



HAL
open science

Synthèse de régulateurs numériques robustes multivariables par optimisation convexe

Hynek Prochazka

► **To cite this version:**

Hynek Prochazka. Synthèse de régulateurs numériques robustes multivariables par optimisation convexe. Automatique / Robotique. Institut National Polytechnique de Grenoble - INPG, 2004. Français. NNT: . tel-00169982

HAL Id: tel-00169982

<https://theses.hal.science/tel-00169982>

Submitted on 14 Sep 2007

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

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

INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

N° attribué par la bibliothèque

|_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/

THESE

pour obtenir le grade de

DOCTEUR DE L'INPG

Spécialité : Automatique – Productique

préparée au ***Laboratoire d'Automatique de Grenoble***
dans le cadre de ***l'Ecole Doctorale Electronique, Electrotechnique, Automatique, Traitement du Signal***

présentée et soutenue publiquement

par

Hynek PROCHÁZKA

le 23 janvier 2004

Titre :

**SYNTHESE DE REGULATEURS NUMERIQUES ROBUSTES
MULTIVARIABLES PAR OPTIMISATION CONVEXE**

Directeur de thèse :

Ioan Doré LANDAU

JURY

M.	A. Barraud	Président
M.	P. Albertos	Rapporteur
M.	H. Bourlès	Rapporteur
M.	I.D. Landau	Directeur de thèse

Résumé

La thèse concerne essentiellement les méthodes de synthèse de régulateurs numériques robustes, monovariables ou multivariables, pour la commande des procédés temps-continu. Pour la synthèse, il est supposé que l'on dispose d'un modèle linéaire échantillonné (discrétisé) du procédé continu à commander. La robustesse de régulateur est traitée par l'analyse fréquentielle des sensibilités (fonctions/matrices de transfert de la boucle fermée). Comme dans le cas de la commande H_∞ , les valeurs singulières des réponses fréquentielles sont examinées pour ces analyses. L'optimisation convexe est utilisée pour simplifier la synthèse et pour la rendre plus appropriée pour la pratique d'ingénierie.

Le mémoire est divisé en cinq parties. La première partie concerne les systèmes monovariables et le reste est concentré sur les problèmes de synthèse dans le domaine multivariable. Les cinq parties traitent les problématiques suivantes:

- La première partie (Chapitre 2) touche la synthèse de régulateurs robustes monovariables par le placement de pôles, le calibrage de sensibilités et l'optimisation convexe [LK98, LL99]. Elle présente une nouvelle approche de la synthèse par placement de pôles en utilisant les filtres bande-étroite du 2^e ordre et un logiciel associé développé dans le cadre de ce travail. Le logiciel reflète les améliorations apportées qui ont radicalement simplifié la procédure de synthèse. La section qui concerne l'optimisation convexe rappelle les principes et introduit la notation utilisée ultérieurement pour la théorie multivariable.
- La deuxième partie (Chapitre 3 et 4) évoque la théorie fondamentale de la commande multivariable et développe l'idée de la synthèse de régulateurs robustes multivariables par placement de pôles et calibrage des sensibilités. Le régulateur a la forme d'observateur avec le retour des états estimés et il est utilisé dans la suite comme le régulateur central pour la synthèse de régulateurs par optimisation convexe.
- La troisième partie (Chapitre 5) est la partie essentielle et elle développe la méthode de synthèse de régulateurs robustes multivariables par calibrage de sensibilités et optimisation convexe. La méthode est basée sur le même principe que celui utilisé dans le domaine monovariable [Lan98, LL99] et elle est réalisée comme une boîte à outils interactive

pour Matlab. Le placement de pôles multivariable est aussi intégré dans cette application. Le logiciel développé fait partie de la thèse et il est brièvement décrit dans ce chapitre.

- La quatrième partie (Chapitre 6) traite la synthèse par optimisation convexe du pré-compensateur multivariable pour la poursuite. Le pré-compensateur nous permet de séparer les spécifications sur la robustesse (rejet de perturbations et traitement des incertitudes) et les spécifications sur la poursuite (temps de montée, dépassement maximal).
- La cinquième partie (Chapitre 7) concerne la technique de réduction des régulateurs multivariables par identification en boucle fermée. La méthode développée fait suite à un approche similaire développée pour les régulateurs monovariables dans [LKC01].

Mots-clés: commande robuste, optimisation convexe, placement de pôles, synthèse de régulateurs, réduction de régulateur, systèmes multivariables, systèmes monovariables, logiciel de synthèse

Abstract

The dissertation is essentially concerned with methods of robust digital multivariable and monovariable controller design. For controller design, a linear discrete-time model of plant to be controlled is used. The controller robustness is treated by sensitivity frequency analysis (sensitivity is transfer function/matrix of closed loop). Correspondingly to H_∞ design, the singular values of frequency responses are used for the analysis. The convex optimization is used to simplified the design and to make it more appropriate for engineering practice.

The work is divided into five parts. The first part concerns monovariable systems and the rest is focussed on multivariable domain. The five parts deal with the following issues:

- *The first part (Chapter 2) is related to robust digital monovariable controller design by pole placement, sensitivity shaping and convex optimization [LK98, LL99]. It presents a new approach by pole placement using 2nd order digital notch filters and an associated software package developed in this framework. The software reflects the improvements*

which drastically simplify the design procedure. The section concerning convex optimization recalls some basic principals and introduces a notation used later for multivariable systems.

- The second part (Chapter 3 and 4) reminds basic theory of multivariable control and it develops the idea of robust multivariable controller design by pole placement and sensitivity shaping. The controller has a form of an observer with an estimated state feedback and it is used later as a central controller for convex optimization controller design.
- The third part (Chapter 5) is the main part of the work and it develops a multivariable controller design method by sensitivity shaping and convex optimisation. The method is based on a similar principles as a monovariate design technique [Lan98, LL99] and it is implemented as an interactive toolbox for Matlab. The multivariable pole placement is implemented in this application as well. The toolbox is a part of the dissertation and it is briefly described in this chapter.
- The fourth part (Chapter 6) treats multivariable pre-compensator tracking part design using convex optimization. The pre-compensator allows to separate robustness specifications (disturbance rejection, uncertainty treating...) and tracking specifications (rising time, maximal overshoot...).
- The fifth part (Chapter 7) concerns multivariable controller reduction technique by identification in closed loop. The developed method is an extension of an approach developed for monovariate controllers [LKC01]

Keywords: *robust control, convex optimization, pole placement, controller design, controller reduction, multivariable systems, monovariate systems, design software*

Bibliography

- [Lan98] J. Langer. *Synthèse de régulateurs numériques robustes - Application aux structures souples*. PhD thesis, Institut National Polytechnique de Grenoble, 1998.
- [LK98] I.D. Landau and A. Karimi. Robust digital control using pole placement with sensitivity function shaping method. *Int. J. Robust and Nonlin. Cont.*, 8:191–210, 1998.
- [LKC01] I.D. Landau, A. Karimi, and A. Constantinescu. Direct controller order reduction by identification in closed loop. *Automatica*, 37(11):1689–1702, 2001.
- [LL99] J. Langer and I.D. Landau. Combined pole placement/sensitivity function shaping method using convex optimisation criteria. *Automatica*, 35:1111–1120, 1999.

Synthèse de régulateurs numériques robustes multivariables par optimisation convexe

Hynek Procházka

15 janvier 2004

Remerciements

Je tiens tout d'abord à remercier le directeur de cette thèse, M. Ioan Doré Landau qui a dirigé cette thèse dans la continuité de mon stage de DEA. Je lui dois de m'avoir accueilli dans son équipe où j'ai pu travailler sur cette thèse et sur bien d'autres sujets passionnants. Il m'a fait découvrir la théorie et la pratique de la commande de systèmes, il a toujours été disponible pour d'intenses et rationnelles discussions. Pour tout cela, je le remercie vivement.

Je remercie les rapporteurs de cette thèse Pedro Albertos et Henri Bourlès pour l'intérêt qu'ils ont porté à mon travail et pour l'attention avec laquelle ils ont lu mon manuscrit. Merci également au président du jury Alain Barraud qui a accepté de juger ce travail.

Je remercie Daniel Rey, Alfons Franco, Giamluca Zito, Aurelian Constantinescu, et Alina Besançon qui m'ont aidé par les discussions que j'ai eu la chance d'avoir avec eux, leurs suggestions ou contributions.

Je tiens à remercier Gérard Scorletti Maître de Conférences à l'Université de Caen et Eduardo F. Camacho directeur del Departamento Ing. de Sistemas y Automatica, Universidad de Sevilla pour ses conseils et discussions fécondes.

Merci enfin à toute ma famille, à mes amis et à Zuzana de m'avoir supporté et aidé les quatre ans de ma vie et de mes études à Grenoble.

Table des matières

Notations	7
Variables et constantes	7
Liste des abréviations	12
1 Introduction	13
1.1 Motivation et contexte	13
1.2 Objectif de travail	17
1.3 Organisation de mémoire	20
2 Rappel sur la méthodologie monovariante	23
2.1 Placement de pôles avec calibrage de sensibilités	24
2.2 Logiciel pour placement de pôles robuste - "PPMaster"	30
2.3 Calibrage de sensibilités par optimisation convexe	31
2.4 Notes et références	36
3 Commande multivariable par calibrage de sensibilités	39
3.1 Notation	39
3.2 Boucle fermée - sensibilités	41
3.3 Calibrage de sensibilités pour la robustesse et la performance	44
3.3.1 Condition de calibrage	45
3.3.2 Spécifications de performance par calibrage	46
3.4 Notes et références	50
4 Placement de pôles MIMO avec calibrage de sensibilités	53
4.1 Structure de régulateur	54

4.2	Placement de pôles multivariable	57
4.3	Placement de pôles avec calibrage de sensibilités	60
4.4	Procédure de synthèse de régulateurs par placement des pôles	64
4.5	Exemples de synthèse	65
4.5.1	Colonne à distiller	65
4.5.2	Autres exemples	69
4.6	Notes et références	69
5	Calibrage de sensibilités par optimisation convexe	73
5.1	Régulateur multivariable pour optimisation convexe	74
5.1.1	Paramétrisation Youla-Kucera des régulateurs	74
5.1.2	Structure de $Q(z^{-1})$	76
5.1.3	Structure complète du régulateur	82
5.2	Optimisation convexe	85
5.2.1	Calibrage de sensibilités	85
5.2.2	Poursuite en boucle fermée	90
5.2.3	δ -stabilité de la boucle fermée	94
5.2.4	Fonctions multi-critère	98
5.2.5	Algorithme d'optimisation	99
5.3	Sommaire de la méthodologie - étapes de conception	101
5.4	Comparaison avec la commande H_∞	104
5.5	Exemples de synthèse	108
5.5.1	Réacteur parfaitement agité (Stirred tank reactor)	108
5.5.2	Dynamique verticale d'avion	115
5.6	Logiciel de synthèse RMC	117
5.7	Notes et références	119
6	Synthèse de pré-compensateur par optimisation convexe	121
6.1	Régulateur à 2-degrés de liberté	121
6.2	Structure du pré-compensateur	122
6.3	Spécifications de performance en poursuite	124
6.4	Critères convexes de performances	126

6.4.1	Condition générale de poursuite	126
6.4.2	Critère pour une fonction de transfert	129
6.4.3	Critère du gain statique	130
6.4.4	Critère de robustesse	130
6.5	Synthèse de pré-compensateur - sommaire	132
6.6	Notes et références	134
7	Réduction de régulateur par identification en boucle fermée	135
7.1	Identification en boucle fermée - rappel	136
7.2	Réduction de régulateurs multivariables	139
7.2.1	Objectif de réduction	139
7.2.2	Réduction par identification en BF	141
7.2.3	Partie fixe de régulateur	143
7.3	Exemples de réduction de régulateurs	143
7.3.1	Réduction du régulateur SISO de la suspension active .	144
7.3.2	Exemple de réduction d'un régulateur MIMO	148
7.4	Notes et références	153
8	Conclusion et perspectives	155
8.1	Conclusion	155
8.2	Perspectives	158
	Bibliographie	160
A	Concepts de base	167
A.1	Systèmes linéaires	167
A.1.1	Décomposition aux valeurs singulières	167
A.1.2	Zéros de transmission	168
A.1.3	Conditions de robustesse	168
A.2	Convexité	170
A.2.1	Fonction convexe	170
A.2.2	Condition convexe	170
A.3	Représentations d'état de matrices de sensibilité	171

B Publications et rapports **175**

- B1. Logiciel pour l'enseignement et le calcul du placement de pôles robuste.
- B2. Pole placement with sensitivity function shaping using 2nd order digital notch filters.
- B3. Comments on the paper "Controller order reduction by identification in closed-loop applied to a benchmark".
- B4. Multivariable robust controller design by pole placement and sensitivity shaping.
- B5. Convex optimisation for robust multivariable controller design as alternative of H_∞ optimisation.
- B6. Multivariable controller reduction by closed-loop identification.
- B7. H.Prochazka. User's Guide for Pole-Placement Master.
- B8. Synthèse du filtre discret bande-étroite 2pôles/2zéros - application au calibrage des fonctions de sensibilité.
- B9. User's Guide for RMC controller design tool.

Notations

$\ M(z^{-1})\ _{\infty}$	norme H_{∞} d'une matrice de transfert $M(z^{-1})$
\hat{x}	valeur estimée de x
$RH_{z\infty}^{m \times n}$	espace des matrices de la dimension $m \times n$ qui sont rationnelles, stables, et discrètes
$\sigma_{max}(M)$	
$\bar{\sigma}(M)$	la plus grande valeur singulière de la décomposition en valeurs singulières d'une matrice complexe constante $M \in C^{m \times n}$
$Re\{a\}$	la partie réelle d'une valeur complexe $a \in C$
$\sup_{\omega_d}(f(\omega_d))$	la valeur maximal de la fonction $f(\omega_d)$ pour $\forall \omega_d$
$\deg(M(z^{-1}))$	ordre/degré du polynôme/matrice de transfert $M(z^{-1})$

Variables et constantes

$A(z^{-1})$	dénominateur polynomial du modèle de procédé SISO
$B(z^{-1})$	numérateur polynomial du modèle de procédé SISO
$A_m(z^{-1})$	dénominateur du modèle de référence pour le régulateur R-S-T
$B_m(z^{-1})$	numérateur du modèle de référence pour le régulateur R-S-T
a_i, b_i	coefficients des polynômes $A(z^{-1})$ et $B(z^{-1})$
$b(t)$	vecteur/scalaire de signaux de bruit de mesure
$A_M, B_M,$ C_M, D_M	matrices constantes de représentation d'états d'une matrice de transfert quelconque $M(z^{-1})$
$d(t)$	vecteur/scalaire de signaux de perturbation sur la commande de la boucle fermée
$d_Q(z^{-1})$	polynôme des pôles (dénominateur commun) de la matrice rationnelle $Q(z^{-1})$
$d_T(z^{-1})$	polynôme des pôles (dénominateur commun) de la matrice rationnelle $T(z^{-1})$

$e(t)$	vecteur/scalaire de signaux d'erreur entre la sortie désirée et la sortie réelle de la boucle fermée
$e_o(t)$	vecteur de signaux de l'erreur entre la sortie estimée la sortie réelle de la boucle fermée
F	matrice de retour d'état estimé
$F_{ij}(\omega_d)$	gabarit (borne supérieure) fréquentiel pour la plus grande valeur singulière ou pour les modules d'une sensibilité S_{ij}
$F_{yp}(z^{-1})$	
$F_{up}(z^{-1})$	filtres bande-étroite de placement de pôles pour les sensibilités S_{yp} et S_{up}
f_e	fréquence d'échantillonnage en Hz
f_n	fréquence naturelle en Hz d'une paire de racines complexes
f_t	fréquence de bande-étroite du filtre en Hz
$G(z^{-1})$	matrice de transfert du modèle échantillonné de procédé
$H_K(z^{-1})$	la partie fixe pré-spécifié du régulateur multivariable K
$H(z^{-1})$	matrice de transfert de la cascade $\mapsto H_K \mapsto G \mapsto$
$H_R(z^{-1})$	la partie fixe du numérateur $R(z^{-1})$ de régulateur R-S-T
$H_S(z^{-1})$	la partie fixe du dénominateur $S(z^{-1})$ de régulateur R-S-T
I	matrice unitaire $I = \text{diag}\{1\}$
$J(t)$	critère temporel d'optimisation
$K(z^{-1})$	matrice de transfert du régulateur MIMO en boucle fermée
$K_0(z^{-1})$	matrice de transfert du régulateur central de la paramétrisation "Youla-Kucera"
$K_c(z^{-1})$	matrice de transfert du régulateur optimisé MIMO
L	matrice de l'observateur d'état
$L(X)$	transformation d'un espace vectoriel à l'espace de matrices rationnelles
l	dimension du vecteur d'optimisation X
M_d	modèle de la perturbation à la commande
M_p	modèle de la perturbation à la sortie
$M_t(z^{-1})$	modèle de poursuite désirée du système complet

M_t	atténuation maximale du filtre bande-étroite
$M_{yy^*}(z^{-1})$	modèle de poursuite de la boucle fermée
n_x, n_y, n_u	nombre des états, nombre des sorties, et nombre d'entrées d'un modèle de procédé
n_A, n_B	les ordres (degrés) des polynômes $A(z^{-1})$ et $B(z^{-1})$
n_R, n_S, n_T	les ordres (degrés) des polynômes $R(z^{-1})$, $S(z^{-1})$, et $T(z^{-1})$
N_q	l'ordre de la paramétrisation du régulateur $K(z^{-1})$
N_t	l'ordre de la paramétrisation du pré-compensateur $T(z^{-1})$
$N_Q(z^{-1})$	la matrice polynômial des numérateurs de la matrice rationnelle $Q(z^{-1})$
$N_T(z^{-1})$	la matrice polynômial des numérateurs du pré-compensateur $T(z^{-1})$
$p(t)$	vecteur/scalaire de signaux de la perturbation sur la sortie
$P(z^{-1})$	polynôme de pôles de la boucle fermée SISO
$P_D(z^{-1})$	polynôme de pôles dominants de la boucle fermée SISO
$P_F^{(', '')}(z^{-1})$	polynôme de pôles auxiliaires de la boucle fermée SISO
$Q(z^{-1})$	matrice/fonction de transfert du paramètre libre de la paramétrisation de "Youla-Kucera"
r_i, s_i, t_i	coefficients des polynômes $R(z^{-1})$, $S(z^{-1})$ et $T(z^{-1})$
$R(z^{-1})$	numérateur du régulateur R-S-T
$R_0(z^{-1})$	la partie du numérateur $R(z^{-1})$ de régulateur R-S-T calculée par placement de pôles
s	opérateur de Laplace ou fréquentiel $s = j\omega$
$S(z^{-1})$	dénominateur du régulateur R-S-T
$S_0(z^{-1})$	la partie du dénominateur $S(z^{-1})$ d'un régulateur SISO calculée par placement de pôles
$T(z^{-1})$	le pré-compensateur SISO ou MIMO pour garantir une poursuite désirée de la boucle fermée
t_m	temps de montée d'un système
$r(t)$	vecteur/scalaire de signaux de la référence (consigne)

$S_{ij}(z^{-1})$	matrice/fonction de sensibilité (de transfert) entre un signal extérieur quelconque $j(t)$ et un signal quelconque $i(t)$ de BF
t	temps en seconds
$T_{1ij}(z^{-1}),$ $T_{2ij}(z^{-1}),$ $T_{3ij}(z^{-1})$	matrices/fonctions de transfert qui décrivent une sensibilité S_{ij} si la paramétrisation "Youla-Kucera" est appliquée au régulateur
T_e	période d'échantillonnage en seconds
$u(t)$	vecteur/scalaire de signaux de la sortie de partie fixe (commande envoyée au procédé)
$u_G(t)$	vecteur/scalaire de signaux de l'entrée du procédé ou du modèle
$u_K(t)$	vecteur de signaux de la commande du régulateur central $K_0(z^{-1})$ ou du régulateur optimisé $K_c(z^{-1})$
v_1, v_2	vecteurs sélecteurs avec tous les coefficients 0 sauf un qui est égale à 1
$W_i(z^{-1})$	matrice de pondération d'une sortie $i(t)$
$W_j(z^{-1})$	matrice de pondération d'une entrée $j(t)$
$X,$ X_1, X_2 X_a, X_b	vecteur réel d'optimisation
$y(t)$	vecteur/scalaire de signaux de la sortie de boucle fermée
$y_G(t)$	vecteur/scalaire de signaux de la sortie de modèle ou de procédé
$y^*(t)$	vecteur/scalair de signaux de la sortie désirée de BF
$y_o(t)$	vecteur de signaux de la sortie estimée du procédé
$y_Q(t)$	vecteur de signaux de la sortie de matrice $Q(z^{-1})$
z^{-1}	opérateur fréquentiel $z^{-1} = e^{-j\omega T_e}$ ou temporel (retard un T_e) ¹

¹Pour une uniformité de la notation, l'opérateur z^{-1} a été utilisé tant pour le domaine fréquentiel que pour le domaine temporel le contexte étant non ambiguë.

$\alpha(z^{-1})$	dénominateur de la fonction de transfert $Q(z^{-1})$ pour l'optimisation SISO
α_i	coefficients de pondération dans les fonctions multi-critères
$\beta(z^{-1})$	numérateur de la fonction de transfert $Q(z^{-1})$ pour l'optimisation SISO
$\gamma(z^{-1}),$ $\gamma_R(z^{-1}),$ $\gamma_S(z^{-1})$	numérateurs des filtres bande-étroite $F(z^{-1}), F_{up}(z^{-1}),$ et $F_{yp}(z^{-1})$
$\delta(z^{-1}),$ $\delta_R(z^{-1}),$ $\delta_S(z^{-1})$	dénominateurs des filtres bande-étroite $F(z^{-1}), F_{up}(z^{-1}),$ et $F_{yp}(z^{-1})$
$\varepsilon_{ij}(t)$	erreur du signal $j(t)$ pour deux systèmes différents
ζ	amortissement d'une paire de racines complexes
ζ_{den}	amortissement du dénominateur de l'équivalent continu du filtre bande-étroite
$\phi_k(X)$	fonction de critère k
$\phi_{OBJ}(X)$	fonction d'objectif
$\phi_{CST}(X)$	fonction de contrainte
ω	pulsation en rad/s
ω_d	pulsation discrète (normalisé) $\omega_d \in \langle 0, \pi \rangle$
ω_e	pulsation d'échantillonnage en rad/s

Liste des abréviations

ARMAX	processus auto-régressif à moyenne ajustée et entrée exogène
BF	Boucle Fermée
BO	Boucle Ouverte
C.S.O.C.	Calibrage de Sensibilités par Optimisation Convexe
GUI	interface graphique d'utilisateur (Graphical User Interface)
HF	hautes fréquences (High Frequency)
LF	basses fréquences (Low Frequency)
LTI	Linéaire et Invariant en Temps (Linear Time Invariant)
LQG	Linéaire Quadratique Gaussian
MCR	Moindres Carrés Récursifs
MIMO	plusieurs entrées, plusieurs sorties (Multi-Input, Multi-Output)
R-S-T	un type de régulateur monovariante numérique (calculé en générale par placement de pôles)
RE	Représentation d'Etat
SBPA	Signal Binaire Pseudo-Aléatoire
SISO	une entrée, une sortie (Single-Input Single-Output)

Chapitre 1

Introduction

1.1 Motivation et contexte

La commande des procédés continus est une problématique concernant toutes les branches de l'industrie, de l'industrie aérospatiale à l'industrie alimentaire. Les machines et les procédés utilisés en pratique sont de plus en plus sophistiqués c'est qui a pour but d'obtenir des performances supérieures. Cela est, entre autre, possible grâce au développement de l'informatique qui est aujourd'hui indispensable pour mettre en oeuvre les algorithmes de la commande. Mais malgré que l'informatique exige des algorithmes numériques souvent les méthodes de synthèse d'algorithmes de commande produisent des algorithmes en temps continu. Ensuite, ils doivent être discrétisés avec tous les problèmes liés à cette opération. Ce mémoire a pour le but de développer des nouveaux outils, méthodes et procédures pour le calcul des régulateurs (régulateur nommé aussi correcteur ou compensateur est un algorithme/loi de commande) numériques linéaires robustes. Trois idées essentielles ont été retenues pour le travail :

- Le régulateur résultant doit être numérique, et il est calculé à partir d'un modèle échantillonné du procédé. Aussi, toutes les méthodes et algorithmes présentés dans ce travail traitent des systèmes numériques. En conséquence, le régulateur peut être appliqué directement et les

fréquences d'échantillonnages sont beaucoup moins élevées que dans le cas des algorithmes discrétisés (pour explication voir [Lan02]).

- Les outils et les méthodes développés sont censés d'être programmés et ensuite utilisés pour la pratique d'ingénierie. En conséquence, il faut automatiser la procédure de calcul des régulateurs, et où ce n'est pas possible, des règles claires et simples doivent être donnés. La commande H_∞ est actuellement utilisée dans la plupart de cas pour le calcul des régulateurs robustes MIMO. Son désavantage principal est justement l'absence des règles claires et d'un certain niveau d'automatisation de la synthèse pour rendre la synthèse plus pratique. Pour quelle raison ? Les moyens d'ajustement du régulateur dans cette méthode : la synthèse des filtres de pondération, le choix de l'objectif de synthèse de régulateur, n'ont pas des liens transparents et logiques avec les spécifications de performance. Par contre pour la méthode de synthèse par calibrage de sensibilités par optimisation convexe [Lan98], les étapes de la synthèse se déduisent directement à partir des spécifications de performance. En plus, une partie importante de la synthèse est effectuée par un algorithme d'optimisation. Ainsi le lien entre l'ajustement du régulateur et les spécifications de performance sont beaucoup plus clairs permettant une synthèse beaucoup plus astucieux.
- Les algorithmes développées doivent être programmés dans le cadre de ce travail et les logiciels résultants avec des guides d'utilisation font partie de cette thèse.

Remarque 1.1 : L'analyse des avantages/ désavantages d'utilisation d'un modèle échantillonné et des conseils concernant le choix de période d'échantillonnage se trouvent par exemple dans [Lan02].

Le développement des méthodes de synthèse des régulateurs robustes linéaires numériques est le sujet principal du travail. La commande des procédés temps-continu en boucle fermée est la tâche de ce régulateurs. La procédure traditionnelle de la synthèse basée sur le modèle de procédé est considérée et

elle est constituée des étapes suivantes :

1. *Identification.* Un modèle(s) linéaire échantillonné du procédé réel est identifié à partir des données réelles. Il est aussi possible de modéliser le procédé et ensuite échantillonné le modèle continu obtenu.
2. *Spécifications de performance et de robustesse.* Les performances et la robustesse désirées pour le système commandé sont définies. Les performances de la boucle fermée en régulation et les performances en poursuite sont souvent spécifiées séparément.
3. *Synthèse de régulateur.* Avec le(s) modèle(s) du procédé, le régulateur est calculé tel qu'il vérifie sur le modèle (en simulation) les spécifications de performance et robustesse.
4. *Test temps-réel.* Le régulateur est testé sur le procédé réel. Si les performances réelles ne correspondent pas aux performances en simulation, il faut améliorer le modèle du procédé par exemple par l'identification en boucle fermée.

Le travail de thèse est consacré essentiellement à la 3^e étape, et surtout, à la problématique de la synthèse des régulateurs robustes linéaires multivariables numériques (système multivariable est un système de plusieurs entrées plusieurs sorties, en anglais multi-input multi-output ou MIMO).

Malgré que la majorité des procédés sont des systèmes non-linéaires, les régulateurs linéaires sont utilisés très souvent, car autour d'un point de fonctionnement la plupart des procédés peuvent être décrits par des modèles dynamiques linéaires. La synthèse de la commande pour ce type de modèles est beaucoup plus simple et aussi les régulateurs obtenus sont moins complexes par rapport aux modèles et régulateurs non-linéaires.

Pour un fonctionnement satisfaisant de la commande linéaire sur des systèmes réels, il faut que le régulateur appliqué soit robuste, car le procédé est affectés par des nombreuses perturbations, par le bruit et par l'incertitude du procédé. En d'autres termes, il doit être capable de fournir les performances désirées en présence de ces effets négatifs. Malheureusement, dans certains cas le comportement du procédé est tel, qu'un seul régulateur

linéaire n'est pas capable de le commander d'une façon satisfaisante. Dans ce cas un autre type de la loi de commande doit être utilisé comme la commande adaptative, ou les commandes non-linéaire, etc. Dans cet ouvrage seule la commande robuste par un régulateur linéaire est considérée.

Aujourd'hui, une majorité des méthodes de synthèse de régulateurs robustes est basé sur l'approche fréquentielle de traitement de robustesse et des performances. Dans cette approche, une analyse fréquentielle des incertitudes, des perturbations, et des performances du système est faite. Ensuite, à la base de ces analyses les propriétés fréquentielles (module, valeur singulière) du système bouclé (et en conséquence du régulateur) sont ajustées pour rendre la boucle fermée moins sensible dans les fréquences où les effets négatifs sont importants. Les exemples typiques de cette approche sont la commande H_∞ [Zho98, Mac89] avec la μ -analyse, le placement de pôles avec calibrage de sensibilités [LK98, PL03], ou le calibrage de sensibilités par optimisation convexe [Lan98].

Ce travail exploite l'approche fréquentielle de traitement de robustesse et plus particulièrement la procédure de la synthèse de régulateur par calibrage des sensibilités et optimisation convexe (C.S.O.C.). Cette procédure a été développée pour les systèmes monovariables (SISO) [Lan98, LK98] et ici elle est étendue dans le domaine multivariable (MIMO). Aujourd'hui dans le domaine multivariable, la procédure de synthèse la plus moderne qui utilise l'approche fréquentielle est la commande H_∞ avec μ -analyse [Mac89, Zho98, DF99]. Malgré son potentiel cette procédure présente un certain nombre de difficultés en pratique :

- La relation entre les filtres de pondération qui ajustent le régulateur et les réponses fréquentielles de sensibilités (elle indiquent les performances du système) est vague. En principe, la réponse fréquentielle de l'inverse du filtre de pondération peut être considéré comme une borne supérieure sur des sensibilités liées au signal filtré par ce filtre.
- En général, la réponse fréquentielle d'une sensibilité fait l'objet de contraintes - des bornes (des gabarits) définies dans des spécifications.

