi, en utilisant la méthode d'Åström [9], on ramène le problème de conception des correcteurs PID à un problème d'optimisation du type *min-max*, puis en utilisant l'approche des scénarios, on le convertit à un problème d'optimisation *standard* pour pouvoir exploiter les méthodes d'optimisation pour le solutionner [10].

Le contenu du mémoire est structuré comme suit :

- Dans le **premier** chapitre, nous présenterons le principe de la commande multiboucle, et nous expliquerons le phénomène d'interaction (couplage) entre les variables du système [3]. Par la suite, nous allons expliquer comment choisir les couples entrées-sorties (meilleure configuration de commande) en utilisant la matrice des gains relatifs (RGA) [3][8]. Puis, nous allons présenter la méthode du plus grand module (BLT) [13] utilisée pour la conception des correcteurs PID dans un environnement multivariable [8].

- Le **deuxième** chapitre est consacré à l'optimisation des fonctions mathématiques en présence des incertitudes sur les paramètres de la fonction. Ainsi, nous avons introduit la notion d'optimisation robuste [12], puis nous nous sommes focalisé sur l'approche des scénarios qui permet de convertir un problème d'optimisation robuste (du type *min-max*) [5][12] en un problème d'optimisation *standard* [10].

- Dans le **troisième** chapitre, on s'intéresse à l'évaluation des performances d'une boucle de commande en utilisant des critères de performances [7][8]. Puis, nous allons utiliser la méthode d'Åström [9] pour formuler le problème de synthèse d'un correcteur PID sous forme d'un problème d'optimisation *standard* [5]. L'approche présentée sera illustrée par un exemple d'application dans le cas d'un système du premier ordre.

- Dans le **quatrième** chapitre, nous généralisons l'approche présentée au Chapitre 3 à la conception d'une commande multiboucle d'un système multivariable incertain. Ainsi, nous allons d'abord introduire la matrice des gains relatifs pour un système incertain. Puis, nous exposons en détail la démarche à suivre pour l'optimisation de la commande multiboucle. Un exemple d'application est donné à la fin du chapitre.

Le mémoire se termine par une conclusion générale sur l'étude réalisée tout en indiquant certaines perspectives qui peuvent être explorées.

Chapitre 1

Commande multiboucle d'un système multivariable

1.1 Introduction

Ce chapitre est consacré à des généralités sur les systèmes multivariables linéaires. Dans ce contexte, on se limitera aux notions utilisées dans l'analyse des interactions. En effet, il existe plusieurs travaux de nature surtout théorique, où une multitude de variantes pour la représentation, l'analyse et la commande des systèmes multivariables, sont développées. La limitation aux notions de bases se justifie par le fait que dans le domaine d'analyse des interactions on n'utilise qu'une petite partie de toutes les notions et les méthodes existantes.

1.2 Systèmes multivariables

1.2.1 Définition

Les systèmes multivariables sont des systèmes à m entrées (u_1, u_2, \dots, u_m) et à r sorties (y_1, y_2, \dots, y_r) (Figure 1.1). Ces derniers ne se réduisent pas à la mise en parallèle de systèmes monovariables, mais, se caractérisent par des phénomènes d'interaction – ou de couplage – dans la mesure où l'application d'un signal sur une entrée entraîne en général une variation de plusieurs ou de toutes les sorties [1][2]. À la notion de couplage s'ajoutent ceux de l'intégrité, de sensibilité et de directionnalité, qui sont des propriétés fondamentales pour l'analyse et la conception de lois de commande pour les systèmes multivariables.

1.2.2 Représentation par matrice de fonctions de transfert

Les matrices de transfert sont utilisées pour représenter les caractéristiques entrées-sorties des systèmes multivariables (Figure 1.1) [3].

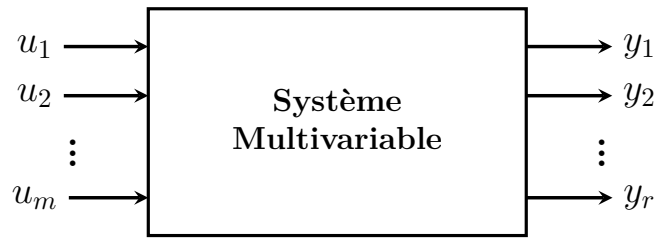


FIGURE 1.1 – Système multivariable.

L'effet de la $j^{\text{ème}}$ entrée sur la $i^{\text{ème}}$ sortie est défini comme suit :

$$y_i(s) = g_{ij}(s) u_j(s), \quad i = 1, \dots, r; j = 1, \dots, m. \quad (1.1)$$

En superposant les effets des diverses entrées, il vient :

$$y_i(s) = \sum_{j=1}^m g_{ij}(s) u_j(s) \quad (1.2)$$

En introduisant le vecteur d'entrée $U(s)$ et le vecteur de sortie $Y(s)$

$$U(s) = [u_1(s), u_2(s), \dots, u_m(s)]^T \quad (1.3)$$

$$Y(s) = [y_1(s), y_2(s), \dots, y_r(s)]^T \quad (1.4)$$

La relation (1.2) peut s'écrire sous la forme matricielle suivante :

$$Y(s) = G(s) U(s) \quad (1.5)$$

La matrice $G(s)$ est appelée matrice de fonctions de transfert, elle est constituée des fonctions de transfert $g_{ij}(s)$, et elle prend la forme générale suivante [3] :

$$G(s) = \begin{bmatrix} g_{11}(s) & g_{12}(s) & \cdots & g_{1m}(s) \\ g_{21}(s) & g_{22}(s) & \cdots & g_{2m}(s) \\ \vdots & \vdots & \ddots & \vdots \\ g_{r1}(s) & g_{r2}(s) & \cdots & g_{rm}(s) \end{bmatrix} \quad (1.6)$$

La matrice de fonctions de transfert est une généralisation naturelle au système multivariable de la notion de la fonction de transfert d'un système monovariable.

Remarque 1.1. La représentation par matrice de fonctions de transfert suppose que les conditions initiales sont nulles. Dans ce qui suit, on s'intéresse aux systèmes **linéaires** multivariables

carrés, c'est-à-dire le nombre d'entrées est le même que celui du nombre de sorties ($m = r$).

1.3 Problème d'interaction dans un système multivariable

Une grande partie des problèmes de commande peut être résolue par des correcteurs classiques de type PID et des correcteurs flous. L'objectif principal de l'analyse des interactions dans les systèmes multivariables est de choisir une configuration de commande dont les interactions entre les boucles sont très faibles, permettant ainsi d'appliquer la commande multiboucle, en s'intéressant au réglage individuel des boucles de commande [3].

Les méthodes d'analyse des interactions permettent d'évaluer le niveau du phénomène d'interaction fréquemment présent dans les systèmes multivariables. Ces méthodes ont donné lieu à des applications intéressantes de la commande multiboucle dans l'industrie dont les résultats sont concluants [3].

En résumé, l'analyse des interactions est une étape d'extrême importance pour la synthèse d'un système de commande multiboucle robuste. Les méthodes d'analyse des interactions développées n'est qu'un moyen pour dégrossir le système multivariable en le réduisant à un ensemble de sous-systèmes monovariabiles en parallèle, par la détermination de la meilleure configuration de commande parmi les $m!$ configurations possibles [3].

Pour éclaircir le phénomène d'interaction dans un système multivariable, considérons le système de la Figure 1.2. Lorsque la perturbation z_{11} affecte la sortie $y_1(t)$, cette dernière s'écarte de sa valeur désirée $y_1^d(t)$. Par conséquent, le correcteur $g_{c11}(s)$ génère une commande $u_1(t)$ pour annuler l'écart observé. Néanmoins, la commande $u_1(t)$ générée affecte également la sortie $y_2(t)$ à travers la transmittance $g_{21}(s)$, donc la sortie $y_2(t)$ s'écarte aussi de sa valeur désirée $y_2^d(t)$. Ceci force le correcteur $g_{c22}(s)$ à générer une commande $u_2(t)$ pour maintenir la sortie $y_2(t)$ à la consigne désirée $y_2^d(t)$. L'action correctrice du correcteur $g_{c22}(s)$ de la deuxième boucle (II) (la commande $u_2(t)$) affecte aussi la sortie $y_1(t)$ à travers la transmittance $g_{12}(s)$. Alors le maintien des sorties $y_1(t)$ et $y_2(t)$ à leurs consignes désirées, en dépit de la perturbation z_{11} qui doit être rejeter par le correcteur $g_{c11}(s)$, est une tâche très difficile.

Par cet exemple, on a montré comment une commande peut affecter plusieurs sorties et comment une perturbation affectant une sortie peut se propager dans le système et perturber les autres sorties. Ceci est dû essentiellement à l'existence des interactions entre les deux boucles (I) et (II) de la configuration de commande. Dans ce cas, les transmittances $g_{12}(s)$ et $g_{21}(s)$ influent sur les performances du système commandé en causant des interactions entre les deux boucles, dont leurs effets sont considérables sur la robustesse de la stratégie de commande considérée.

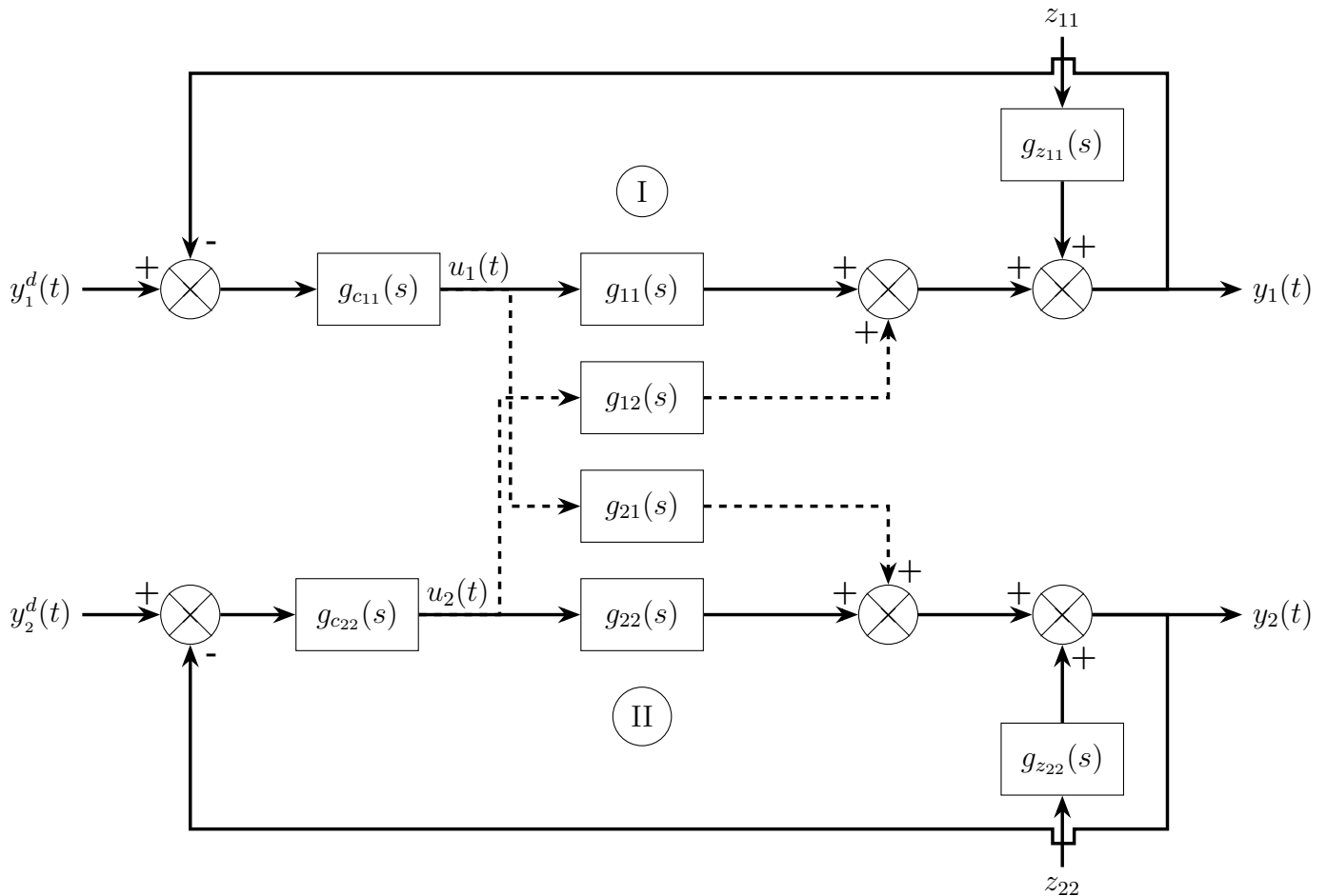


FIGURE 1.2 – Commande multiboucle.

Par conséquent, le phénomène d'interaction est une contrainte qui pose de sérieux problèmes pour la commande des systèmes multivariables, c'est pourquoi l'analyse des interactions joue un rôle de premier plan lors de la synthèse d'un système de commande multivariable, en particulier pour la conception d'une commande multiboucle.

1.4 Commande multiboucle

La commande multiboucle en automatique fait référence à un système de commande constituée de plusieurs boucles de commande monovariante dont le but est de commander plusieurs variables d'un système multivariable. Cela permet de prendre en compte les interactions entre les différentes variables et d'optimiser les performances du système. Par exemple, une commande multiboucle peut être utilisée pour commander la température, la pression et le débit dans un processus de chauffage.

La commande optimale est utilisée dans des applications industrielles où les variables du processus sont interdépendantes et nécessitent une régulation simultanée et coordonnée.

Pour la conception d'une commande multiboucle, on suit les étapes suivantes :

Étape 1 : Choix du couple, c'est-à-dire de la configuration de commande.

Étape 2 : Synthèse des correcteurs (détermination des paramètres).

1.4.1 Choix des couples

Lors de la conception d'une commande multiboucle pour un procédé à m entrées et m sorties, il existe $m!$ combinaisons possibles de couples entrée/sortie. Le choix des couples entrées/sorties, c'est-à-dire de la configuration de commande, est primordial et déterminant pour les performances recherchées ainsi que pour la stabilité du système en boucle fermée.

L'analyse par les gains relatifs permet d'évaluer le niveau d'interaction et de choisir la meilleure configuration de commande parmi les $m!$ configurations de commande possibles. La matrice des gains relatifs (RGA), en anglais Relative Gain Array, a été développée par **Bristol** en 1966. La RGA permet de déterminer la meilleure configuration de commande avec un faible niveau d'interaction.

Pour déterminer la RGA, on suit les étapes suivantes [3] :

Étape 1 : Calculer la matrice des gains statiques $G(0)$

$$G(0) = \lim_{s \rightarrow 0} G(s) \quad (1.7)$$

Étape 2 : Calculer la matrice des gains relatifs (RGA) comme suit :

$$RGA = \begin{bmatrix} \lambda_{11} & \lambda_{12} & \dots & \lambda_{1m} \\ \lambda_{21} & \lambda_{22} & \dots & \lambda_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_{m1} & \lambda_{m2} & \dots & \lambda_{mm} \end{bmatrix} = G(0) .* [G(0)^{-1}]^T \quad (1.8)$$

où $.*$ est le produit élément par élément (produit de Hadamard).

Remarque 1.2.

1. Les gains relatifs λ_{ij} ($i, j = 1, \dots, m$) sont adimensionnels (sans dimension). Par conséquent, le choix des unités des variables d'entrées et de sorties n'a aucune importance [8].

2. La somme des gains relatifs sur une ligne ou sur une colonne est égale à un, c'est-à-dire [8]

$$\sum_{i=1}^m \lambda_{ij} = \sum_{j=1}^m \lambda_{ij} = 1 \quad (1.9)$$

3. Si le gain relatif λ_{ij} d'un couple est négatif alors le système en boucle fermée peut devenir instable. Le couple entrée-sortie choisi doit donc présenter un gain relatif positif et supérieur à 0,5 [8].

4. Un gain relatif élevé ($\lambda_{ij} \gg 1$) indique une forte interaction (couplage) [8].

Une fois la matrice des gains relatifs est obtenue, et en tenant compte des points de la Remarque 1.2, pour le choix de la meilleure configuration de commande, on doit choisir des couples dont les gains relatifs sont *positifs* et *proches* de **1**.

1.4.2 Synthèse des correcteurs

Une fois la configuration de commande est déterminée, on passe à la synthèse des correcteurs des boucles de commande. Ainsi, en considérant les fonctions de transfert correspondantes aux couples entrée/sortie de la configuration, on synthétise pour chaque couple (fonction de transfert) un correcteur permettant de réaliser les spécifications en performances du cahier des charges. Pour ce faire, on suit les étapes suivantes :

Étape 1 : Choix de technique de commande et la structure du correcteur.

Étape 2 : Détermination des paramètres du correcteur.

Dans le milieu industriel, souvent on utilise des correcteurs PID. L'une des méthodes pratiques utilisées pour la conception d'une commande multiboucle est la technique du plus grand module (BLT) basée sur la méthode de **Ziegler-Nichols**.

Méthode de Ziegler-Nichols

La méthode de **Ziegler-Nichols** est une approche de réglage des correcteurs PID (Proportionnel-Intégral-Dérivé). Elle implique l'augmentation progressive du gain proportionnel jusqu'à ce que le système atteigne le point d'oscillation, puis la détermination des paramètres PID en fonction de cette réponse.

Pour déterminer les paramètres du correcteur PID, on suit les étapes suivantes [4] :

1. Initialisation du régulateur avec un gain proportionnel faible,
2. Augmentation progressive du gain jusqu'à l'apparition d'oscillations,

3. Utilisation des données de ces oscillations pour définir les paramètres du correcteur PID. Ces paramètres peuvent être déterminés à partir des valeurs critiques telles que le gain critique et la période d'oscillation ultime en utilisant les formules résumées dans le Tableau 1.1.

Remarque 1.3. La méthode de **Ziegler-Nichols** peut conduire à des réglages agressifs, et son application nécessite de la prudence pour éviter des réponses instables pour certains systèmes.

Ainsi, pour concevoir un correcteur du type PID pour un système monovarié décrit par une fonction de transfert $g(s)$. On détermine le gain critique K_{cr} du système en utilisant la formule suivante :

$$K_{cr} = \frac{1}{|g(j \omega_{cr})|} \quad (1.10)$$

et la période T_{cr} des oscillations entretenues en utilisant la formule suivante :

$$T_{cr} = \frac{2\pi}{1,2 \omega_{cr}} \quad (1.11)$$

où ω_{cr} est la pulsation critique du système qu'on détermine par la formule suivante :

$$\angle g(j \omega_{cr}) = -\pi \quad (1.12)$$

Méthode du plus grand module (BLT)

En présence des interactions, les correcteurs déterminés en utilisant la méthode de **Ziegler-Nichols** conduisent généralement à une instabilité en boucle fermée. Ainsi, les gains des correcteurs doivent être ajustés pour garantir la stabilité. Pour ce faire, on utilise la méthode du plus grand module.

La méthode du plus grand module, dite BLT (Biggest Log modulus Tuning), permet d'ajuster les paramètres des correcteurs monovariés du type PI en prenant en considération le phénomène d'interaction entre les boucles de la configuration choisie.

Correcteur	K_c	T_i	K_d
P	$0,5 K_{cr}$		
PI	$0,45 K_{cr}$	$T_{cr}/1,2$	
PID	$0,6 K_{cr}$	$T_{cr}/2$	$T_{cr}/8$

TABLE 1.1 – Réglages d'un PID recommandés par **Ziegler-Nichols**.

La BLT a l'avantage d'être à la fois facile à comprendre et à utiliser, il suffit de suivre les étapes suivantes [13] :

Étape 1 : Calculer les paramètres du correcteur PI pour chaque boucle individuelle en utilisant la méthode de **Ziegler-Nichols**. D'après le Tableau 1.1, on a :

$$K_{ZN} = 0,45 K_{cr} \quad (1.13)$$

$$T_{ZN} = \frac{T_{cr}}{1,2} \quad (1.14)$$

Étape 2 : Ajuster les paramètres du correcteurs PI en utilisant les expressions suivantes :

$$K_{BLT} = \frac{K_{ZN}}{F} \quad (1.15)$$

$$T_{BLT} = F \times T_{ZN} \quad (1.16)$$

où F est un facteur de dérèglage à déterminer en utilisant la formule suivante :

$$\max_{\omega} \left\{ 20 \log_{10} \left| \frac{W(j\omega)}{1 + W(j\omega)} \right| \right\} = 2m \quad (1.17)$$

avec

$$W(s) = 1 + |I + G_c(s) G(s)| \quad (1.18)$$

où $G_c(s)$ est la matrice de fonctions de transfert du correcteur multiboucle donnée comme suit :

$$G_c(s) = \begin{bmatrix} g_{c11}(s) & 0 & \dots & 0 \\ 0 & g_{c22}(s) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & g_{cmm}(s) \end{bmatrix} \quad (1.19)$$

Remarque 1.4. La résolution de l'équation (1.17) par rapport à F est souvent difficile. En pratique, F est compris entre 1,5 et 4 .

1.5 Exemple d'application

On considère le système multivariable suivant :

$$G(s) = \begin{bmatrix} \frac{12,8 e^{-s}}{16,7 s + 1} & -\frac{18,9 e^{-3s}}{21 s + 1} \\ \frac{6,6 e^{-7s}}{10,9 s + 1} & \frac{19,4 e^{-3s}}{14,4 s + 1} \end{bmatrix} \quad (1.20)$$

pour lequel on désire concevoir une commande multiboucle en utilisant la méthode BLT.

Étape 1 : Cette étape consiste à déterminer la meilleure configuration de commande pour notre système. Ainsi, en utilisant la relation (1.7), on obtient la matrice des gains statiques du système comme suit :

$$G(0) = \begin{bmatrix} 12,8 & -18,9 \\ 6,6 & 19,4 \end{bmatrix} \quad (1.21)$$

et en utilisant l'équation (1.8), il vient

$$RGA = G(0) \cdot [G(0)^{-1}]^T \quad (1.22)$$

$$= \begin{bmatrix} 0,6656 & 0,3344 \\ 0,3344 & 0,6656 \end{bmatrix} \quad (1.23)$$

On remarque que les gains relatifs de la diagonale de la matrice RGA sont positifs et proches de 1, par conséquent la meilleure configuration de commande est $[u_1 - y_1]$ et $[u_2 - y_2]$.

Étape 2 : Cette étape consiste à déterminer les paramètres des correcteurs PI. On commence par la détermination des paramètres des correcteurs PI en utilisant la méthode de **Ziegler-Nichols**. Pour notre exemple, pour les couples de la configuration choisie, la fonction de transfert est de la forme :

$$g(s) = \frac{K_s e^{-\theta s}}{1 + \tau s}$$

où K_s est le gain statique, θ est le retard, et τ est la constante de temps.

La réponse en fréquence est obtenue en posant $s = j\omega$, ce qui donne :

$$g(j\omega) = \frac{K_s e^{-\theta j\omega}}{1 + \tau j\omega} \quad (1.24)$$

Le gain critique K_{cr} pour le système, déterminé en utilisant la relation (1.10), est :

$$K_{cr} = \frac{\sqrt{1 + (\tau \omega_{cr})^2}}{K_s} \quad (1.25)$$

où la pulsation critique ω_{cr} est déterminé en utilisant la relation **(1.12)**, c'est-à-dire en résolvant l'équation algébrique suivante :

$$\angle K_s - \theta \omega_{cr} - \arctan(\tau \omega_{cr}) = 0 \quad (1.26)$$

Déterminons les correcteurs PI pour la configuration de commande choisie $[u_1 - y_1]$ et $[u_2 - y_2]$.

Pour la première boucle (le couple $[u_1 - y_1]$), on a la fonction de transfert :

$$g_{11}(s) = \frac{12,8 e^{-s}}{16,7 s + 1} \quad (1.27)$$

et par identification, on obtient $K_s = 12,8$, $\theta = 1$ et $\tau = 16,7$. Ainsi, en utilisant la relation **(1.26)**, on obtient :

$$\omega_{cr_1} = 1,6080 \text{ rad/s} \quad (1.28)$$

et la relation **(1.25)** donne

$$K_{cr_1} = 2,0994 \quad (1.29)$$

Les relations **(1.13)** et **(1.14)** donne pour la première boucle le réglage de Ziegler-Nichols suivant :

$$K_{ZN_1} = 0,9543 \quad (1.30)$$

$$T_{ZN_1} = 3,2562 \text{ s} \quad (1.31)$$

Pour la deuxième boucle (le couple $[u_2 - y_2]$), on a

$$g_{22}(s) = \frac{19,4 e^{-3s}}{14,4 s + 1} \quad (1.32)$$

Par identification, on a $K_s = 19,4$, $\theta = 3$ et $\tau = 14,4$. Ainsi, en utilisant la relation **(1.26)**, on obtient :

$$\omega_{cr_2} = 0,5644 \text{ rad/s} \quad (1.33)$$

et la relation **(1.25)** donne

$$K_{cr_2} = 0,4221 \quad (1.34)$$

et les relations (1.13) et (1.14) donne

$$K_{z_{N_2}} = 0,1919 \quad (1.35)$$

$$T_{z_{N_2}} = 9,2770 \text{ s} \quad (1.36)$$

Pour déterminer les correcteurs PI de la commande multiboucle, c'est-à-dire

$$G_c(s) = \begin{bmatrix} K_{11} \left(1 + \frac{1}{T_{11} s}\right) & 0 \\ 0 & K_{22} \left(1 + \frac{1}{T_{22} s}\right) \end{bmatrix}, \quad (1.37)$$

on utilise les relations (1.15) et (1.16), ce qui donne

$$K_{11} = \frac{K_{z_{N_1}}}{F} = \frac{0,9543}{F} \quad (1.38)$$

$$T_{11} = F T_{z_{N_1}} = 3,2562 F \quad (1.39)$$

$$K_{22} = \frac{K_{z_{N_2}}}{F} = \frac{0,1919}{F} \quad (1.40)$$

$$T_{22} = F T_{z_{N_2}} = 9,2770 F \quad (1.41)$$

Ainsi, en prenant $F = 2.55$, on obtient

$$g_{c_1}(s) = 0,3742 \left(1 + \frac{1}{8,3032 s}\right) \quad (1.42)$$

$$g_{c_2}(s) = 0,0752 \left(1 + \frac{1}{23,6563 s}\right) \quad (1.43)$$

Pour évaluer les performances du correcteur multiboucle, nous appliquerons les consignes désirées suivantes :

$$y_1^d(s) = \begin{cases} 0 & t < 200 \text{ s} \\ 1 & t \geq 200 \text{ s} \end{cases} \quad (1.44)$$

$$y_2^d(s) = 1 \quad (1.45)$$

Le résultat de la simulation obtenu est donné par la Figure 1.3. On remarque que chaque sortie suit la consigne imposée. On remarque aussi que l'effet de perturbation dû au changement de consigne de la sortie $y_1(t)$, effectué à l'instant $t = 200 \text{ s}$, est rejeté par le correcteur $g_{c_2}(s)$. En conclusion, en dépit des fortes interactions entre les variables du système, le correcteur multiboucle

permet d'assurer la poursuite de consigne et le rejet de perturbation.

1.6 Conclusion

Dans ce chapitre, nous avons présenté le principe de la commande multiboucle qui consiste à commander un système multivariable en utilisant des correcteurs multivariables. Pour la synthèse de cette technique de commande, l'étape d'analyse des interactions joue un rôle primordial pour le choix de la configuration de commande (couples entrées-sorties). Dans cette optique, nous avons présenté une des méthodes la plus utilisée qui est la matrice des gains relatifs (RGA).

Pour la conception des correcteurs PI dans un environnement multivariable, nous avons présenté la méthode du plus grand module (BLT). Le principe de cette dernière consiste à ajuster les paramètres des correcteurs PI, déterminés par la méthode de **Ziegler-Nichols**, en utilisant le facteur de déréglage. La détermination de ce paramètre passe par la résolution d'un problème d'optimisation souvent difficile particulièrement en présence des incertitudes paramétriques.