Typiquement le filtre de pondération est calculé comme l'inverse d'un gabarit et il faut chercher une approximation analytique et stable d'un tel gabarit.

- Un filtre de pondération influence toutes les sensibilités liées au signal filtré. Pour éliminer des influences indésirables sur certaines sensibilités, il faut ajouter des filtres supplémentaires qui accroissent la complexité du régulateur et compliquent la synthèse.
- Si le régulateur n'a pas une complexité suffisante pour obtenir les performances désirées, il faut augmenter l'ordre des filtres de pondération. En conséquences, il faut re-toucher les filtres sans des règles claires.

Tous les problèmes de la commande H_∞ indiqués peuvent être évités ou simplifiés avec la méthode de synthèse par le calibrage de sensibilités par optimisation convexe. Cette méthode dans une première étapes détermine quelles sensibilités seront ajustées et des gabarits (sur les modules) pour chaque de sensibilités choisies. Les gabarits sont déterminés à partir des spécifications de performance et de robustesse et ils peuvent être définies simplement par un ensemble des points. Ensuite le régulateur central est synthétisé utilisant les techniques de placement de pôles avec calibrage de sensibilités [Lan02, PL03], LQG, ou une autre technique. Finalement, un bloc des paramètres libres est ajouté au régulateur central et les paramètres sont optimisés pour obtenir les performances et la robustesse spécifiées.

1.2 Objectif de travail

L'objectif de ce travail est de développer une procédure de la synthèse des régulateurs robustes numériques linéaires temps-invariants pour des procédés continus multivariables caractérisés par un modèle échantillonné. La synthèse sera basée sur l'analyse fréquentielle de la boucle fermée (l'outil moderne de traitement de robustesse et de performances [Mac89, GL95, Zho98, DF99, Lan02]) et elle nécessite un modèle échantillonné linéaire temps-invariant du procédé.

La procédure développée comporte le placement de pôles [LK98, Lan02], le calibrage des sensibilités [LK98] et l'optimisation convexe [BB91, LL99, Lan98]. Le placement de pôles calcule le régulateur central nécessaire dans la paramétrisation de régulateur utilisée. Le calibrage des sensibilité désigne une approche de synthèse de régulateur robustes et performants par l'analyse fréquentielle de la boucle fermée. L'optimisation convexe basée sur la paraméterisation "Youla-Kucera" [Kuc79, BB91, Zho98] de tous régulateurs stabilisant devrait simplifier et automatiser la synthèse de régulateur.

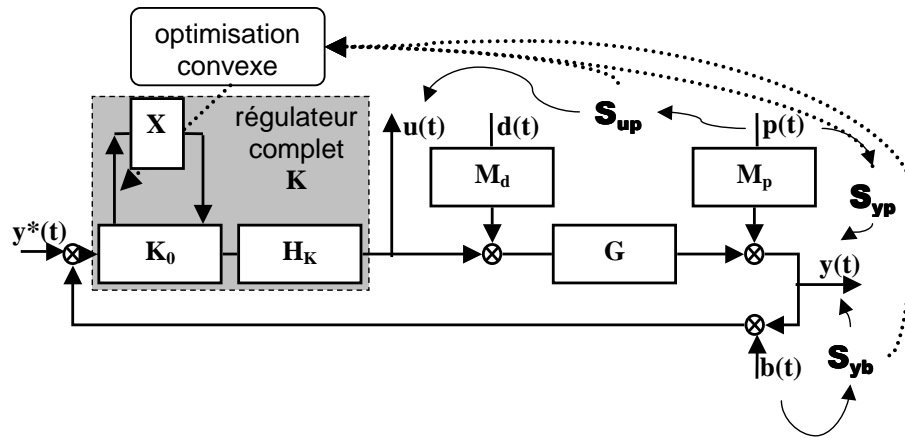


FIG. 1.1 – Principe de l'optimisation du régulateur en boucle fermée

Le schéma donné dans la Fig.1.1 représente le principe de la procédure de synthèse. Le régulateur à calculer est en boucle fermée avec le modèle échantillonné G du procédé où les perturbations $d(t)$ sur l'entrée du procédé, les perturbations $p(t)$ sur la sortie du procédé et le bruit de mesure $b(t)$ sont supposés d'agir sur le système bouclé. L'algorithme d'optimisation modifie la partie à optimiser du régulateur (décrit ici comme un vecteur X des coefficients inconnus) afin de donner une forme appropriée aux sensibilités S_{yp} , S_{yb} , S_{up} , etc. Les sensibilités sont des matrices de transfert qui décrivent l'influence des perturbations sur la commande $u(t)$, sur la sortie $y(t)$ ou sur un autre signal du système. Dans le cas du calibrage de sensibilités, elles servent comme une indication de la robustesse et de la performance de la boucle

fermée. Le principe du calibrage est le suivant : D'abord, les spécifications de performance et robustesse sont transformées en gabarits imposés aux sensibilités. Ensuite, le régulateur est ajusté pour faire entrer les sensibilités dans de ces gabarits. La structure du régulateur de la Fig.1.1 est déterminée par la paramérisation appliquée où le régulateur central K_0 est, dans notre cas, un observateur d'états avec le retour d'état estimé et H_K est la partie fixe du régulateur imposée par des spécifications avant l'optimisation.

Le régulateur complet K montré dans la Fig.1.1 est le régulateur de la boucle fermée. Il assure des performances désirées en régulation et éventuellement en poursuite ainsi que les spécifications de robustesse. Si le comportement du système en poursuite n'est pas satisfaisant, un pré-compensateur T peut être calculé (voir Fig.1.2) pour améliorer le comportement. Une partie de ce travail propose une procédure de synthèse de pré-compensateur basée sur l'optimisation convexe. L'idée de la procédure est illustrée dans la Fig.1.2. L'algorithme d'optimisation ajuste le vecteur des coefficients X qui forme le pré-compensateur, afin d'assurer une poursuite désirée définie par un modèle de poursuite M_t . Les propriétés de poursuite du système sont caractérisées par la sensibilité S_{yr} (matrice de transfert entre la référence $r(t)$ et la sortie $y(t)$) et l'algorithme minimise la différence entre S_{yr} et M_t . Les deux autres sensibilités S_{er} et S_{ur} peuvent être concernées pour obtenir une meilleure robustesse du pré-compensateur.

L'optimisation convexe, comme d'autres méthodes sophistiquées de synthèse, donne souvent des régulateurs très complexes. Des régulateurs d'ordre élevé sont souvent non-souhaitable. Soit la taille du régulateur est limitée par des contraintes matérielles, soit le régulateur simple est plus facile à implémenter est à comprendre. Il est donc important un pratique de mettre en oeuvre des régulateurs d'une complexité réduite. Pour ces raisons, une partie de ce travail concerne aussi la réduction de régulateurs. Une approximation simple de la matrice de transfert du régulateur par une autre matrice d'ordre plus petite ne peut pas être utilisée. Elle ne prend pas en compte des effets d'un telle réduction sur des performances de la boucle fermée. Des méthodes

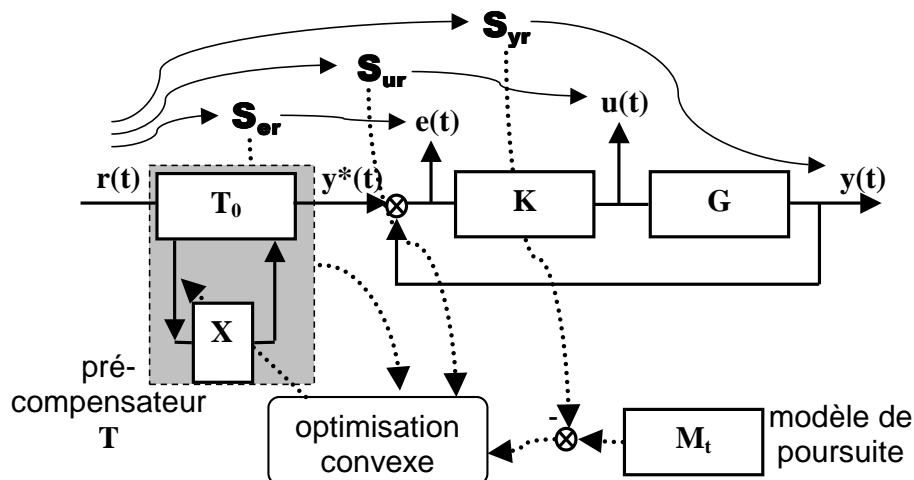


FIG. 1.2 – Principe de l'optimisation du pré-compensateur

pour la réduction de régulateurs multivariables qui préservent dans une large mesure les performances désirées de la boucle fermée ont été développées dans le cadre de cette étude.

Pour la réduction des régulateurs multivariables nous allons explorer l'idée de la réduction par l'identification en boucle fermée présentée dans [LKC01] pour les systèmes monovariables. En bref, cette réduction utilise un algorithme modifié d'identification en boucle fermée pour identifier le régulateur d'ordre réduit. Nous allons utiliser une approche similaire où pour l'identification un algorithme standard d'identification est exploité sans des modifications majeurs.

1.3 Organisation de mémoire

Le mémoire concerne principalement les systèmes MIMO sauf le Chapitre 2 qui traite la synthèse de régulateurs robustes SISO. Le texte consiste de 7 parties avec les contenus suivants :

- Le Chapitre 2 rappelle la synthèse de régulateurs robustes monovariables par le placement de pôles, le calibrage de sensibilités et l'optimi-

sation convexe. Aussi, il présente une nouvelle approche de la synthèse par placement de pôles en utilisant les filtres bande-étroite du 2^e ordre et un logiciel associé développé en cadre de ce travail.

- Le Chapitre 3 évoque la théorie fondamentale de la commande multivariable et développe l'idée de la synthèse de régulateurs robustes multivariables par calibrage de sensibilité.
- Le Chapitre 4 présente la procédure développée de la synthèse de régulateur MIMO par placement de pôles et calibrage de sensibilité. Le régulateur a la forme d'observateur avec retour d'état et il est utilisé par la suite comme le régulateur central pour la synthèse de régulateurs par optimisation convexe.
- Le Chapitre 5 est la partie essentielle et elle développe la méthode de synthèse de régulateurs robustes multivariables par calibrage de sensibilités et optimisation convexe. La méthode est basée sur la même procédure réalisée dans le domaine monovariante et elle est réalisée comme une boîte à outils interactive pour Matlab.
- Le Chapitre 6 traite la synthèse de pré-compensateur multivariable de la poursuite par optimisation convexe. Le pré-compensateur nous permet de séparer les spécifications en régulation et les spécifications en poursuite.
- Le Chapitre 7 concerne la technique de réduction de régulateur multivariable par identification en boucle fermée. La méthode développée fait suite à un approche similaire développé pour les régulateurs monovariants.
- Le Chapitre 8 donne les perspectives de ce travail et tire quelques conclusions.

Chapitre 2

Rappel sur la méthodologie monovariabile

Dans ce chapitre la méthode de synthèse de régulateurs SISO par placement de pôles avec calibrage des fonctions de sensibilité [LK98] est brièvement rappelée. Également, les améliorations apportées à cette méthode sont présentées [PL03]. Ensuite, la théorie du calibrage des fonctions de sensibilité par l'optimisation convexe [Lan98] est introduite avec la notation utilisée ultérieurement. Cette théorie monovariabile nous serve comme le point de départ pour le développement de la méthodologie multivariabile.

In this chapter the pole placement with sensitivity shaping controller design method [LK98] is recalled briefly. The improvements of the method made by the author [PL03] are presented as well. Next, the theory of sensitivity shaping by convex optimisation [Lan98] is introduced with notations employed later. This monovariabile theory will serve us as a starting point for multivariabile methodology development.

2.1 Placement de pôles avec calibrage de sensibilités

La configuration utilisée (utilisée aussi par [Lan98, Lan02]) de la boucle fermée avec un régulateur polynômial numérique R-S-T est montrée dans la Fig.2.1. Le modèle échantillonné du procédé $G(z^{-1})$ et les polynômes du régulateur ont les formes suivantes :

$$G(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})} = \frac{z^{-d}(b_1z^{-1} + b_2z^{-2} + \dots + b_{n_B}z^{-n_B})}{1 + a_1z^{-1} + a_2z^{-2} + \dots + a_{n_A}z^{-n_A}} \quad (2.1)$$

et

$$R(z^{-1}) = r_0 + r_1z^{-1} + r_2z^{-2} + \dots + r_{n_R}z^{-n_R} \quad (2.2)$$

$$S(z^{-1}) = s_0 + s_1z^{-1} + s_2z^{-2} + \dots + s_{n_S}z^{-n_S} \quad (2.3)$$

$$T(z^{-1}) = t_0 + t_1z^{-1} + t_2z^{-2} + \dots + t_{n_T}z^{-n_T} \quad (2.4)$$

avec les degrés n_A, n_B, n_R, n_S, n_T des polynômes et le retard pur d du modèle. La période d'échantillonnage en seconds du système est T_e et z^{-1} est soit l'opérateur temporel de retard, soit l'opérateur fréquentiel avec $\omega_d \in \langle 0, \frac{\pi}{T_e} \rangle$ la pulsation discrète normalisée (voir le Chapitre 3).

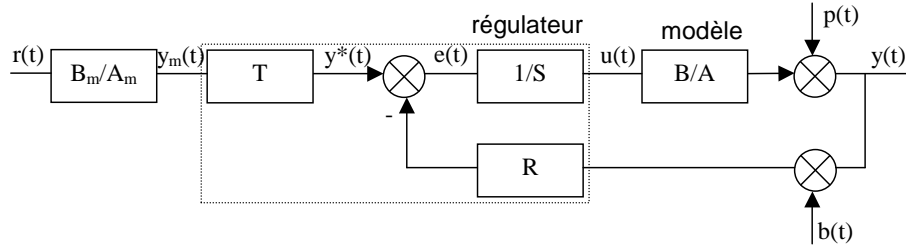


FIG. 2.1 – Boucle fermée avec un régulateur R-S-T

Le régulateur R-S de la boucle fermée est paramétrisé de la façon suivante :

$$\frac{R(z^{-1})}{S(z^{-1})} = \frac{R_0(z^{-1})H_R(z^{-1})}{S_0(z^{-1})H_S(z^{-1})} \quad (2.5)$$

où $H_R(z^{-1})$ et $H_S(z^{-1})$ sont les parties fixes (soit imposées par les spécifications de synthèse, soit fixées pour calibrer des fonctions de sensibilité). Les

polynômes $R_0(z^{-1})$, $S_0(z^{-1})$ sont les solutions de l'équation de Bezout (2.6) (pôles de la boucle fermée) de placement de pôles.

$$A(z^{-1})S_0(z^{-1})H_S(z^{-1}) + B(z^{-1})R_0(z^{-1})H_R(z^{-1}) = \underbrace{P_D(z^{-1})P_F(z^{-1})}_{P(z^{-1})} \quad (2.6)$$

Dans l'équation (2.6), le polynôme $P(z^{-1})$ représente les pôles désirés de la boucle fermée, où $P_D(z^{-1})$ définit les pôles dominants (les plus lents) et $P_F(z^{-1})$ les pôles auxiliaires. La partie gauche de (2.6) est le dénominateur des fonctions de sensibilité. Les pôles désirés doivent être fixés afin de satisfaire les spécifications de performance et de robustesse imposées. L'équation de Bezout a une solution unique minimale sous les conditions suivantes :

$$\begin{aligned} \deg(P(z^{-1})) &= n_P \leq n_A + n_{H_S}n_B + n_{H_R} + d - 1 \\ \deg(S(z^{-1})) &= n_S = n_B + n_{H_R} + d - 1 \\ \deg(R(z^{-1})) &= n_R = n_A + n_{H_S} - 1 \end{aligned} \quad (2.7)$$

La partie T de la poursuite est utilisée dans le régulateur pour compenser la dynamique de la boucle fermée afin que la fonction de transfert entre $r(t)$ et $y(t)$ s'approche au modèle de la référence $\frac{B_m(z^{-1})}{A_m(z^{-1})}$. Le polynôme $T(z^{-1})$ est considéré d'avoir une des trois structures suivantes :

- Le polynôme $T(z^{-1})$ contient tous les pôles de la boucle fermée donnée par le polynôme $P(z^{-1}) = A(z^{-1})S(z^{-1}) + B(z^{-1})R(z^{-1})$ et son gain statique est ajusté pour que le gain statique du système complet corresponde au $\frac{B_m(1)}{A_m(1)}$. Par conséquent $T(z^{-1})$ a la forme :

$$T(z^{-1}) = \frac{P(z^{-1})}{B(1)} \quad (2.8)$$

et la fonction de transfert entre la référence $r(t)$ et la sortie $y(t)$ devient :

$$S_{yr}(z^{-1}) = \frac{B_m(z^{-1})B(z^{-1})}{A_m(z^{-1})B(1)} \quad (2.9)$$

- Le polynôme $T(z^{-1})$ contient les pôles dominants de la boucle fermée donnés par $P_D(z^{-1})$ et son gain statique est ajusté. Ainsi, la partie de

la poursuite a la forme suivante :

$$T(z^{-1}) = \frac{P_D(z^{-1})P_F(1)}{B(1)} \quad (2.10)$$

et la fonction de transfert entre la référence $r(t)$ et la sortie $y(t)$ est maintenant :

$$S_{yr}(z^{-1}) = \frac{B_m(z^{-1})B(z^{-1})P_F(1)}{A_m(z^{-1})P_F(z^{-1})B(1)} \quad (2.11)$$

- La partie T est un gain avec la valeur $T = \frac{P(1)}{B(1)}$ La fonction de transfert entre la référence $r(t)$ et la sortie $y(t)$ est la suivante :

$$S_{yr}(z^{-1}) = \frac{B_m(z^{-1})B(z^{-1})P(1)}{A_m(z^{-1})P(z^{-1})B(1)} \quad (2.12)$$

Le modèle de référence $\frac{B_m}{A_m}$ est généralement une fonction de transfert du 2^e ordre avec la dynamique désirée de la poursuite.

La méthode SISO de placement de pôles avec calibrage de fonctions de sensibilité considère principalement la perturbation de sortie $p(t)$, et le bruit de mesure $b(t)$ agissant sur le système bouclé - voir Fig.2.1. L'influence de ces signaux sur la sortie $y(t)$ et sur la commande $u(t)$ est exprimée par les fonctions de sensibilité (2.13), (2.14), (2.15). Il est démontré [LK98, Lan02] qu'une bonne robustesse et la plupart des performances peuvent être garanties en imposant les gabarits (bornes) supérieurs appropriés sur ces fonctions.

Dans le placement de pôles SISO avec calibrage de fonctions de sensibilité [Lan02, LLRB96, LK98], surtout ces trois fonctions de sensibilité sont prises en compte :

- fonction de sensibilité perturbation-sortie $y(z^{-1}) = S_{yp}(z^{-1})p(z^{-1})$:

$$S_{yp}(z^{-1}) = \frac{A(z^{-1})S_0(z^{-1})H_S(z^{-1})}{P(z^{-1})} \quad (2.13)$$

- fonction de sensibilité perturbation-entrée $u(z^{-1}) = S_{up}(z^{-1})p(z^{-1})$:

$$S_{up}(z^{-1}) = \frac{-A(z^{-1})R_0(z^{-1})H_R(z^{-1})}{P(z^{-1})} \quad (2.14)$$

– fonction de sensibilité bruit-sortie $y(z^{-1}) = S_{yb}(z^{-1})b(z^{-1})$:

$$S_{yb}(z^{-1}) = \frac{-B(z^{-1})R_0(z^{-1})H_R(z^{-1})}{P(z^{-1})} \quad (2.15)$$

Afin de mieux ajuster (calibrer) les fonctions de sensibilité la paramétrisation suivante de fonctions de sensibilité $S_{yp}(z^{-1})$ et $S_{up}(z^{-1})$ est proposée (voir aussi l'Annexe B2, B8, ou [PL03]) :

$$H_R(z^{-1}) = H'_R(z^{-1})\gamma_R(z^{-1}) \quad \text{avec} \quad P_F(z^{-1}) = P'_F(z^{-1})\delta_R(z^{-1})$$

ou

$$H_S(z^{-1}) = H'_S(z^{-1})\gamma_S(z^{-1}) \quad \text{avec} \quad P_F(z^{-1}) = P''_F(z^{-1})\delta_S(z^{-1})$$

Avec cette notation les modules de fonctions de sensibilité deviennent :

$$|S_{yp}(z^{-1})| = \left| \frac{A(z^{-1})S_0(z^{-1})H'_S(z^{-1})}{P''_F(z^{-1})} \right| \left| \frac{\gamma_S(z^{-1})}{\delta_S(z^{-1})} \right| \quad (2.16)$$

$$|S_{up}(z^{-1})| = \left| \frac{A(z^{-1})R_0(z^{-1})H'_R(z^{-1})}{P'_F(z^{-1})} \right| \left| \frac{\gamma_R(z^{-1})}{\delta_R(z^{-1})} \right| \quad (2.17)$$

où les filtres $F_{yp}(z^{-1}) = \frac{\gamma_S(z^{-1})}{\delta_S(z^{-1})}$ et $F_{up}(z^{-1}) = \frac{\gamma_R(z^{-1})}{\delta_R(z^{-1})}$ contiennent plusieurs filtres 2zéros/2pôles facile à ajuster. Il s'agit des filtres bande étroite avec deux zéros et deux pôles pour lesquels nous avons développé un algorithme de calcul (voir l'Annexe B2, B8 ou [PL03, Lan02]). Les polynômes H'_R et H'_S représentent les parties fixes (intégrateur, ouverture de boucle, etc.) déterminées en fonction des spécifications.

Le filtre bande étroite a la forme suivante :

$$F(z^{-1}) = \frac{b_{z0} + b_{z1}z^{-1} + b_{z2}z^{-2}}{a_{z0} + a_{z1}z^{-1} + a_{z2}z^{-2}} = \frac{\gamma(z^{-1})}{\delta(z^{-1})} \quad (2.18)$$

et sa réponse fréquentielle est montrée dans la Fig.2.2. Pour déterminer le filtre il suffit de spécifier les paramètres suivants (voir aussi la Fig.2.2) :

- M_t - atténuation désirée maximale ($M_t < 0$) en dB
- f_t - fréquence centrale de la bande étroite en Hz
- ζ_{den} - l'amortissement du dénominateur du filtre analogique équivalent (généralement entre 0.3 et 1). La valeur est liée à la méthode de construction du filtre, voir l'Annexe B2, B8 ou [PL03, Lan02].

A partir de ces paramètres un filtre analogique est construit est ensuite discrétisé par la transformation bilinéaire¹. L'avantage de cette transformation est qu'elle préserve la réponse fréquentielle, c'est-à-dire les réponses du filtre analogique et du filtre discrétisé sont très similaires seulement "warping" de fréquences doit être considéré.

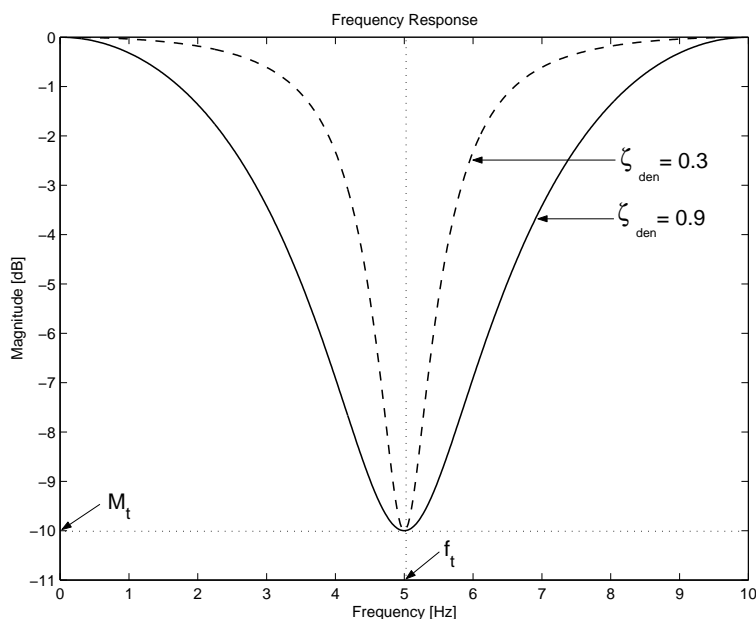


FIG. 2.2 – Exemple de réponses fréquentielles du filtre 2zéros/2pôles bande étroite

L'avantage de ce type de filtre est qu'il atténue seulement dans la zone désirée de sensibilité. L'influence sur les zones adjacentes est très limitée et dans des zones éloignées la sensibilité reste inchangée. Un exemple d'application d'un filtre bande étroite ($f_t = 1\text{Hz}$, $M_t = -5.4\text{dB}$, $\zeta_{den} = 0.8$) placé dans la fonction de sensibilité S_{yp} est présenté dans la Fig.2.3. Nous pouvons observer que l'action du filtre est limitée sur la zone autour de la fréquence de bande-étroite f_t . Particulièrement, le filtre n'a pas produit l'effet de "waterbed", qui est typique pour le calibrage de sensibilité S_{yp} .

¹Pour $f_t < 0.15f_e$ la synthèse de ces filtre peut se faire directement au discret, voir [PL03, Lan02] ou Annexe B2.

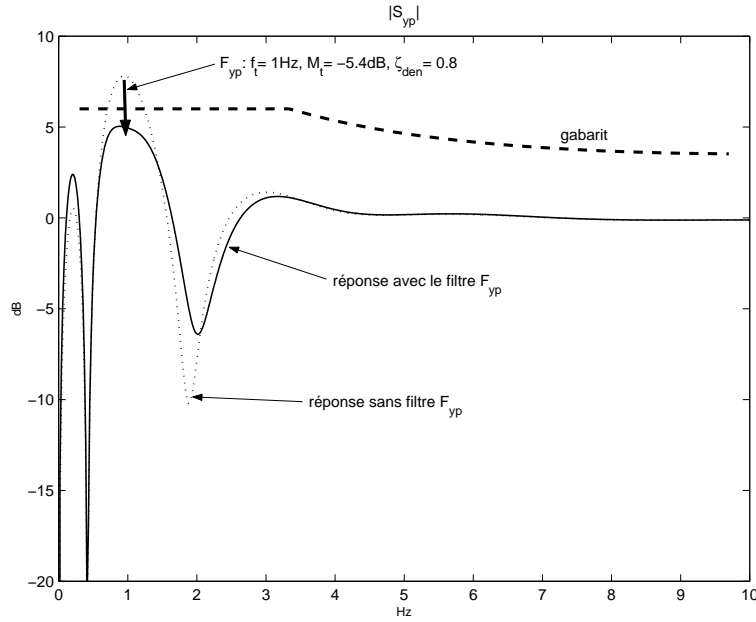


FIG. 2.3 – La réponse de $|S_{yp}|$ sans et avec le filtre 2zéros/2pôles bande étroite.

En conclusion, pour la synthèse de régulateur R-S nous disposons de ces paramètres ajustables de synthèse :

- $P_D(z^{-1})$ polynôme de pôles dominants désirés
- $P'_F(z^{-1})$ polynôme de pôles auxiliaires
- $H'_S(z^{-1})$ partie fixe du dénominateur S
- $H'_R(z^{-1})$ partie fixe du numérateur R
- $F_{yp}(z^{-1})$ filtres bande étroite appliqués sur S_{yp}
- $F_{up}(z^{-1})$ filtres bande étroite appliqués sur S_{up}

Le polynôme $T(z^{-1})$ assure les performances en poursuite (tracking) du système complet dans le cas où la dynamique désirée de la poursuite ne corresponde pas à la dynamique obtenue par le régulateur R-S (conçu pour réaliser une stabilisation robuste, rejet de perturbation, etc.). Plusieurs alternatives sont proposées pour la construction du pré-filtre $T(z^{-1})$ [Lan02] basées sur le principe d'élimination de la dynamique de BF avec des zéros de pré-filtre (voir (2.8) à (2.12)).

Afin d'ajuster les fonctions de sensibilité $S_{yp}(z^{-1})$, $S_{up}(z^{-1})$, et $S_{yb}(z^{-1})$, il est nécessaire de connaître les influences des différents paramètres de synthèse (P_D , H_S , ...) sur ces fonctions. Pour cela un ensemble des outils, des recommandations, et des règles [Lan02] et un logiciel [PL02] ont été développés pour simplifier la synthèse. Des exemples pratiques de calibrage de sensibilités par placement de pôles se trouvent à [Lan02, PL02, PL03].

2.2 Logiciel pour placement de pôles robuste - "PPMaster"

Dans la cadre de ce travail un logiciel de synthèse de régulateurs robustes SISO par placement de pôles avec calibrage de sensibilités a été développé [PL02]. Il s'agit d'une application interactive réalisée dans l'environnement de MATLAB[®] 5.3 en utilisant les boîtes à outils de la programmation générale, de l'interface d'utilisateur (graphical user interface - GUI), des polynômes et du traitement de signaux.

L'approche interactive est utilisée pour réaliser la synthèse du régulateur. Les modifications des paramètres ajustables de synthèse provoquent le réaffichage des propriétés du système bouclé. Les paramètres ajustables de synthèse sont les suivants (voir aussi la section 2.1) : pôles dominants définis par P_D , pôles auxiliaires définis par P'_F , les parties fixes du régulateur H'_S et H'_R , et les filtres bande-étroite F_{yp} , F_{up} . Les structures des paramètres ajustables disponibles dans le logiciel sont présentées dans Tab.2.1. Les modifications sont réalisées soit par une ligne de texte, soit par un "slider".

Les propriétés de la boucle fermée et du régulateur qui peuvent être surveillées lors de la synthèse sont résumées dans Tab. 2.2. Trois graphes de la fenêtre principale permettent de afficher les réponses fréquentielles et les cartes de pôles-zéros (voir la Fig.2.4). Une description détaillée du logiciel est donnée dans l'Annexe B7. Un exemple d'utilisation est dans l'Annexe B1 ou dans [PL02].