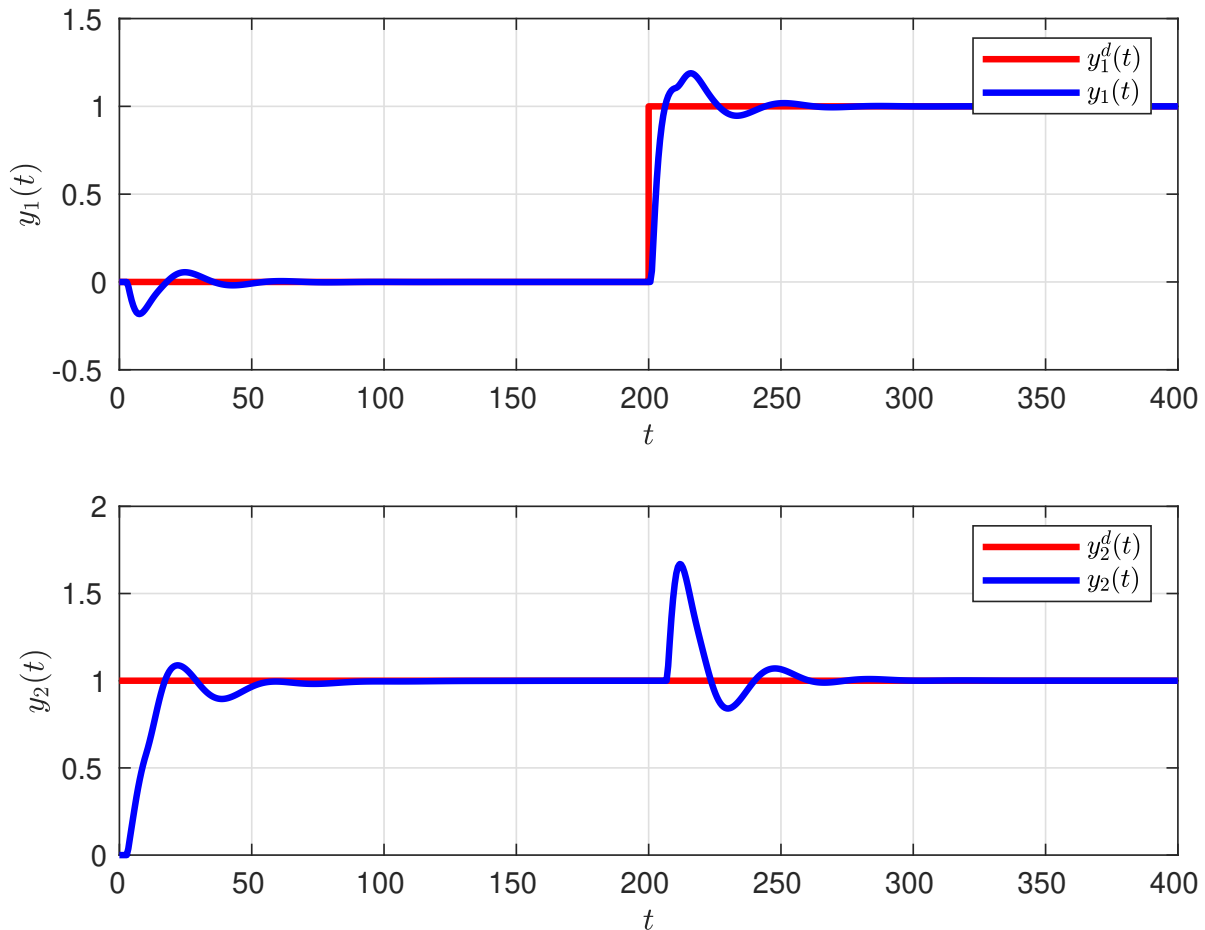


FIGURE 1.3 – Évolutions temporelles des sorties y_1 et y_2 .

Dans le chapitre suivant, nous nous intéressons à l'optimisation des fonctions mathématiques en présence des incertitudes sur les paramètres de la fonction.

Chapitre 2

Optimisation des fonctions en présence d'incertitudes

2.1 Introduction

Dans ce chapitre, nous allons introduire la notion d'optimisation robuste, c'est-à-dire l'optimisation en présence d'incertitudes sur les paramètres de la fonction objectif et/ou les contraintes.

On a commencé par définir c'est quoi un problème d'optimisation et la nature de l'optimum. Puis, on introduit la formulation mathématique d'un problème d'optimisation, et la classification des problèmes d'optimisation et des méthodes d'optimisation. A la fin du chapitre, nous allons introduire l'optimisation en présence d'incertitudes (optimisation robuste), et l'approche des scénarios utilisée pour résoudre ce type de problèmes.

2.2 Définition de l'optimisation

L'optimisation est un ensemble de techniques visant à obtenir la meilleure performance possible et cela en maximisant ou minimisant une fonction de coût avec ou sans contraintes. Dans le domaine de l'automatique, l'optimisation est une pratique visant à améliorer le fonctionnement d'un système ou d'un processus en termes de performances, de coûts, de productivité et de réduction des erreurs. Elle peut être appliquée à différents domaines et se fait en utilisant des outils logiciels, des méthodes mathématiques ou des techniques informatiques.

2.3 Nature de l'optimum

Un **optimum** ou **extremum** d'une fonction $f(x)$ est un maximum ou un minimum de la fonction (Figure 2.1), c'est-à-dire la valeur la plus grande ou la plus petite que prend la fonction sur

son ensemble de définition D .

Soit f une fonction de $D \subset \mathbb{R}^n$ ($f : D \subset \mathbb{R}^n \rightarrow \mathbb{R}$). On distingue trois types pour l'optimum.

1. Optimum local : f admet un minimum (resp. un maximum) local en $\hat{x} \in D$, si et seulement s'il existe un voisinage $V_\varepsilon(\hat{x})$ de \hat{x} , tel que :

$$\forall x \in V_\varepsilon(\hat{x}), \quad f(\hat{x}) \leq f(x) \quad (\text{resp. } f(\hat{x}) \geq f(x))$$

2. Optimum local strict : f admet un minimum (resp. un maximum) local au sens strict en $\tilde{x} \in D$, si et seulement s'il existe un voisinage $V_\varepsilon(\tilde{x})$ de \tilde{x} , tel que :

$$\forall x \in V_\varepsilon(\tilde{x}), \quad x \neq \tilde{x}, \quad f(\tilde{x}) < f(x) \quad (\text{resp. } f(\tilde{x}) > f(x))$$

3. Optimum global : f admet un minimum (resp. un maximum) global en $x^* \in D$ si :

$$\forall x \in D, \quad f(x^*) \leq f(x) \quad (\text{resp. } f(x^*) > f(x))$$

Remarque 2.1.

- Si f est convexe sur D , alors tout minimum local est un minimum global.
- Si f est strictement convexe, alors tout minimum local devient non seulement global mais aussi unique.

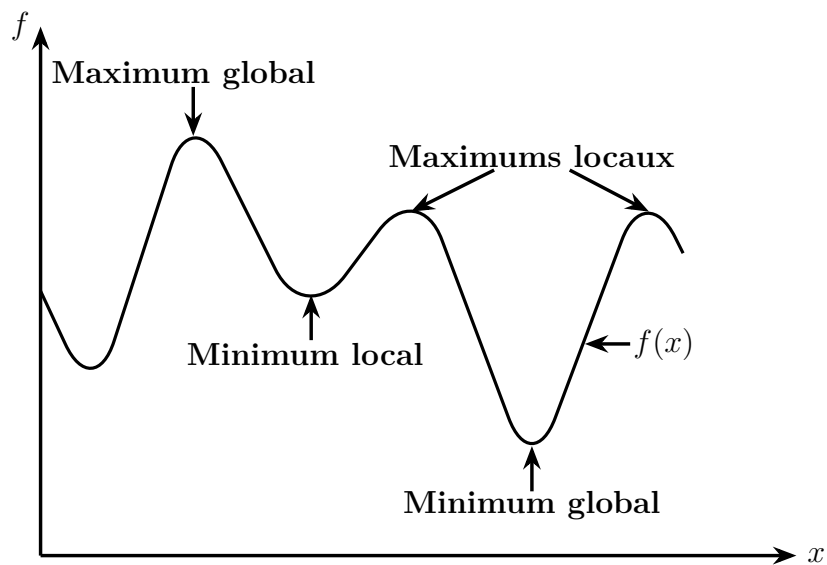


FIGURE 2.1 – Minima et maxima d'une fonction $f(x)$.

2.4 Forme mathématique d'un problème d'optimisation

Considérons une fonction scalaire de plusieurs variables x_1, x_2, \dots, x_n , appelées variables de décision, notée $f(x)$, et appelée fonction objectif ou critère. Le vecteur de variables de décision, noté $x = (x_1, x_2, \dots, x_n)^T$, doit appartenir à un domaine donné $D \in \mathbb{R}^n$. Ce dernier est défini par des relations de contraintes du type égalité [6]

$$d_i(x) = 0, \quad i = 1, \dots, p \quad (2.1)$$

et/ou inégalité

$$h_j(x) \leq 0, \quad j = 1, \dots, q \quad (2.2)$$

Nous présentons un problème d'optimisation mathématiquement sous la forme suivante [6] :

$$\begin{aligned} & \min_x f(x) \\ \text{sujet à :} & \\ & d_i(x) = 0, \quad i = 1, \dots, p \\ & h_j(x) \leq 0, \quad j = 1, \dots, q \end{aligned} \quad (2.3)$$

où

- $x = (x_1, x_2, \dots, x_n)$ vecteur de variables de décision (ou d'optimisation), et n est le nombre de variable de décision.
- $f(x)$ fonction objectif que nous souhaitons optimiser.
- $d_i(x)$ contraintes du type égalité.
- $h_j(x)$ contraintes du type inégalité.

Remarque 2.2. Moyennant le changement de fonctions $f(x)$ en $-f(x)$, on peut passer d'un problème de minimisation à un problème de maximisation et vice versa. Minimiser $f(x)$ revient à maximiser $-f(x)$, c'est-à-dire [6] :

$$\min_x f(x) = -\max_x [-f(x)] \quad (2.4)$$

2.5 Classification des problèmes d'optimisation

Suivant la nature de la fonction objectif et des contraintes qui définissent le domaine des solutions admissibles D , le problème d'optimisation correspondant porte des noms divers. On distingue plus particulièrement les cas résumés dans le Tableau 2.1 [6]. Dans ce qui suit, on présente des exemples pour chaque problème d'optimisation.

2.5.1 Problème d'optimisation linéaire

Dans un problème d'optimisation linéaire, les fonctions $f(x)$, $d_i(x)$ et $h_j(x)$ sont toutes linéaires, c'est-à-dire des polynômes d'ordre 1, de la forme [10] :

$$\sum_{i=1}^{n-1} a_i x_i + b. \tag{2.5}$$

Ce problème peut être résolu efficacement avec des algorithmes tels que la méthode du simplexe ou la méthode du point intérieur [10]. Par exemple :

$$\begin{aligned} & \min_x \quad 2x_1 - x_2 \\ & \text{sujet à :} \\ & \quad 3x_1 - 2x_2 \geq -2 \\ & \quad 2x_1 - 4x_2 \leq 3 \\ & \quad x_1, x_2 \geq 0 \end{aligned} \tag{2.6}$$

2.5.2 Problème d'optimisation non linéaire

Un problème d'optimisation est dit non linéaire si au moins une des fonctions du problème n'est pas linéaire. Les méthodes courantes pour résoudre ce problème sont la méthode de Newton, la méthode de quasi-Newton et la méthode du gradient dans le cas où le problème est sans contraintes. Dans le cas d'un problème d'optimisation avec contraintes, on peut utiliser la méthode des multiplicateurs de Lagrange (contraintes du type égalité) et la méthode de Kuhn-Tucker (contraintes du type inégalité) [10]. Par exemple :

$$\begin{aligned} & \min_x \quad (x_1 - 0,5)^2 + (x_2 - 1)^2 \\ & \text{sujet à :} \\ & \quad x_1^2 + 2x_2^2 - 4 \leq 0 \\ & \quad -2 \leq x_1 \leq 2; \quad -\sqrt{2} \leq x_2 \leq \sqrt{2} \end{aligned} \tag{2.7}$$

		Fonction objectif $f(x)$		
		Linéaire	Quadratique	Non linéaire
Contraintes $d_i(x), h_j(x)$	Linéaires	Linéaire	Quadratique	Non linéaire
	Quadratiques	Non linéaire	Non linéaire	Non linéaire
	Non linéaires	Non linéaire	Non linéaire	Non linéaire

TABLE 2.1 – Classification des problèmes d'optimisation.

2.5.3 Problème d'optimisation quadratique

Le problème d'optimisation quadratique est un problème où la fonction $f(x)$ est quadratique, c'est-à-dire un polynôme d'ordre 2 dont la forme générale est $x^T Q x$ (Q est une matrice de dimension $n \times n$) et les contraintes $d_i(x)$ et $h_j(x)$ sont des fonctions linéaires. Le problème d'optimisation quadratique est un cas particulier du problème d'optimisation non linéaire. Ce problème peut être résolu analytiquement [10]. Par exemple :

$$\begin{aligned} & \min_x \quad 2x_1^2 + 3x_2 \\ \text{sujet à :} & \\ & x_1 + 4x_2 \leq 10 \\ & x_1, x_2 \geq 0 \end{aligned} \tag{2.8}$$

2.5.4 Problème d'optimisation convexe

Un problème d'optimisation est dit convexe si la fonction $f(x)$ et les contraintes $d_i(x)$ et $h_j(x)$ sont convexes. La convexité est souvent utilisée pour garantir l'unicité et l'existence de la solution. Une fonction est dite convexe, si la ligne entre deux points du graphe de la fonction se situe entièrement au-dessus du graphe. De manière équivalente, un ensemble est convexe si pour tout couple de points de l'ensemble, le segment de droite les reliant est entièrement contenu dans l'ensemble [11]. Par exemple :

$$\begin{aligned} & \min_x \quad 2x_1^2 + 5x_2^2 \\ \text{sujet à :} & \\ & x_1 + 2x_2 \geq 3 \\ & x_1, x_2 \geq 0 \end{aligned} \tag{2.9}$$

2.6 Classification des méthodes d'optimisation

Pour l'optimisation continue, on sépare sommairement le cas linéaire (qui relève notamment de la programmation linéaire pour lequel la solution est facile à obtenir) du cas non linéaire. Pour certains problèmes qui vérifient la propriété de convexité, la solution peut être obtenue en utilisant une méthode locale qui exploite, ou non, les gradients de la fonction objectif. Néanmoins, la plupart des problèmes d'optimisation sont classés difficiles, car le nombre de minima locaux est très élevé, alors le recours à une méthode globale s'impose.

Pour le traitement des contraintes, on a recours à la méthode des multiplicateurs de Lagrange dans le cas des contraintes du type égalité, et la méthode de Kuhn-Tucker ou la méthode de fonctions d'écart dans le cas de contraintes du type inégalité. La méthode de fonctions de pénalisation

constitue une approche élégante pour traiter les deux cas. Cette méthode permet de ramener un problème d'optimisation posé avec contraintes à un problème sans contraintes, puis d'appliquer les méthodes d'optimisation numériques.

Dans la littérature, on distingue trois classes pour les méthodes d'optimisation numériques :

- Les méthodes exactes (appelées encore classiques ou déterministes).
- Les méthodes heuristiques (appelées encore méthodes stochastiques).
- Les méthodes hybrides (coopération entre une méthode exacte et une méthode heuristique).

2.6.1 Méthodes exactes

Ces méthodes requièrent des propriétés mathématiques restrictives de la fonction objectif à optimiser telle que la continuité, la différentiabilité, et la convexité. Ces méthodes permettent de localiser la solution avec une très grande précision au prix d'un nombre d'itérations important, c'est-à-dire elles nécessitent un temps de calcul parfois énorme. Comme exemples de méthodes exactes, on peut citer les méthodes du gradient et de Newton, et leurs variantes. Néanmoins, en pratique, il est très difficile de savoir si la fonction objectif satisfait ou non de telles propriétés mathématiques. De plus, la plupart des fonctions sont multimodales (plusieurs maximums et minimums), discontinues et non dérivables. Le principe de ces méthodes est résumé par l'algorithme suivant [6] :

Étape 1 : Initialisation $k = 0$, $x^{(k)} = x^{(0)}$ et calcul de $f(x^{(0)})$.

Étape 2 : Pour $k = k + 1$, on calcule la variation de $x^{(k)}$ donnée par le vecteur suivant :

$$\Delta x^{(k)} = \begin{bmatrix} \Delta x_1^{(k)} \\ \Delta x_2^{(k)} \\ \vdots \\ \Delta x_n^{(k)} \end{bmatrix} \quad (2.10)$$

en utilisant une procédure appropriée selon la méthode utilisée (par exemple la méthode du gradient, méthode du gradient conjugué ou la méthode de Newton).

Étape 3 : Calcul $x^{(k)} = x^{(k-1)} + \Delta x^{(k)}$

Étape 4 : Calcul de $f(x^{(k)})$ et son gradient noté $\nabla_x f(x^{(k)})$, et vérification de la convergence en utilisant un critère approprié (par exemple la norme du gradient de la fonction à optimiser doit être inférieur à une valeur ε , c'est-à-dire $(\|\nabla_x f(x^{(k)})\| < \varepsilon)$. Si la convergence est assurée on passe à l'étape 5, sinon on revient à l'étape 2.

Étape 5 : La solution est $x^* = x^{(k)}$.

Par exemple, dans le cas de la méthode de Newton, la variation de $x^{(k)}$ notée $\Delta x^{(k)}$, est donnée comme suit :

$$\Delta x^{(k)} = - \left[\nabla_x^2 (f(x^{(k)})) \right]^{-1} \nabla_x (f(x^{(k)})) \quad (2.11)$$

2.6.2 Méthodes heuristiques

Ces dernières années, on marque l'arrivée d'une nouvelle classe de méthodes, nommées méthodes heuristiques ou stochastiques qui constituent une alternative très intéressante aux méthodes exactes. Les méthodes heuristiques sont un ensemble d'algorithmes d'optimisation visant à résoudre des problèmes d'optimisation difficile. Ces méthodes vont chercher un optimum de façon aléatoire (au hasard). En général, elles procèdent par voisinages successifs. De façon plus précise, une solution étant obtenue à l'itération k , à l'itération $k + 1$ on cherche au hasard une solution meilleure dans un voisinage précédent. Les itérations successives permettent de passer d'une solution à une meilleure solution. L'algorithme s'arrête après avoir vérifié un critère d'arrêt, généralement le nombre d'itérations ou une précision souhaitée.

Parmi les heuristiques, on retrouve les heuristiques de *voisinage*, qui font progresser une seule solution à la fois (par exemple la méthode de recuit simulé et la recherche tabou), et les heuristiques *distribuées*, qui manipulent en parallèle toute une population de solutions (par exemples les algorithmes génétiques et les essaims de particules). L'inconvénient majeur de ces méthodes est, qu'on ne peut garantir leur convergence vers la solution globale que d'une manière asymptotique, c'est-à-dire elles permettent de localiser un $x \in V_\varepsilon(x^*)$ (voisinage de l'optimum global). Bien que ces méthodes puissent être adaptées à tout type de problème d'optimisation, mais elles sont souvent moins puissantes que les méthodes exactes sur certains types de problèmes. Elles ne garantissent pas non plus la localisation de l'optimum global en un temps fini. Cependant, un grand nombre de problèmes réels n'est pas optimisable efficacement par des approches purement mathématiques, les heuristiques peuvent alors être utilisées avec profit, par exemple, dans le cas où l'expression analytique de la fonction objectif à optimiser est inconnue ou cette dernière est non différentiable [6].

2.6.3 Méthodes hybrides

Les méthodes hybrides profitent des avantages des méthodes exactes et heuristiques. Le principe de ces méthodes consiste à coopérer une méthode heuristique et une méthode exacte pour localiser l'optimum global recherché avec une précision déterminée. En général, la recherche de l'optimum global commence par l'utilisation d'une heuristique qui localise un x appartenant au voisinage de l'optimum globale, c'est-à-dire $V_\varepsilon(x^*)$, puis on passe le relais à une méthode d'optimisation exacte qui permet de localiser l'optimum globale x^* avec une bonne précision [6].

2.7 Optimisation en présence des incertitudes

L'optimisation en présence d'incertitudes est un domaine d'étude qui vise à trouver des solutions robustes et fiables malgré les incertitudes dans les données du problème d'optimisation. Pour résoudre ce type de problème, plusieurs méthodes peuvent être utilisées, parmi lesquelles, on peut citer :

2.7.1 Méthode de Monte Carlo

Cette méthode est utilisée pour générer des échantillons aléatoires à partir de distributions probabilistes, permettant ainsi d'estimer les performances des solutions dans différentes conditions.

2.7.2 Optimisation par intervalles

Plutôt que de travailler avec des valeurs exactes, l'optimisation par intervalles utilise des intervalles pour représenter les incertitudes, ce qui permet de trouver des solutions qui sont robustes dans un large éventail de situations.

2.7.3 Optimisation multi-objectif

Au lieu de chercher une solution optimale, cette approche cherche un ensemble de solutions qui représentent un compromis entre plusieurs objectifs souvent contradictoires. Cela peut aider à trouver des solutions qui sont plus résilientes face à l'incertitude.

2.7.4 Méthode du pire cas

Dans ce cas, on essaie de prendre en considération la totalité du domaine de l'incertitude. Cette approche consiste à résoudre un problème d'optimisation *semi-infini* qui est en général très difficile à résoudre (classé NP difficile).

Ces méthodes, parmi d'autres, peuvent être utilisées individuellement ou combinées pour résoudre des problèmes d'optimisation dans des environnements où les incertitudes sont présentes. Pour notre étude, on s'intéresse à une autre méthode bien particulière qui est la **méthode des scénarios**. Avant de parler de cette dernière, on va d'abord introduire la forme mathématique d'un problème d'optimisation robuste.

2.8 Optimisation robuste

Elle consiste à trouver la solution d'un problème d'optimisation dont les paramètres sont incertains mais bornés. Ces incertitudes peuvent provenir de divers facteurs tels que les erreurs de mesure, les variations des conditions environnementales, etc. Contrairement à l'optimisation classique qui suppose que les données sont parfaitement connues, l'optimisation robuste cherche à trouver une solution qui sera satisfaisante même en présence d'incertitudes.

Un problème d'optimisation en présence d'incertitudes peut être formulé de manière générale comme un problème *min-max* dont l'objectif est d'optimiser le critère dans le pire scénario possible pour les incertitudes. Le problème prend la forme suivante [5] :

$$\begin{aligned} & \min_x \max_{\delta} f(x, \delta) \\ \text{sujet à :} & \\ & d_i(x, \delta) = 0, \quad i = 1, \dots, p \\ & h_j(x, \delta) \leq 0, \quad j = 1, \dots, q \\ & \delta \in I_n \end{aligned} \tag{2.12}$$

où δ est le vecteur des paramètres incertains du système, et I_n est l'ensemble des incertitudes.

2.8.1 Méthode des scénarios

La méthode des scénarios consiste à relaxer le problème d'optimisation robuste en considérant un ensemble fini de scénarios représentatifs des incertitudes possibles. Les scénarios sont ensuite utilisés pour transformer le problème d'optimisation *min-max* en un problème d'optimisation *standard*, facile à résoudre. Le principe consiste à optimiser la fonction objectif pour l'ensemble de scénarios considérés tout en garantissant la robustesse de la solution face aux incertitudes non prises en compte lors de la résolution. Les étapes à réaliser pour résoudre ce type de problème sont [5] :

Étape 1 : Formulation du problème d'optimisation *min-max* sous forme d'un problème d'optimisation *semi-infini* : Le nouveau problème d'optimisation est caractérisé par un nombre de variables de décision fini et un nombre de contraintes infini, d'où l'appellation optimisation *semi-infini*. En général, ce problème d'optimisation prend la forme suivante :

$$\begin{aligned} & \min_{x, z} z \\ \text{sujet à :} & \\ & f(x, \delta) \leq z \\ & d_i(x, \delta) = 0, \quad i = 1, \dots, p \\ & h_j(x, \delta) \leq 0, \quad j = 1, \dots, q \end{aligned} \tag{2.13}$$

où z est une nouvelle variable de décision.

Ce type de problème est impossible à résoudre vu le nombre de contraintes à considérer qui est infini. Pour cela, on doit le convertir en un problème d'optimisation *standard* avec un nombre fini de contraintes.

Étape 2 : Conversion du problème d'optimisation *semi-infini* en un un problème d'optimisation *standard* : Cette étape consiste à transformer le problème d'optimisation *semi-infini*, avec un nombre infini de contraintes, en un autre problème mais avec un nombre fini de contraintes. Cette transformation implique la sélection de N échantillons indépendants et identiquement distribués $(\delta^{(1)}, \delta^{(2)}, \dots, \delta^{(N)})$ à partir de l'ensemble des incertitudes I_n , suivant une probabilité de distribution donnée. Ensuite, on ne considère que les N contraintes correspondantes à ces échantillons pour formuler le problème d'optimisation *standard* qui est de la forme générale

$$\begin{aligned} & \min_{x,z} z \\ \text{sujet à :} & \\ & f(x, \delta^{(i)}) \leq z, \quad i = 1, \dots, N \\ & d_i(x, \delta^{(k)}) = 0, \quad i = 1, \dots, p; k = 1, \dots, N \\ & h_j(x, \delta^{(k)}) \leq 0, \quad j = 1, \dots, q; k = 1, \dots, N \end{aligned} \tag{2.14}$$

Étape 3 : Résolution du problème d'optimisation *standard*. Pour résoudre ce dernier, on utilise les méthodes d'optimisation en présence de contraintes du type inégalité [10].

Remarque 2.3. Pour que la solution du problème d'optimisation *standard* soit une solution du problème d'optimisation *semi-infini* original, le nombre d'échantillons N à considérer doit être choisi en utilisant la formule suivante [12] :

$$N \geq \frac{2}{\varepsilon} \left(\ln \left(\frac{1}{\gamma} \right) + n \right), \quad 0 < \varepsilon, \gamma < 1 \tag{2.15}$$

où n est le nombre de variables de décision, γ est le paramètre de confiance de la solution, et ε est le paramètre de risque de violation des contraintes.

2.8.2 Exemple d'application

$$\min_x \max_{\delta} f(x, \delta) = (x_1 - a)^2 + 2(x_2 - b)^2 \tag{2.16}$$

avec

$$\delta = \begin{bmatrix} a \\ b \end{bmatrix}, \quad 1 \leq a \leq 5, \text{ et } -4 \leq b \leq -1 \quad (2.17)$$

Étape 1 : Formulation du problème d'optimisation *min-max* sous forme d'un problème d'optimisation *semi-infini* :

$$\begin{aligned} & \min_{x,z} z \\ \text{sujet à :} & \\ & (x_1 - a)^2 + 2(x_2 - b)^2 \leq z \\ & 1 \leq a \leq 5 \\ & -4 \leq b \leq -1 \end{aligned} \quad (2.18)$$

Étape 2 : Conversion du problème d'optimisation *semi-infini* en un un problème d'optimisation *standard* :

$$\begin{aligned} & \min_{x,z} z \\ \text{sujet à :} & \\ & (x_1 - a^{(i)})^2 + 2(x_2 - b^{(i)})^2 \leq z, \quad i = 1, \dots, N \end{aligned} \quad (2.19)$$

Étape 3 : Résolution du problème d'optimisation *standard*. Pour cela on s'appuie sur la relation (2.15), en prenant $\varepsilon = 0,01$ et $\gamma = 10^{-10}$. Dans notre exemple, on a trois variables de décision (x_1, x_2 et z), par conséquent $n = 3$. La formule (2.15) donne

$$N \geq \frac{2}{0,01} \left(\ln \left(\frac{1}{10^{-10}} \right) + 3 \right) \quad (2.20)$$

$$N \geq 5206$$

En prenant $N = 5206$, la résolution du problème d'optimisation (2.19) en utilisant la fonction **ga** (algorithme génétique) de Matlab, on obtient la solution suivante :

$$x_1^* = 4,8858, x_2^* = -3,425 \text{ et } z^* = 0,029 \quad (2.21)$$

2.9 Conclusion

Ce chapitre a été consacré à l'optimisation des fonctions mathématiques. Nous avons présenté les différentes notions de base relatives à l'optimisation *standard*. Ensuite, nous avons présenté les différents types de problèmes d'optimisation et les différentes méthodes de résolution de problèmes d'optimisation.

Après, nous avons introduit l'optimisation robuste, c'est-à-dire en présence des incertitudes sur les données du problème. Ce type de problème d'optimisation *semi-infini* peut être résolu par l'approche des scénarios dont le principe consiste à le convertir en un problème d'optimisation *standard* qu'on peut résoudre par les méthodes classiques.

Dans le chapitre suivant, nous allons formuler la conception d'un correcteur PID sous forme d'un problème d'optimisation en utilisant la méthode d'Åström.

Chapitre 3

Conception d'un correcteur PID basée sur l'optimisation

3.1 Introduction

En pratique, le correcteur PID reste le plus utilisée pour la commande des systèmes dynamique. Dans ce chapitre, on s'intéresse à la conception de correcteurs PID basée sur l'optimisation de critères de performances.

On commence par la présentation des critères utilisées pour évaluer les performances d'un asservissement. Par la suite, nous nous intéressons au choix des actions du correcteur PID, et à l'optimisation de ses paramètres basée sur la méthode d'Åström. A la fin du chapitre, nous présentons un exemple d'application.

3.2 Critères de performance

L'objectif fondamental d'un correcteur dans un système en boucle fermée est de minimiser l'écart entre la sortie du système et une valeur de consigne préétablie. Cet écart peut résulter d'un changement de consigne ou de perturbations extérieures affectant le système. Pour déterminer un réglage optimal du régulateur, il est crucial d'évaluer à la fois l'amplitude maximale de l'écart et la durée nécessaire à sa réduction après une perturbation ou un changement de consigne. Ainsi, le correcteur joue un rôle essentiel en garantissant que la réponse du système présente des caractéristiques dynamiques et stationnaires adéquates.