Symbole	Nom	Structures prévues	paramètres à ajuster
P_D	pôles dominants	i) paire des pôles complexes	f_n, ζ
P_{Fi}	pôles auxiliaires	i) paires des pôles complexes ii) pôle multiple $(1 - \alpha z^{-1})^N$	f_n, ζ α, N
H_{Ri}	partie fixe de R	i) zéro simple $(1 - \alpha z^{-1})$ ii) paire des zéros complexes	α f_n, ζ
H_{Si}	partie fixe de S	i) pôle simple $(1 - \alpha z^{-1})$ ii) paire des zéros complexes	α f_n, ζ
F_{ypi}	filtre bande étroite sur S_{yp}	i) fonctions de transfert	M_t, f_t, ζ_{den}
F_{upi}	filtre bande étroite sur S_{up}	i) fonctions de transfert	M_t, f_t, ζ_{den}

TAB. 2.1 – Paramètres ajustables de synthèse et les structures disponibles (f_n est la fréquence naturelle et ζ est l’amortissement d’une paire de racines complexes).

2.3 Calibrage de sensibilités par optimisation convexe

Le placement de pôles avec calibrage de fonctions de sensibilité par l’optimisation convexe est une méthode de synthèse de régulateurs numériques SISO introduite par [Lan98, LL99]. Elle est basée sur le placement de pôles avec calibrage de fonction de sensibilité [LK98, Lan02] et sur l’optimisation convexe [BB91]. L’optimisation automatise l’ajustement des sensibilités et elle est implémentée par la paramétrisation de ”Youla-Kucera” (nommé aussi Q-paramétrisation) [YJB76, Mac89, BB91], etc. Le régulateur central, calculé par le placement de pôles classique, est augmenté par la paramétrisation et la partie rajoutée du régulateur est optimisée afin d’obtenir des sensibilités

Nom	Description	Paramètres pour calibrer la propriété
$ S_{yp} $	v.abs. de la fonction de sensibilité perturbation-sortie + son gabarit	P_D, P_F H_S, F_{yp}
$ S_{up} $	v.abs. de la fonction de sensibilité perturbation-entrée + son gabarit	P_D, P_F H_R, F_{up}
$ S_{yb} $	v.abs. de la fonction de sensibilité bruit-sortie + son gabarit	P_D, P_F H_R
S	pôles du régulateur	H_S, F_{yp}
P	pôles de la boucle fermée	P_D, P_F F_{yp}, F_{up}

TAB. 2.2 – Les propriétés visualisées par le logiciel

désirées.

La paramétrisation du régulateur R-S est la suivante :

$$\frac{R(z^{-1})}{S(z^{-1})} = \frac{H_R(z^{-1})(R_0(z^{-1}) + A(z^{-1})H_S(z^{-1})Q(z^{-1}))}{H_S(z^{-1})(S_0(z^{-1}) - B(z^{-1})H_R(z^{-1})Q(z^{-1}))} \quad (2.19)$$

et il s'agit de la paramétrisation de Youla-Kucera [YJB76, Kuc79]. $Q(z^{-1})$ est la fonction de transfert rationnelle stable et propre (partie à optimiser rajoutée au régulateur) et $\frac{R_0(z^{-1})}{S_0(z^{-1})}$ est le régulateur central calculé par le placement de pôles². Ce régulateur est calculé par l'équation de Bezout (2.6).

Supposons la fonction de transfert $Q(z^{-1})$ de la forme :

$$Q(z^{-1}) = \frac{\beta(z^{-1})}{\alpha(z^{-1})} \quad (2.20)$$

avec le numérateur $\beta(z^{-1})$ et le dénominateur $\alpha(z^{-1})$. Dans ce cas, le régulateur optimisé peut avoir la forme suivante :

$$\frac{R(z^{-1})}{S(z^{-1})} = \frac{H_R(z^{-1})(R_0(z^{-1})\alpha(z^{-1}) + A(z^{-1})H_S(z^{-1})\beta(z^{-1}))}{H_S(z^{-1})(S_0(z^{-1})\alpha(z^{-1}) - B(z^{-1})H_R(z^{-1})\beta(z^{-1}))} \quad (2.21)$$

²D'autres méthodes peuvent être utilisés comme par exemple LQG ou la commande prédictive.

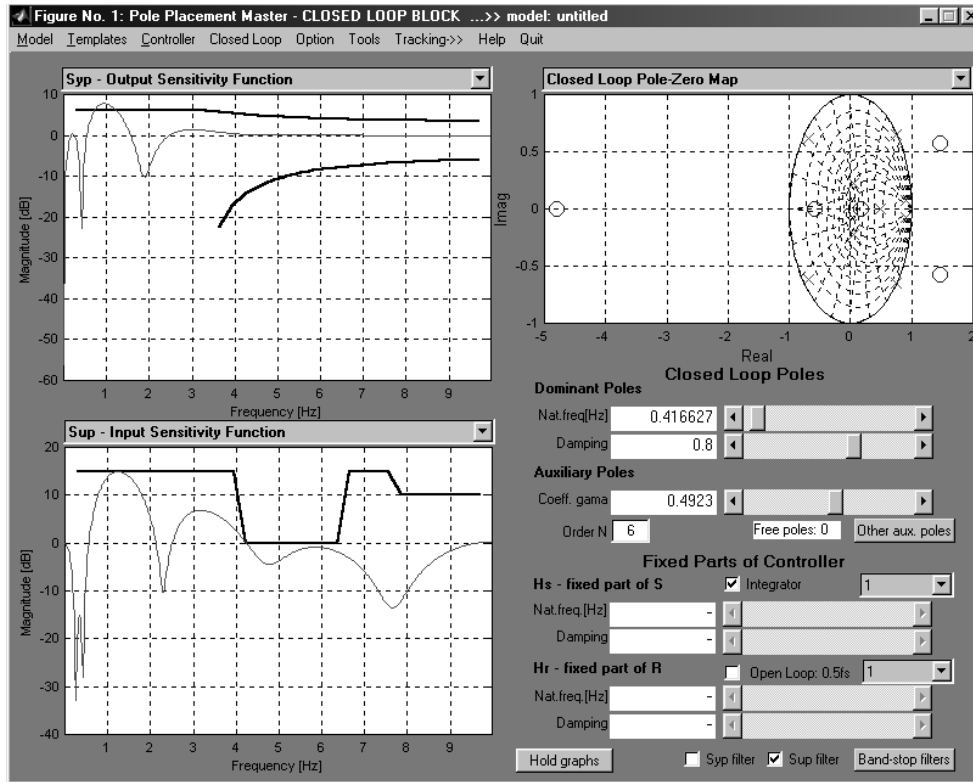


FIG. 2.4 – Fenêtre principale du logiciel

et les pôles de la boucle fermée sont :

$$A(z^{-1})S(z^{-1}) + B(z^{-1})R(z^{-1}) = P_D(z^{-1})P_F(z^{-1})\alpha(z^{-1}) = P(z^{-1}) \quad (2.22)$$

Les pôles de la boucle fermée sont ceux définis par le régulateur central (calculés par placement de pôles) et ceux placés par l'optimisation. C'est une propriété intéressante de la paramétrisation de "Youla-Kucera" avec $Q(z^{-1})$ sous la forme (2.20), car les pôles de $Q(z^{-1})$ apparaissent dans les pôles de la boucle fermée. En optimisant les pôles de $Q(z^{-1})$ et donc une partie de pôles de la boucle fermée, l'ajustement des fonctions de sensibilité devient très efficace. L'autre propriété importante de cette paramétrisation est la forme convexe de toutes les fonctions de sensibilité par rapport à $Q(z^{-1})$. Avec la Q-paramétrisation les fonctions de sensibilité ont la forme

générale suivante (dans le cas SISO) :

$$S_{ij}(z) = T_{1ij}(z^{-1}) + T_{2ij}(z^{-1})Q(z^{-1}) = T_{1ij}(z^{-1}) + T_{2ij}(z^{-1})\frac{\beta(z^{-1})}{\alpha(z^{-1})} \quad (2.23)$$

avec $T_{1ij}(z^{-1})$ et $T_{2ij}(z^{-1})$ des fonctions de transfert dépendantes de la boucle fermée. Les indices i, j signifient les signaux d'entrée et de sortie d'une fonction de sensibilité. Par exemple pour $S_{yp}(z^{-1})$, $i \equiv y$, $j \equiv p$, et à partir de (2.13), (2.19) nous avons :

$$T_{1ij} = \frac{A(z^{-1})H_S(z^{-1})S_0(z^{-1})}{P_D(z^{-1})P_F(z^{-1})} \quad T_{2ij} = -\frac{A(z^{-1})H_S(z^{-1})B(z^{-1})H_R(z^{-1})}{P_D(z^{-1})P_F(z^{-1})}$$

Afin d'ajuster les fonctions de sensibilité (2.23) nous imposons un gabarit supérieur sur le module de ces fonctions, et cela se traduit en la condition suivante :

$$\left| T_{1ij}(e^{-j\omega_d}) + T_{2ij}(e^{-j\omega_d})\frac{\beta(e^{-j\omega_d})}{\alpha(e^{-j\omega_d})} \right| \leq F_{ij}(\omega_d) \quad \forall \omega_d \quad (2.24)$$

où $F_{ij}(\omega_d)$ est le gabarit supérieur sur $|S_{ij}|$, qui dans notre cas n'est pas nécessairement rationnel et il est considéré positif $\forall \omega_d$. La pulsation $\omega_d = \omega T_e$ est une pulsation discrète normalisée $\omega_d \in \langle 0, \pi \rangle$.

La condition (2.24) est convexe par rapport à $Q(z^{-1})$ (pour la définition de convexité voir Annexe A.2) mais elle n'est pas par rapport à $\alpha(z^{-1})$ et $\beta(z^{-1})$. Nous allons multiplier l'inégalité (2.24) par $|\alpha(z^{-1})|$ et nous remplaçons cet élément par le terme $Re\{\alpha(z^{-1})\}$ dans la partie droite de l'inégalité. La condition devient :

$$\left| T_{1ij}(e^{-j\omega_d})\alpha(e^{-j\omega_d}) + T_{2ij}(e^{-j\omega_d})\beta(e^{-j\omega_d}) \right| \leq Re\{\alpha(e^{-j\omega_d})\}F_{ij}(\omega_d) \quad \forall \omega_d \quad (2.25)$$

Si la condition (2.25) est vérifiée, la condition (2.26) est également remplie. A noter, que (2.26) garantie $\alpha(e^{-j\omega_d})$ strictement positif réel.

$$Re\{\alpha(e^{-j\omega_d})\} > 0 \quad (2.26)$$

La condition (2.25) est convexe par rapport à $\alpha(z^{-1})$ et $\beta(z^{-1})$ [Lan98, RM94]. Malheureusement, la dernière opération (remplacement) n'est pas

sans coût, l'inégalité (2.25) est plus conservatrice que celle d'origine (2.24). Par contre l'avantage c'est que la condition (2.25) garantie la stabilité du polynôme $\alpha(z^{-1})$, c'est à dire la stabilité des pôles optimisés de la boucle fermée (voir (2.22)). La stabilité est assurée parce que en vérifiant la condition (2.25), le polynôme $\alpha(z^{-1})$ est strictement positif réel [Lan98].

Pour employer l'optimisation convexe, une paramétrisation de $\alpha(z^{-1})$ et de $\beta(z^{-1})$ est nécessaire. Dans [BB91] l'approximation de dimension finie de Ritz est proposée et elle est également utilisée par [Lan98]. Avec cette approximation le numérateur $\beta(z^{-1})$ et le dénominateur $\alpha(z^{-1})$ du paramètre $Q(z^{-1})$ ont la structure suivante :

$$\alpha(z^{-1}, X_a) = 1 + \sum_{k=1}^{N_q} x_{ak} \left(\frac{z_0 - z^{-1}}{1 - z_0 z^{-1}} \right)^k \quad (2.27)$$

$$X_a = [x_{a1}, x_{a2}, \dots, x_{aN_q}]^T$$

$$\beta(z^{-1}, X_b) = x_{b0} + \sum_{k=1}^{N_q} x_{bk} \left(\frac{z_0 - z^{-1}}{1 - z_0 z^{-1}} \right)^k \quad (2.28)$$

$$X_b = [x_{b0}, x_{b1}, x_{b2}, \dots, x_{bN_q}]^T$$

où $X = [X_a^T X_b^T]^T$ est le vecteur d'optimisation, N_q est l'ordre d'approximation et $z_0 \in (-1, 1)$ est le paramètre de l'approximation. Remarquons que le dénominateur $\alpha(z^{-1})$ et le numérateur $\beta(z^{-1})$ sont définie par (2.27) et (2.28) comme les fonctions rationnelles avec un dénominateur commun $d_c(z^{-1}) = (1 - z_0 z^{-1})^N$. Ce dénominateur s'élimine dans le bloc $Q(z^{-1})$.

L'analyse de l'influence du paramètre z_0 est faite dans [Lan98]. En bref, les valeurs de z_0 vers -1 placent les pôles de $\alpha(z^{-1})$ plus dans le demi-cercle gauche, les valeurs autour de 1 dans le demi-cercle droit et les valeurs autour de 0 placent les pôles dans tout le cercle d'unité.

La condition (2.25) est testée sur une grille fréquentielle de 32 points. La fonction critère correspondante peut être par exemple la valeur maximale de l'ensemble des points évalués. La fonction $\max(\cdot)$ est utilisée pour réunir la condition (2.25) définie pour plusieurs fonctions de sensibilité. Le logiciel OPTREG [Ada99] réalise l'optimisation convexe décrite dans cette section.

Dans [LL99] deux conditions convexes de la δ -stabilité des pôles optimisés sont aussi définies. Avec ses conditions la position des pôles est restreinte sur une région δ placé à l'intérieur du cercle d'unité. Le région est délimité par un cercle avec le centre sur l'axe réel (restriction de l'amortissement et de la vitesse des pôles) ou/et par une droite parallèle avec l'axe imaginaire (restriction de la vitesse des pôles). Les deux conditions développées sont relativement conservatrices et en général elles sont utilisées pour définir une fonction critère de contrainte.

Remarque 2.1 : Un système linéaire a la propriété de δ -stabilité si tous les pôles de ce système sont placé dans un région δ . Le région δ est situé à l'intérieur du région de stabilité (au cercle d'unité pour les systèmes discrets, au demi-plan gauche pour les systèmes continus).

2.4 Notes et références

Références : Le placement de pôles SISO avec calibrage de fonctions de sensibilité a été introduit dans [LK98]. L'amélioration par le calibrage avec des filtres bande étroite est présentée dans [PL03] ou à Annexe B2. Le rapport complet est dans l'Annexe B8. Des exemples pratiques d'application sont donnés dans [LLRB96, PL03, Pro03] et dans Annexe B2 et B3. La présentation intégrale de la méthode se trouve dans [Lan02]. Le guide d'utilisateur pour le logiciel PPMaster de placement de pôles avec calibrage de sensibilités est donné dans l'Annexe B7. La méthode de calibrage par optimisation convexe est introduite dans [Lan98, LL99]. La théorie concernant l'optimisation convexe de régulateurs est dans [BB91, Mac89, Zho98, RM94] et la paramétrisation de "Youla-Kucera" est donné par [BB91, YJB76, Kuc79]. Des exemples pratiques de la synthèse sont donnés par [LL99, LC99, CL03]

Fichiers et logiciels : Le logiciel PPMaster se trouve sur le CD de thèse dans le répertoire : \\Software \PPMaster\. M-fonctions de Matlab pour la commande est l'analyse des systèmes SISO programmées dans le cadre

de ce thèse sont dans le répertoire `\M_functions\` (pour la description voir le fichier `Contents.m`). Toutes les figures de graphes sont enregistrées sous format de Matlab dans le dossier `:\Figures\`. Les figures Fig. 2.2 et Fig. 2.3 correspondent respectivement aux fichiers `filter_specif.fig` et `Fyp_influence.fig`.

References : *The SISO pole placement with sensitivity shaping is introduced in [LK98]. In this thesis developed improvement of the design method by shaping with notch-filters is presented in [PL03] or in Appendix B2. A complete report is in Appendix B8. Some practical examples of application are given in [LLRB96, PL03, Pro03] and in Appendix B2 and B3. A detailed presentation of the design procedure can be found in [Lan02]. A user's guide of the PPMaster software design tool for pole placement with sensitivity function shaping is in Appendix B7. The sensitivity shaping by convex optimization method is introduced in [Lan98, LL99]. A theory concerning the convex optimization of controllers is in [BB91, Mac89, Zho98, RM94] and the "Youla-Kucera" parameterizations is given in [BB91, YJB76, Kuc79]. Some practical examples of the design by convex optimization are given in [LL99, LC99, CL03].*

Files and software : *Complete PPMaster application can be found in the directory `:\Software \PPMaster\`. All programmed m-functions for SISO system control and analysis are in the subdirectory `\M_functions\` (for the m-functions description see `Contents.m`). All presented graphs are saved in Matlab format in the directory `:\Figures\`. The graphs in Fig. 2.2 and Fig. 2.3 correspond respectively to the files `filter_specif.fig` and `Fyp_influence.fig`.*

Chapitre 3

Commande multivariable par calibrage de sensibilités

Dans ce chapitre la notation utilisée pour les systèmes multivariables est introduit. Ensuite, le problème général de la commande multivariable d'un modèle échantillonné du procédé continu MIMO est présenté. Une discussion préalable nous permet d'introduire l'approche du calibrage de sensibilités. Cet approche est une des possibilités pour obtenir la robustesse et les performances désirées du système.

In this chapter the used notation for multivariable systems is introduced. Next, the problem of multivariable robust control for continuous-time plants described by discrete-time models is presented. A preliminary discussion allows us to introduce the sensitivity shaping approach. This approach is one of possibilities for obtaining required system robustness and performance.

3.1 Notation

Deux représentations des systèmes linéaires discrets MIMO invariants dans le temps sont utilisées dans ce travail : 1) matrice de transfert, 2) représentation d'état.

Considérons un vecteur des signaux d'entrée $u(t)$ et un vecteur des signaux de sortie du système $y(t)$ d'un système discret M . Les deux formes de représentation sont :

- *Matrice de transfert* - une matrice de fonctions rationnelles où chaque coefficient de la matrice est une fonction rationnelle discrète en l'opérateur z^{-1} . Le système est décrit par l'équation suivante :

$$y(t) = M(z^{-1})u(t) \quad (3.1)$$

et dans ce cas $M(z^{-1})$ est un opérateur de transfert, ou par :

$$y(z^{-1}) = M(z^{-1})u(z^{-1}) \quad (3.2)$$

dans ce cas $M(z^{-1})$ est une matrice de transfert.

- *Représentation d'état* - le système est décrit par 4 matrices constantes de l'espace d'état A_M, B_M, C_M, D_M :

$$x_M(t+1) = A_M x_M(t) + B_M u(t) \quad (3.3)$$

$$y(t) = C_M x_M(t) + D_M u(t) \quad (3.4)$$

où $x_M(t)$ est le vecteur des états du système.

Le degré $\text{deg}(M)$ d'un système linéaire $M(z^{-1})$ décrit par (3.1), (3.2), ou (3.3) - (3.4) est le nombre des états nécessaires pour décrire ce système par représentation d'état. Ainsi, il indique aussi le nombre des pôles du systèmes.

L'opérateur z^{-1} représente soit l'opérateur fréquentiel

$$z^{-1} = e^{-j\omega T_e} = e^{-j\omega_d} \quad (3.5)$$

soit l'opérateur temporel de retard

$$z^{-1}y(t) = y(t-1) \quad (3.6)$$

La pulsation ω de l'opérateur fréquentiel z^{-1} est définie sur l'intervalle :

$$\omega \in \left\langle 0, \frac{\omega_e}{2} \right\rangle \quad (3.7)$$

avec ω_e la pulsation d'échantillonnage

$$\omega_e = 2\pi \frac{1}{T_e} \quad (3.8)$$

en rad/s. T_e est la période d'échantillonnage en secondes, la variable t est le temps discret $t = kT_e \mid k = 0, 1, 2, \dots$ et la pulsation ω_d est la pulsation discrète normalisée :

$$\omega_d = \omega T_e \mid \omega_d \in \langle 0, \pi \rangle \quad (3.9)$$

La fréquence d'échantillonnage est définie en Hz par $f_e = T_e^{-1}$.

Les systèmes considérés à commander sont des procédés multivariables réels continus approximatés par un modèle échantillonné $G(z^{-1})$ qui représente le procédé uniquement aux instants d'échantillonnage. Le modèle échantillonné peut être décrit par la représentation d'état suivante :

$$\begin{aligned} x_G(t+1) &= Ax_G(t) + Bu_G(t) \\ y_G(t) &= Cx_G(t) + Du_G(t) \end{aligned} \quad (3.10)$$

avec les matrices de représentation d'état $A \in R^{n_x \times n_x}$, $B \in R^{n_x \times n_u}$, $C \in R^{n_y \times n_x}$, $D \in R^{n_y \times n_u}$. $u_G(t)$, $y_G(t)$, et $x_G(t)$ sont les vecteurs des entrées, des sorties, et des états respectivement. Les variables n_u , n_y , n_x représentent les nombres des entrées, des sorties, et des états. La matrice D est égale à zéro dans le cas des systèmes continus échantillonnés et puisque seulement ces systèmes sont considérés dans notre cas, elle est supposé $D = 0$. La matrice (l'opérateur) de transfert correspondante au modèle $G(z^{-1})$ est définie comme suit :

$$G(z^{-1}) = C \left(\frac{1}{z^{-1}} I - A \right)^{-1} B \quad (3.11)$$

avec la sortie du modèle

$$y_G(t) = G(z^{-1})u_G(t) = C \left(\frac{1}{z^{-1}} I - A \right)^{-1} Bu_G(t) \quad (3.12)$$

3.2 Boucle fermée - sensibilités

Le schéma considéré pour la commande est la boucle fermée donnée dans la Fig.3.1. Le bloc K correspond au correcteur assurant les performances de

la boucle fermée comme la stabilité robuste, rejet de perturbation etc. Le pré-compensateur T permet d'assurer les performances désirées en poursuite (tracking) si elles ne sont pas garanties par le correcteur K . Les deux blocs sont décrits par ses représentations d'état avec les matrices A_K , B_K , C_K , D_K , et A_T , B_T , C_T , D_T .

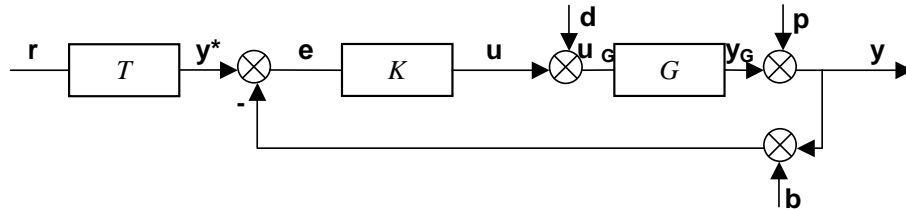


FIG. 3.1 – Configuration considérée de la boucle fermée

Les vecteurs des signaux $r(t)$, $y^*(t)$, $e(t)$, $u(t)$, $y(t)$, $d(t)$, $p(t)$, et $b(t)$ qui apparaissent dans la Fig. 3.1 sont :

- $r(t)$ vecteur de références (de consignes).
- $y^*(t)$ vecteur des sorties désirées
- $e(t) = y^*(t) - y(t)$ vecteur des erreurs entre la sortie désirée et la sortie réelle
- $u(t)$ vecteur de commandes envoyées au procédé par le régulateur
- $y(t) = y_G(t) + p(t)$ vecteur de sorties réelles du système bouclé
- $d(t)$ vecteur de perturbations agissant sur l'entrée du procédé (sur la commande)
- $p(t)$ vecteur de perturbations sur la sortie du procédé
- $b(t)$ vecteur de bruits de mesure

La référence $r(t)$, la perturbation $p(t)$, le bruit de mesure $b(t)$ et la perturbation $d(t)$ sont les signaux extérieurs qui agissent sur le système bouclé. Le signal $r(t)$ est généralement la consigne générée qui dirige la sortie $y(t)$. Les signaux $p(t)$, $b(t)$, et $d(t)$ sont en général les signaux non-désirés dont l'influence sur le système devrait être limitée (différentes méthodes permettent éliminer les perturbations : commande H_∞ , "loop-shaping", calibrage de sensibilités, etc. [GL95, MZ89, Lan02]). Les perturbations sont souvent utilisées

juste pour exprimer une certaine robustesse du système par exemple vis-a-vis des changements de paramètres du procédé sans existence d'une vraie perturbation agissante sur la boucle. La dernière idée est typiquement utilisée dans la commande H_∞ ou le calibrage de sensibilités et elle va être explorée dans ce travail.

Les influences des signaux extérieurs sur le système peuvent être interprétées par les fonctions de sensibilité ou, dans le cas MIMO, à l'aide des matrices de sensibilité $S_{ij}(z^{-1})$ (les indices ij représentent le paire des vecteurs de signaux yp , up , yb , yd , ud). Les sensibilités S_{ij} , qui sont les matrices de transfert de la boucle fermée, jouent le rôle crucial dans l'analyse de robustesse et des performances d'un système en boucle fermée (voir [Lan02, Zho98, Mac89, GL95] etc.). Les sensibilités suivantes sont typiquement considérées :

- $S_{yp}(z^{-1})$: Matrice de sensibilité perturbation de sortie - sortie

$$S_{yp}(z^{-1}) = \frac{y(z^{-1})}{p(z^{-1})} = (I + G(z^{-1})K(z^{-1}))^{-1} \quad (3.13)$$

- $S_{up}(z^{-1})$: Matrice de sensibilité perturbation de sortie - commande

$$S_{up}(z^{-1}) = \frac{u(z^{-1})}{p(z^{-1})} = -K(z^{-1})(I + G(z^{-1})K(z^{-1}))^{-1} \quad (3.14)$$

- S_{yb} : Matrice de sensibilité (complémentaire) bruit - sortie

$$S_{yb}(z^{-1}) = \frac{y(z^{-1})}{b(z^{-1})} = -G(z^{-1})K(z^{-1})(I + G(z^{-1})K(z^{-1}))^{-1} \quad (3.15)$$

- S_{yd} : Matrice de sensibilité perturbation de commande - sortie

$$S_{yd}(z^{-1}) = \frac{y(z^{-1})}{d(z^{-1})} = G(z^{-1})(I + K(z^{-1})G(z^{-1}))^{-1} \quad (3.16)$$

- S_{ud} : Matrice de sensibilité perturbation de commande - commande

$$S_{ud}(z^{-1}) = \frac{u(z^{-1})}{d(z^{-1})} = -K(z^{-1})G(z^{-1})(I + K(z^{-1})G(z^{-1}))^{-1} \quad (3.17)$$

Les représentations d'état de toutes ces sensibilités avec les matrices de représentation d'état A_{ij} , B_{ij} , C_{ij} , et D_{ij} sont données dans l'Annexe A.

Toutes les autres sensibilités possibles sont équivalentes aux sensibilités (3.13)-(3.17). Il s'agit des sensibilités suivantes :

$$S_{ey^*}(z^{-1}) = S_{yp}(z^{-1}) \quad (3.18)$$

$$S_{uy^*}(z^{-1}) = S_{up}(z^{-1}) \quad (3.19)$$

$$S_{yy^*}(z^{-1}) = -S_{yb}(z^{-1}) \quad (3.20)$$

La sensibilité S_{yb} est parfois nommée la sensibilité complémentaire, car elle possède de la propriété suivante :

$$S_{yp}(z^{-1}) + S_{yb}(z^{-1}) = I \quad (3.21)$$

3.3 Calibrage de sensibilités pour la robustesse et la performance

Dans la synthèse d'une commande, les spécifications de performance et de robustesse sont définies dans un première temps. L'approche fréquentielle de la synthèse d'une commande robuste transforme toutes les spécifications en contraintes imposées sur les réponses fréquentielles des sensibilités. Ceci est le cas tant pour la commande H_∞ (méthode de synthèse la plus utilisée) que pour la méthode de calibrage de sensibilités, qui est étudiée dans ce travail. Les spécifications de performance et en conséquence aussi les contraintes sur les sensibilités sont typiquement en conflit, c'est-à-dire toutes les performances ne peuvent pas en même temps atteindre la meilleure valeur. Si les spécifications sont trop exigeantes, la synthèse de commande est une recherche d'un compromis pour satisfaire au mieux les spécifications de performances et c'est le cas dans beaucoup des problèmes pratiques de la commande.

La commande H_∞ contraint les sensibilités indirectement. Elle minimise une certaine norme de la réponse fréquentielle d'une sensibilité pondérée par

un filtre où le filtre de pondération devrait correspondre aux spécifications. Par contre la méthode de calibrage de sensibilités utilise une approche plus logique. Elle définit un gabarit pour chaque sensibilité considérée et ensuite les réponses fréquentielles de sensibilités sont ajuster afin qu'elles entrent dans les gabarits.

3.3.1 Condition de calibrage

Pour analyser une sensibilité $S_{ij}(z^{-1})$, l'évolution de la plus grande valeur singulière $\bar{\sigma}(S_{ij}(e^{-j\omega_d}))$ est considérée (pour la définition voir l'Annexe A). La valeur maximale de cette valeur singulière donne la norme H_∞ de la matrice de sensibilité :

$$\|S_{ij}(e^{-j\omega_d})\|_\infty \doteq \sup_{\omega_d} \bar{\sigma}(S_{ij}(e^{-j\omega_d})) \quad (3.22)$$

Dans la commande H_∞ [Zho98, Mac89, GL95], une condition typique sur $S_{ij}(z^{-1})$, pour garantir des performances ou la robustesse, a la forme suivante :

$$\|W_j(e^{-j\omega_d})S_{ij}(e^{-j\omega_d})W_i(e^{-j\omega_d})\|_\infty \leq \gamma \quad (3.23)$$

où γ est une constante suffisamment petite afin d'assurer les performances désirées. Les matrices W_i , W_j sont les matrices des filtres de pondération utilisées pour accentuer certaines zones des fréquences. Remplaçant la norme H_∞ par l'expression équivalente de la valeur singulière (voir l'Annexe A), la condition (3.23) devient :

$$\bar{\sigma}(W_j(e^{-j\omega_d})S_{ij}(e^{-j\omega_d})W_i(e^{-j\omega_d})) \leq \gamma \mid \forall \omega_d \quad (3.24)$$

Les signaux extérieurs d'entrée sont pondérés par la matrice W_j . Afin de simplifier l'écriture considérons la matrice $W_j(e^{-j\omega_d}) = I$ et cela nous donne :

$$\bar{\sigma}(S_{ij}(e^{-j\omega_d})W_i(e^{-j\omega_d})) \leq \gamma \mid \forall \omega_d \quad (3.25)$$

Notons que nous n'avons pas perdu la généralité de la condition. En principe, le filtre W_j peut être ajouter a n'importe quel moment car il ne dépende pas du régulateur K . Le filtre peut être considéré comme une pondération

de la sensibilité S_{ij} . Pour les opérations appliquées sur la condition dans la suite (paramétrisation, etc.) le filtre ne modifie pas la propriété de convexité car il s'agit d'une simple multiplication de sensibilité avec une matrice de transfert connue.