Théoriquement, des critères sont utilisés pour évaluer l'efficacité d'un réglage particulier. Parmi les plus fréquemment utilisés, on peut citer [7] :

- **Intégrale du carré de l'erreur (ICE)** : Ce critère est généralement moins influencé par les petites erreurs que par les fortes erreurs, ce qui se traduit souvent par une réponse caractérisée par un faible dépassement mais un temps de stabilisation relativement long. Mathématiquement, le critère prend la forme

$$\text{ICE} = \int_0^{+\infty} e^2(t) dt \quad (3.1)$$

- **Intégrale de la valeur absolue de l'erreur (IAE)** : Comparé au critère ICE, l'IAE accorde une importance accrue aux erreurs de faible amplitude. Il est exprimé comme suit

$$\text{IAE} = \int_0^{+\infty} |e(t)| dt \quad (3.2)$$

- **Intégrale de la valeur absolue de l'erreur pondérée par le temps (ITAE)** : Ce critère favorise la réduction de l'erreur statique au détriment d'un dépassement initial potentiellement notable, vu qu'il a lieu pour de petites valeurs de t . Mathématiquement, il est exprimé comme suit

$$\text{ITAE} = \int_0^{+\infty} t |e(t)| dt \quad (3.3)$$

Ainsi, le problème avec la conception du correcteur PID revient alors à sélectionner le type de correcteur (les actions) ainsi que ses paramètres afin de minimiser l'un des critères de performances présentés. Ces critères peuvent être classés selon leurs caractéristiques spécifiques [8] :

- L'utilisation de l'IAE conduit à des systèmes bien amortis; pour un système de second ordre, le facteur d'amortissement résultant serait d'environ 0,7; de même que pour l'ITAE. En revanche, avec l'ICE, il serait d'environ 0,5. Ce dernier est peu sensible aux variations de paramètres, contrairement à l'ITAE qui l'est.
- Pour éliminer les grandes erreurs (valeur > 1), l'ICE est préférable à l'IAE car le terme d'erreur est pris en compte au carré.
- Pour éliminer les petites erreurs (valeur < 1), l'IAE est préférable à l'ICE.
- Pour éliminer les erreurs persistantes sur une longue durée, l'ITAE est le critère optimal car le terme t amplifie les petites erreurs persistantes. C'est souvent le critère privilégié car il est le plus conservateur, généralement associé à des dépassements et des oscillations moins prononcés.

3.3 Optimisation d'un correcteur PID

L'optimisation d'un correcteur PID est un processus crucial dans la conception de systèmes de commande efficaces. Elle vise à déterminer les valeurs optimales des coefficients proportionnel,

intégral et dérivé du correcteur PID afin d'atteindre les performances souhaitées du système de commande.

3.3.1 Choix du correcteur

Le choix du correcteur détermine en grande partie les performances du système de commande. Les correcteurs PID sont largement utilisés en raison de leur simplicité et de leur efficacité pratique. Cependant, d'autres types de correcteurs, tels que les correcteurs PID fractionnaires ou adaptatifs, et flous peuvent également être envisagés en fonction des caractéristiques spécifiques du système à commander et des exigences de performance.

Dans ce qui suit, on présente les contributions de chaque action d'un correcteur PID (Figure 3.1).

1. Action proportionnelle

- Accélère la réponse du système,
- Produit un écart à l'état stationnaire sauf pour les systèmes qui possèdent un terme intégrateur ($1/s$) dans leurs fonctions de transfert.

2. Action intégrale

- Élimine l'écart à l'état stationnaire,
- L'élimination de l'écart se fait en général au prix de dépassements souvent importants,
- Les réponses sont lentes, avec de longues oscillations,
- L'augmentation du gain K rend le comportement plus oscillatoire et peut conduire à des instabilités.

3. Action dérivée

- Anticipe les erreurs futures,
- Introduit un effet stabilisant dans la réponse en boucle fermée ,
- Rend le système en boucle fermée rapide en augmentant la bande passante. Par conséquent, le système devient sensible aux bruits et aux signaux de hautes fréquences.

3.3.2 Directives pour le choix du type du correcteur

1. Lorsque l'écart à l'état stationnaire est acceptable ou lorsque le processus intègre un terme intégral, on peut utiliser un correcteur P [8].

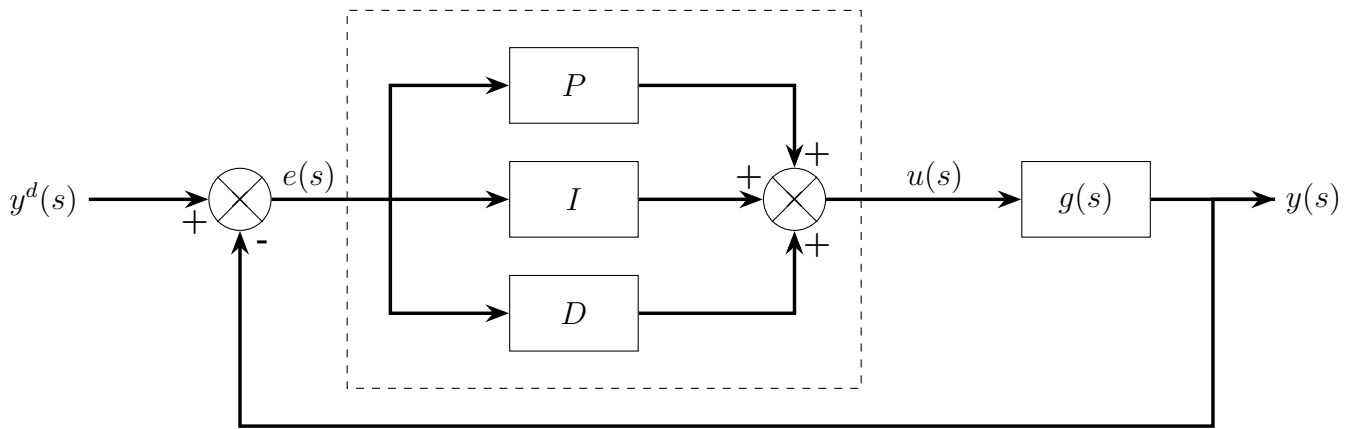


FIGURE 3.1 – Boucle de régulation avec un régulateur PID.

2. Lorsque l'écart à l'état stationnaire est trop important, et les retards induits par l'action intégrale ne sont pas préjudiciables, on peut opter pour une correction du type PI. Généralement, ce type de correcteur est recommandé pour les systèmes de dynamique de premier ordre [8].
3. Dans les autres cas, un correcteur PID est recommandé. Ce dernier permet d'avoir une réponse en boucle fermée plus rapide et une robustesse du régulateur. Ce type de correcteur convient bien pour la commande des systèmes de dynamique de deuxième ordre, qui peuvent être difficiles à déterminer, ou présentant des constantes de temps d'ordres de grandeur différents. L'action dérivée permet de limiter le gain [8].
4. Les systèmes présentant les difficultés suivantes sont moins adaptés aux correcteurs PID [8] :
 - Les systèmes avec un retard pur.
 - Les systèmes avec des modes oscillatoires.
 - Les systèmes avec des paramètres à fortes variations.
 - Les systèmes pour lesquels on cherche à contrôler une variable de qualité.

Il existe plusieurs méthodes pour optimiser les paramètres d'un correcteur PID. Parmi les plus couramment utilisées, on peut citer [7][8] :

- Réglage manuel.
- Méthode de Ziegler-Nichols.
- Méthode de Tyreus-Luyben.
- Méthode de Cohen-Coon.
- Méthode de Takahashi.
- Méthode de Graham-Lathroup.

Dans ce qui suit, on s'intéresse à la conception d'un correcteur PID basée sur l'optimisation du critère ICE.

3.3.3 Optimisation d'un correcteur PID basée sur la méthode d'Åström

La méthode d'Åström, nommée d'après le célèbre ingénieur de contrôle Karl Johan Åström, est une approche bien connue pour l'évaluation analytique de certains critères de performances. Cette méthode repose sur le calcul de la valeur du critère en fonction des paramètres du système à commander. Bien que cette méthode puisse être efficace, elle nécessite souvent un processus itératif et peut être sujette à des limitations en termes de complexité du système et de ressources de calcul disponibles.

La combinaison d'une sélection judicieuse du type du correcteur et de la méthode d'Åström peut conduire à des systèmes de contrôle robustes et performants, répondant aux exigences de précision et de stabilité dans une variété d'applications pratiques.

Pour expliquer le principe de cette méthode, on considère le cas de la minimisation du critère ICE donné par l'équation (3.1). La formulation du problème d'optimisation est la suivante :

$$\begin{aligned} \min_{K_c, T_i, T_d} \text{ICE} &= \int_0^{+\infty} e^2(t) dt \\ \text{sujet à :} & \\ K_c &> 0 \\ T_i &> 0 \\ T_d &> 0 \end{aligned} \tag{3.4}$$

Pour convertir ce problème d'optimisation dynamique en un problème d'optimisation statique, on propose d'évaluer analytiquement le critère ICE en utilisant la méthode d'Åström.

La méthode d'Åström permet un calcul rapide et précis de la valeur de l'ICE à partir de la transformée de Laplace de l'erreur de poursuite $e(s)$, facilitant ainsi l'optimisation des paramètres du correcteur [5]. Pour appliquer cette méthode, on suit les étapes suivantes [9] :

Étape 1 : On commence par déterminer la fonction de transfert en boucle ouverte du système $g(s)$, et celle du correcteur $g_c(s)$ à utiliser pour la commande du système. Ces dernières sont données comme suit :

$$g(s) = \frac{y(s)}{u(s)} \tag{3.5}$$

$$g_c(s) = \frac{u(s)}{e(s)} = K_c \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (3.6)$$

où y , u , et e sont respectivement la sortie, la commande du système à commander, et l'erreur de poursuite.

Étape 2 : Ensuite, on calcule l'erreur de poursuite $e(s)$ du système en boucle fermée.

$$e(s) = y^d(s) - y(s) = y^d(s) - e(s) g_c(s) g(s) \quad (3.7)$$

où $y^d(s)$ est la sortie désirée.

De l'équation (3.7), on a

$$e(s) = \frac{y^d(s)}{1 + g_c(s) g(s)} \quad (3.8)$$

Étape 3 : On exprime $e(s)$ sous la forme suivante :

$$e(s) = \frac{b_1 s^{n-1} + b_2 s^{n-2} + \dots + b_{n-1} s + b_n}{a_0 s^n + a_1 s^{n-1} + \dots + a_{n-1} s + a_n} \quad (3.9)$$

Étape 4 : Par la suite, pour le numérateur et le dénominateur de l'équation (3.9), on construit les Tableaux 3.1 et 3.2. Pour établir les Tableaux 3.1 et 3.2, on utilise les formules suivantes :

$$a_i^{n-1} = \begin{cases} a_{i+1}^n & \text{si } i \text{ est pair} \\ a_{i+1}^n - \frac{a_0^n a_{i+2}^n}{a_1^n} & \text{si } i \text{ est impair} \end{cases} \quad i = 0, \dots, n-1 \quad (3.10)$$

$$b_j^{n-1} = \begin{cases} b_{j+1}^n & \text{si } j \text{ est pair} \\ b_{j+1}^n - \frac{b_1^n a_{j+1}^n}{a_1^n} & \text{si } j \text{ est impair} \end{cases} \quad j = 1, \dots, n-1 \quad (3.11)$$

Étape 5 : On détermine la valeur de l'ICE en se basant sur les Tableaux 3.1 et 3.2 à l'aide de la formule suivante [9] :

$$J(K_c, T_i, T_d) = \sum_{k=1}^n \frac{(b_1^k)^2}{2 a_0^k a_1^k} \quad (3.12)$$

Ainsi, la détermination des paramètres optimaux du correcteur PID revient à résoudre le problème d'optimisation suivant :

Numérateur					
b_1^n	b_2^n	b_3^n	b_4^n	\dots	b_n^n
a_1^n	0	a_3^n	0	a_5^n	\dots
	b_1^{n-1}	b_2^{n-1}	b_3^{n-1}	\dots	b_{n-1}^{n-1}
	a_1^n	0	a_3^n	0	\dots
		\dots	\vdots	\vdots	\vdots
			b_1^3	b_2^3	b_3^3
			a_1^3	0	0
				b_1^2	b_2^2
				a_1^2	0
					b_1^1

TABLE 3.1 – Tableau d'Åström généralisé pour le numérateur de $e(s)$

Dénominateur					
a_0^n	a_1^n	a_2^n	a_3^n	\dots	a_n^n
a_1^n	0	a_3^n	0	a_5^n	\dots
	a_0^{n-1}	a_1^{n-1}	a_2^{n-1}	\dots	a_{n-1}^{n-1}
	a_1^n	0	a_3^n	0	\dots
		\dots	\vdots	\vdots	\vdots
			a_0^2	a_1^2	a_2^2
			a_1^2	0	0
				a_0^1	a_1^1
				a_1^1	0
					a_0^0

TABLE 3.2 – Tableau d'Åström généralisé pour le dénominateur de $e(s)$

$$\min_{K_c, T_i, T_d} J(K_c, T_i, T_d) = \sum_{k=1}^n \frac{(b_1^k)^2}{2 a_0^k a_1^k} \quad (3.13)$$

sujet à :

$$K_c > 0 \quad (3.14)$$

$$T_i > 0 \quad (3.15)$$

$$T_d > 0 \quad (3.16)$$

Notons que les coefficients a_0^k , a_1^k , et b_1^k sont en fonction des paramètres du correcteur PID, c'est-à-dire en fonction de K_c , T_i , et T_d .

3.4 Exemple d'application

On considère le système de second ordre suivant :

$$g(s) = \frac{K}{\frac{1}{\omega^2} s^2 + \frac{2\xi}{\omega} s + 1} \quad (3.17)$$

où K est le gain statique du système, ω est la pulsation propre non amortie, et ξ est le coefficient d'amortissement.

Les paramètres du système sont : $K = 10, \omega = 5$, et $\xi = 0,7$. Pour commander le système, on propose d'utiliser un correcteur du type PI dont la fonction de transfert est :

$$g_c(s) = K_c \left(1 + \frac{1}{T_i s} \right) \quad (3.18)$$

L'erreur de poursuite $e(s)$, dans le domaine de Laplace, est :

$$e(s) = \frac{T_i s^2 + 2 T_i \xi \omega s + T_i \omega^2}{T_i s^3 + 2 T_i \xi \omega s^2 + (T_i \omega^2 + K K_c T_i \omega^2) s + K K_c \omega^2} \quad (3.19)$$

En utilisant les formules (3.10) et (3.11), on construit les tableaux d'Åström pour le numérateur (Tableau 3.3) et le dénominateur (Tableau 3.4).