Sachant qu'une des propriétés de la valeur singulière c'est :

$$\bar{\sigma}(AB) \leq \bar{\sigma}(A)\bar{\sigma}(B) \mid \forall A, B \in C \quad (3.26)$$

une condition qui vérifie (3.25), mais qui est plus conservatrice a la forme suivante :

$$\bar{\sigma}(S_{ij}(e^{-j\omega_d}))\bar{\sigma}(W_i(e^{-j\omega_d})) \leq \gamma \mid \forall \omega_d \quad (3.27)$$

A partir de (3.27) nous avons la condition :

$$\bar{\sigma}(S_{ij}(e^{-j\omega_d})) \leq \gamma \bar{\sigma}(W_i(e^{-j\omega_d}))^{-1} \mid \forall \omega_d \quad (3.28)$$

qui peut être écrite comme suit :

$$\bar{\sigma}(S_{ij}(e^{-j\omega_d})) \leq F_{ij}(\omega_d) \mid \forall \omega_d \quad (3.29)$$

où $F_{ij}(\omega_d) > 0 \mid \forall \omega_d$ est un gabarit supérieur (bornes) imposé sur la valeur singulière de sensibilité. La condition (3.29) est non-convexe par rapport à la matrice du régulateur $K(z^{-1})$. Pour la rendre convexe, il faut une paramétrisation appropriée du régulateur $K(z^{-1})$.

Concernant le passage de (3.24) à (3.29), la "taille" du conservatisme apporté dépend de la forme du filtre W_i et de la sensibilité S_{ij} . Néanmoins dans ce cas, le passage n'est pas vraiment utilisé en pratique, il sert juste pour montrer la similitude entre la commande H_∞ et le calibrage de sensibilités et aussi pour présenter une possibilité comment le calibrage de sensibilités peut être vu.

3.3.2 Spécifications de performance par calibrage

La robustesse et les performances désirées de la boucle fermée peuvent être garanties par le calibrage de sensibilités. Une liste de possibles performances et robustesse désirées avec les règles de calibrage est donnée ci-dessous :

- *Stabilité robuste de BF* - Si la stabilité robuste est mesurée par les marges de stabilité, la marge de phase est donnée par le maximum de $\bar{\sigma}(S_{yb}(z^{-1}))$ [GL95]. La marge du module et la marge du retard peut être assurée, dans le cas SISO, par un gabarit qui limite $|S_{yp}(z^{-1})| < 6\text{dB}$ en basses fréquences et $< 3\text{dB}$ en hautes fréquences [LK98, Lan02]. Le même gabarit sera considéré ici dans le domaine MIMO pour l'évolution de la valeur singulière $\|S_{yp}(z^{-1})\|_{\infty}$ ¹. Il est supposé que si la plus grande valeur singulière de matrice de sensibilité est sous le gabarit, automatiquement le module de chaque fonction de la matrice de sensibilité est sous ce gabarit et que ceci est suffisant pour garantir la robustesse correspondant aux marges définies. Concernant la marge de retard multivariable, elle a été traitée par plusieurs auteurs, voir par exemple [KNR98]. Le gabarit et un exemple de la réponse de $S_{yp}(z^{-1})$ est montré dans la Fig.3.2.

D'autre part la stabilité robuste peut être évaluée pour des modèles d'incertitude non-structuré (pour des cas où l'incertitude est connue, l'incertitude structurée est utilisée). Les incertitudes additives et multiplicatives sont généralement considérées. Afin de rendre le système insensible à l'incertitude additive, il faut restreindre $\bar{\sigma}(S_{up}(z^{-1}))$. Dans le cas de l'incertitude multiplicative, la valeur de $\bar{\sigma}(S_{yb}(z^{-1}))$ doit être bornée. Pour les détails sur les conditions de robustesse voir [Mac89, GL95, Zho98], Tab.3.1 et l'Annexe A.

- *Poursuite (tracking) de BF* - Les propriétés de la poursuite comportent le temps de montée, et le dépassement de réponse en échelon des canaux directs (fonctions de transfert entre i^{eme} référence et i^{eme} sortie) de la boucle fermée et le découplage (interaction). Pour la définition voir le Chapitre 6. Les propriétés de poursuite sont approximativement caractérisées par la réponse $|S_{yb}(z^{-1})|$ (un exemple de réponse est montré dans la Fig.3.3). Le dépassement maximal correspond au

¹Dans beaucoup des exemples de la commande robuste MIMO, la valeur de $\|S_{yp}(z^{-1})\|_{\infty}$ est $\leq 4\text{dB}$, voir exemples à [Mac89, GL95]

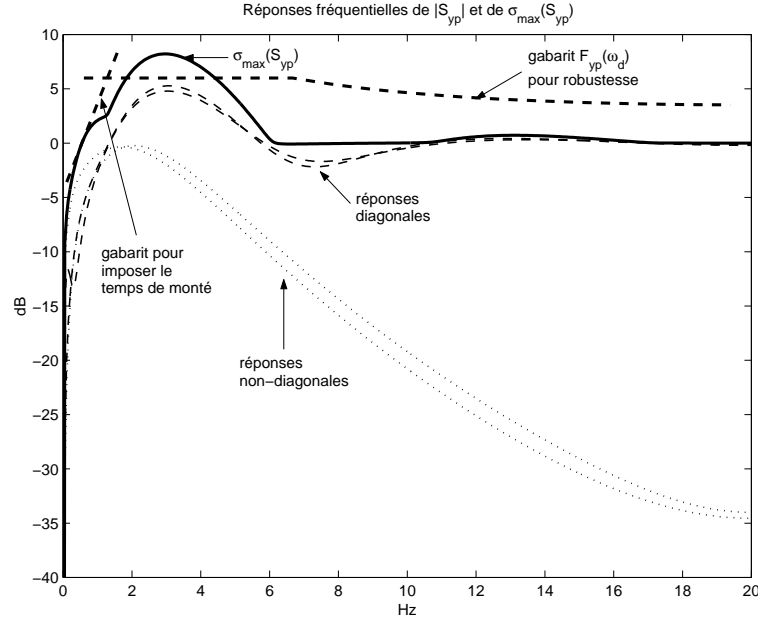


FIG. 3.2 – Performance et robustesse caractérisées par un gabarit sur $S_{yp}(e^{-j\omega d})$

maximum en basse fréquences de la réponse fréquentielle correspondante - un dépassement temporel de 0%, 10%, 20% donnent pour la réponse fréquentielle diagonale un maximum autour de 0dB, 0.9dB, 1.6dB. Le temps de montée est déterminé, comme dans le cas SISO, par la fréquence f_{0dB} où la réponse descend à 0dB comme c'est montré dans la Fig. 3.3 (réponses diagonales $|S_{yb11}|$ et $|S_{yb22}|$). Ces propriétés de poursuite sont approximativement observable à partir de $\bar{\sigma}(S_{yb}(z^{-1}))$ (sous la condition que les réponses non-diagonale ont les modules sensiblement plus petits par rapport aux modules des réponses diagonales). Le découplage est donné par les maximums de réponses non-diagonale de $|S_{yb}(z^{-1})|$ (voir Fig.3.3). Avec un maximum de -10dB, -20dB l'interaction entre les canaux est 0.3 et 0.1.

Le dépassement maximal et l'interaction peuvent être garantis par des gabarits (bornes) supérieurs imposés à $|S_{yb}(z^{-1})|$ ou à $\bar{\sigma}(S_{yb}(z^{-1}))$. Ceci n'est pas possible dans le cas du temps de montée déterminé par la

bande passante de BF. Pour garantir un temps de montée désiré, un gabarit en forme d'une rampe doit être imposé à $\bar{\sigma}(S_{yp}(z^{-1}))$ en basses fréquences (voir Fig.3.2). L'effet négatif de cette accélération est une augmentation du maximum de $\bar{\sigma}(S_{yp}(z^{-1}))$ (stabilité robuste) et de $\bar{\sigma}(S_{yb}(z^{-1}))$ (stabilité robuste et dépassement maximal).

Remarque 3.1 : Le temps de montée peut être évalué à partir d'une approximation par un système du 2^e ordre avec une fréquence naturelle f_{0dB} et un amortissement $\zeta = 0.9$ (cas assez typique) [Lan02] : $t_m = \frac{3.5}{2\pi f_{0dB}}$.

- *Atténuation du bruit en hautes fréquences (roll-off)* - Généralement, une pente avec -20dB (-40dB) par décade (dans l'échelle logarithmique) est demandée sur toutes les réponses de $|S_{yb}(z^{-1})|$, voir Fig.3.3. Pour assurer l'atténuation, il est possible d'imposer cette pente à la plus grand valeur singulière $\bar{\sigma}(S_{yb}(z^{-1}))$. La pente est utilisé surtout pour les systèmes continu où les réponses sont dans l'échelle logarithmique. Pour les systèmes numériques, la pente doit être ré-calculée dans l'échelle linéaire.
- *Gain de régulateur (solicitation d'actionneur)* - Il peut être caractérisé par la réponse $S_{up}(e^{-j\omega d})$. Si une réduction du gain de régulateur est exigée dans une certaine zone de fréquences, il est nécessaire d'atténuer $\bar{\sigma}(S_{up}(e^{-j\omega d}))$ dans cette région. Les cas plus typiques sont : L'atténuation du gain de régulateur en haute fréquences pour réduire les sollicitations des actionneurs et améliorer la marge de retard, l'atténuation dans une certaine zone de fréquences où l'incertitude est présente dans le modèle, la limitation du gain du régulateur aux fréquences où le gain du modèle est faible². Le gabarits pour la sensibilité S_{up} est généralement ajusté au cours de la synthèse selon le comportement des régulateurs obtenus.

²Pour les fréquences du rejet parfait de perturbation $p(t)$ [Lan02]

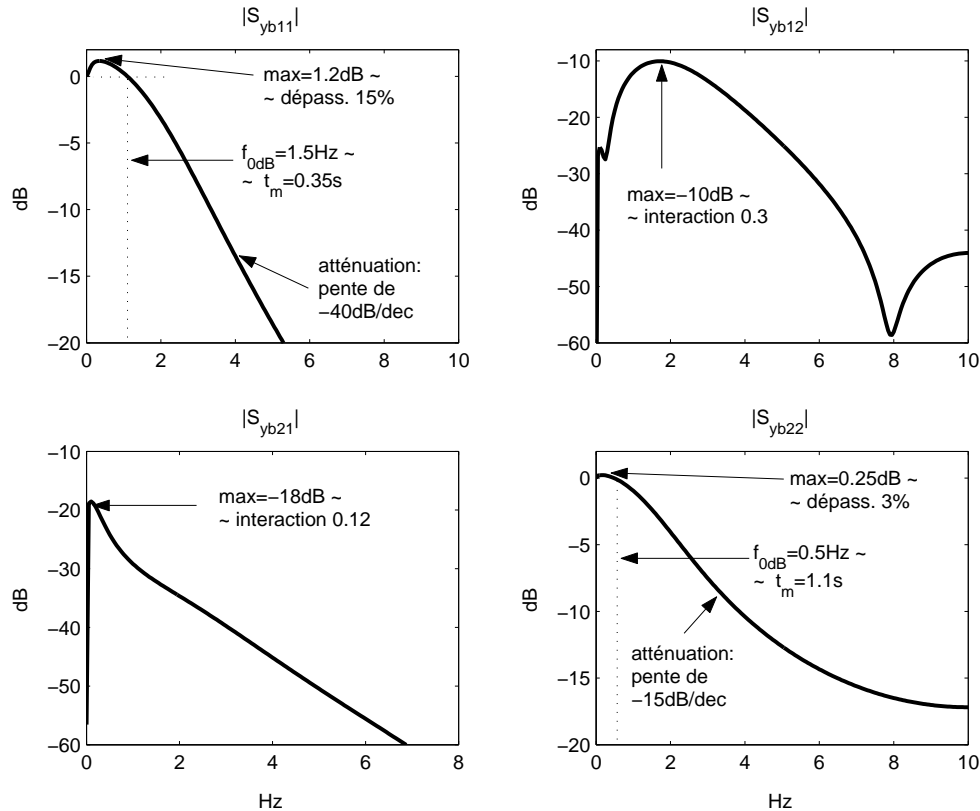


FIG. 3.3 – Propriétés en poursuite d'un système MIMO

Le Tab.3.1 représente une guide qui permet de choisir la sensibilité à traiter selon des incertitudes ou des perturbations rencontrées dans le système, pour obtenir la performance et la robustesse désirées (voir [Zho98]).

3.4 Notes et références

Références : La théorie de systèmes multivariables linéaires est traitée dans nombreux ouvrages comme par exemple [Gan66, Kai80, Wol74, Zho98, GL95]. Pour la synthèse de la commande linéaire multivariable un grand nombre d'approches a été développé comme [GL95, Zho98, Mac89] (commande robuste H_∞ ou LQG), [BB91] (optimisation convexe), ou [Mag02] (commande modale). En se qui concerne le calibrage de sensibilités multiva-

riables, des règles vagues liés au choix des filtres de pondération se trouvent surtout dans les ouvrages et les articles sur la commande H_∞ orientés vers la pratique comme [Mac89, GL95, MZ89] etc. L'article de l'auteur sur le calibrage de sensibilités est dans l'Annexe B4.

Fichiers et logiciels : M-fonctions de Matlab pour la commande est l'analyse des systèmes MIMO programmées dans le cadre de ce thèse sont dans le répertoire `\M_functions\` (pour la description voir le fichier `Contents.m`). Toutes les figures de graphes sont enregistrées sous format de Matlab dans le dossier `:\Figures\`. Les figures Fig. 3.2 et Fig. 3.3 correspondent respectivement aux fichiers `syp_robustness.fig` et `syb_tracking.fig`.

References : *The theory of linear multivariable systems is treated in a lot of books for example in [Gan66, Kai80, Wol74, Zho98, GL95]. For high-performance linear multivariable control design a number of approaches is developed : [GL95, Zho98, Mac89] (robust H_∞ control or LQG), [BB91] (convex optimization), or [Mag02] (modal control). Concerning the multivariable sensitivity shaping, some vague rules related to choice of weighting filters can be found mainly in the books and papers on H_∞ control oriented to practice [Mac89, GL95, MZ89] etc. The paper of the author on multivariable sensitivity shaping is in Appendix B.*

Files and software : *All programmed m-functions for MIMO system control and analysis are in the subdirectory `\M_functions\` (for the m-functions description see `Contents.m`). All presented graphs are saved in Matlab format in the directory `:\Figures\`. The graphs in Fig. 3.2 and Fig. 3.3 correspond respectively to the files `syp_robustness.fig` and `syb_tracking.fig`.*

condition de performance et de robustesse $\forall \omega_d$	Incertitudes / perturbations dans le système
$\bar{\sigma}(S_{yp}(e^{-j\omega_d})) < F_{yp}(\omega_d)$	erreurs de LF des paramètres, perturbation $p(t)$ sur la sortie pôles instables incertaines incert. multiplicative inversée de sortie
$\bar{\sigma}(S_{yb}(e^{-j\omega_d})) < F_{yb}(\omega_d)$	erreurs de mesures $b(t)$ (bruit), dynamique négligée de HF zéros instables incertaines incertitude multiplicative de sortie
$\bar{\sigma}(S_{up}(e^{-j\omega_d})) < F_{up}(\omega_d)$	incertitude additive dynamique de HF négligée zéros/pôles incertaines bruit de mesure $b(t)$
$\bar{\sigma}(S_{ud}(e^{-j\omega_d})) < F_{ud}(\omega_d)$	erreurs d'entrée (d'actionneur) dynamique négligée de HF perturbation $d(t)$ sur l'entrée incert. multiplicative d'entrée
$\bar{\sigma}(S_{yd}(e^{-j\omega_d})) < F_{yd}(\omega_d)$	erreurs de LF des paramètres incert. multiplicative inversée d'entrée

TAB. 3.1 – Conditions de performances et de robustesse (LF=basses fréquences, HF= hautes fréquences).

Chapitre 4

Placement de pôles MIMO avec calibrage de sensibilités

La procédure de synthèse de régulateur multivariable par placement de pôles avec calibrage de sensibilités est développée dans ce chapitre. La procédure étend une approche simple et efficace de synthèse de régulateur SISO introduit par [LK98, Lan02], améliorée en cadre de ce travail (voir [PL03] ou l'Annexe B2) et appliquée avec succès sur les systèmes réels [LLRB96, Pro03]. Une brève introduction de cette méthodologie est donnée dans le Chapitre 2. Dans le domaine multivariable, la procédure est moins efficace, car il n'existe pas une solution unique du placement de pôles pour une configuration de pôles de BF. Néanmoins, nous nous intéressons à cette approche pour les raisons suivantes :

- Elle permet de calculer un régulateur multivariable du type observateur avec le retour d'état qui peut servir comme le régulateur central pour la synthèse de régulateurs par l'optimisation convexe (voir le Chapitre 5).
- La procédure de synthèse est cohérente avec la méthode d'optimisation convexe (les spécifications de synthèse sont les mêmes, les moyens d'ajustement du régulateur sont similaires). Le régulateur central obtenu par placement de pôles peut avoir des performances qui sont proches

des performances désirées du régulateur optimisé.

- La procédure conserve une compréhension du concept élémentaire du régulateur multivariable.

A procedure of multivariable controller design by pole placement with sensitivity shaping is developed in this chapter. The procedure extends a simple and efficient SISO controller design approach introduced in [LK98, Lan02], improved in this work (see [PL03] or Appendix B2) and successfully applied on real systems [LLRB96, Pro03]. A short introduction of this methodology is given in Chapter 2. In the multivariable domain, the approach is less efficient than in the SISO domain, since there is not a unique solution of pole assignment for one closed loop pole configuration. Nevertheless, we are interested in this approach for the following reasons :

- *It allows to compute a multivariable observer with state feedback controller, which can serve as a central controller for controller design by convex optimization.*
- *The procedure is coherent with the convex optimization method (design specifications are the same, means of controller adjustment are similar). Consequently, an obtained controller to be optimized can have its performance close to a desired performance of convex optimization controller.*
- *The procedure conserve comprehension of elementary multivariable controller concept.*

4.1 Structure de régulateur

Supposons que le régulateur final $K(z^{-1})$ (voir Fig.3.1) est une cascade d'une matrice pré-spécifiée $H_K(z^{-1})$ (une matrice de transfert stable quelconque) et d'un correcteur central $K_0(z^{-1})$. La matrice de transfert du régulateur $K(z^{-1})$ a la forme suivante :

$$K(z^{-1}) = H_K(z^{-1})K_0(z^{-1}) \quad (4.1)$$

Remarque 4.1 : La matrice fixe $H_K(z^{-1})$ contient typiquement un modèle d'une perturbation à éliminer (principe du modèle interne [Lan02]). Par exemple, pour éviter l'erreur statique, elle contient un intégrateur.

En plus il est supposé que le correcteur central $K_0(z^{-1})$ a la structure utilisée par la commande LQG ou par le placement de pôles MIMO [Kai80, Wol74] (voir Fig. 4.1). Le régulateur est constitué de l'observateur (nommé aussi l'observateur de Luenberger) avec la matrice d'observateur L et du retour d'état avec la matrice de retour d'état F . La complexité du régulateur final, donnée par le nombre des états, est dans ce cas : $\deg(K(z^{-1})) = \deg(G(z^{-1})) + 2\deg(H_K(z^{-1}))$.

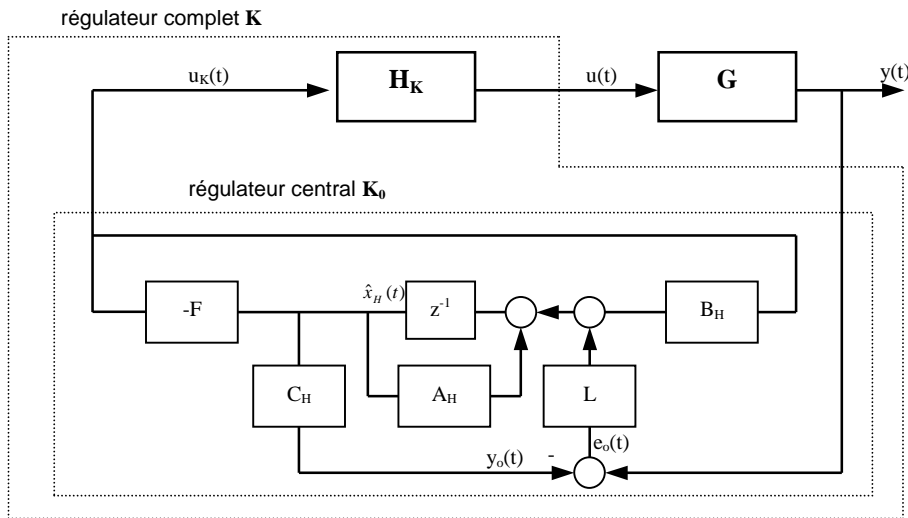


FIG. 4.1 – Structure de régulateur

Les vecteurs des signaux $y_o(t)$ et $e_o(t)$ de la Fig. 4.1 sont respectivement le vecteur des sorties estimées et le vecteur des erreurs d'estimation. Les matrices constantes A_H , B_H , C_H , et D_H sont les matrices de représentation d'état (RE) de la connexion cascade $H(z^{-1}) = G(z^{-1})H_K(z^{-1})$ (modèle et la matrice pré-spécifiée) avec le vecteur des états $x_H(t) = [x_{H_K}^T(t), x_p^T(t)]^T$. Les RE suivantes définies $H(z^{-1})$ et $H_K(z^{-1})$:

$$\begin{aligned} x_H(t+1) &= A_H x_H(t) + B_H u_K(t) \\ y(t) &= C_H x_H(t) \end{aligned}$$

$$\begin{aligned}
x_{H_K}(t+1) &= A_{H_K}x_{H_K}(t) + B_{H_K}u_K(t) \\
u(t) &= C_{H_K}x_{H_K}(t) + D_{H_K}u_K(t)
\end{aligned} \tag{4.2}$$

où les matrices de la connexion cascade ont les formes suivantes :

$$\begin{aligned}
A_H &= \begin{bmatrix} A_{H_K} & 0 \\ B_p C_{H_K} & A_p \end{bmatrix}, B_H = \begin{bmatrix} B_{H_K} \\ B_p D_{H_K} \end{bmatrix} \\
C_H &= \begin{bmatrix} 0 & C_p \end{bmatrix}, D_H = 0
\end{aligned} \tag{4.3}$$

Le régulateur $K_0(z^{-1})$ calculé par le placement de pôles a la représentation d'état suivante :

$$\begin{aligned}
\hat{x}_H(t+1) &= (A_H - B_H F - LC_H)\hat{x}_H(t) + Ly(t) \\
u_K(t) &= -F\hat{x}_H(t)
\end{aligned} \tag{4.4}$$

où $\hat{x}_H(t)$ est l'estimation du vecteur des états $x_H(t)$. La représentation d'état du régulateur complet (4.1) est de la forme :

$$\begin{aligned}
x_K(t+1) &= A_K x_K(t) + B_K y(t) \\
u(t) &= C_K x_K(t)
\end{aligned} \tag{4.5}$$

avec le vecteur d'état $x_K(t) = [\hat{x}_H^T(t), x_{H_K}^T(t)]^T$ et avec les matrices

$$\begin{aligned}
A_K &= \begin{bmatrix} A_H - B_H F - LC_H & 0 \\ -B_{H_K} F & A_{H_K} \end{bmatrix}, B_K = \begin{bmatrix} L \\ 0 \end{bmatrix} \\
C_K &= \begin{bmatrix} -D_{H_K} F & C_{H_K} \end{bmatrix}
\end{aligned} \tag{4.6}$$

Les matrices constantes L et F dans (4.4), (4.5), (4.6) déterminent le placement des pôles désirés de la boucle fermée. Nous pouvons distinguer deux types des pôles :

- *pôles d'observateur* - déterminés par la matrice d'observateur L . Ces pôles correspondents aux valeurs propres de la matrice $(A_H - LC_H)$.
- *pôles de retour d'état* - déterminés par la matrice de retour d'état F . Ces pôles correspondents aux valeurs propres de la matrice $(A_H - B_H F)$.

Le nombre des pôles à placer de chaque type est égal à

$$n_o = n_{re} = \deg(P(z^{-1})) + \deg(H_K(z^{-1})) \quad (4.7)$$

(somme des états du modèle et des états de la matrice pré-spécifiée) où n_o , n_{re} correspondent respectivement au nombre des pôles d'observateur et au nombre des pôles de retour d'état. La complexité du régulateur complet est $n_K = \deg(P(z^{-1})) + 2\deg(H_K(z^{-1}))$.

Deux méthodes essentielles existent pour le calcul des matrices L et F :

- *Placement de pôles* (eigenstructure assignment) dans l'espace d'état [Kai80, Wol74]. Dans le cas multivariable, les pôles et aussi les vecteurs propres de boucle fermée doivent être placés.
- *LQG* (linear-quadratic-gaussian) techniques présentées dans [Mac89] etc. Il s'agit du filtre de Kalman comme l'observateur avec le retour d'état optimal.

Ici, la méthode de placement de pôles est considérée mais il serait intéressante d'explorer aussi la deuxième possibilité.

4.2 Placement de pôles multivariable

Le problème classique de placement de pôles défini dans l'espace d'état est posé comme suit :

Disposant d'une paire de matrices (A_H, B_H) avec des dimensions appropriées et d'un ensemble des valeurs propres $\lambda_i, i = 1, 2, \dots, n_x$, trouver une matrice F telle que la matrice $(A_H - B_H F)$ a ces valeurs propres placées à λ_i . Pour la matrice d'observateur L , le problème est dual et il peut être défini de la même façon en remplaçant respectivement A_H, B_H , et F par A_H^T, C_H^T , et L^T .

Pour les systèmes multivariables, le problème de placement de pôles comme nous l'avons défini n'implique pas une solution unique pour la matrice F (L^T) [Kai80]. La solution unique est obtenue si et seulement si tous les vecteurs propres (eigenvectors) sont spécifiés. En principe tous les vecteurs propres

(left and right eigenvectors) de la boucle fermée peuvent être placés librement, mais ils doivent appartenir au sous-espace déterminé par les matrices d'état du modèle et par les pôles fixés de la boucle fermée. Nous sommes ainsi limités dans le choix de ces vecteurs. Actuellement, deux approches principales existent pour déterminer ces vecteurs propres : recherche des vecteurs propres par une optimisation, et choix explicite des ces vecteurs.

Afin de simplifier et automatiser la méthodologie de synthèse de régulateur (notre objectif est de développer une méthodologie convenable pour la pratique d'ingénierie), la méthode de choix explicite des vecteurs propres ne sera pas considérée dans ce travail. Néanmoins, elle reste une option intéressante de placement de pôles multivariables.

Les deux approches de placement de pôles multivariable ont les propriétés suivantes :

- *Recherche par optimisation* - un critère est défini permettant d'obtenir un placement robuste des pôles de BF ¹, une norme minimale de la matrice $F(L)$, etc. Ensuite, les vecteurs propres sont cherchés par une optimisation appropriée. Trois algorithmes efficaces basés sur cette approche [KND85, TY96, Var00] sont considérés dans ce travail :
- Méthode de Kautsky [KND85] - Cet algorithme donne des solutions qui peuvent sérieusement varier avec des modifications négligeables (d'ordre 10^{-4}) d'emplacement désiré des pôles de la boucle fermée (la méthode est mal-conditionnée). Afin d'obtenir une solution satisfaisante, parfois il est nécessaire d'itérer autour des positions des pôles de BF (il faut connaître approximativement les positions correctes). L'algorithme est très rapide, le temps de calcul est d'une fraction de seconde, et ainsi il est convenable pour les itérations nécessaires. Nous avons utilisé sa réalisation programmée sous "Matlab" pour "Control Toolbox" - la m-fonction "place.m"
- Méthode de Tits [TY96] - L'algorithme de Tits est une version amé-

¹Placement robuste signifie qu'un petit changement de la position des pôles du modèle ne produit qu'un petit déplacement des pôles de BF.

liorée de l'algorithme de Kautsky. Il donne souvent des solutions proches aux meilleures solutions obtenues par la méthode de [KND85]. En plus, l'algorithme est mieux conditionné et ainsi des itérations (essentiels dans la méthode de [KND85]) ne sont pas nécessaires. Néanmoins, il est plus lent et dans certains cas la solution obtenue n'est pas la meilleure (l'algorithme de Kautsky peut donner des meilleurs résultats). La meilleure approche est de commencer avec la méthode de Tits pour trouver un emplacement initial des pôles de BF et ensuite de itérer autour de ces positions utilisant la méthode de Kautsky qui permet d'améliorer les résultats. Pour appliquer l'algorithme de Tits nous avons utilisé la m-fonction "robpole.m" programmée sous Matlab par A.L.Tits.

- Méthode de Varga [Var00] - Cet algorithme est basé sur la solution de l'équation de Sylvester. Comme l'algorithme de Tits, il est mieux conditionné que l'algorithme de Kautsky, mais il est plus lent (le calcul prend quelques secondes). Les résultats obtenus sont souvent de même qualité ou légèrement détériorés par rapport aux résultats donnés par l'algorithme de Tits. La m-fonction appliquée "sylvplace.m" est développée par A.Varga.

Il n'est pas possible d'établir quel est le meilleur approche, car leur applicabilité dépend du problème à traiter.

- *Choix explicite des vecteurs propres* - Les vecteurs propres de la boucle fermée qui correspondent aux pôles désirés de la boucle fermée sont choisis (tous ou une partie) explicitement "à la main". La commande modale [Mag02] ou les méthodes présentées dans [SSj94] sont des exemples représentatifs de tel approche. Cet approche est visiblement plus complexe du point de vue de l'utilisateur - les vecteurs propres à fixer sont des paramètres de synthèse en plus.