En utilisant la formule (3.12) et les résultats des Tableaux 3.3 et 3.4, on obtient :

$$J(K_c, T_i) = \frac{1}{4 \xi \omega} + \frac{T_i (1 + K K_c - 4 \xi^2)^2}{8 \xi^2 (2 T_i \xi \omega - K K_c + 2 K K_c T_i \xi \omega)} + \frac{T_i^2 \xi \omega (K^2 K_c^2 - 4 K K_c \xi^2 + 2 K K_c T_i \xi \omega + 2 T_i \omega \xi)^2}{K K_c (2 T_i \xi \omega - K K_c + 2 K K_c T_i \xi \omega)^3} \quad (3.20)$$

Numérateur		
T_i	$2 T_i \xi \omega$	$T_i \omega^2$
$2 T_i \xi \omega$	0	$K K_c \omega^2$
	$2 T_i \xi \omega - \frac{(T_i \omega^2 + K K_c T_i \omega^2)}{(2 \xi \omega)}$	$T_i \omega^2$
	$T_i \omega^2 + K K_c T_i \omega^2 - \frac{(K K_c \omega)}{(2 \xi)}$	0
		$\frac{T_i \omega^2 (K^2 K_c^2 - 4 K K_c \xi^2 + 2 T_i \omega K K_c \xi + 2 T_i \omega \xi)}{2 T_i \omega \xi - K * K_c + 2 T_i \omega K K_c \xi}$

TABLE 3.3 – Tableau d'Åström pour le numérateur de l'erreur (3.19)

Dénominateur			
T_i	$2 T_i \xi \omega$	$T_i \omega^2 + K K_c T_i \omega^2$	$K K_c \omega^2$
$2 T_i \xi \omega$	0	$K K_c \omega^2$	0
	$2 T_i \xi \omega$	$T_i \omega^2 + K K_c T_i \omega^2 - \frac{(K K_c \omega)}{(2 \xi)}$	$K K_c \omega^2$
	$T_i \omega^2 + K K_c T_i \omega^2 - \frac{(K K_c \omega)}{(2 \xi)}$	0	0
		$T_i \omega^2 + K K_c T_i \omega^2 - \frac{(K K_c \omega)}{(2 \xi)}$	$K K_c \omega^2$
		$K K_c \omega^2$	0
			$K K_c \omega^2$

TABLE 3.4 – Tableau d'Åström pour le dénominateur de l'erreur (3.19)

En somme, le problème d'optimisation à résoudre est formulé comme suit :

$$\begin{aligned}
 & \min_{K_c, T_i} J(K_c, T_i) \\
 & \text{sujet à :} \\
 & K_c > 0 \\
 & T_i > 0
 \end{aligned} \tag{3.21}$$

La résolution du problème d'optimisation (3.21) en utilisant la fonction **ga** (algorithme génétique) de Matlab conduit à la solution suivante :

$$K_c^* = 0,13308 \text{ et } T_i^* = 0,23834 \tag{3.22}$$

Ainsi, le correcteur PI optimal est :

$$g_c(s) = 0,13308 \left(1 + \frac{1}{0,23834 s} \right) \tag{3.23}$$

Pour évaluer les performances du correcteur, nous considérons une consigne désirée $y^d(s) = 1$. Les résultats de la simulation obtenus sont donnés par les Figures 3.2 et 3.3. On remarque que la sortie présente une dynamique d'un système de deuxième ordre amortie, et suit la consigne imposée (écart statique nul). De plus, la commande présente des fluctuations acceptables. En conclusion, le correcteur PI permet d'assurer la poursuite de consigne.

3.5 Conclusion

Ce chapitre a été dédié à l'optimisation d'un correcteur PID basée sur la méthode d'Åström. Nous avons exposé les différents critères de performance utilisés essentiellement pour évaluer l'ef-

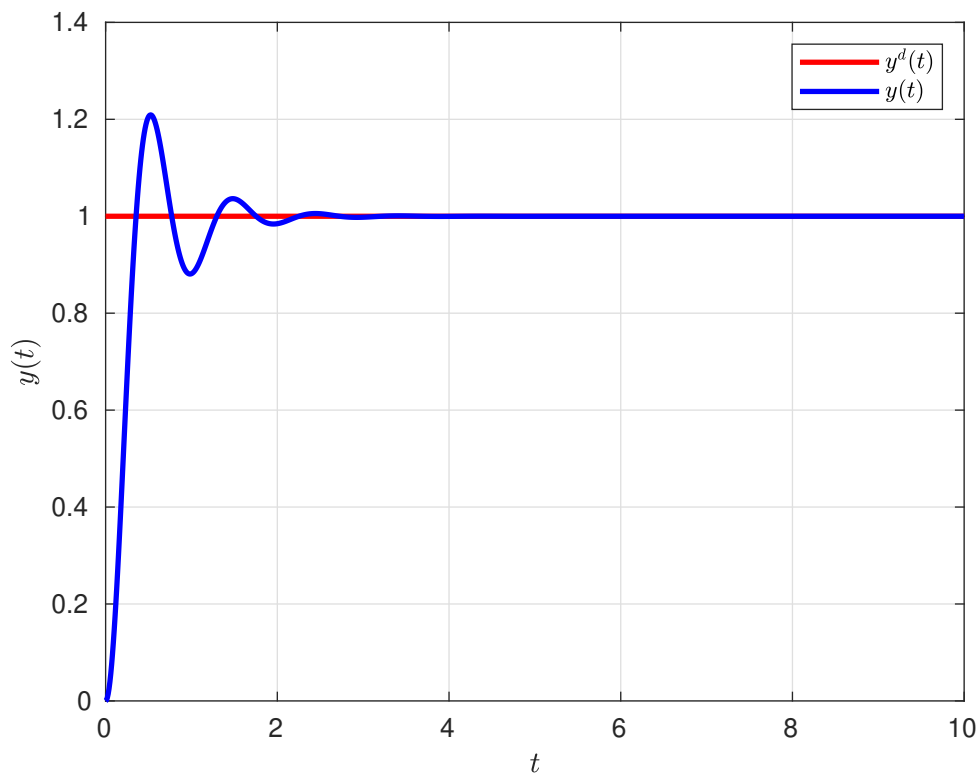


FIGURE 3.2 – Évolution temporelle de la sortie y .

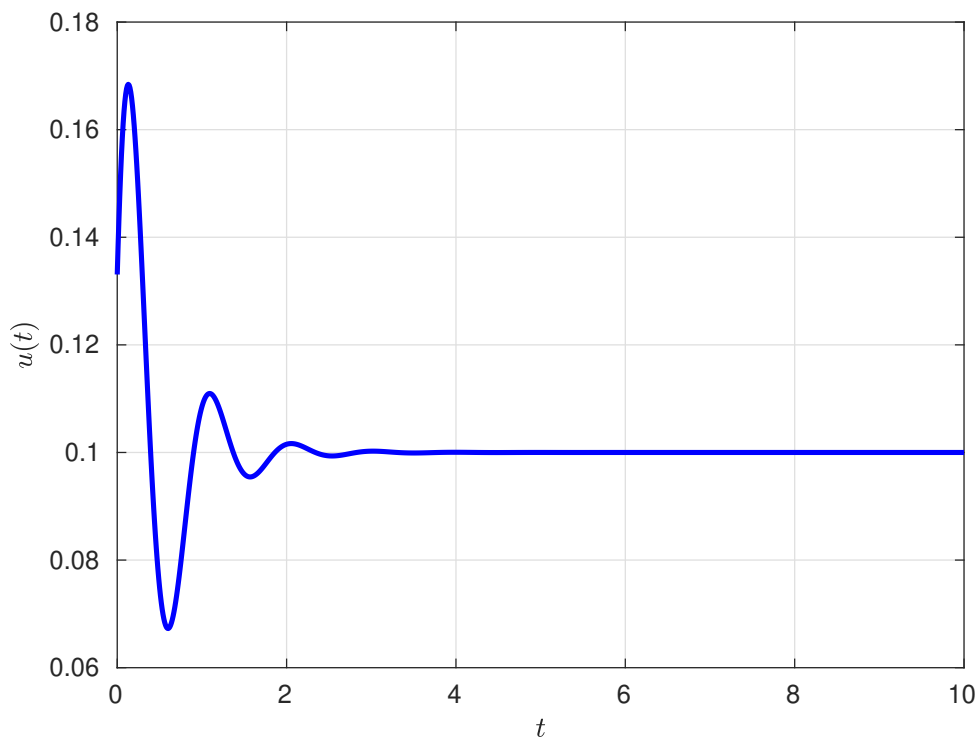


FIGURE 3.3 – Évolution temporelle de la commande u .

efficacité des correcteurs. Ensuite, nous avons présenté quelques directives pour le choix du type de correcteur à utiliser.

Pour optimiser les paramètres du correcteur PID, nous avons présenté la méthode d'Åström qui permet de convertir l'optimisation d'un critère de performance en un problème d'optimisation *standard* qu'on peut résoudre par des méthodes d'optimisation. Un exemple d'application a été introduit à la fin du chapitre pour expliquer les différentes étapes de l'approche d'optimisation du correcteur PID basée sur la méthode d'Åström. Des résultats de simulation ont été présentés pour démontrer l'efficacité de l'approche.

Cette approche d'optimisation sera généralisée, dans le chapitre suivant, pour la conception d'une commande multiboucle d'un système multivariable incertain.

Chapitre 4

Synthèse d'une commande multiboucle pour un système incertain

4.1 Introduction

La conception d'un correcteur multiboucle pour un système incertain est une tâche complexe nécessitant une approche de synthèse garantissant la stabilité et les performances désirées. En effet, la présence des incertitudes paramétriques influe considérablement sur les interactions entre les variables du système, par conséquent sur les performances en boucle fermée.

Dans ce chapitre, nous allons étendre l'approche de synthèse d'un correcteur PID optimal, présentée au Chapitre 3, aux systèmes multivariables incertains. L'objectif est de concevoir une commande multiboucle robuste. L'idée consiste à utiliser la stratégie multiboucle, et à déterminer les paramètres des correcteurs PID en utilisant l'approche des scénarios.

4.2 Formulation du problème de commande

Les systèmes multivariables incertains sont courants dans les applications industrielles, où les paramètres du système peuvent varier en raison de modifications dans l'environnement ou des conditions de fonctionnement.

Pour simplifier la présentation, on considère le cas d'un système multivariable à deux entrées et deux sorties $m = 2$. Le développement qui suit reste valable pour le cas général $m \geq 3$. Soit le système multivariable incertain suivant :

$$G(s) = \begin{bmatrix} g_{11}(s) & g_{12}(s) \\ g_{21}(s) & g_{22}(s) \end{bmatrix} \quad (4.1)$$

où les fonctions de transfert $g_{ij}(s)$ sont incertaines et définies comme suit

$$g_{ij}(s) = \frac{\left[\frac{b_p^{ij}}{a_n^{ij}}, \frac{\bar{b}_p^{ij}}{\bar{a}_n^{ij}} \right] s^p + \left[\frac{b_{p-1}^{ij}}{a_{n-1}^{ij}}, \frac{\bar{b}_{p-1}^{ij}}{\bar{a}_{n-1}^{ij}} \right] s^{p-1} + \dots + \left[\frac{b_0^{ij}}{a_0^{ij}}, \frac{\bar{b}_0^{ij}}{\bar{a}_0^{ij}} \right]}{\left[\frac{b_p^{ij}}{a_n^{ij}}, \frac{\bar{b}_p^{ij}}{\bar{a}_n^{ij}} \right] s^n + \left[\frac{b_{n-1}^{ij}}{a_{n-1}^{ij}}, \frac{\bar{b}_{n-1}^{ij}}{\bar{a}_{n-1}^{ij}} \right] s^{n-1} + \dots + \left[\frac{b_0^{ij}}{a_0^{ij}}, \frac{\bar{b}_0^{ij}}{\bar{a}_0^{ij}} \right]}, \quad p < n, \quad i, j = 1, 2 \quad (4.2)$$

où le symbole $\underline{\quad}$ et $\bar{\quad}$ sont respectivement les valeurs minimale ou maximale du paramètre incertain.

L'objectif est de concevoir un correcteur multiboucle qui permet d'assurer les performances désirées, en dépit des incertitudes, et des perturbations. Pour un système multivariable avec $m = 2$, le correcteur multiboucle est de la forme

$$G_c(s) = \begin{bmatrix} g_{c_1}(s) & 0 \\ 0 & g_{c_2}(s) \end{bmatrix} \quad (4.3)$$

où g_{c_j} ($j = 1, 2$) sont des correcteurs PID de la forme

$$g_{c_j}(s) = K_c^j \left(1 + \frac{1}{T_i^j s} + T_d^j s \right) \quad (4.4)$$

où K_c^j , T_i^j , et T_d^j sont respectivement le gain, la constante intégrale et la constante dérivée du j -ème correcteur PID.

4.3 Conception du correcteur multiboucle

La première étape consiste à sélectionner la meilleure configuration de commande. Pour ce faire, on propose d'utiliser la matrice des gains relatifs (RGA). La matrice des gains statiques du système est donnée comme suit

$$G(0) = \begin{bmatrix} \left[\frac{b_0^{11}}{a_0^{11}}, \frac{\bar{b}_0^{11}}{\bar{a}_0^{11}} \right] & \left[\frac{b_0^{12}}{a_0^{12}}, \frac{\bar{b}_0^{12}}{\bar{a}_0^{12}} \right] \\ \left[\frac{b_0^{21}}{a_0^{21}}, \frac{\bar{b}_0^{21}}{\bar{a}_0^{21}} \right] & \left[\frac{b_0^{22}}{a_0^{22}}, \frac{\bar{b}_0^{22}}{\bar{a}_0^{22}} \right] \end{bmatrix} \quad (4.5)$$

et en utilisant l'équation (1.8), on obtient la RGA suivante

$$RGA = \begin{bmatrix} \left[\frac{b_0^{11}}{a_0^{11}}, \frac{\bar{b}_0^{11}}{\bar{a}_0^{11}} \right] & \left[\frac{b_0^{12}}{a_0^{12}}, \frac{\bar{b}_0^{12}}{\bar{a}_0^{12}} \right] \\ \left[\frac{b_0^{21}}{a_0^{21}}, \frac{\bar{b}_0^{21}}{\bar{a}_0^{21}} \right] & \left[\frac{b_0^{22}}{a_0^{22}}, \frac{\bar{b}_0^{22}}{\bar{a}_0^{22}} \right] \end{bmatrix} \cdot \left[\begin{bmatrix} \left[\frac{b_0^{11}}{a_0^{11}}, \frac{\bar{b}_0^{11}}{\bar{a}_0^{11}} \right] & \left[\frac{b_0^{12}}{a_0^{12}}, \frac{\bar{b}_0^{12}}{\bar{a}_0^{12}} \right] \\ \left[\frac{b_0^{21}}{a_0^{21}}, \frac{\bar{b}_0^{21}}{\bar{a}_0^{21}} \right] & \left[\frac{b_0^{22}}{a_0^{22}}, \frac{\bar{b}_0^{22}}{\bar{a}_0^{22}} \right] \end{bmatrix}^{-1} \right]^T \quad (4.6)$$

Comme les éléments de la matrice $G(0)$ sont des intervalles, alors pour déterminer la matrice RGA, on utilise les opérations sur les intervalles. Pour les deux intervalles *positifs* $\alpha = [\underline{\alpha}, \bar{\alpha}]$ et $\beta = [\underline{\beta}, \bar{\beta}]$, on a

- $\alpha + \beta = [\underline{\alpha} + \underline{\beta}, \bar{\alpha} + \bar{\beta}]$,
- $\alpha - \beta = [\underline{\alpha} - \bar{\beta}, \bar{\alpha} - \underline{\beta}]$,
- $\alpha \times \beta = [\min E, \max E]$ avec $E = \{\underline{\alpha}\underline{\beta}, \underline{\alpha}\bar{\beta}, \bar{\alpha}\underline{\beta}, \bar{\alpha}\bar{\beta}\}$,
- $\frac{1}{\alpha} = \left[\frac{1}{\bar{\alpha}}, \frac{1}{\underline{\alpha}} \right]$,
- $\frac{\alpha}{\beta} = \alpha \times \frac{1}{\beta}$.

En utilisant ces opérations, on obtient la RGA sous la forme suivante :

$$RGA = \begin{bmatrix} [\underline{\lambda}_{11}, \bar{\lambda}_{11}] & [\underline{\lambda}_{12}, \bar{\lambda}_{12}] \\ [\underline{\lambda}_{21}, \bar{\lambda}_{21}] & [\underline{\lambda}_{22}, \bar{\lambda}_{22}] \end{bmatrix} \quad (4.7)$$

et pour déterminer la meilleure configuration de commande, on peut considérer la matrice nominale des gains relatifs donnée comme suit

$$RGA = \frac{1}{2} \begin{bmatrix} \underline{\lambda}_{11} + \bar{\lambda}_{11} & \underline{\lambda}_{12} + \bar{\lambda}_{12} \\ \underline{\lambda}_{21} + \bar{\lambda}_{21} & \underline{\lambda}_{22} + \bar{\lambda}_{22} \end{bmatrix} \quad (4.8)$$

Pour continuer le développement, on suppose que la meilleure configuration de commande est celle définie par les couples $[u_1 - y_1]$ et $[u_2 - y_2]$. Ainsi, pour concevoir les correcteurs PID, on propose de minimiser le critère suivant :

$$J(x, \delta) = J_1(x, \delta) + J_2(x, \delta) \quad (4.9)$$

où x est le vecteur de variables d'optimisation qui regroupe les paramètres des correcteurs PID, c'est-à-dire

$$x = \left[K_c^1 \quad K_c^2 \quad T_i^1 \quad T_i^2 \quad T_d^1 \quad T_d^2 \right]^T \quad (4.10)$$

et δ le vecteur des paramètres incertains du système, c'est-à-dire

$$\delta = \begin{bmatrix} \delta_a \\ \delta_b \end{bmatrix} \quad (4.11)$$

avec

$$\delta_a = \left[a_0^{11} \quad \dots \quad a_n^{11} \quad a_0^{12} \quad \dots \quad a_n^{12} \quad a_0^{21} \quad \dots \quad a_n^{21} \quad a_0^{22} \quad \dots \quad a_n^{22} \right]^T \quad (4.12)$$

$$\delta_b = \left[b_0^{11} \quad \dots \quad b_n^{11} \quad b_0^{12} \quad \dots \quad b_n^{12} \quad b_0^{21} \quad \dots \quad b_n^{21} \quad b_0^{22} \quad \dots \quad b_n^{22} \right]^T \quad (4.13)$$

et

$$J_1(x, \delta) = \int_0^{+\infty} (y_1^d(t) - y_1(t))^2 dt \quad (4.14)$$

$$J_2(x, \delta) = \int_0^{+\infty} (y_2^d(t) - y_2(t))^2 dt \quad (4.15)$$

Pour exprimer les critères J_1 et J_2 en fonction de x et δ , on utilise la méthode d'Åström. En boucle fermée, la matrice de fonctions de transfert entre le vecteur des erreurs de poursuite e et le vecteur des sorties commandées y est donnée comme suit :

$$E(s) = [I_2 + G_c(s, x) G(s, \delta)]^{-1} Y^d(s) \quad (4.16)$$

avec

$$E(s) = \begin{bmatrix} e_1(s) \\ e_2(s) \end{bmatrix}, \quad Y^d(s) = \begin{bmatrix} y_1^d(s) \\ y_2^d(s) \end{bmatrix} \quad (4.17)$$

où E est le vecteur des erreurs de poursuite, Y^d est le vecteur des consignes désirées.

Ainsi, pour des consignes du type échelon unité, il vient pour les erreurs en boucle fermée

$$E_j(s) = \frac{b_{j,1} s^{n-1} + b_{j,2} s^{n-2} + \dots + b_{j,n-1} s + b_{j,n}}{a_{j,0} s^n + a_{j,1}^j s^{n-1} + \dots + a_{j,n-1} s + a_{j,n}}, \quad j = 1, 2 \quad (4.18)$$

où les coefficients $a_{j,0}, \dots, a_{j,n}$, et $b_{j,1}, \dots, b_{j,n}$ ($j = 1, 2$) sont en fonctions de x et de δ .

En appliquant la méthode d'Åström, détaillée dans le Chapitre 3, pour la fonction de transfert (4.18), on obtient

$$J_j(x, \delta) = \sum_{k=1}^n \frac{(b_{j,1}^k)^2}{2 a_{j,0}^k a_{j,1}^k}, \quad j = 1, 2 \quad (4.19)$$

Remarque 4.1. Les expressions des paramètres $a_{i,j}$, $b_{i,j}$ et J_j sont obtenues en utilisant la boîte à outils Symbolic de Matlab. Elles ne sont pas données car elles sont très longues.

Pour déterminer les paramètres optimaux des correcteurs PID, on résout le problème d'optimisation du type *min-max* suivant

$$\min_x \max_{\delta} J(x, \delta) = J_1(x, \delta) + J_2(x, \delta) \quad (4.20)$$

sujet à :

$$K_c^j, T_i^j, T_d^j > 0, \quad j = 1, 2 \quad (4.21)$$

Pour résoudre ce problème d'optimisation, on utilise l'approche des scénarios présentée au Chapitre 2. Ainsi, en considérant N échantillons pour les incertitudes, c'est-à-dire $\delta^{(i)}$, $i = 1, \dots, N$, le problème d'optimisation relaxé prend la forme suivante

$$\min_x z \tag{4.22}$$

sujet à :

$$J(x, \delta^{(k)}) \leq z, \quad k = 1, \dots, N \tag{4.23}$$

$$K_c^j, T_i^j, T_d^j > 0, \quad j = 1, 2 \tag{4.24}$$

Rappelons que le nombre d'échantillons N à considérer est déterminé en utilisant la formule (2.15).

4.4 Exemple d'application

On considère un système multivariable à deux entrées et deux sorties de matrice de fonctions de transfert

$$G(s) = \begin{bmatrix} \frac{K_{11}}{\tau_{11}s + 1} & \frac{K_{12}}{\tau_{12}s + 1} \\ \frac{K_{21}}{\tau_{21}s + 1} & \frac{K_{22}}{\tau_{22}s + 1} \end{bmatrix} \tag{4.25}$$

Les paramètres du système sont donnés comme suit :

$$K_{12} = K_{21} = K_{22} = 4, \quad \tau_{11} = \tau_{12} = \tau_{21} = 10$$

avec les deux paramètres incertains suivant

$$2 \leq K_{11} \leq 5, \quad 9 \leq \tau_{22} \leq 10$$

Pour déterminer la meilleure configuration de commande, on utilise la RGA. La matrice des gains statiques du système est donnée comme suit

$$G(0) = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} = \begin{bmatrix} [2, 5] & 4 \\ 4 & 4 \end{bmatrix} \tag{4.26}$$

D'après l'équation (4.6), la RGA est

$$\begin{aligned} RGA &= G(0) \cdot [G(0)^{-1}]^T \\ &= \begin{bmatrix} [2, 5] & 4 \\ 4 & 4 \end{bmatrix} \cdot \left[\begin{bmatrix} [2, 5] & 4 \\ 4 & 4 \end{bmatrix}^{-1} \right]^T \end{aligned} \quad (4.27)$$

avec

$$G(0)^{-1} = \frac{\begin{bmatrix} 4 & -4 \\ -4 & [2, 5] \end{bmatrix}}{|G(0)|} \quad (4.28)$$

et

$$\begin{aligned} |G(0)| &= ([2, 5] \times 4) - (4 \times 4) \\ &= [8, 20] - 16 \\ &= [-8, 4] \end{aligned} \quad (4.29)$$

Le calcul de $G(0)^{-1}$ donne

$$G(0)^{-1} = \begin{bmatrix} \left[-\frac{1}{2}, 1 \right] & \left[-1, \frac{1}{2} \right] \\ \left[-1, \frac{1}{2} \right] & \left[-\frac{5}{8}, \frac{5}{4} \right] \end{bmatrix} \quad (4.30)$$

Ainsi, la RGA est

$$RGA = \begin{bmatrix} \left[-\frac{5}{2}, 5 \right] & [-4, 2] \\ [-4, 2] & \left[-\frac{5}{2}, 5 \right] \end{bmatrix} \quad (4.31)$$

Pour déterminer la meilleure configuration de commande, on calcule la matrice des gains relatifs nominaux en utilisant la formule (4.8), ce qui donne :

$$RGA = \begin{bmatrix} 1, 25 & -1 \\ -1 & 1, 25 \end{bmatrix} \quad (4.32)$$

D'après les éléments de la RGA, on remarque que les éléments de la diagonale sont plus proches de un. De plus, les éléments hors diagonale sont négatifs, par conséquent la meilleure configuration de commande est $[u_1 - y_1]$ et $[u_2 - y_2]$.

Pour la commande multiboucle du système, on utilise des correcteurs du type PI dont les

fonctions de transfert sont :

$$g_{c_1}(s) = K_c^1 \left(1 + \frac{1}{T_i^1} s \right) \quad (4.33)$$

$$g_{c_2}(s) = K_c^2 \left(1 + \frac{1}{T_i^2} s \right) \quad (4.34)$$

Ainsi, le vecteur de variables de décision x est défini comme suit

$$x = \begin{bmatrix} K_c^1 \\ T_i^1 \\ K_c^2 \\ T_i^2 \end{bmatrix}$$

et le vecteur de paramètres incertain δ est donné comme suit

$$\delta = \begin{bmatrix} K_{11} \\ \tau_{22} \end{bmatrix}$$

Le problème d'optimisation à résoudre est donné comme suit

$$\min_x z \quad (4.35)$$

sujet à :

$$J(x, K_{11}^{(k)}, \tau_{22}^{(k)}) \leq z, \quad k = 1, \dots, N \quad (4.36)$$

$$K_c^1 > 0 \quad (4.37)$$

$$T_i^1 > 0 \quad (4.38)$$

$$K_c^2 > 0 \quad (4.39)$$

$$T_i^2 > 0 \quad (4.40)$$

Le nombre d'échantillons N est déterminé en utilisant la relation **(2.15)**. Ainsi, en prenant $\varepsilon = 0,01$ et $\gamma = 10^{-10}$, et comme le nombre de variables de décision $n = 5$ ($K_c^1, T_i^1, K_c^2, T_i^2, z$), alors $N = 5606$. La résolution du problème d'optimisation en utilisant la fonction **ga** (algorithme génétique) de Matlab, donne les résultats suivants :

$$K_c^{1*} = 244,971; \quad T_i^{1*} = 17,950; \quad K_c^{2*} = 216,397; \quad T_i^{2*} = 38,352; \quad \text{et } z^* = 0,551 \quad (4.41)$$

Ainsi, les correcteurs PI optimaux sont :

$$g_{c_1}(s) = 244,971 \left(1 + \frac{1}{17,950 s} \right) \quad (4.42)$$

$$g_{c_2}(s) = 216,397 \left(1 + \frac{1}{38,352 s} \right) \quad (4.43)$$

Pour évaluer les performances du correcteur multiboucle, nous appliquerons les consignes désirées suivantes :

$$y_1^d(s) = \begin{cases} 0 & t < 1,5 \text{ s} \\ 1 & t \geq 1,5 \text{ s} \end{cases} \quad (4.44)$$

$$y_2^d(s) = 1 \quad (4.45)$$

et on considère 57 scénarios pour les incertitudes.

Le résultat de la simulation obtenu est donné par les Figures 4.1 et 4.2. On remarque que chaque sortie suit la consigne imposée. On remarque aussi que l'effet de perturbation dû au changement de consigne de la sortie y_1 , effectué à l'instant $t = 1,5 \text{ s}$, est rejeté par le correcteur $g_{c_2}(s)$. Les fluctuations observées dans les commandes u_1 et u_2 (Figure 4.2) indiquent des interactions fortes entre les boucles de commande.

Un autre test de simulation été effectué en considérant les consignes suivantes

$$y_1^d(s) = \begin{cases} 0 & t < 1,5 \text{ s} \\ 1 & 1,5 \text{ s} \leq t < 4,5 \text{ s} \\ 0,5 & t \geq 4,5 \text{ s} \end{cases} \quad (4.46)$$

$$y_2^d(s) = \begin{cases} 1 & 0 \leq t < 3 \text{ s} \\ 0,5 & 3 \text{ s} \leq t < 6 \text{ s} \\ 0 & t \geq 6 \text{ s} \end{cases} \quad (4.47)$$

et on considère le même nombre de scénarios.

Les Figures 4.3 et 4.4 donnent les résultats de simulation obtenus. On remarque que les sorties suivent leurs consignes et les perturbations dues aux changements de consignes sont rejetées par les correcteurs. Les variations des commandes restent acceptables (Figure 4.4).

En conclusion, en dépit des fortes interactions entre les variables du système et les incertitudes, le correcteur multiboucle permet d'assurer la poursuite de consigne et le rejet de perturbation.

4.5 Conclusion

Dans ce chapitre, nous avons présenté une méthode de synthèse d'une commande multiboucle à base des correcteurs PID pour un système multivariable incertain. Ainsi, en utilisant la méthode d'Åström, nous avons formulé le problème de conception du correcteur multiboucle sous forme d'un problème d'optimisation du type *min-max*, puis pour le résoudre, nous avons opté pour la méthode des scénarios.