4.3 Placement de pôles avec calibrage de sensibilités

Utilisant les approches [KND85, TY96, Var00] de placement de pôles multivariable, le régulateur (4.4) dépend seulement du choix approprié des pôles désirés de la boucle fermée. Comme dans le cas de la méthodologie SISO de placement de pôles avec calibrage de sensibilités [LK98], nous avons développé l'ensemble des règles, afin de placer correctement (selon des spécifications de synthèse) les pôles de la boucle fermée. Les règles suivantes sont proposées :

- *Robustesse* - Généralement, la meilleure robustesse est obtenue si les pôles du modèle (3.10) (si stables est bien atténués) et ses vecteurs propres sont fixés comme les pôles (vecteurs propres) désirés de la boucle fermé [Mag02]. Les vecteurs propres sont placés par l'algorithme d'optimisation dans notre cas, ainsi il n'est pas sûr d'obtenir les vecteurs propres du modèle dans la boucle fermée. Néanmoins, si c'est le cas, les pôles du modèle du procédé apparaissent comme les zéros de transmission du régulateur² (pour la définition de zéros de transmission voir l'Annexe A). Dans la commande H_∞ sans pondération sur S_{yd} , le régulateur calculé a justement cette structure (voir par exemple dans [Mac89] ou l'exemple de la commande H_∞ dans Matlab). Les règles qui mènent à la commande robuste peuvent être résumés comme suit :
 - fixer tous les pôles stables du modèle comme les pôles désirés de l'**observateur**. Si certains pôles sont peu atténués, modifier l'atténuation à une valeur acceptable (entre 0.7 et 1). Cela est essentiellement le cas des pôles en bases fréquences. Dans les hautes fréquences les pôles peuvent rester peu atténués, surtout si ils sont bien au delà de la bande-passante souhaitée de la boucle fermée.

²Il s'agit de la commande à modèle interne qui suppose de conserver les pôles stables du modèle dans la boucle fermée. Dans le cas de la commande à modèle interne SISO par placement de pôles [Lan02], les pôles du modèle vont apparaître au numérateur (comme zéros) du régulateur

- placer tous les pôles instables du modèle dans les pôles désirés de **l'observateur** avec la même fréquence et avec l'amortissement positif dont la valeur correspond à la valeur absolue de l'amortissement de pôles instables (sous condition que l'amortissement est suffisant). Ce choix de pôles stables est la meilleure approximation des pôles instables dans le domaine fréquentiel. En principe il s'agit d'une réflexion des pôles instables vis à vis du cercle d'unité [Lan02]³. Cette approximation apporte meilleur comportement du régulateur et de la boucle fermée : le régulateur a le gain plus faible, les modules de sensibilités sont moins élevés. L'autre terme pour ce choix de pôles est le placement à énergie minimale (minimum energy assignment) [Mag02].

Exemple : Un modèle échantillonné (période d'échantillonnage $T_e = 1s$) contient une paire des pôles instables : $1 \pm j0.5$. La paire a un équivalent continu avec la fréquence naturelle $f_n = 0.0759Hz$ et l'amortissement $\zeta = -0.234$. Pour obtenir le placement à énergie minimale, il faut placer une paire des pôles complexes dans les pôles désirés d'observateur avec la fréquence $f_n = 0.0759Hz$ et l'amortissement $\zeta = 0.234$. Mais parce qu'il s'agit des pôles lents un meilleur choix d'amortissement pourrait être entre 0.7 et 1.

- placer les zéros de transmission basses fréquences (si stables est bien amorties) du modèle comme les pôles désirés de **l'observateur**, si c'est nécessaire et si il reste des pôles non-spécifiés⁴. Ces zéros apparaissent en générale comme les pôles du régulateur et c'est une configurations similaire au cas SISO où ces pôles éliminent complètement l'influence des zéros du modèle. En multivariable l'élimination totale n'est pas possible.

³Pour obtenir les pôles stables correspondants il est aussi possible de calculer les pôles du polynôme réciproque (l'ordre inversé des coefficients) obtenu à partir du polynôme définie par la paire des pôles instables [Lan02].

⁴Le nombre des pôles de la boucle fermée est donné par le degré du modèle G et le degré du bloc pré-spécifié H_K

- *Gain de régulateur* - Il y a deux demandes principales imposées sur le gain du régulateur : Il devrait être le plus petit possible (question économique de conservation d'énergie) et il doit être suffisamment bas en hautes fréquences (pour éviter des sollicitations indésirables des actionneurs). Pour accomplir ces deux tâches nous proposons :
 - fixer un ou plusieurs pôles réels multiples dans les pôles désirés de **retour d'état** ou **d'observateur** de la forme :

$$p_{rm}(z^{-1}) = (1 - \alpha z^{-1})^{N_m} \quad (4.8)$$

La multiplicité $N_m \in \{1, 2, \dots, n_y\}$ est en général fixée à la multiplicité maximale n_y (nombre de sorties) permise par les algorithmes [KND85, TY96, Var00] de placement de pôles. La position $\alpha \in (0, 1)$ des pôles dépend de l'atténuation désirée. Plus il faut atténuer des basses fréquences, plus la valeur est plus proche de 1 (les valeurs typiques sont entre 0.2 et 0.5).

Remarque 4.1 : Le pôle multiple fixé représente un filtre passe-bas ajouté dans la dynamique de boucle fermée avec sa bande-passante déterminée par la position α . Cet outil caractéristique de calibrage de sensibilités est repris de la méthode de calibrage SISO [LK98]. Le polynôme (4.8) est connu aussi comme le filtre de robustification [MZ89].

- introduire par la partie fixe H_K du régulateur dans chaque canal un zéro à -1 . Dans ce cas H_K va avoir la forme suivante :

$$H_K(z^{-1}) = \begin{pmatrix} (1 + z^{-1}) & 0 & \dots & 0 \\ 0 & (1 + z^{-1}) & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & (1 + z^{-1}) \end{pmatrix} \quad (4.9)$$

Il s'agit de l'ouverture de boucle à la fréquence de Nyquist $\frac{f_e}{2}$, un outil de calibrage utilisé dans la synthèse SISO [Lan02]. Les zéros

effectuent l'atténuation de S_{up} à très hautes fréquences (à $\frac{f_c}{2}$ Hz le module est nul). L'outil influence aussi les autres sensibilités, S_{yb} est atténuée comme S_{up} , et S_{yp} va avoir son module égale à 1 à la fréquence $\frac{f_c}{2}$ Hz. Le désavantage de ce filtre très efficace est qu'il augmente la complexité du régulateur.

- *Dynamique (bande passante) désirée de boucle fermée* - si les pôles de l'observateur sont fixés pour obtenir la robustesse, nous avons les pôles du modèle du procédé dans les pôles de la boucle fermée. Ainsi, il faut prendre en compte la vitesse de ces modes qui peut être dramatiquement inférieure par rapport à la vitesse de la dynamique désirée (du temps de montée désiré) de boucle fermée. Pour garantir une dynamique (temps de montée) désirée de boucle fermée, nous proposons :
 - fixer n_y paires de pôles complexes conjugués dans les pôles désirés de **retour d'état**. La fréquence naturelle de ces paires de pôles doit correspondre à la dynamique désirée ou elle doit être plus haute dans le cas d'une dynamique très lente des pôles d'observateur. L'amortissement est entre 0.75 (dépassement vers 5%) et 1 (dépassement de 0%).
 - assurer que tous les pôles multiples fixés soient plus rapides que la dynamique désirée.
 - accélérer les pôles fixés d'observateur (surtout les plus lents) en conservant la structure des ces pôles (pôles complexes restent complexes, pôles réels restent réels). Ce choix détériore la robustesse - les maximums de $\|S_{yb}(z^{-1})\|$, $\|S_{yp}(z^{-1})\|$, et $\|S_{up}(z^{-1})\|$ sont plus élevés. Ainsi il est conseillé d'accélérer modérément.
- *Atténuation en hautes fréquences (roll-off)* - Pour obtenir une atténuation de hautes fréquences en boucle fermée, il faut :
 - introduire des pôles réels multiples dans les pôles désirés **d'observateur** ou de **retour d'état**. Comme dans le cas de calibrage du gain de régulateur, il s'agit du filtre de robustification (4.8).
 - placer un filtre de l'ouverture de boucle (4.9) dans le bloc pré-spécifié

du régulateur H_K .

Remarque 4.2 : Pour les systèmes échantillonnés, la fréquence maximale est $0.5f_e$, et ainsi le terme de "hautes fréquences" signifie les fréquences entre $0.25f_e$ et $0.5f_e$.

- *Interaction (découplage)* - La question de découplage est toujours ouverte dans le cas du placement de pôles multivariable par l'optimisation des vecteurs propres. La deuxième méthode de placement de pôles (méthode de choix explicite de vecteurs propres) propose quelques possibilités de placement de vecteurs propres afin d'améliorer le découplage [Mag02].

4.4 Procédure de synthèse de régulateurs par placement des pôles

La procédure de synthèse de régulateurs robustes MIMO par placement de pôles est similaire à la procédure proposée dans [LK98] pour les systèmes SISO. Elle permet de synthétiser un régulateur central qui peut être utilisé pour l'optimisation. Premièrement, les gabarits sur les réponses fréquentielles des sensibilités (3.13) - (3.17) sont définies afin de garantir des spécifications de synthèse (il n'est pas nécessaire d'utiliser toujours toutes les sensibilités). Après, un processus itératif est appliqué pour entrer les sensibilités choisies dans les gabarits fixés. La procédure de synthèse comporte les étapes suivantes :

1. Choisir des matrices de sensibilité à ajuster et définir les gabarits pour les réponses fréquentielles de ces matrices. Les sensibilités $S_{yp}(z^{-1})$ (pour la stabilité robuste), $S_{up}(z^{-1})$ (pour limiter le gain du régulateur), et $S_{yb}(z^{-1})$ (pour la stabilité robuste et pour garantir la dynamique désirée) sont typiquement pris en compte. Les gabarits définis limitent les valeurs singulières ou les modules des réponses fréquentielles de ces sensibilités. Pour les détails voir la section 3.2.

2. Si nécessaire, définir le bloc pré-spécifié $H_K(z^{-1})$. Cette matrice de transfert est utilisée pour garantir certaines spécifications de synthèse. L'élimination de l'erreur statique est, par exemple, effectuée en introduisant un intégrateur multivariable dans cette matrice, i.e. les éléments diagonales de $H_K(z^{-1})$ contiennent chacun un intégrateur. A voir aussi le Remarque 4.1.
3. Fixer les pôles désirés de la boucle fermée en utilisant les règles proposés, afin de faire entrer les sensibilités dans les gabarits définis.
4. Évaluer les matrices L et F du régulateur $K_0(z^{-1})$ de (4.4), reconstruire le régulateur complet $K(z^{-1})$ d'après (4.5), (4.6) et calculer les réponses fréquentielles calibrées.

Remarque 4.3 : Pour l'évaluation des matrices F , L , il est avantageux d'utiliser les méthodes de Tits ou Varga d'abord. Elles donnent souvent des résultats raisonnables sans itération et ainsi il est possible d'estimer si les positions choisies des pôles de la boucle fermée sont correctes. Néanmoins, dans la phase finale, quand on connaît les valeurs des pôles, il peut être utile d'essayer la méthode de Kautsky. Elle peut donner des meilleurs résultats en ajustant les positions des pôles.

5. Vérifier les réponses des matrices de sensibilité, si elles entrent dans les gabarits. Si les résultats ne sont pas satisfaisants, retourner au pas 3 ou éventuellement 2.

4.5 Exemples de synthèse

4.5.1 Colonne à distiller

Le procédé à commander est une colonne à distiller avec le modèle linéaire continue simplifié suivant [Lim91] :

$$G(s) = \frac{1}{75s + 1} \begin{bmatrix} 0.878 & -0.864 \\ 1.082 & -1.096 \end{bmatrix} \quad (4.10)$$

où l'opérateur $s = j\omega$ de (4.10) est en rad/min.

Les spécifications de synthèse sont les suivantes :

1. Pour la réponse à l'échelon de chaque canal, la sortie doit atteindre la valeur 0.9 dans 30 min maximum, avec le dépassement maximal 10% et l'interaction maximal 0.5.
2. L'erreur statique maximale doit être 1%.

Nous allons illustrer sur cet exemple l'effet de différents choix des pôles désirés de la boucle fermée.

La synthèse comporte les pas suivants :

- *Modèle échantillonné* Il faut discrétiser le modèle nominal continu $G_n(s)$. Nous utiliserons la période d'échantillonnage $T_e = 2\text{min}$ et la représentation d'état du modèle $G_n(z^{-1})$ est calculée. Les deux pôles du modèle ont la valeur 0.973686.
- *Partie fixe* Pour garantir la spécification 2, un intégrateur MIMO est introduit dans la matrice pré-spécifiée :

$$H_K(z^{-1}) = \begin{bmatrix} \frac{1}{1-z^{-1}} & 0 \\ 0 & \frac{1}{1-z^{-1}} \end{bmatrix}$$

- *Placement de pôles 1* Pour satisfaire la spécification 1, deux paires de pôles complexes à $f_n = 0.025\text{Hz}$ avec l'amortissement $\zeta = 0.9$ sont placées comme les pôles du retour d'état. La fréquence naturelle 0.025Hz et l'amortissement de ces pôles correspond bien à la vitesse et le dépassement demandés. Tous les pôles non-spécifiés sont dans la phase initiale placés autour de l'origine, ainsi leur influence est négligeables. Les valeurs singulières des sensibilités $S_{yp}(z^{-1})$, $S_{up}(z^{-1})$ et $S_{yb}(z^{-1})$ sont montrées dans Fig. 4.2, Fig. 4.3, et Fig. 4.4 (courbes en pointillé). Nous observons, que les maximums de $\bar{\sigma}(S_{yp}(z^{-1}))$ et de $\bar{\sigma}(S_{yb}(z^{-1}))$ sont assez élevés (problème de robustesse) et que $\bar{\sigma}(S_{up}(z^{-1}))$ est grand sur toutes les fréquences sauf les plus basses (solicitation importante de l'actionneur).
- *Placement de pôles 2* Pour améliorer la robustesse, les deux pôles du modèle $(1 - 0.973686z^{-1})^2$ sont placés comme les pôles de l'observa-

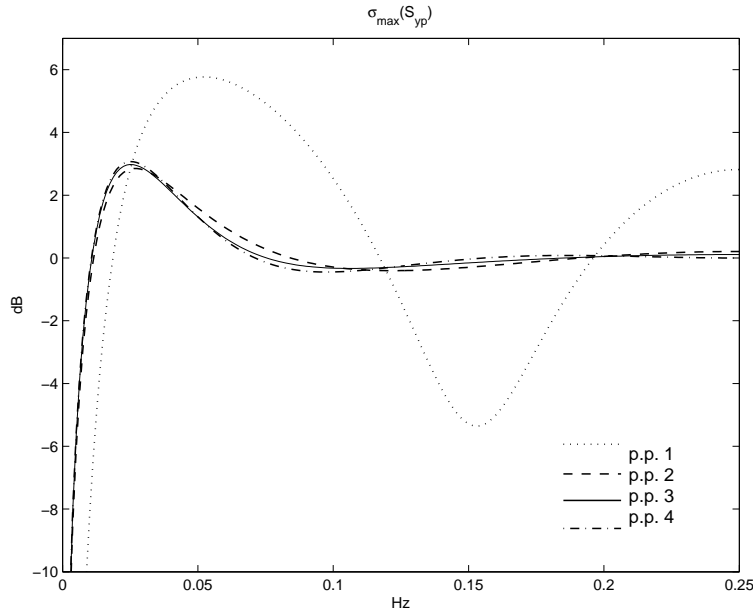


FIG. 4.2 – Réponses fréquentielles de $\bar{\sigma}(S_{yp}(z^{-1}))$ pour trois configurations de pôles désirés de BF.

teur. Cela va réduire le maximum de $S_{yp}(z^{-1})$ et $S_{yb}(z^{-1})$. Le gain du régulateur caractérisé par $S_{up}(z^{-1})$ est aussi diminué. De l'autre côté, le temps de montée de la réponse en échelon est allongé. Les valeurs singulières des sensibilités sont aussi représentées dans Fig. 4.2, Fig. 4.3, et Fig. 4.4 (courbes en tirets).

- *Placement de pôles 3* Il reste deux pôles non-spécifiés à placer (par défaut ils sont placés proche de l'origine). Nous plaçons un pôle réel avec la multiplicité 2 : $(1 - 0.3z^{-1})^2$ comme deux pôles de l'observateur. Les pôles augmentent l'atténuation des hautes fréquences, c'est qui réduit la sollicitation de l'actionneur et améliore la robustesse. L'effet est visible dans Fig. 4.2, Fig. 4.3, et Fig. 4.4 (courbes pleines). Plus le pôle réel est placé vers 1, plus il atténue les hautes fréquences, mais aussi allonge le temps de montée.
- *Placement de pôles 4* Pour illustrer l'effet de l'ouverture de boucle, à la place du pôle réel multiple placé dans le "placement de pôles 3" nous

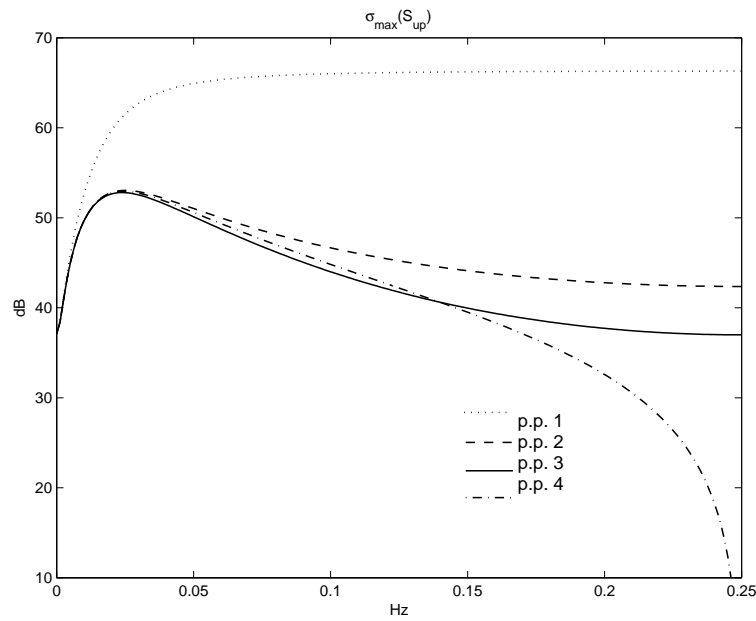


FIG. 4.3 – Réponses fréquentielles de $\bar{\sigma}(S_{up}(z^{-1}))$ pour trois configurations de pôles désirés de BF.

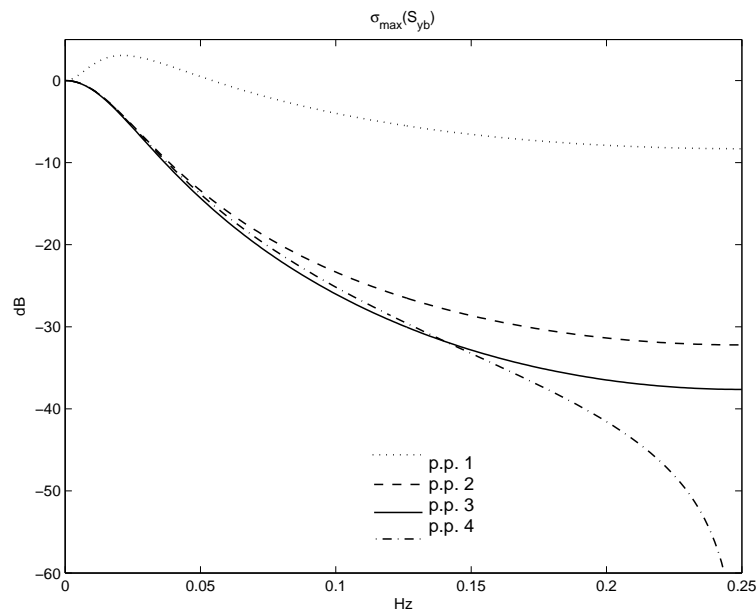


FIG. 4.4 – Réponses fréquentielles de $\bar{\sigma}(S_{yb}(z^{-1}))$ pour trois configurations de pôles désirés de BF.

allons ajouter l'ouverture de boucle dans la partie fixe H_K . Les pôles de boucle fermée sont placés comme dans le "placement de pôles 2" et le bloc pré-spécifié a la structure suivante :

$$H_K(z^{-1}) = \begin{bmatrix} \frac{1+z^{-1}}{1-z^{-1}} & 0 \\ 0 & \frac{1+z^{-1}}{1-z^{-1}} \end{bmatrix}$$

Les valeurs singulières des sensibilités sont données dans Fig. 4.2, Fig. 4.3, et Fig. 4.4 (courbes en point-tirets). Visiblement, les très hautes fréquences sont bien atténuées dans le cas de $S_{yb}(z^{-1})$ et $S_{up}(z^{-1})$, par contre l'effet sur $S_{yp}(z^{-1})$ est, comme pour le pôle réel multiple, moins important.

Pour optimiser les vecteurs propres nous avons utilisé l'algorithme de placement de pôles de Kautsky. L'algorithme de Varga a donné des sensibilités similaires, mais l'algorithme de Tits n'a pas réussi de trouver une solution.

4.5.2 Autres exemples

Un exemple de synthèse du régulateur multivariable par placement de pôles se trouve dans l'Annexe B4. Le placement de pôles est utilisé pour la synthèse des régulateurs centraux pour l'optimisation convexe des régulateurs. D'autres exemples d'application de placement de pôles se trouvent dans l'exemple de synthèse du Chapitre 5 et dans l'Annexe B5.

4.6 Notes et références

Références : La théorie de base du placement de pôles multivariable est donnée par exemple dans [Wol74]. Le placement de pôles par optimisation est traité par [KND85, TY96, Var00]. L'idée de placer aussi les vecteurs propres est développée dans [SSj94, Mag02]. Le placement de pôles multivariable avec calibrage de sensibilités est présenté (avec un exemple d'application) dans l'article de l'Annexe B4.

Fichiers et logiciels : La technique du placement de pôles multivariable avec calibrage de sensibilités est introduit dans la boîte à outils pour Matlab RMC (Robust Multivariable Control) programmée dans le cadre de ce travail. Elle est accessible dans la section de la synthèse de régulateur central pour l'optimisation convexe. La boîte à outils RMC se trouve sur le CD de la thèse dans le répertoire `\Software\RMC_tool\`. Le guide d'utilisateur du logiciel est donné dans l'Annexe B9 et aussi dans le répertoire de la boîte à outils. D'autres m-fonctions programmées pour la commande est l'analyse des systèmes MIMO sont dans le dossier `\M_functions\` (pour la description voir le fichier `Contents.m`). Les fichiers correspondants au exemple de la synthèse se trouvent dans le répertoire `\Design_examples\sec451_Colonne_\`. Pour la description des fichiers lisez le fichier `_README.txt`. Toutes les figures de graphes sont enregistrées sous format de Matlab dans le dossier `:\Figures\`. Les figures Fig. 4.2, Fig. 4.3 et Fig. 4.4 correspondent respectivement aux fichiers `syp_distill_pp.fig`, `sup_distill_pp.fig`, et `syb_distill_pp.fig`.

References : *Basic theory on multivariable pole-placement are given for example in [Wol74]. The pole placement (or eigenstructure assignment) are treated in [KND85, TY96, Var00]. The idea to place also the eigenvectors is developed in [SSj94, Mag02]. The multivariable pole-placement with sensitivity shaping is presented (with one example of application) in the paper Appendix B4.*

Files and software : *The multivariable pole placement with sensitivity shaping is introduced to the RMC Matlab toolbox developed in this work. It is accessible in the central controller design section of the toolbox. The toolbox RMC is on thesis CD in the directory \Software\RMC_tool\. A corresponding user's guide is given in Appendix B9 and also in the toolbox directory. Other programmed m-functions for MIMO system control and analysis are in the subdirectory \M_functions\ (for the m-functions description see Contents.m). The files corresponding to the presented design example are placed in the directory \Design_examples\sec451_Colonne_\. For detailed*

files description see the file `_README.txt`. All presented graphs are saved in Matlab format in the directory `:\Figures\`. The graphs in Fig. 4.2, Fig. 4.3 and Fig. 4.4 correspond respectively to the files `syp_distill_pp.fig`, `sup_distill_pp.fig`, and `syb_distill_pp.fig`.

Chapitre 5

Calibrage de sensibilités par optimisation convexe

Dans ce chapitre la méthode de synthèse de régulateurs multivariables par le calibrage de sensibilités avec l'optimisation convexe est introduite. La structure utilisée de régulateur est présentée. Il s'agit de la paramétrisation "Youla-Kucera" de tous régulateurs stabilisants. Le régulateur comporte un régulateur central et une matrice de transfert Q qui est un paramètre libre à ajuster. Le régulateur central doit être calculé avant l'optimisation, et nous allons employer le régulateur de placement de pôles dans le Chapitre 4 (l'observateur avec le retour d'état). Ensuite, les critères convexes à optimiser sont définis pour le calibrage des sensibilités et pour la δ -stabilité de la BF. Finalement, la procédure de synthèse de régulateurs par le calibrage de sensibilités avec l'optimisation convexe est résumée dans quelques étapes brièvement commentées. Un exemple de synthèse est donné à la fin de ce chapitre.

In this chapter, a multivariable controller design method by sensitivity shaping and convex optimization is introduced. A used controller structure is presented. The controller is parameterized by "Youla-Kucera" parameterization of all stabilizing controllers. The controller contains a central controller and a transfer matrix Q which is a free parameter to be adjusted. The cen-

tral controller has to be designed before the optimization and we will employ the pole placement controller from Chapter 4 (observer with state feedback). Next, all convex criteria for optimization are defined for sensitivity shaping and for closed-loop δ -stability. Finally, a procedure of controller design by sensitivity shaping with convex optimization is resumed to a few briefly commented steps. One design example is given at the end of this chapter.

5.1 Régulateur multivariable pour optimisation convexe

La paramétrisation de régulateur à développer doit être une transformation de la forme : $L(X) : R^l \rightarrow H_{z\infty}^{m \times n}$ (d'un espace des vecteurs R^l à un espace de matrices de transfert temps-discret $H_{z\infty}^{m \times n}$) qui rend la condition (3.29) de calibrage convexe par rapport au vecteur d'optimisation $X \in R^l$. En plus, la transformation doit assurer la stabilité de la boucle fermée et si possible la stabilité du régulateur. De point de vue d'optimisation, le régulateur à optimiser se présente comme suit :

$$K(z^{-1}) = L(X) \tag{5.1}$$

où une transformation $L(X)$ satisfaisant nos exigences est développer dans les sections suivantes.

5.1.1 Paramétrisation Youla-Kucera des régulateurs

La paramétrisation utilisée est la fameuse paramétrisation "Youla-Kucera (nommée aussi Q-paramétrisation) [BB91, Zho98, Mac89] introduite pour la première fois par [YJB76, Kuc79]. Supposons un modèle LTI quelconque $G(z^{-1})$ connecté en boucle fermée comme dans la Fig.3.1. Supposons en plus que ce modèle est stabilisé en boucle fermée par un régulateur central $K_0(z^{-1})$. Le modèle et le régulateur peuvent être factorisés par la factorisa-

tion coprime suivante :

$$\begin{aligned} G(z^{-1}) &= N(z^{-1})M^{-1}(z^{-1}) = \tilde{M}(z^{-1})^{-1}\tilde{N}(z^{-1}) \\ K_0(z^{-1}) &= U_0(z^{-1})V_0^{-1}(z^{-1}) = \tilde{V}_0^{-1}(z^{-1})\tilde{U}_0(z^{-1}) \end{aligned}$$

avec $U_0(z^{-1})$, $\tilde{U}_0(z^{-1})$, $V_0(z^{-1})$, $\tilde{V}_0(z^{-1})$, $M(z^{-1})$, $\tilde{M}(z^{-1})$, $N(z^{-1})$, $\tilde{N}(z^{-1})$ des matrices rationnelles stables et tel que

$$\begin{bmatrix} \tilde{V}_0(z^{-1}) & -\tilde{U}_0(z^{-1}) \\ -\tilde{N}(z^{-1}) & \tilde{M}(z^{-1}) \end{bmatrix} \begin{bmatrix} M(z^{-1}) & U_0(z^{-1}) \\ N(z^{-1}) & V_0(z^{-1}) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Ensuite, tous les régulateurs qui stabilisent la boucle fermée peuvent s'écrire sous la forme fractionnelle suivante :

$$K(z^{-1}) = (U_0(z^{-1}) + M(z^{-1})Q(z^{-1}))(V_0(z^{-1}) - N(z^{-1})Q(z^{-1}))^{-1} \quad (5.2)$$

pour $Q(z^{-1})$ une matrice rationnelle et stable.

Les propriétés essentielles de cette paramétrisation sont :

- le régulateur nominal $K_0(z^{-1})$ peut avoir la forme de représentation d'état d'un observateur avec le retour d'état [BB91, Mac89]. Ultérieurement, cette forme de $K_0(z^{-1})$ sera utilisée pour la mise en oeuvre du régulateur optimisé $K(z^{-1})$.
- toutes les matrices de la boucle fermée (les sensibilités) peuvent s'écrire comme suit :

$$S_{ij}(z) = T_{1ij}(z^{-1}) + T_{2ij}(z^{-1})Q(z^{-1})T_{3ij}(z^{-1}) \quad (5.3)$$

où la matrice $Q(z^{-1})$ est le paramètre libre à optimiser et les matrices $T_{1ij}(z^{-1})$, $T_{2ij}(z^{-1})$, $T_{3ij}(z^{-1})$ sont des matrices de transfert dépendantes du système bouclé. Avec les matrices de sensibilité sous la forme (5.3) la condition (3.29) de calibrage de sensibilités devient :

$$\bar{\sigma} (T_{1ij}(e^{-j\omega_d}) + T_{2ij}(e^{-j\omega_d})Q(e^{-j\omega_d})T_{3ij}(e^{-j\omega_d})) \leq F_{ij}(\omega_d) \quad \forall \omega_d \quad (5.4)$$

et cela représente un ensemble des conditions convexes par rapport à la matrice $Q(z^{-1})$.

- Les pôles de $Q(z^{-1})$ sont en même temps des pôles de la boucle fermée. Ainsi pour avoir une optimisation efficace, il faut optimiser aussi ces pôles.

Malheureusement, il ne s'agit pas d'une transformation du type $L(X) : R^l \rightarrow H_{z\infty}^{m \times n}$ (d'un espace vectoriel à un espace de matrices rationnelles de transfert), convenable pour l'optimisation, mais il s'agit de la transformation du type : $L(H(z^{-1})) : H_{z\infty}^{m \times n} \rightarrow H_{z\infty}^{m \times n}$ (d'un espace de matrices rationnelles à un autre espace du même type). En conséquence, pour implémenter cette paramétrisation il nous faut une structure appropriée de la forme $L(X) : R^l \rightarrow RH_{z\infty}^{m \times n}$ pour la matrice $Q(z^{-1})$. Cette structure doit :

- Conserver la convexité de la condition (5.4)
- Assurer la stabilité de la matrice $Q(z^{-1})$. Cette propriété est nécessaire pour garantir la stabilité de la boucle fermée (voir la définition de Q-paramétrisation).
- Optimiser les pôles de $Q(z^{-1})$ et ainsi en conséquences les pôles de la boucle fermée.

5.1.2 Structure de $Q(z^{-1})$

Comme structure générale de la matrice $Q(z^{-1})$ nous allons utiliser la forme suivante d'une matrice rationnelle quelconque :

$$Q(z^{-1}) = d^{-1}(z^{-1})N(z^{-1}) \quad (5.5)$$

où $N(z^{-1})$ est une matrice des polynômes et $d(z^{-1})$ est le polynôme contenant tous les pôles de la matrice de transfert $Q(z^{-1})$.

Il reste à déterminer la structure du polynôme $d(z^{-1})$ et de la matrice des polynômes $N(z^{-1})$. Nous avons considéré 2 cas possibles :

- *Structure standard* - considère directement les coefficients des polynômes comme les éléments du vecteur optimisé X .
- *Structure basée sur l'approximation de Ritz* - dans ce cas chaque polynôme est une combinaison linéaire des fonctions rationnelles stables et propres.

Remarque 5.1 : La structure proposée de $Q(z^{-1})$ n'est pas minimale, car le plus petit dénominateur commun $d(z^{-1})$ est sorti de la matrice $Q(z^{-1})$. Il serait donc intéressant d'analyser la matrice $Q(z^{-1})$ après l'optimisation, si elle ne contient pas des couples de pôles-zéros qui s'éliminent. Cela pourrait réduire la complexité de la matrice $Q(z^{-1})$.

Structure standard de $d(z^{-1})$ et $N(z^{-1})$

Dans cette forme de $d(z^{-1})$ et $N(z^{-1})$, les coefficients des tous polynômes sont directement les éléments correspondants du vecteur d'optimisation X . La paramétrisation de la matrice $Q(z^{-1})$ est définie comme suit :

$$Q(z^{-1}) = L_{S_{N_q}}(X) : R^l \rightarrow H_{z^\infty}^{n_u \times n_y} \mid l = n_u * n_y * (N_q + 1) + N_q \quad (5.6)$$

avec

$$\begin{aligned} X &= [x_0, x_1, x_2, \dots, x_l] \\ L_{S_{N_q}}(X) &= d_s^{-1}(X) N_s(X) \\ d_s(X) &= x_0 + x_1 z^{-1} + x_2 z^{-2} + \dots + x_{N_q} z^{-N_q} \\ N_s(X) &= \begin{bmatrix} n_{11}(z^{-1}) & n_{12}(z^{-1}) & \dots \\ \dots & n_{ij}(z^{-1}) & \dots \\ \dots & \dots & n_{n_u n_y}(z^{-1}) \end{bmatrix} \mid i = 1, \dots, n_u; j = 1, \dots, n_y \\ n_{ij}(z^{-1}) &= x_k + x_{k+1} z^{-1} + \dots + x_{k+N_q} z^{-N_q} \\ k &= N_q + (i - 1) * n_y * (N_q + 1) + (j - 1) * (N_q + 1) + 1, \\ &N_q + (i - 1) * n_y * (N_q + 1) + (j - 1) * (N_q + 1) + 2, \dots, l \end{aligned} \quad (5.7)$$

où N_q est l'ordre de la paramétrisation qui correspond à l'ordre des tous numérateurs et du dénominateur de la matrice $Q(z^{-1})$.

Remarque 5.2 : Il n'est pas nécessaire d'avoir le même degré des polynômes $n_{ij}(z^{-1})$ et du polynôme $d_s(X)$. Il est donc possible, par exemple : D'optimiser un coefficient dans les numérateurs $n_{ij}(z^{-1})$ (toutes les zéros à

l'origine) et N_q pôles de $d_s(X)$. L'autre variante est de fixer les pôles de $d_s(X)$ (pôles de la BF) et d'optimiser seulement la matrice $N(z^{-1})$ de numérateurs. etc.

Approximation de Ritz de $d(z^{-1})$ et de $N(z^{-1})$

Cette forme de $d(z^{-1})$ et de $N(z^{-1})$ considère le polynôme $d(z^{-1})$ et les polynômes $n_{ij}(z^{-1})$ de la matrice $N(z^{-1})$ comme les fonctions de transfert avec le même dénominateur commun $d_c(z^{-1})$. En plus, un nouveau paramètre z_0 est introduit. La même paramétrisation est implementée pour le cas SISO dans [RM94, Lan98] et pour une matrice de transfert dans [BB91]. La paramétrisation de la matrice $Q(z^{-1})$ est définie comme suit :

$$Q(z^{-1}) = Lr_{N_q}(X) : R^l \rightarrow H_{z\infty}^{n_u \times n_y} \mid l = n_u * n_y * (N_q + 1) + N_q \quad (5.8)$$

avec

$$\begin{aligned} X &= [x_0, x_1, x_2, \dots, x_l] \\ Lr_{N_q}(X) &= d_r^{-1}(X)N_r(X) \\ d_r(X) &= x_0 + x_1\alpha_1 + x_2\alpha_2 + \dots + x_{N_q}\alpha_{N_q} \\ N_r(X) &= \begin{bmatrix} n_{11}(z^{-1}) & n_{12}(z^{-1}) & \dots \\ \dots & n_{ij}(z^{-1}) & \dots \\ \dots & \dots & n_{n_u n_y}(z^{-1}) \end{bmatrix} \mid i = 1, \dots, n_u; j = 1, \dots, n_y \\ n_{ij}(z^{-1}) &= x_k + x_{k+1}\alpha_1 + \dots + x_{k+N_q}\alpha_{N_q} \\ \alpha_p &= \left(\frac{z_0 - z^{-1}}{1 - z_0 z^{-1}} \right)^p \mid p = 1, 2, \dots, N_q; z_0 \in \langle -1, 1 \rangle \\ k &= N_q + (i - 1) * n_y * (N_q + 1) + (j - 1) * (N_q + 1) + 1, \\ &N_q + (i - 1) * n_y * (N_q + 1) + (j - 1) * (N_q + 1) + 2, \dots, l \end{aligned} \quad (5.9)$$

d'où $d_r(X)$ et $N_r(X)$ ont un dénominateur commune $d_c(z^{-1})$ avec la structure :

$$d_c(z^{-1}) = (1 - z_0 z^{-1})^{N_q} \quad (5.10)$$

Le paramètre z_0 influence les résultats d'optimisation. L'analyse de l'influence de z_0 pour le cas SISO est faite dans [Lan98]. En principe plus la valeur de z_0 approche 1, plus les pôles optimisés de $Q(z^{-1})$ sont situés vers la partie droite (bases fréquences) du cercle d'unité et vice versa.

Remarque 5.3 : Contrairement à la structure standard, les termes $d_r(X)$ et $n_{ij}(z^{-1})$ sont ici considérés comme les fonctions de transfert. Remarque 5.2 concerne aussi l'approximation de Ritz.

Représentation d'état de la matrice Q - carrée

Considérons la matrice $Q(z^{-1})$ carrée, c'est-à-dire $n_u = n_y$ (pour la forme non-carrée $n_u \neq n_y$ il faut utiliser d'autres méthodes de conversion) avec la structure (5.5). La matrice de numérateur $N(z^{-1})$ et le dénominateur $d(z^{-1})$ sont paramétrisés soit par (5.7), soit par (5.9). Considérons maintenant la représentation d'état de la matrice $Q(z^{-1})$:

$$\begin{aligned} x_Q(t+1) &= A_Q x_Q(t) + B_Q e_o(t) \\ y_Q(t) &= C_Q x_Q(t) + D_Q e_o(t) \end{aligned} \quad (5.11)$$

Les matrices constantes de représentation d'état $A_Q \in R^{N_q \times N_q}$, $B_Q \in R^{N_q \times n_y}$, $C_Q \in R^{n_u \times N_q}$, $D_Q \in R^{n_u \times n_y}$ d'une forme standard contrôlable peuvent être obtenues comme suit [Kuc91] :

Matrice D_Q - Elle correspond à :

$$D_Q = Q(z^{-1})|_{z=0} = \underbrace{[n_{ij}(z^{-1})]}_{N(X)}|_{z=0} \quad (5.12)$$

Matrice A_Q - Supposons une matrice polynômiale diagonale du dénominateur $d(z^{-1})$ de (5.5) suivante :

$$\text{diag}\{d(z^{-1})\} = I + A'_2 \begin{bmatrix} z^{-Nq} & 0 & \cdots \\ z^{-Nq+1} & & \\ \vdots & & \\ z^{-1} & & \\ & \ddots & \\ 0 & & z^{-Nq} \\ \vdots & & z^{-Nq+1} \\ & & \vdots \\ 0 & & z^{-1} \end{bmatrix} \quad (5.13)$$

Décrivons les coefficients de la matrice constante A'_2 comme suit :

$$A'_2 = \begin{bmatrix} \alpha_{11}^1 & \cdots & \alpha_{11}^{Nq} & \cdots & \alpha_{1n_u}^1 & \cdots & \alpha_{1n_u}^{Nq} \\ & & & \ddots & & & \\ \alpha_{n_u 1}^1 & \cdots & \alpha_{n_u 1}^{Nq} & \cdots & \alpha_{n_u n_u}^1 & \cdots & \alpha_{n_u n_u}^{Nq} \end{bmatrix} \quad (5.14)$$

Ensuite, la matrice A_Q a la forme suivante :

$$A_Q = \begin{bmatrix} F_{11} & \cdots & F_{1n_u} \\ \vdots & & \vdots \\ F_{n_u 1} & \cdots & F_{n_u n_u} \end{bmatrix} \quad (5.15)$$

où les matrices F_{ij} sont les matrices "companiones" de la dimension $N_q \times N_q$. Pour $i = j$ elles sont de la forme :

$$F_{ii} = \begin{bmatrix} 0 & 1 & 0 & \cdots \\ \vdots & & \ddots & \\ 0 & & & 1 \\ -\alpha_{ii}^1 & -\alpha_{ii}^2 & \cdots & -\alpha_{ii}^{Nq} \end{bmatrix} \quad (5.16)$$

et pour $i \neq j$ de la forme :

$$F_{ij} = \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & & \vdots \\ 0 & \cdots & 0 \\ -\alpha_{ij}^1 & \cdots & -\alpha_{ij}^{N_q} \end{bmatrix} \quad (5.17)$$

Matrice B_Q - Elle a la forme suivante :

$$B_Q = \begin{bmatrix} B'_1 \\ \vdots \\ B'_{n_u} \end{bmatrix} \quad (5.18)$$

où les blocs B'_i sont définis comme suit :

$$B'_i = \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & & \vdots \\ 0 & \cdots & 0 \\ e_1 & e_2 & \cdots & e_{n_u} \end{bmatrix} \quad (5.19)$$

où l'élément e_i est égal à $e_i = 1$ et tous les autres sont égaux à 0.

Matrice C_Q - Définissons une matrice polynômiale $C_2(z^{-1})$:

$$C_2(z^{-1}) = [n_{ij}(z^{-1})] - D_Q \text{diag}\{d(z^{-1})\} \quad (5.20)$$

où $N_{s(r)} = [n_{ij}(z^{-1})]$ et $d(z^{-1})$ sont la matrice polynômiale et le dénominateur commun de la matrice $Q(z^{-1})$ dans (5.5). La matrice constante C_Q peut être

déterminée à partir de la relation suivante :

$$C_2(z^{-1}) = C_Q \begin{bmatrix} z^{-N_q} & 0 \\ z^{-N_q+1} & \vdots \\ \vdots & \\ z^{-1} & \\ 0 & \ddots & 0 \\ \vdots & & z^{-N_q} \\ & & z^{-N_q+1} \\ & & \vdots \\ 0 & & z^{-1} \end{bmatrix} \quad (5.21)$$

5.1.3 Structure complète du régulateur

Supposons que la structure du régulateur est donnée par la paramétrisation Youla-Kucera où la matrice $Q(z^{-1})$ a la forme (5.5) et avec $d(z^{-1})$ et $N(z^{-1})$ caractérisés soit par la structure (5.7), soit par l'approximation de Ritz (5.9). Supposons aussi que le régulateur peut contenir une matrice de transfert pré-spécifiée nommée $H_K(z^{-1})$. La matrice H_K permet de vérifier certaines spécifications de performance (pour les détails voir Remarque 4.1). Le régulateur complet $K(z^{-1})$ est de la forme :

$$\begin{aligned} K(z^{-1}) &= H_K(z^{-1})K_c(z^{-1}) = \\ &= H_K(z^{-1}) (U_0(z^{-1}) + M(z^{-1})Q(z^{-1})) (V_0(z^{-1}) - N(z^{-1})Q(z^{-1}))^{-1} \end{aligned} \quad (5.22)$$

avec

$$\begin{aligned} G(z^{-1}) &= N(z^{-1})M^{-1}(z^{-1}) \\ K_0(z^{-1}) &= U_0(z^{-1})V_0^{-1}(z^{-1}) \end{aligned}$$

Le régulateur optimisé $K_c(z^{-1})$ est construit à partir du modèle $G(z^{-1})$, du régulateur central $K_0(z^{-1})$ et de la matrice de transfert optimisé $Q(z^{-1})$.

En pratique, il sera réalisé comme un observateur avec le retour d'état estimé, où la matrice $Q(z^{-1})$ est connectée entre l'erreur d'estimation de sortie $e_o(t)$ et la commande $u_o(t)$ envoyée au filtre $H_K(z^{-1})$. La configuration complète est montrée dans la Fig.5.1.

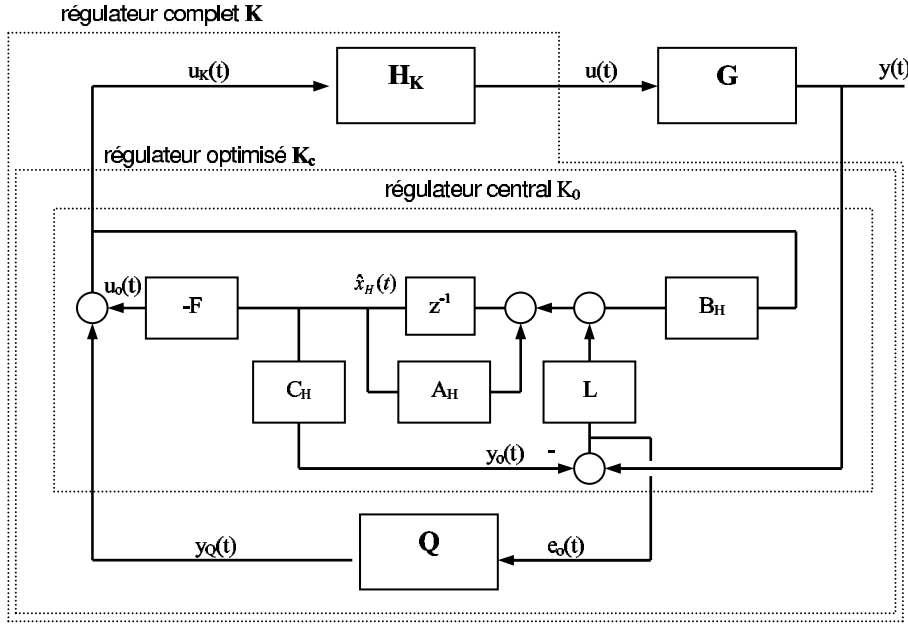


FIG. 5.1 – Structure du régulateur complet

Supposons le modèle du procédé $G(z^{-1})$ et la partie fixe $H_K(z^{-1})$ décrits avec les équations d'état suivantes :

$$\begin{aligned}
 x_G(t+1) &= Ax_G(t) + Bu(t) \\
 y(t) &= Cx_G(t) \\
 x_{H_K}(t+1) &= A_{H_K}x_{H_K}(t) + B_{H_K}u_o(t) \\
 u(t) &= C_{H_K}x_{H_K}(t) + D_{H_K}u_o(t)
 \end{aligned} \tag{5.23}$$

La cascade des matrices de transfert $H_K(z^{-1}) \rightarrow G(z^{-1})$ nous donne un système $H(z^{-1}) = G(z^{-1})H_K(z^{-1})$ avec le vecteur d'état $x_H(t) = [x_{H_K}^T(t), x_G^T(t)]^T$ décrit par :

$$\begin{aligned}
 x_H(t+1) &= A_Hx_H(t) + B_Hu_o(t) \\
 y(t) &= C_Hx_H(t)
 \end{aligned} \tag{5.24}$$

où les matrices de représentation d'état ont la structure (4.3).

Le régulateur optimisé $K_c(z^{-1})$ est ensuite décrit par les équations :

$$\begin{aligned}\hat{x}_H(t+1) &= A_H\hat{x}_H(t) + Le_o(t) + B_Hu_K(t) \\ y_o(t) &= C_H\hat{x}_H(t)\end{aligned}\quad (5.25)$$

avec \hat{x}_H l'estimation du vecteur d'état de la cascade (5.24). Le retour d'état $u_K(t)$ et l'erreur d'estimation $e_o(t)$ sont donnée par :

$$\begin{aligned}u_K(t) &= -F\hat{x}(t) + y_Q(t) \\ e_o(t) &= y(t) - y_o(t)\end{aligned}\quad (5.26)$$

La matrice L est la matrice d'observateur et F est la matrice de retour d'état. Elles déterminent le placement d'une partie des pôles de boucle fermée où $(A_H - LC_H)$ définit les pôles d'observateur et $(A_H - B_HF)$ définit les pôles de retour d'état. Les deux matrices L et F constituent le régulateur central $K_0(z^{-1})$. Plus des détails sur le calcul du régulateur central utilisé dans ce travail se trouvent dans le Chapitre 4.

Rajoutons les relations (5.26) et les équations d'état de $Q(z^{-1})$ (5.11) dans les équations (5.25) du régulateur $K_c(z^{-1})$. Nous obtenons l'expression complète des équations d'état du régulateur optimisé $K_c(z^{-1})$:

$$\begin{aligned}x_c(t+1) &= A_cx_c(t) + B_cy(t) \\ u_K(t) &= C_cx_c(t) + D_cu_K(t)\end{aligned}\quad (5.27)$$

avec le vecteur d'état x_c et les matrices A_c, B_c, C_c, D_c suivantes :

$$\begin{aligned}x_c &= \begin{bmatrix} \hat{x} \\ x_Q \end{bmatrix}, A_c = \begin{bmatrix} A_H - B_HF - LC_H - B_HD_QC_H & B_HC_Q \\ -B_QC_H & A_Q \end{bmatrix} \\ B_c &= \begin{bmatrix} B_HD_Q + L \\ B_Q \end{bmatrix}, C_c = \begin{bmatrix} -F - D_QC_H & C_Q \end{bmatrix}, D_c = D_Q\end{aligned}\quad (5.28)$$

Finalement le régulateur complet $K(z^{-1})$ est obtenu par la connection en cascade (5.22) du régulateur $K_c(z^{-1})$ et de la partie fixe $H_K(z^{-1})$.

Remarque 5.4 : Le régulateur central influence dramatiquement les propriétés du régulateur optimisé $K_c(z^{-1})$. Ainsi, le régulateur central doit être attentivement synthétisé de telle façon que ses propriétés (comme robustesse et performance) approchent les propriétés désirées du régulateur optimisé. Ceci va simplifier la tâche de l'optimisation et va réduire la complexité du régulateur obtenu.

5.2 Optimisation convexe

L'optimisation convexe peut être appliquée seulement si nous avons une fonction critère convexe. Dans la synthèse de régulateurs la fonction critère doit refléter les spécifications et les contraintes sur la performance de régulation et de poursuite et sur la robustesse. Dans cette section nous allons développer les fonctions critère qui permettent d'optimiser trois types de propriétés de la boucle fermée :

- *Calibrage de sensibilités* - A partir de la condition pour le calibrage de sensibilités (3.29) ou (5.4) nous allons développer les fonctions critères convexes pour imposer des gabarits supérieurs sur les sensibilités¹.
- *Poursuite en boucle fermée* - Une fonction critère qui évalue la similitude entre la sensibilité de poursuite S_{yy^*} et un modèle de référence spécifié est développée.
- *δ -stabilité de boucle fermée* - Un critère pour imposer une région δ sur l'emplacement des pôles de boucle fermée.

5.2.1 Calibrage de sensibilités

Utilisant la forme (5.5) de la matrice $Q(z^{-1})$, la condition (5.4) de calibrage de sensibilités devient :

$$\bar{\sigma} \left(T_{1ij}(e^{-j\omega_d}) + T_{2ij}(e^{-j\omega_d})d^{-1}(e^{-j\omega_d})N(e^{-j\omega_d})T_{3ij}(e^{-j\omega_d}) \right) \leq F_{ij}(\omega_d) \quad \forall \omega_d \quad (5.29)$$

¹Les matrices de transfert de la boucle fermée comme S_{yp} , S_{up} , S_{yb} , S_{yd} , ou S_{ud}

Sachant que tous les éléments de la condition (5.29) sont dépendants de la pulsation ω_d , simplifions l'écriture en supprimant $(e^{-j\omega_d})$. La condition s'écrit :

$$\bar{\sigma}(T_{1ij} + T_{2ij}d^{-1}NT_{3ij}) \leq F_{ij} \mid \forall \omega_d \quad (5.30)$$

Pour mettre cette condition convexe par rapport aux $d(z^{-1})$ et $N(z^{-1})$, nous la développons comme suit :

$$\begin{aligned} \bar{\sigma}(T_{1ij} + T_{2ij}d^{-1}NT_{3ij}) &\leq F_{ij} \equiv \\ \equiv |d|\bar{\sigma}(T_{1ij} + T_{2ij}d^{-1}NT_{3ij}) &\leq |d|F_{ij} \equiv \\ \equiv \bar{\sigma}(dT_{1ij} + T_{2ij}NT_{3ij}) - |d|F_{ij} &\leq 0 \mid \forall \omega \in \langle 0, \frac{\omega_e}{2} \rangle \end{aligned} \quad (5.31)$$

c'est-à-dire la condition est multipliée avec la valeur absolue du dénominateur $|d(z^{-1})|$ est la partie droit de la condition est déplacée à gauche.

L'inégalité (5.31) est encore non-convexe à cause de l'élément $-|d|F_{ij}$. Remplaçant la valeur absolue $|d|$ par la partie réel $Re\{d\}$, elle devient convexe (voir le Preuve 5.1). Nous avons ainsi la condition de la forme suivante :

$$\bar{\sigma}(T_{1ij}d + T_{2ij}NT_{3ij}) - Re\{d\}F_{ij} \leq 0 \mid \forall \omega_d \quad (5.32)$$

Remarque 5.5 : Notons que la condition (5.32) reste suffisante mais elle n'est plus nécessaire, la condition devient plus conservatrice par rapport à la condition (5.31). En plus, la multiplication par $|d|$, qui est un terme dépendant de la fréquence, modifie la forme de la condition dans le domaine fréquentiel. Cela apporte des désavantages : en optimisant la condition (5.32) les solutions obtenues sont sub-optimales par rapport à la condition initiale (3.29) et le maximum de la condition (3.29) n'est pas à la fréquence où se trouve le maximum de la condition (5.32).

L'avantage de la dernière opération c'est que la condition (5.32) assure en même temps la stabilité du polynôme $d(z^{-1})$ et ainsi de la matrice de transfert $Q(z^{-1})$. C'est parce que le polynôme $d(z^{-1})$ doit être forcément strictement positif réel².

²Vérifiant la condition (5.32) nous avons $Re\{d(z^{-1})\} > 0 \mid \forall \omega_d$ et c'est une condition suffisant de la stabilité de polynômes (voir [Lan98])

L'évaluation de l'inégalité convexe (5.32) exige la connaissance des matrices $T_{1ij}(z^{-1})$, $T_{2ij}(z^{-1})$, $T_{3ij}(z^{-1})$. Afin d'éviter le calcul de ces matrices nous utiliserons la propriété que la valeur singulière $\bar{\sigma}(T_{1ij}d + T_{2ij}NT_{3ij})$ représente la valeur singulière de la matrice de sensibilité appropriée multipliée avec la valeur absolue du dénominateur : $|d|\bar{\sigma}(S_{ij})$. En conséquence, il est possible de réécrire l'inégalité (5.32) comme suit :

$$|d|\bar{\sigma}(S_{ij}) - Re\{d\}F_{ij} \leq 0 \quad \forall \omega_d \quad (5.33)$$

et cette forme de la condition du calibrage de sensibilités sera mise en place dans l'optimisation.

Une condition similaire peut être développée pour chaque élément (fonction de transfert) d'une matrice de sensibilité $S_{ij}(z^{-1})$. Dans la condition (5.29), la valeur singulière $\bar{\sigma}(\cdot)$ est remplacée par le module $|\cdot|$. Ensuite, en multipliant chaque coté de la matrice de sensibilité par un vecteur de sélection v_1, v_2 , il est possible d'isoler un élément de cette matrice. Les deux vecteurs ont tous ses éléments sauf un égaux à zéro. L'élément non-nul est égal à 1 et sa position détermine quelle fonction de transfert est isolée. La condition de calibrage pour un élément de la matrice de sensibilité peut être décrite par :

$$|v_1^T(T_{1ij}d + T_{2ij}NT_{3ij})v_2| - Re\{d\}F_{ij} \leq 0 \quad \forall \omega_d \quad (5.34)$$

ou par la forme utilisée pour les calculs³ :

$$|dv_1^T S_{ij}v_2| - Re\{d\}F_{ij} \leq 0 \quad \forall \omega_d \quad (5.35)$$

Définissons maintenant à partir de (5.32), (5.34) les fonctions critère $\phi_{ij}(X)$, $\phi_{ijv_1v_2}(X)$ à optimiser. Pour la structure standard (5.7) de $d(z^{-1})$ et $N(z^{-1})$ les fonctions critère sont définies comme suit :

$$\phi_{ij}(X) = \sup_{\omega_d} [\bar{\sigma}(T_{1ij}d_s(X) + T_{2ij}N_s(X)T_{3ij}) - Re\{d_s(X)\}F_{ij}] \quad (5.36)$$

$$\phi_{ijv_1v_2}(X) = \sup_{\omega_d} [|v_1^T(T_{1ij}d_s(X) + T_{2ij}N_s(X)T_{3ij})v_2| - Re\{d_s(X)\}F_{ij}] \quad (5.37)$$

³Rappelons que $(T_{1ij}d + T_{2ij}NT_{3ij}) = |d^{-1}||S_{ij}|$.

Utilisant les conditions de calibrage (5.33), (5.35), les fonctions critère deviennent :

$$\phi_{ij}(X) = \sup_{\omega_d} [|d_s(X)|\bar{\sigma}(S_{ij}) - \operatorname{Re}\{d_s(X)\}F_{ij}] \quad (5.38)$$

$$\phi_{ijv_1v_2}(X) = \sup_{\omega_d} [|d_s(X)v_1^T S_{ij}v_2| - \operatorname{Re}\{d_s(X)\}F_{ij}] \quad (5.39)$$

Proposition 5.1 : *Les fonctions critère développées (5.36), (5.37), (5.38), et (5.39) sont les fonctions convexes par rapport au vecteur d'optimisation X .*

Preuve 5.1 : Une fonction $\phi(X) : R^l \rightarrow R$ est convexe si et seulement si [Roc82, SW70]

$$\phi(\lambda X_1 + (1 - \lambda)X_2) \leq \lambda\phi(X_1) + (1 - \lambda)\phi(X_2) \mid \forall \lambda \in \langle 0, 1 \rangle; \forall X_1, X_2 \in R^l$$

Dans notre cas la condition à vérifier a la forme générale suivante :

$$\begin{aligned} & \sup_{\omega_d} [\bar{\sigma}(T_{1ij}d_s(\lambda X_1 + (1 - \lambda)X_2) + T_{2ij}N_s(\lambda X_1 + (1 - \lambda)X_2)T_{3ij}) - \\ & - \operatorname{Re}\{d_s(\lambda X_1 + (1 - \lambda)X_2)\}F_{ij}] \leq \\ & \leq \lambda \sup_{\omega_d} [\bar{\sigma}(T_{1ij}d_s(X_1) + T_{2ij}N_s(X_1)T_{3ij}) - \operatorname{Re}\{d_s(X_1)\}F_{ij}] + \\ & + (1 - \lambda) \sup_{\omega_d} [\bar{\sigma}(T_{1ij}d_s(X_2) + T_{2ij}N_s(X_2)T_{3ij}) - \operatorname{Re}\{d_s(X_2)\}F_{ij}] \end{aligned} \quad (5.40)$$

Utilisant les propriétés de la valeur singulière et de la fonction $\sup[\cdot]$:

$$\begin{aligned} \bar{\sigma}(H_1 + H_2) & \leq \bar{\sigma}(H_1) + \bar{\sigma}(H_2) \mid \forall H_1, H_2 \in C^{m \times n} \\ \bar{\sigma}(\lambda H) & \leq \lambda \bar{\sigma}(H) \mid \forall H \in C^{m \times n}, \lambda \in \langle 0, 1 \rangle \\ \sup_x [f(x) + g(x)] & \leq \sup_x [f(x)] + \sup_x [g(x)] \\ \sup_x [\lambda f(x)] & = \lambda \sup_x [f(x)] \mid \lambda \in \langle 0, 1 \rangle \end{aligned} \quad (5.41)$$

et sachant que pour $X_1, X_2, X \in R^l$ et $\lambda \in \langle 0, 1 \rangle$ les égalités suivantes sont vrai :

$$d_s(X_1 + X_2) = d_s(X_1) + d_s(X_2)$$

$$\begin{aligned}
N_s(X_1 + X_2) &= N_s(X_1) + N_s(X_2) \\
d_s(\lambda X) &= \lambda d_s(X) \\
N_s(\lambda X) &= \lambda N_s(X)
\end{aligned} \tag{5.42}$$

développons la partie gauche de la condition de convexité (5.40) :

$$\begin{aligned}
& \sup_{\omega_d} [\bar{\sigma}(T_{1ij}d_s(\lambda X_1 + (1 - \lambda)X_2) + T_{2ij}N_s(\lambda X_1 + (1 - \lambda)X_2)T_{3ij}) \\
& - Re\{d_s(\lambda X_1 + (1 - \lambda)X_2)\}F_{ij}] \\
= & \\
& \sup_{\omega_d} [\bar{\sigma}(T_{1ij}d_s(\lambda X_1) + T_{1ij}d_s((1 - \lambda)X_2) + T_{2ij}N_s((1 - \lambda)X_2)T_{3ij} + \\
& + T_{2ij}N_s(\lambda X_1)T_{3ij}) - Re\{d_s(\lambda X_1) + d_s((1 - \lambda)X_2)\}F_{ij}] \\
= & \\
& \sup_{\omega_d} [\bar{\sigma}(\lambda T_{1ij}d_s(X_1) + \lambda T_{2ij}N_s(X_1)T_{3ij} + (1 - \lambda)T_{1ij}d_s(X_2) + \\
& + (1 - \lambda)T_{2ij}N_s(X_2)T_{3ij}) - Re\{\lambda d_s(X_1) + (1 - \lambda)d_s(X_2)\}F_{ij}] \\
\leq & \\
& \sup_{\omega_d} [\lambda \bar{\sigma}(T_{1ij}d_s(X_1) + T_{2ij}N_s(X_1)T_{3ij}) - \lambda Re\{d_s(X_1)\}F_{ij} + \\
& + (1 - \lambda) \bar{\sigma}(T_{1ij}d_s(X_2)T_{2ij}N_s(X_2)T_{3ij}) - (1 - \lambda) Re\{d_s(X_2)\}F_{ij}] \\
\leq & \\
& \sup_{\omega_d} [\lambda \bar{\sigma}(T_{1ij}d_s(X_1) + T_{2ij}N_s(X_1)T_{3ij}) - \lambda Re\{d_s(X_1)\}F_{ij}] + \\
& + \sup_{\omega_d} [(1 - \lambda) \bar{\sigma}(T_{1ij}d_s(X_2) + T_{2ij}N_s(X_2)T_{3ij}) - (1 - \lambda) Re\{d_s(X_2)\}F_{ij}] \\
= & \\
& \lambda \sup_{\omega_d} [\bar{\sigma}(T_{1ij}d_s(X_1) + T_{2ij}N_s(X_1)T_{3ij}) - Re\{d_s(X_1)\}F_{ij}] + \\
& + (1 - \lambda) \sup_{\omega_d} [\bar{\sigma}(T_{1ij}d_s(X_2) + T_{2ij}N_s(X_2)T_{3ij}) - Re\{d_s(X_2)\}F_{ij}]
\end{aligned}$$

et évidemment, le dernier terme est égal à la partie droite de la condition (5.40).