Un exemple d'application est présenté à la fin du chapitre pour illustrer la démarche de conception, et démontrer les performances de la commande multiboucle dans le cas d'un système incertain. Les résultats obtenus sont concluants puisque le correcteur multiboucle permet d'assurer la poursuite de trajectoires tout en rejetant les perturbations internes dues aux interactions et aux incertitudes paramétriques.

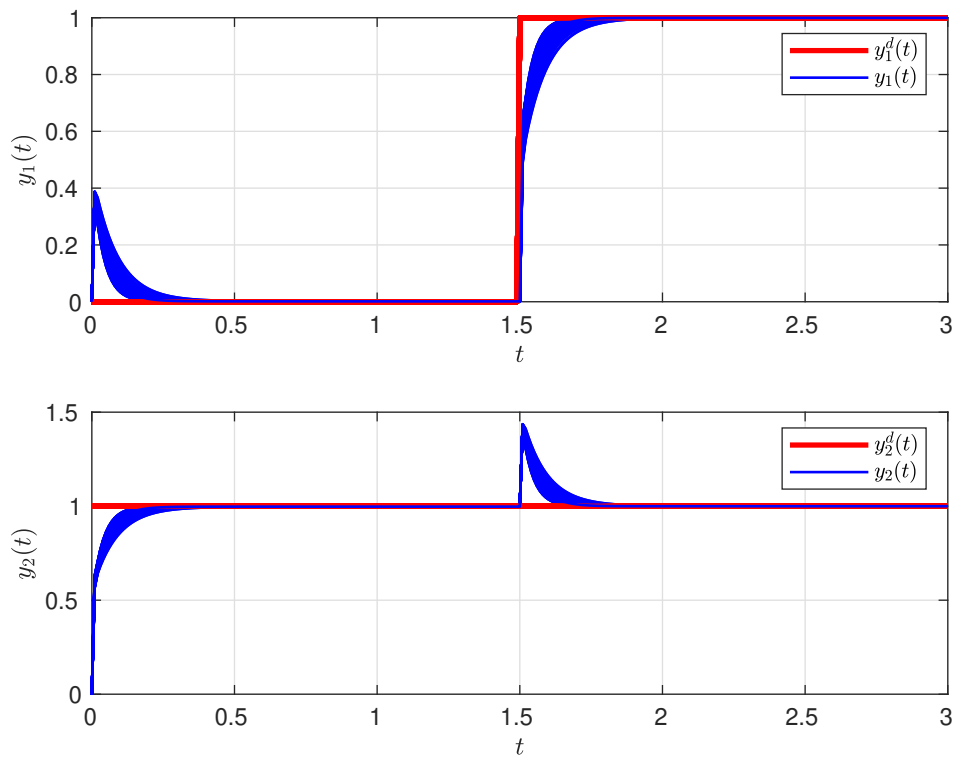


FIGURE 4.1 – Évolution temporelle des sorties y_1 et y_2 .

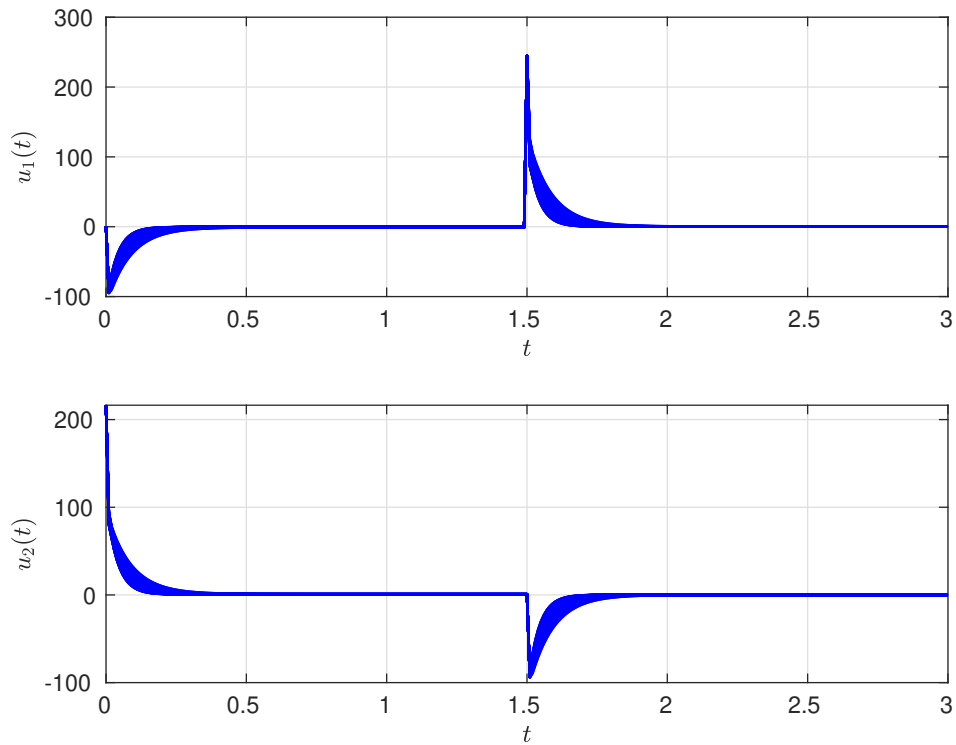


FIGURE 4.2 – Évolution temporelle des commandes u_1 et u_2 .

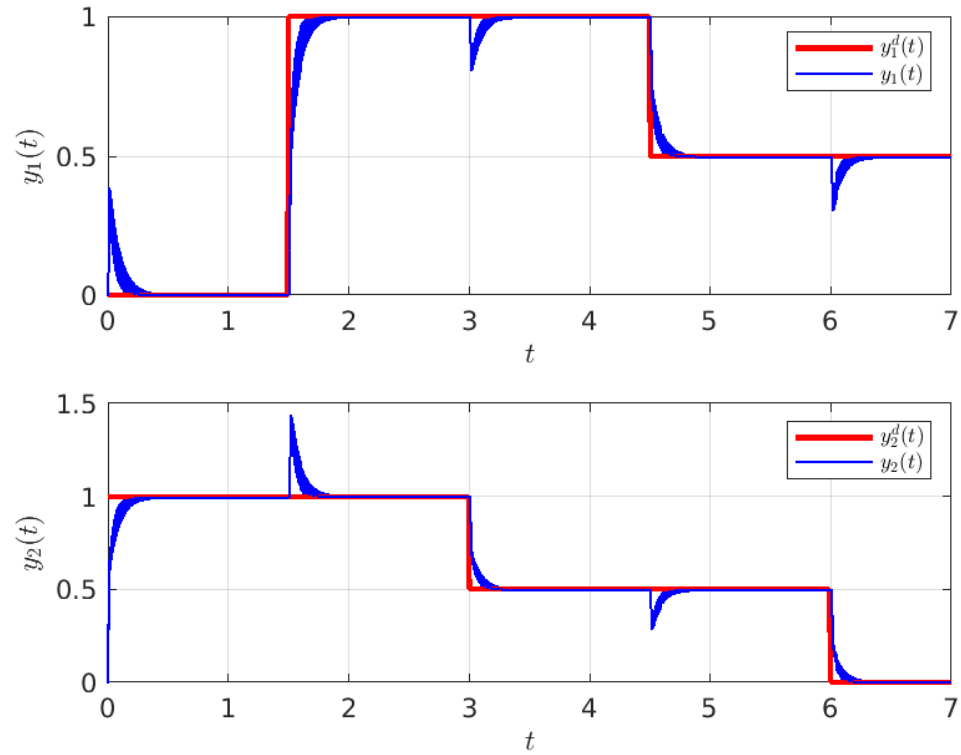


FIGURE 4.3 – Évolution temporelle des sorties y_1 et y_2 pour des consignes variables.

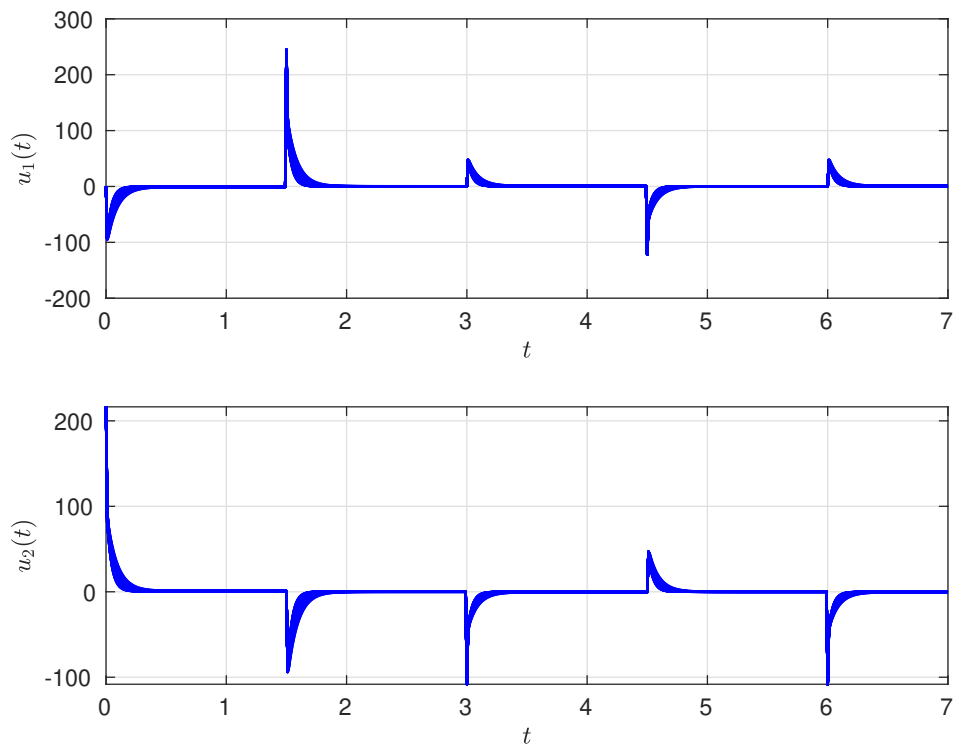


FIGURE 4.4 – Évolution temporelle des commandes u_1 et u_2 pour des consignes variables.

Conclusion générale

Le travail présenté dans ce mémoire porte sur la commande multiboucle des systèmes multivariables. L'objectif consiste à proposer une approche de synthèse d'une commande multiboucle en présence des incertitudes paramétriques.

Ainsi, après avoir présenté des généralités sur les systèmes multivariables en particulier le phénomène des interactions, nous avons expliqué le principe de la commande multiboucle et ses étapes de synthèse. Par la suite, nous avons introduit la notion de l'optimisation robuste, c'est-à-dire l'optimisation d'une fonction mathématique dont les paramètres sont incertains. Pour résoudre un tel problème d'optimisation, nous avons opté à une méthode plus pratique qui est l'approche des scénarios. Cette dernière a été utilisée pour concevoir une commande multiboucle pour un système multivariable incertain.

A la lumière de l'étude réalisée, on peut affirmer que l'utilisation de la méthode d'Åström permet de convertir un problème de conception d'un correcteur monovarié ou multiboucle en un problème d'optimisation *standard* qu'on peut facilement résoudre par une panoplie de méthodes d'optimisation. Cette approche peut être aussi appliquée pour les systèmes incertains. Dans ce cas, le problème d'optimisation résultant est du type *min-max* qu'on peut relaxer en utilisant l'approche des scénarios, et appliquer par la suite les méthodes d'optimisation classiques pour le solutionner.

En somme, nous avons étendu la matrice des gains relatifs aux systèmes incertains, et développer une approche de synthèse d'une commande multiboucle robuste en combinant la méthode d'Åström et l'approche des scénarios. Un exemple d'application sur un système multivariable caractérisé par de fortes interactions est présenté. Les résultats de simulation obtenus ont démontré les performances de la commande multiboucle synthétisée. En effet, en dépit des incertitudes paramétriques, les deux correcteurs assurent une poursuite de consigne tout en rejetant l'effet de perturbation dû au changement des consignes.

L'étude présentée dans ce mémoire s'est limitée à la synthèse d'une commande multiboucle dans le domaine temporel, ainsi comme perspectives du travail, on peut penser à :

- Considérer d'autres critères de performances pour la synthèse de la commande multiboucle, et comparer les résultats on utilisant différents critères.
- Utiliser des critères multi-objectifs en incluant des contraintes de robustesse.

Bibliographie

- [1] A. FOSSARD. *Systèmes Multientrées-Multisorties*. Technique de l'ingénieur, traité Mesures et Contrôle, No. 139, R 7 220, 1997.
- [2] A. FOSSARD ET C. GUEGUEN. *Commande des systèmes multidimensionnels*. Edition Dunod, Paris, 1972.
- [3] A. MAIDI. *Analyse de interactions dans les systèmes à grande échelle : application à la commande multiboucle des systèmes fortement couplés*. Mémoire de magistère, Département Automatisation des Procédés Industriels et Électrification, Université M'hamed Bougara de Boumerdès, 2001.
- [4] A. POMERLEAU. *La commande de procédés industriels*. Edition Hermes, Paris, 1997.
- [5] A. TRIKI. *Synthèse de correcteurs optimaux pour les systèmes à paramètres distribués*. Thèse de doctorat, Département Automatique, Université Mouloud Mammeri de Tizi-Ouzou, 2018.
- [6] G. IDIRI. *Commande prédictive des systèmes non linéaires dynamiques*. Mémoire de Magister, Département Automatique, Université Mouloud Mammeri de Tizi-Ouzou, 2011.
- [7] J. M. FLAUS. *La régulation industriel*. Edition Hermes, Paris, 1994.
- [8] J. P. CORRIOU. *Commande des procédés*. Edition Lavoisier, Paris, 2003.
- [9] K. J. ÅSTRÖM. *Introduction to stochastic control theory*. Edition Academic Press, New York, 1970.
- [10] M. BIERLAIRE. *Introduction à l'optimisation différentiable*. Edition Presses Polytechniques et Universitaires Romandes, Lausanne, 2016.
- [11] R. BOUDAREL, J. DELMAS ET P. GUICHET. *Commande optimale des processus*. Edition Dunod, Paris, 1968.
- [12] R. TEMPO, G. CALAFIORE ET F. DABBENE. *Randomized Algorithms for Analysis and Control of Uncertain Systems*. Edition Springer-Verlag, London, 2013.
- [13] W. L. LUYBEN. *Simple method for tuning SISO controllers in multivariable systems*. Industrial & Engineering Chemistry Process Design and Development, Vol. 25, No. 3, pp. 654–660, 1986.