□

Les propriétés (5.42) sont valide également pour la structure (5.9) de $d(z^{-1})$, $N(z^{-1})$ basée sur l'approximation de Ritz, ainsi aussi dans ce cas le même preuve peut être fait en remplaçant $d_s(z^{-1})$, $N_s(z^{-1})$ par $d_r(z^{-1})$,

$N_r(z^{-1})$.

L'évaluation des fonctions critère (5.38) ou (5.39) est faite sur une grille de fréquences de 0 à $\frac{\omega_e}{2}$ suffisamment dense afin de ne pas dégrader la fonction $\sup[\cdot]$. Plus la boucle fermée est complexe, plus il faut avoir des points, sachant que le nombre plus grand des points de fréquences ralentit l'optimisation.

Remarque 5.6 : Concernant la fonction critère (5.37) pour calibrer un élément d'une matrice de sensibilité, la preuve de convexité est la même. Les raisons principales sont :

- La valeur singulière $\bar{\sigma}(\cdot)$ et le module $|\cdot|$ possèdent des mêmes propriétés (5.41).
- La multiplication de sensibilité à droite et à gauche par les vecteurs constants v_1, v_2 ne change pas la structure de la fonction critère.

5.2.2 Poursuite en boucle fermée

L'utilisation des gabarits supérieurs sur les valeurs singulières ou sur les modules de sensibilités ne permet pas de garantir des performances en poursuite (sauf le découplage). Pourtant, les performances en poursuite sont très souvent exigées. Pour gérer ces spécifications, la sensibilité $S_{yy^*} = -S_{yb}$ doit être traitée. Nous allons définir un modèle de poursuite désirée M_{yy^*} , qui interprète les spécifications de la poursuite. Généralement, il s'agit d'une matrice de transfert diagonale avec des fonctions de transfert du 1^{er} ou 2^e ordre sur la diagonale :

$$M_{yy^*}(z^{-1}) = \text{diag} \left\{ \begin{array}{ccc} \frac{N_1(z^{-1})}{D_1(z^{-1})} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{N_{n_y}(z^{-1})}{D_{n_y}(z^{-1})} \end{array} \right\} \quad (5.43)$$

Pour développer un critère de poursuite nous allons commencer à partir de la condition suivante (la différence entre la boucle fermée et le modèle de poursuite) :

$$|(S_{yy^*}(e^{-j\omega_d}) - M_{yy^*}(e^{-j\omega_d}))| \leq \gamma \quad \forall \omega_d \quad (5.44)$$

Remarque 5.7 : Une condition similaire est utilisée dans la commande H_∞ pour introduire le critère des performances en poursuite surtout dans le cas de la synthèse de régulateurs à 2-degrés de liberté [LKP93].

Remarque 5.8 : Si les performances en poursuite ne sont pas satisfaisant même en optimisant la condition (5.44), il faut utiliser un pré-compensateur (voir le Chapitre 6).

Supposons la Q-paramétrisation du régulateur de la boucle fermée et supposons la matrice Q décrite par (5.5). La condition (5.44) devient :

$$\bar{\sigma} (T_{1yy^*} + T_{2yy^*}d^{-1}NT_{3yy^*} - M_{yy^*}) - \gamma \leq 0 \mid \forall \omega_d \quad (5.45)$$

où la notation $(e^{-j\omega_d})$ a été supprimée.

Comme dans le cas de la conditions de calibrage (5.32), multiplions la condition par $|d|$ et remplaçons ce terme par $Re\{d\}$ auprès γ . La condition (5.45) devient maintenant :

$$\bar{\sigma} (dT_{1yy^*} + T_{2yy^*}NT_{3yy^*} - dM_{yy^*}) - Re\{d\}\gamma \leq 0 \mid \forall \omega_d \quad (5.46)$$

ou utilisant la notation de S_{yy^*} :

$$\bar{\sigma} (dS_{yy^*} - dM_{yy^*}) - Re\{d\}\gamma \leq 0 \mid \forall \omega_d \quad (5.47)$$

Remarque 5.9 : L'opération a certainement déformé la condition (5.44) dans le domaine fréquentiel, en plus elle est devenue plus restrictive (voir Remarque 5.5). Cela réduit l'efficacité de la condition, car elle devrait minimiser la différence entre deux matrices de transfert ($S_{yy^*}(z^{-1})$ et $M_{yy^*}(z^{-1})$) dans le domaine fréquentiel.

Les fonctions critère correspondantes aux conditions (5.46) et (5.47) sont pour la paramétrisation (5.7) de $N(z^{-1})$ et $d(z^{-1})$:

$$\phi_{yy^*}(X) = \sup_{\omega_d} [\bar{\sigma} (d_s(X)(T_{1yy^*} - M_{yy^*}) + T_{2yy^*}N_s(X)T_{3yy^*}) - Re\{d_s(X)\}\gamma] \quad (5.48)$$

et

$$\phi_{yy^*}(X) = \sup_{\omega_d} [\bar{\sigma}(d_s(X)S_{yy^*} - d_s(X)M_{yy^*}) - \text{Re}\{d_s(X)\}\gamma] \quad (5.49)$$

Les mêmes fonctions sont obtenues pour la paramétrisation (5.9), il suffit de substituer $N_s(X) = N_r(X)$ et $d_s(X) = d_r(X)$.

Proposition 5.2 *La fonction critère (5.48) et de ce fait aussi la fonction critère (5.49) sont convexes par rapport au vecteur d'optimisation X .*

Preuve 5.2 : Une fonction $\phi(X) : R^l \rightarrow R$ est convexe si et seulement si [Roc82, SW70] :

$$\phi(\lambda X_1 + (1 - \lambda)X_2) \leq \lambda\phi(X_1) + (1 - \lambda)\phi(X_2) \mid \forall \lambda \in \langle 0, 1 \rangle; \forall X_1, X_2 \in R^l$$

Pour la fonction critère (5.48), la condition à vérifier aura la forme suivante :

$$\begin{aligned} & \sup_{\omega_d} [\bar{\sigma}(d_s(\lambda X_1 + (1 - \lambda)X_2)(T_{1yy^*} - M_{yy^*}) \\ & \quad + T_{2yy^*}N_s(\lambda X_1 + (1 - \lambda)X_2)T_{3yy^*}) \\ & \quad - \text{Re}\{d_s(\lambda X_1 + (1 - \lambda)X_2)\}\gamma] \\ & \leq \\ & \lambda \sup_{\omega_d} [\bar{\sigma}(d_s(X)(T_{1yy^*} - M_{yy^*}) + T_{2yy^*}N_s(X)T_{3yy^*}) - \text{Re}\{d_s(X)\}\gamma] + \\ & + (1 - \lambda) \sup_{\omega_d} [\bar{\sigma}(d_s(X)(T_{1yy^*} - M_{yy^*}) + T_{2yy^*}N_s(X)T_{3yy^*}) - \text{Re}\{d_s(X)\}\gamma] \end{aligned} \quad (5.50)$$

Utilisant les propriétés de la valeur singulière et de la fonction $\sup[\cdot]$:

$$\begin{aligned} \bar{\sigma}(H_1 + H_2) & \leq \bar{\sigma}(H_1) + \bar{\sigma}(H_2) \mid \forall H_1, H_2 \in C^{m \times n} \\ \bar{\sigma}(\lambda H) & \leq \lambda \bar{\sigma}(H) \mid \forall H \in C^{m \times n}, \lambda \in \langle 0, 1 \rangle \\ \sup_x [f(x) + g(x)] & \leq \sup_x [f(x)] + \sup_x [g(x)] \\ \sup_x [\lambda f(x)] & = \lambda \sup_x [f(x)] \mid \lambda \in \langle 0, 1 \rangle \end{aligned} \quad (5.51)$$

et sachant que pour $X_1, X_2, X \in R^l$ et $\lambda \in \langle 0, 1 \rangle$ les égalités suivantes sont vrai :

$$\begin{aligned}
d_s(X_1 + X_2) &= d_s(X_1) + d_s(X_2) \\
N_s(X_1 + X_2) &= N_s(X_1) + N_s(X_2) \\
d_s(\lambda X) &= \lambda d_s(X) \\
N_s(\lambda X) &= \lambda N_s(X)
\end{aligned} \tag{5.52}$$

développons la partie gauche de la condition de convexité :

$$\begin{aligned}
& \sup_{\omega_d} [\bar{\sigma}(d_s(\lambda X_1 + (1 - \lambda)X_2)(T_{1yy^*} - M_{yy^*}) + \\
& + T_{2yy^*}N_s(\lambda X_1 + (1 - \lambda)X_2)T_{3yy^*}) - Re\{d_s(\lambda X_1 + (1 - \lambda)X_2)\}\gamma] \\
& = \\
& \sup_{\omega_d} [\bar{\sigma}(\lambda d_s(X_1)(T_{1yy^*} - M_{yy^*}) + (1 - \lambda)d_s(X_2)(T_{1yy^*} - M_{yy^*}) + \\
& + \lambda T_{2yy^*}N_s(X_1)T_{3yy^*} + (1 - \lambda)T_{2yy^*}N_s(X_2)T_{3yy^*}) - \\
& - Re\{\lambda d_s(X_1) + (1 - \lambda)d_s(X_2)\}\gamma] \\
& \leq \\
& \sup_{\omega_d} [\lambda \bar{\sigma}(d_s(X_1)(T_{1yy^*} - M_{yy^*}) + T_{2yy^*}N_s(X_1)T_{3yy^*}) - \lambda Re\{d_s(X_1)\}\gamma + \\
& + (1 - \lambda)\bar{\sigma}(d_s(X_2)(T_{1yy^*} - M_{yy^*}) + T_{2yy^*}N_s(X_2)T_{3yy^*}) - \\
& - (1 - \lambda)Re\{d_s(X_2)\}\gamma] \\
& \leq \\
& \lambda \sup_{\omega_d} [\bar{\sigma}(d_s(X_1)(T_{1yy^*} - M_{yy^*}) + T_{2yy^*}N_s(X_1)T_{3yy^*}) - Re\{d_s(X_1)\}\gamma] + \\
& + (1 - \lambda) \sup_{\omega_d} [\bar{\sigma}(d_s(X_2)(T_{1yy^*} - M_{yy^*}) + T_{2yy^*}N_s(X_2)T_{3yy^*}) - \\
& - Re\{d_s(X_2)\}\gamma]
\end{aligned}$$

et le dernier terme est égal à la partie droite de la condition (5.50).

□

Remarque 5.10 : La constante γ indique le niveau acceptable de la différence entre les deux matrices de transfert. Elle n'est pas nécessaire si la

fonction $\phi_{tBF}(X)$ sert comme un des objectifs de l'optimisation, dans ce cas elle peut être mise à zéro et la fonction devient suffisante et nécessaire (non-conservatrice). Néanmoins, la fonction restera toujours déformée (voir Remarque 5.9).

Les conditions de poursuite (5.46) ou (5.47) peuvent être définies aussi pour une fonction de transfert de la boucle fermée, comme dans le cas des conditions de calibrage (5.34) et (5.35). Il suffi de remplacer la valeur singulière $\bar{\sigma}(\cdot)$ par le module $|\cdot|$ et multiplier le module avec deux vecteurs-sélecteurs v_1, v_2 (voir l'explication pour les conditions (5.34) et (5.35)). Les deux conditions équivalentes de poursuite (5.46) et (5.47) devient :

$$v_1^T |dT_{1yy^*} + T_{2yy^*}NT_{3yy^*} - dM_{yy^*}| v_2 - Re\{d\}\gamma \leq 0 \quad \forall \omega_d \quad (5.53)$$

et

$$v_1^T |dS_{yy^*} - dM_{yy^*}| v_2 - Re\{d\}\gamma \leq 0 \quad \forall \omega_d \quad (5.54)$$

A partir des conditions (5.53), (5.54), les fonctions critère convexes de poursuite pour une fonction de transfert de boucle fermée ont les formes suivantes :

$$\begin{aligned} \phi_{yy^*v_1v_2}(X) = \sup_{\omega_d} [v_1^T |d_s(X)(T_{1yy^*} - M_{yy^*}) + T_{2yy^*}N_s(X)T_{3yy^*}| v_2 - \\ - Re\{d_s(X)\}\gamma] \end{aligned} \quad (5.55)$$

et

$$\phi_{yy^*v_1v_2}(X) = \sup_{\omega_d} [v_1^T |d_s(X)S_{yy^*} - d_s(X)M_{yy^*}| v_2 - Re\{d_s(X)\}\gamma] \quad (5.56)$$

Les deux fonctions critère sont convexes par rapport au vecteur d'optimisation X . Le preuve est le même comme pour la fonction critère (5.48) (Preuve 5.2, pour l'explication voir Remarque 5.6).

5.2.3 δ -stabilité de la boucle fermée

Dans la boucle fermée avec le régulateur optimisé, il y a trois types des pôles :

- *Pôles de l'observateur* - déterminés par la matrice d'observateur L et définis par les valeurs propres de la matrice $(A_H - LC_H)$
- *Pôles de retour d'état* - déterminés par la matrice de retour d'état F et définis par la matrice $(A_H - B_H F)$
- *Pôles optimisés* - définies par les racines du polynôme $d(z^{-1})$ de l'équation (5.5). Ils sont également les pôles de la matrice $Q(z^{-1})$ (paramètre libre de Q-paramétrisation).

Les pôles d'observateur et de retour d'état sont fixés lors du calcul du régulateur central $K_0(z^{-1})$. Ainsi, pour assurer la δ -stabilité, il suffit de borner l'emplacement des pôles optimisés de $d(z^{-1})$. Rappelons déjà que la vérification de la condition (5.32) sur le calibrage de sensibilités assure la stabilité du polynôme $d(z^{-1})$, car la condition :

$$-Re\{d(z^{-1})\} < 0 \quad (5.57)$$

est remplie et elle garantit d'avoir $d(z^{-1})$ strictement positif réel (une condition suffisante de la stabilité).

Remarque 5.11 : Nous avons remarqué, que la vérification de la condition (5.57) a garanti dans tous les cas rencontrés la stabilité du régulateur $K(z^{-1})$. Cela est une propriété importante, car en pratique, sauf des cas "pathologiques" où le procédé ne peut pas être stabilisé par un régulateur stable, il faut toujours utiliser des régulateurs stables.

Basé sur la condition (5.57) [Lan98, LL99] ont proposé deux types des bornes pour la construction du région δ :

- *Cercle avec le centre sur l'axe réel* - pour assurer une vitesse et un amortissement des pôles.
- *Droite parallèle avec l'axe imaginaire* - cette borne délimite un demi-plan gauche pour fixer une certaine vitesse minimale des pôles.

Ici nous allons considérer seulement la première borne, qui est plus appropriée pour les systèmes échantillonnés. Avec le cercle il est possible de limiter l'amortissement minimal (par la valeur du rayon de cercle) et la vi-

tesse (fréquence) minimale (par la position et le rayon du cercle) des pôles optimisés.

A partir de (5.57) nous allons définir une condition qui va garantir que tous les pôles optimisés sont dans un cercle avec le rayon r et le centre sur l'axe réel à c .

Proposition 5.3 *La condition*

$$-Re\{\bar{d}(z^{-1})\} < 0 \mid \forall \omega_d \quad (5.58)$$

où

$$\bar{d}(z^{-1}) = z^{-N_q} d(\bar{z})|_{\bar{z}=rz+c} \quad (5.59)$$

est vérifiée (est vrai) si tous les pôles optimisés (racines du dénominateur $d(z^{-1})$) sont à l'intérieur du cercle avec le centre sur l'axe réel placé à $c \in \langle r-1, 1-r \rangle$ et avec le rayon $r \in (0, 1)$. Ce cercle porte le nom "région δ " et la propriété garantie par (5.58) est la δ -stabilité des pôles optimisés.

Preuve 5.3 : Considérons un dénominateur de $Q(z^{-1})$ quelconque $d(z^{-1}) = z^{-N_q} d(z)$ avec les zéros $p_1, p_2, \dots, p_{N_q} \in C$, qui sont en même temps les racines de $d(z)$:

$$d(z^{-1}) = (1 - p_1 z^{-1})(1 - p_2 z^{-1}) \dots (1 - p_{N_q} z^{-1}) \quad (5.60)$$

où la condition (5.57) nous garantie la stabilité des racines p_1, \dots, p_{N_q} , c'est-à-dire leurs emplacement dans le cercle unitaire.

Modifions les racines $p_i \mid i = 1, \dots, N_q$ utilisant le rayon $r \in (0, 1)$ et le centre $c \in \langle r-1, 1-r \rangle$ de telle façon qu'ils devient $\bar{p}_i = \frac{p_i - c}{r}$. Définissons le polynôme $\bar{d}(z^{-1})$ contenant les racines \bar{p}_i :

$$\bar{d}(z^{-1}) = \left(1 - \frac{p_1 - c}{r} z^{-1}\right) \left(1 - \frac{p_2 - c}{r} z^{-1}\right) \dots \left(1 - \frac{p_n - c}{r} z^{-1}\right) \quad (5.61)$$

où c est la position du centre d'un cercle sur l'axe réel et r et le rayon du cercle.

En appliquant la condition (5.57) au polynôme $\bar{d}(z^{-1})$, nous obtenons la condition (5.58). Sachant que la nouvelle condition garantit que les racines $\bar{p}_i \mid i = 1, \dots, N_q$ sont à l'intérieur du cercle unitaire, c'est-à-dire :

$$|\bar{p}_i| < 1 \mid \forall i = 1, \dots, N_q \quad (5.62)$$

il est évident que la condition est aussi vérifiée si les racines originales $p_i \mid i = 1, \dots, N_q$ sont à l'intérieur du cercle avec le rayon r et avec le centre sur l'axe réel à c , car (5.62) implique :

$$|p_i - c| < r \mid i = 1, \dots, N_q \quad (5.63)$$

Il reste à démontrer que le polynôme $\bar{d}(z^{-1})$ est en effet le dénominateur $d(z^{-1})$ en opérateur modifié $\bar{z} = rz + c$ (la relation (5.59)). Dans (5.61), mettons au dénominateur commun les intérieurs de parenthèses :

$$\bar{d}(z^{-1}) = \left(\frac{rz + c - p_1}{rz} \right) \left(\frac{rz + c - p_2}{rz} \right) \dots \left(\frac{rz + c - p_{N_q}}{rz} \right) \quad (5.64)$$

Sortons le dénominateur commun rz des parenthèses et appliquons la substitution $\bar{z} = rz + c$:

$$\bar{d}(z^{-1}) = \frac{z^{-N_q}}{r^{N_q}} (\bar{z} - p_1)(\bar{z} - p_2) \dots (\bar{z} - p_{N_q}) \quad (5.65)$$

et évidemment cela peut s'écrire sous la forme :

$$\bar{d}(z^{-1}) = \frac{z^{-N_q}}{r^{N_q}} (\bar{z}^{N_q} + d_1 \bar{z}^{N_q-1} + \dots + d_{N_q}) \quad (5.66)$$

c'est qui correspond au terme (5.59) de la Proposition 5.3 :

$$\bar{d}(z^{-1}) = z^{-N_q} d(\bar{z}) \quad (5.67)$$

avec $d(\bar{z}) = \frac{1}{r^{N_q}} (\bar{z}^{N_q} + d_1 \bar{z}^{N_q-1} + \dots + d_{N_q})$.

Ainsi la condition (5.58) vérifiée garanti que les racines de $d(\bar{z}) \mid \bar{z}=rz+c$ sont à l'intérieur du cercle défini par c et r .

□

Comme dans le cas des conditions pour le calibrage de sensibilités, une fonction critère ϕ_δ est construite à partir de la condition (5.58) :

$$\phi_\delta(X) = \sup_{\omega_d} [-Re\{\bar{d}_{s/r}(X)\}] \quad (5.68)$$

le terme $\bar{d}_{s/r}(X)$ signifie que le polynôme $d(z^{-1})$ a la structure décrite par (5.7) ou la structure (5.9). La fonction $\phi_\delta(X)$ est convexe par rapport au vecteur X et l'évaluation de cette fonction est la même comme dans le cas de la fonction (5.36) de calibrage de sensibilités (évaluation sur une grille des points de fréquence).

Remarque 5.12 : A partir du Preuve 5.3 il est aussi évident que, si l'opérateur $z \equiv e^{j\omega_d}$ est considéré à la place de $z^{-1} \equiv e^{-j\omega_d}$ dans (5.58), (5.59), les deux relations sont remplacées par :

$$\begin{aligned} -Re\{\bar{d}(z)\} &< 0 \mid \forall \omega_d \\ \bar{d}(z) &= d(\bar{z})|_{\bar{z}=rz+c} \end{aligned}$$

5.2.4 Fonctions multi-critère

Pour traiter plusieurs spécifications contradictoires de synthèse, il faut souvent optimiser plusieurs fonctions critère de la forme (5.38), (5.49) ou (5.68) simultanément. Une telle optimisation porte le nom d'optimisation multi-objectifs. Pour cette optimisation, il faut une fonction qui ressemble plusieurs fonctions critère. Dans notre cas nous utilisons la fonction $\max(\cdot)$ qui conserve la convexité sous condition que toutes les fonctions introduites soient convexes. La fonction multi-critère est définie comme suit :

$$\phi_{mc}(X) = \max_i [\alpha_i \phi_i(X)] \quad (5.69)$$

où l'indice i correspond à δ , yp , up , yb ... selon la fonction critère introduite. Le coefficient $\alpha_i \in (0, 1)$ est une valeur de pondération attribuant les poids aux fonctions critère pour équilibrer les valeurs des critères.

Exemple : Typiquement, pour les modèles avec un gain non-unitaire faible⁴, le régulateur et en conséquence aussi la sensibilité S_{up} ont des modules élevés (d'ordre 10, 100, ...) par rapport au modules des sensibilité S_{yp} , S_{yb} (d'ordre 1 pour des systèmes robuste). Dans ce cas, pour optimiser tout les trois sensibilités simultanément, il faut donner plus de poids aux sensibilités avec des modules faibles.

5.2.5 Algorithme d'optimisation

L'algorithme d'optimisation convexe utilisé est un algorithme ellipsoïde classique présenté par exemple dans [BB91] et utilisé aussi dans l'approche SISO [Lan98, LL99]. Il est basé sur le calcul des gradients et il permet d'optimiser une fonction convexe d'objectif sous contrainte d'une autre fonction convexe (fonction de contrainte). Pour l'optimisation de nos fonctions critère nous disposons donc de deux fonctions multi-objectif à minimiser :

- *Fonction d'objectif* $\phi_{OBJ}(X)$ - elle est minimisée sous la condition que la fonction de contrainte est négative.
- *Fonction de contrainte* $\phi_{CST}(X)$ - pendant l'optimisation de fonction d'objectif ϕ_{OBJ} elle doit être négative, si ce n'est pas le cas elle est minimisée à la place de ϕ_{OBJ} pour la rendre négative.

La procédure d'optimisation d'une fonction critère (d'objectif ou de contrainte) comporte les étapes suivantes :

1. Un ellipsoïde initial E est défini. La taille est donnée par une matrice diagonale A , le centre est un vecteur $X_c \in R^l$. Cet ellipsoïde doit contenir le minimum recherché, ainsi il faut surtout choisir la taille suffisamment grande (choix général est $A = \text{diag}\{1000\}$). Le centre est généralement placé aléatoirement mais proche d'origine.
2. Le gradient $g_\phi(X_c)$ est calculé au centre d'ellipsoïde.
3. Le centre X_c est déplacé selon la direction du gradient et une nouvelle

⁴Le modèle qui a la grandeur des signaux de sortie $y(t)$ largement inférieur à la grandeur des signaux de l'entrée $u(t)$ ($\|y(t)\| \ll \|u(t)\|$)

taille A d'ellipsoïde, plus petite que l'ancienne, est calculée. Le nouveau ellipsoïde devrait toujours contenir le minimum.

4. Un critère d'arrêt est testé si le centre X_c est déjà suffisamment proche du minimum ou si le nombre maximal d'itérations est dépassé. Si ce n'est pas le cas, la procédure revient au pas 2.

La Fig. 5.2 montre un exemple d'évolution de la valeur de fonction d'objectif $\phi_{OBJ}(X_c)$ lors de l'optimisation. Chaque fois quand le centre X_c est déplacé les contraintes sont vérifiées et la fonction d'objectif est évaluée avec ce nouveau centre. Comme l'optimisation avance, le centre s'approche plus en plus à l'optimum est finalement quand la taille d'ellipsoïde est très petite il semble que le centre ne bouge plus et que la valeurs de $\phi_{OBJ}(X_c)$ reste inchangée.

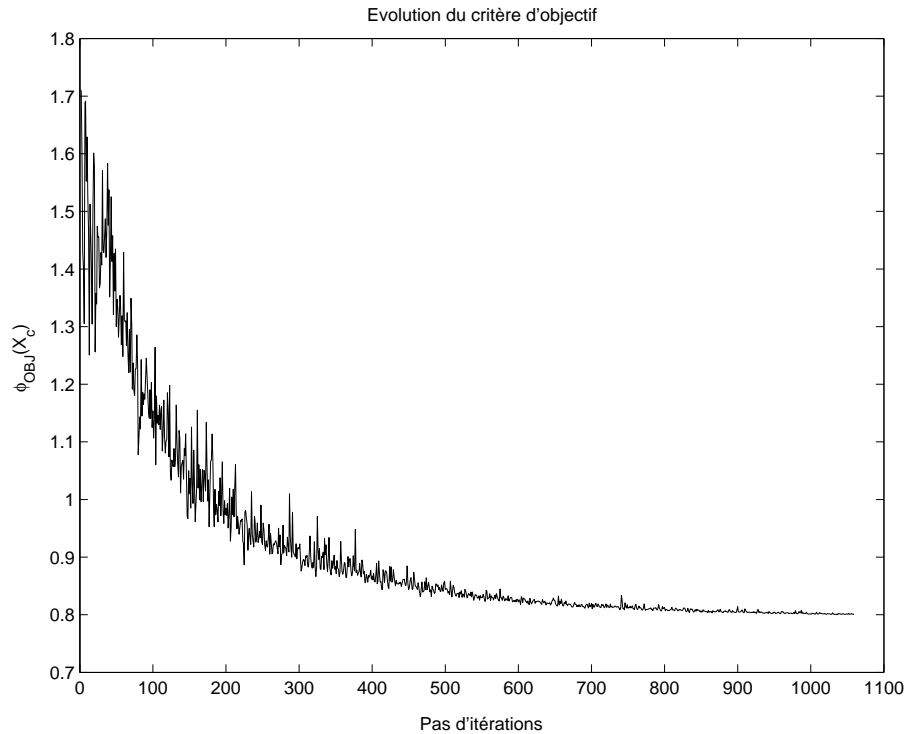


FIG. 5.2 – L'évolution de la valeur de fonction d'objectif lors de l'optimisation.

Le gradient $g_\phi(X_p)$ de la fonction critère $\phi_{mc}(X)$ dans le point $X_p \in R^l$ est évalué avec une fraction suffisamment petite $dX = [dx, dx \dots, dx]^T$ de

l'espace. Ainsi, le gradient est donnée par la relation suivante :

$$g_\phi(X_1) = \frac{1}{dx} \begin{bmatrix} \phi_{mc}(X_p + dX_1) \\ \phi_{mc}(X_p + dX_2) \\ \vdots \\ \phi_{mc}(X_p + dX_l) \end{bmatrix} \quad (5.70)$$

où dX_i est un vecteur avec tous ses éléments misent à zéros sauf i^e élément qui est égale à dx . La taille de dx devrait être plus petite possible, mais il faut éviter les erreurs de calculs numérique. Généralement, nous utilisons la valeur $dx \leq 10^{-6}$.

5.3 Sommaire de la méthodologie - étapes de conception

La procédure complète de la synthèse de régulateur MIMO par placement de pôles avec calibrage de sensibilités par l'optimisation convexe est résumée dans la Fig.5.3. La procédure comporte les pas suivants :

1. Transformer les spécifications désirées de la boucle fermée (cahier de charges) aux spécifications de synthèse compatible avec la méthodologie développée. Comme il été montré les deux types de spécifications sont considérés :
 - Gabarits sur des sensibilités - Il faut décider quelles sensibilités seront traitées et il faut déterminer les gabarits (bornes) $F_{ij}(\omega_d)$ pour ces sensibilités. La fonction critère (5.38) de calibrage de valeur singulière ou la fonction critère (5.39) de calibrage du module d'un élément de sensibilité sont appliquées. Cette spécification est capable d'assurer la stabilité robuste, la réponse temporelle, rejet de perturbations, traitement des incertitudes, etc. Pour plus de détail sur le calibrage voir le Chapitre 4.
 - Région de δ -stabilité pour les pôles optimisés - Elle nous permet de restreindre l'amortissement et la dynamique des pôles. La fonction

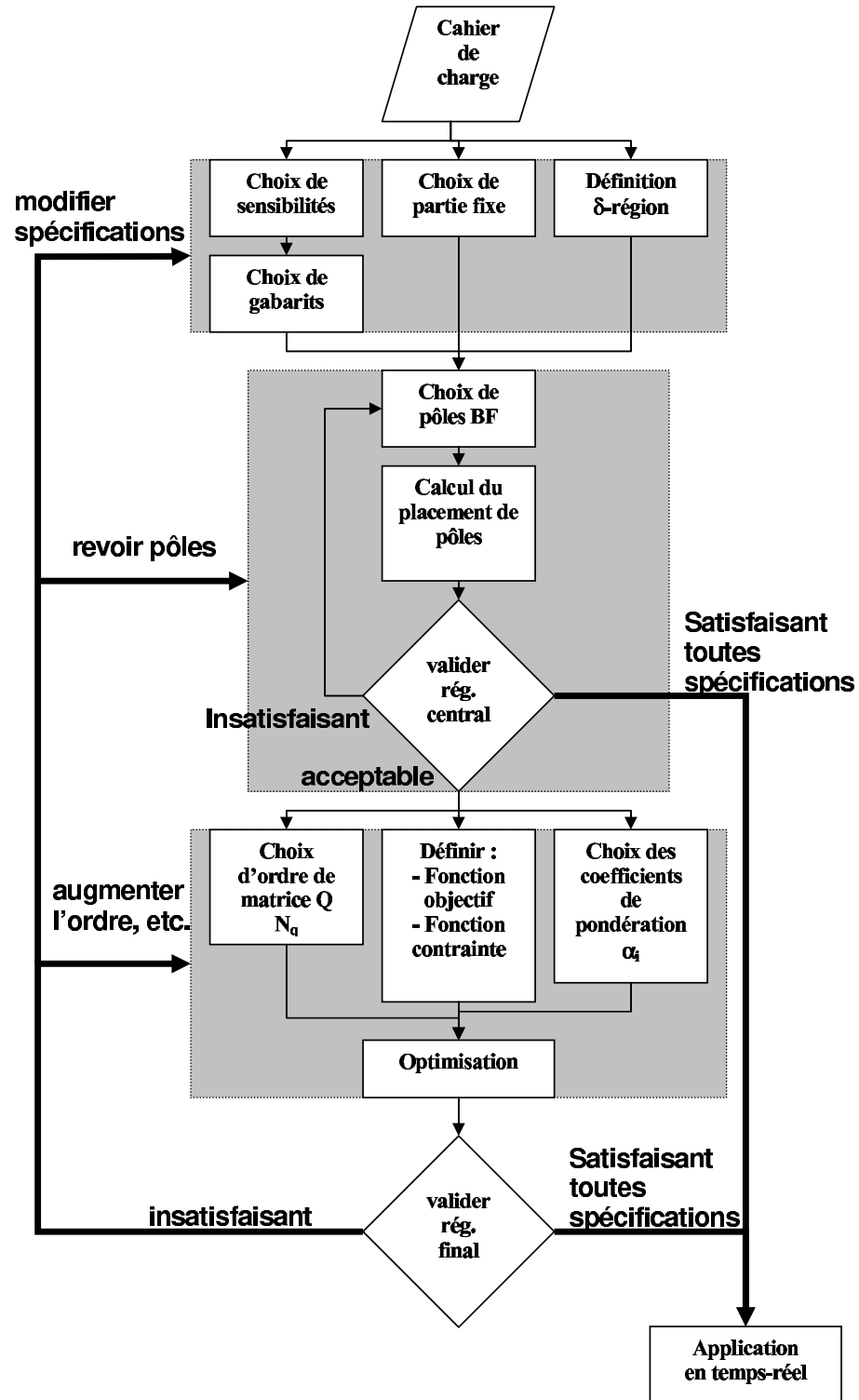


FIG. 5.3 – Procédure de la synthèse de régulateur

critère (5.68) est appliquée.

2. Choisir le bloc pré-spécifié $H_K(z^{-1})$. Quelques performances désirées peuvent être assurées en incorporant un filtre fixé dans le régulateur. Le bloc $H_K(z^{-1})$ contient souvent le modèle interne d'une perturbation ou un filtre d'ouverture de boucle à une certaine fréquence. Un exemple typique est un intégrateur MIMO pour éliminer l'erreur statique ou l'ouverture de boucle à la plus haute fréquence $\frac{f_c}{2}$ (voir aussi le Chapitre 4).
3. Calculer le régulateur central $K_0(z^{-1})$. Les performances de ce régulateur influencent radicalement les performances finales du régulateur optimisé. Pour obtenir un régulateur optimisé satisfaisant (performant et pas très complexe), les performances du régulateur $K_0(z^{-1})$ devraient refléter les spécifications désirées.

La méthode de calcul est dans notre cas le placement de pôles multivariable avec l'optimisation des vecteurs propres (voir le Chapitre 4). Ainsi, il faut déterminer les positions désirées des pôles de la boucle fermée (des pôles d'observateur et des pôles de retour d'état). À noter que d'autres méthodes de la synthèse de régulateur peuvent être appliquées comme LQG, placement de pôles et des vecteurs propres, etc. sous condition que le régulateur obtenu peut s'écrire sous la forme d'observateur avec le retour d'état.

4. Définir des paramètres d'optimisation. Il faut d'abord décider quelles fonctions critère seront considérées comme contrainte et quelles seront considérées comme objectif. Ensuite, les coefficients de pondération α_i dans (5.69) doivent être choisis (si plusieurs fonctions critère sont optimisées). Aussi, l'ordre N_q de paramétrisation (5.7) ou (5.9) doit être fixé. En général on commence avec $N_q = 1$ et on augmente l'ordre jusqu'à ce que les résultats soient satisfaisants. Concernant l'algorithme d'optimisation, il faut choisir la grandeur de l'ellipsoïde initial, la taille du gradient et un critère d'arrêt.
5. Appliquer l'optimisation et vérifier le régulateur obtenu. Si les résultats

ne sont pas satisfaisant, revenir aux pas précédents.

5.4 Comparaison avec la commande H_∞

Supposons le problème général de la synthèse de régulateurs par la commande H_∞ comme il est défini dans le schéma Fig.5.4. Les blocs G et K correspondent respectivement au modèle du procédé et au régulateur de la commande H_∞ . Une optimisation H_∞ cherche un régulateur K tel que le maximum de la plus grande valeur singulière de la matrice M_{BF} où $[y, u]^T = M_{BF}[p, d, b, r]^T$ soit minimal. Les éléments W_i sont les matrices (filtres) de pondération qui permettent d'ajuster le résultat de l'optimisation et de ce fait le régulateur obtenu. En conséquence les matrices de pondération doivent refléter les spécifications désirées de performance. Or, la procédure de la synthèse par la commande H_∞ est un processus itératif qui consiste en l'ajustement des filtres W_i et de l'optimisation effectuée après chaque ajustement. Chaque fois le régulateur obtenu est étudié, et si il ne satisfait pas les spécifications les pondérations W_i sont réajustées.

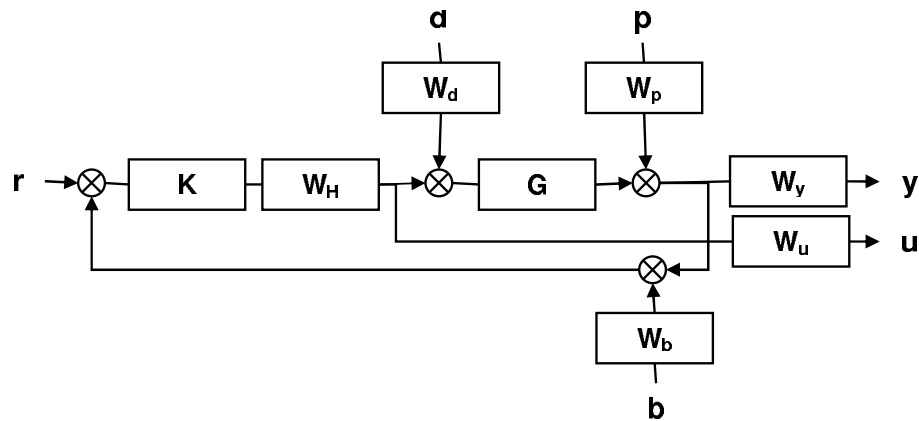


FIG. 5.4 – Configuration de synthèse par la commande H_∞

Les similitudes et les différences entre la synthèse par la commande H_∞ et par le calibrage de sensibilités avec optimisation convexe (C.S.O.C.) sont les suivantes :

- Pour C.S.O.C. la configuration du système bouclé est proche de la configuration H_∞ (Fig. 5.4). Néanmoins, les filtres de pondération W_y , W_u , W_d , W_p , et W_b sont remplacés par les gabarits sur les sensibilités correspondants comme suit :

$$\begin{aligned}
W_y, W_d &\approx F_{yd} \\
W_y, W_p &\approx F_{yp} \\
W_y, W_b &\approx F_{yb} \\
W_u, W_d &\approx F_{ud} \\
W_u, W_p &\approx F_{up} \\
W_u, W_b &\approx F_{ub}
\end{aligned} \tag{5.71}$$

En plus, dans C.S.O.C. les filtres sur les entrées peuvent apparaître quand même, mais comme des modèles des perturbations. La pondération W_H correspond à la matrice H_K , mais l'ajustement du gain de H_K n'est pas nécessaire, car l'équilibre des gains des sensibilités peut être obtenu avec les coefficients de pondération α_{ij} .

- La fonction critère à minimiser de la commande H_∞ peut être décrite comme suit :

$$\phi_{H_\infty} = \left\| \begin{array}{ccc} W_y S_{yd} W_d & W_y S_{yp} W_p & W_y S_{yb} W_b \\ W_u S_{ud} W_d & W_u S_{up} W_p & W_u S_{ub} W_b \end{array} \right\|_\infty - \gamma \tag{5.72}$$

avec la constante γ suffisamment petite. De l'autre côté, pour C.S.O.C. nous avons deux fonctions de critère suivantes⁵ :

$$\phi_{OBJ}(X) = \max_{ij} [\alpha_{ij} \phi_{ij}(X)] \tag{5.73}$$

$$\phi_{CST}(X) = \max_{ij} [\alpha_{ij} \phi_{ij}(X)] \tag{5.74}$$

où les indices i et j représentent les sorties et les entrées appropriées. La fonction $\phi_{ij}(X)$ peut avoir la forme (5.38), (5.39), ou (5.68).

- Le gabarit (borne) sur le module ou valeur singulière d'une sensibilité est très proche à la pratique de l'ingénierie où typiquement les

⁵La fonction critère de contrainte et la fonction critère d'objectif

spécifications de performance limite ces valeurs. Par contre dans le cas de la commande H_∞ le lien entre les spécifications et les pondérations est beaucoup moins évident. Les gabarits sur les sensibilités sont souvent utilisés pour trouver les pondérations de H_∞ .

- Il n'est pas nécessaire d'avoir une expression rationnelle des gabarits, il suffit un ensemble des points dans le domaine fréquentiels. De l'autre coté, la commande H_∞ demande d'exprimer pondérations fréquentielles sous forme des matrices de transfert rationnelles.
- Dans la synthèse C.S.O.C., chaque sensibilité a son propre gabarit. En plus, n'importe quelle combinaison des sensibilités bornées est possible. La commande H_∞ est beaucoup moins flexible de ce point de vue.
- Le choix des coefficients de pondération α_{ij} (important surtout pour la fonction d'objectif) correspond approximativement (dans la synthèse par la commande H_∞) au choix du gain de filtres de pondération. Dans les deux cas le but est de créer un équilibre entre des différents critères de sensibilité.

Exemple : Dans le cas du régulateur avec un gain élevé ($\gg 1$), la sensibilité S_{up} a son module aussi élevé. Par contre S_{yp} (dans le cas de la commande robuste) a son module toujours d'ordre 1. Sans ajouter les coefficients α_{ij} et sans des modifications du gain de filtres de pondération, la sensibilité S_{up} va dominer dans les fonctions de critère (5.72), (5.73), et (5.74) et en conséquence la robustesse (décrite entre autre par S_{yp}) est très défavorisée. Pour l'éviter, il suffit, par exemple, de fixer α_{up} suffisamment petit ou dans le cas de la commande H_∞ d'ajuster le gain de W_H .

- Avec (5.39) un gabarit fréquentiel peut être imposé sur une fonction (un élément) d'une matrice de sensibilité S_{ij} et cela n'est pas possible dans le cas de la commande H_∞ . Par exemple pour imposer le découplage, il suffit de fixer un ensemble de gabarits bas sur tous les éléments non-diagonales de S_{yb} .
- La fonction de critère $\phi_{ij}(X)$ est plus conservatrice que la fonction de

critère de la commande H_∞ .

- La complexité du régulateur peut être augmentée par une simple incrémentation de l'ordre de paramétrisation N_q dans (5.7) ou (5.9), si des spécifications de la synthèse le demandent. Dans la synthèse H_∞ , les filtres de pondération doivent être modifiés dans ce cas ce qui est moins logique et plus compliqué.
- La paramétrisation de C.S.O.C. a besoin un régulateur central, ainsi avant l'optimisation ce régulateur doit être calculé. La complexité du régulateur central (nombre des états) est $\deg(K_0) = \deg(G) + 2\deg(H_K)$ et cela correspond au régulateur H_∞ calculé avec des filtres de pondération dont la somme des degrés est $\sum_k \deg(W_k) = 2\deg(H_K)$. Dans la procédure de C.S.O.C. la matrice H_K contient en générale seulement l'intégrateur. Dans ce cas, la valeur est relativement petite $2\deg(H_K) = 2n_u$.
- La méthode de C.S.O.C. utilise directement la représentation échantillonnée du modèle de procédé. Cela permet d'utiliser les modèles échantillonnés identifiés par des algorithmes classiques et en plus le régulateur obtenu est aussi numérique. En conséquences, nous évitons la discrétisation du régulateur qui conduit, en général, à une fréquence d'échantillonnage très élevée et demande une plus grande puissance de calcul [Lan02].
- En comparant de une façon générale les méthodes de C.S.O.C. et de la commande H_∞ , la méthode de C.S.O.C. remplace le choix heuristique et itératif des filtres de pondération par une procédure automatisée de l'optimisation convexe. Néanmoins, un régulateur central doit être calculé d'abord. Il est évident, qu'un tel approche est utile seulement pour des problèmes complexes où un régulateur simple ne peut pas satisfaire toutes les spécifications de synthèse.

5.5 Exemples de synthèse

5.5.1 Réacteur parfaitement agité (Stirred tank reactor)

Le procédé est un réacteur qui comporte un bac avec un liquide agité où le bac est refroidit de l'extérieur par un autre liquide. Le modèle échantillonné du procédé a été utilisé dans [CB95] pour illustrer la commande prédictive avec la période d'échantillonnage $T_e = 0.01$ min. (le modèle continue original est avec l'opérateur $s = j\omega$ en rad/min.). Le modèle a 2 entrées (le débit du liquide dans le réacteur et le débit du liquide refroidissant), 2 sorties (écoulement du liquide de réacteur et la température dans le réacteur) et 4 états. Il y a 4 pôles réels à 0.90484, 0.95805, 0.92774, et 0.94176.

Selon les résultats obtenus dans [CB95] nous avons définies les spécifications de synthèse suivantes :

1. Temps de montée 3 échantillons maximum⁶
2. Dépassement maximal 5% et le temps d'établissement doit être proche du temps de montée.
3. Erreur statique nulle.
4. Interaction faible 20% au maximum⁷.
5. Une bonne robustesse, c'est-à-dire $\bar{\sigma}(S_{yp})$ inférieur à 6dB en basses fréquences est à 3.5dB en hautes fréquences, et $\bar{\sigma}(S_{yb}) \leq 2$ dB.

La synthèse du régulateur comporte trois étapes - la transformation des spécifications de performances en spécifications de synthèse et en paramètres de synthèse, calcul du régulateur central, et l'optimisation du régulateur complet (voir Fig. 5.3).

⁶Il s'agit d'un choix limite de la période d'échantillonnage, une période plus courte serait plus appropriée.

⁷Pour la définition de l'interaction voir le Chapitre 6

Transformation de spécifications de performance en spécifications de synthèse

Il faut passer par les trois étapes suivantes :

- *Choix des sensibilités et gabarits* -
 - Sensibilité S_{yp} (fonction critère $\phi_{yp}(X)$) : La sensibilité S_{yp} sera traitée pour assurer la robustesse (spécification 5) et aussi pour obtenir la vitesse désirée de la réponse en échelon (spécification 1). Le gabarit correspondant est montré dans la Fig. 5.5 (le trait épais en tirets dans le graphe de $\bar{\sigma}(S_{yp})$). La rampe en basses fréquences garanti la vitesse, le reste permet de vérifier la robustesse demandée.
 - Sensibilité S_{up} (fonction critère $\phi_{up}(X)$) : Nous allons utiliser cette sensibilité pour limiter le gain du régulateur. Néanmoins, il est difficile d'estimer le gabarit approprié à partir des spécification de synthèse. Ce gabarit dépend principalement du gain de modèle et des performances de la boucle fermée. Le gabarits de cette sensibilité sera choisi au cours de synthèse du régulateur central est ajuster lors de l'optimisation si la commande n'est pas satisfaisante.
 - Sensibilité S_{yy^*} (fonction critère $\phi_{yy^*v1v2_{i,i}}(X) \mid i = 1, 2$) : Pour obtenir une poursuite désirée, la sensibilité $S_{yy^*} = -S_{yb}$ est ajustée. Pour l'ajustement, la fonction critère de poursuite de la boucle fermée sera employée. En conséquence, il faut définir un modèle de poursuite M_{yy^*} qui reflète les performances désirées en poursuite. Nous allons traiter séparément les fonctions de transfert diagonales $S_{yy^*_{i,i}}(z^{-1})$, $i = 1, 2$ de la matrice de sensibilité $S_{yy^*}(z^{-1})$ et le modèle de poursuite pour toutes les fonctions diagonales aura la forme suivante :

$$m_{yy^*_{i,i}}(z^{-1}) = \frac{0.38993}{1 - 0.85484z^{-1} + 0.244771z^{-2}} \quad (5.75)$$

où le dénominateur contient une paire de pôles complexes à 14Hz avec l'amortissement $\zeta = 0.8$ et le gain statique est égal à 1 (pour la réponse fréquentielle voir la Fig. 5.9, le temps de montée de ce système est bien 3 échantillons).

- Sensibilité S_{yb} (fonction critère $\phi_{ybv1v2_{i,j}}(X) \mid i \neq j$) : La spécification 4, impose une interaction maximale de 20%, cela impose un gabarit supérieur sur le module des fonctions de transfert non-diagonales de la matrice $S_{yb}(z^{-1})$. La valeur maximale du module doit être $\leq 0.2 \approx 20 \log_{10}(0.2) \approx -14\text{dB}$. Notons, que l'interaction peut être traitée aussi par la fonction critère de poursuite (sensibilité S_{yy^*}).
- *Choix de région de δ -stabilité* - Il n'y a pas des contraintes sur les amortissements ou sur les fréquences naturelles des pôles de la boucle fermée. En conséquence, la région de δ -stabilité comportera tout le région de stabilité = cercle d'unité avec $c = 0, r = 1$.
- *Choix de partie fixe H_K* - Pour vérifier la spécification 3, l'intégrateur MIMO est placé dans la matrice fixe $H_K(z^{-1})$:

$$H_K(z^{-1}) = \begin{bmatrix} \frac{1}{(1-z^{-1})} & 0 \\ 0 & \frac{1}{(1-z^{-1})} \end{bmatrix} \quad (5.76)$$

Synthèse du régulateur central K_0

La procédure de placement de pôles avec le calibrage de sensibilités du Chapitre 4 est employée.

- *Placement de pôles 1* - Nous allons choisir les deux types de pôles désirés de la boucle fermée comme suit :
 - Pôles de l'observateur : Pour garantir la robustesse, les pôles du modèle sont placé comme les pôles désirés de l'observateur.
 - Pôles de retour d'état : Les deux premières spécifications correspondent à une réponse à échelon d'un système du 2^e ordre avec les pôles entre $14 \div 18\text{Hz}$ et avec l'amortissement entre $\zeta = 0.8 \div 1$. Nous allons utilisé 2 paires de pôles à 16Hz avec $\zeta = 0.8$ comme pôles désirés de retour d'état.

Les 4 pôles libres sont placés dans la phase initiale proche de l'origine et ainsi leurs influence peut être négligée. Le régulateur, calculé par la méthode de Kautsky [KND85], a de très mauvaises propriétés, les maximums de $\bar{\sigma}(S_{yp})$ et $\bar{\sigma}(S_{yb})$ sont très élevés (voir Fig. 5.5 - le trait

en pointillés).

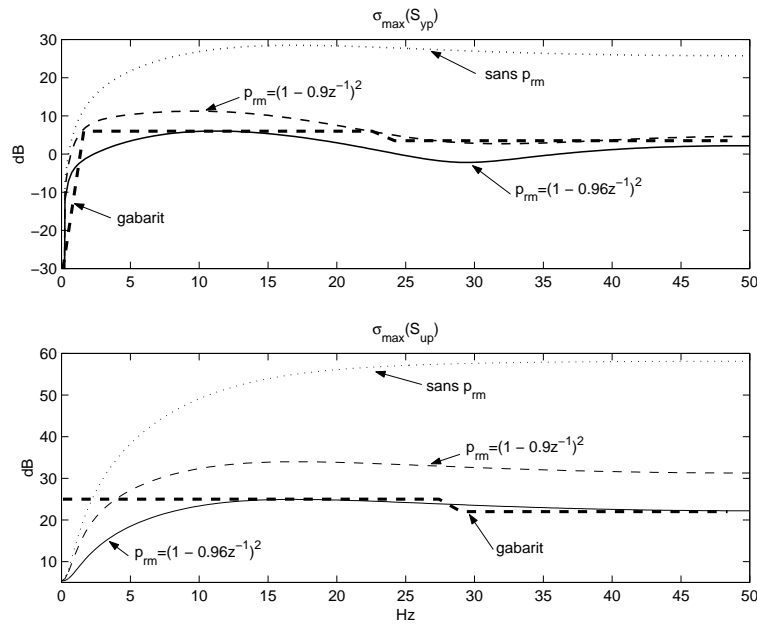


FIG. 5.5 – Les réponses fréquentielles de $\bar{\sigma}(S_{yp})$ et $\bar{\sigma}(S_{up})$ pour le régulateur central

- *Placement de pôles 2* - Pour améliorer la robustesse du régulateur nous allons placer un pôle réel multiple en basses fréquences comme pôle de retour d'état. D'un coté il va ralentir la réponse de BF, de l'autre coté il va atténuer les hautes fréquences. La multiplicité du pôle est $N = 2$ qui correspond au nombre de sorties du système n_y (voir le Chapitre 4 pour l'explication). Deux placements différents du pôle sont présentés, un à 0.9 : $p_{rm}(z^{-1}) = (1 - 0.9z^{-1})^2$, l'autre à 0.96 : $p_{rm}(z^{-1}) = (1 - 0.96z^{-1})^2$. Les sensibilités qui résultent de ce deux choix différents sont montrées dans la Fig.5.5 (le trait en tirets et le trait plein). A noter que le deuxième choix donne les meilleurs résultats en robustesse et il sera utiliser dans la suite.
- *Choix de gabarit sur S_{up}* - Avec le régulateur central obtenu, la sensibilité S_{up} a le maximum de la valeur singulière $\bar{\sigma}(S_{up}(z^{-1}))$ autour de 25dB et elle est à 22dB à la fréquence de Nyquist ($\frac{f_s}{2}$). Selon l'évolution

de $\bar{\sigma}(S_{up}(z^{-1}))$, le gabarit choisi pour cette sensibilité est à 25dB pour les fréquences basse et à 22dB pour les hautes fréquences - voir Fig. 5.5 bas, le trait épais en tirets.

Le régulateur central obtenu a la complexité (nombre des états) $deg(K_0) = 8$. Il est relativement robuste ($\|S_{yp}\|_\infty = 6\text{dB}$ et $\|S_{yb}\|_\infty = 2.3\text{dB}$), mais les propriétés en poursuite de la boucle fermée sont insatisfaisantes avec l'interaction près de 0.4, dépassement maximal pour le premier canal de 20% et le temps de montée de 0.25s pour le deuxième canal. Les réponses directes de BF sont en effet différentes pour chaque canal - le premier canal a une réponse suffisamment rapide mais avec le dépassement, le deuxième canal donne par contre une réponse trop lente, voir la Fig. 5.7. Ainsi, il faut procéder à l'optimisation.

Optimisation convexe

L'étape de l'optimisation convexe est une procédure qui consiste de : 1) définition des paramètres d'optimisation, 2) l'optimisation et 3) vérification des résultats. Dans cet exemple, nous avons procédé de la façon suivante :

- *Définition de paramètres d'optimisation* - Nous allons définir :
- Fonctions d'objectif et de contrainte - La fonction d'objectif est définie comme suit :

$$\phi_{OBJ}(X) = \max\{\phi_{yy^*v1v2_{1,1}}(X), \phi_{yy^*v1v2_{2,2}}(X)\} \quad (5.77)$$

c'est-à-dire la plus grande différence entre la sensibilité S_{yy^*} et le modèle de poursuite pour les fonctions de transfert diagonales est minimisée. La constante γ des fonctions de critère est mise à zéro.

La fonction suivante de contrainte est utilisée :

$$\phi_{CST}(X) = \max\{\phi_{up}(X), \phi_{yp}(X), \phi_\delta(X), \phi_{ybv1v2_{i,j}}\} \quad (5.78)$$

Cette fonction garantie les valeurs singulières des sensibilités S_{yp} , S_{up} sous les gabarits définis, les pôles de la boucle fermée stables, et le module des fonctions de transfert non-diagonale de S_{yb} inférieur à -14dB.

- Coefficients de pondération α_{ij} - Tous les coefficients de pondérations α_{ij} sont égales à 1. Cela est possible grâce à la fonction d'objectif uniforme (les fonctions critère dans $\phi_{OBJ}(X)$ donnent des valeurs équivalentes).
- Ordre de paramétrisation N_q - Le premier choix est $N_q = 1$, et au cours de l'optimisation l'ordre sera augmenté si les spécifications ne sont pas satisfaites.
- *Optimisation convexe 1* - L'optimisation est lancée avec l'ordre de paramétrisation $N_q = 1$ et $N_q = 2$. Pour ces valeurs de N_q , l'optimisation n'a pas pu trouver une solution qui satisfait les contraintes imposées.
- *Optimisation convexe 2* - L'ordre de paramétrisation est fixé $N_q = 3$. L'optimisation trouve une solution qui satisfait les contraintes et avec laquelle la valeur de fonction d'objectif est vers $\phi_{OBJ} \cong 0.9$. Les valeurs singulières des sensibilités S_{yp} et S_{up} sont montrées dans la Fig. 5.6. Les modules des réponses fréquentielles de sensibilité S_{yy^*} sont dans la Fig. 5.9, et nous pouvons constater que les réponses diagonales sont proches de la réponse du modèle de poursuite. En plus, les réponses non-diagonales sont largement sous le gabarit imposé pour limiter l'interaction. Les réponses à échelon pour le régulateur optimisé sont présentées dans la Fig. 5.8. Pour visualiser l'amélioration significative apporté par l'optimisation, la Fig. 5.7 montre les réponses à échelon de la boucle fermée avec le régulateur central.

Le régulateur optimisé a la complexité $deg(K) = 14$. Il est plus robuste que le régulateur central ($\|S_{yp}\|_\infty = 4.45\text{dB}$ et $\|S_{yb}\|_\infty = 0.35\text{dB}$) et aussi les performances en poursuite sont très satisfaisantes avec l'interaction maximale 0.06 et le temps de montée de 3 échantillons (0.03s). Les pôles optimisés de la boucle fermée sont les suivants : 2 paires des pôles complexes à 27.9Hz avec $\zeta = 0.2$ et 1 pôle réel multiple $(1 - 0.936z^{-1})^2$.

Les pôles et les zéros de transmission du régulateur sont dans le Tab. 5.1. Remarquons les quatre zéros réel qui correspondent exactement au pôles du modèle et qui suppriment la dynamique du modèle. Ceci est résultat de

racine	valeur abs.	ζ [-]	f_n [Hz]
PÔLES			
1.000000	1.000000	1.000000	0
1.000000	1.000000	1.000000	0
0.988850	0.988850	1.000000	0.178457
0.910974	0.910974	1.000000	1.483984
$0.942266 \pm 0.045962j$	0.943386	0.767102	1.209165
$-0.112475 \pm 0.704289j$	0.713214	0.191826	28.041177
$-0.127020 \pm 0.629282j$	0.641974	0.242905	29.039661
$-0.092130 \pm 0.282402j$	0.297050	0.541177	35.698260
$-0.114634 \pm 0.076019j$	0.137549	0.613122	51.495041
ZÉROS			
0.958710	0.958710	1.000000	0.671106
0.945140	0.945140	1.000000	0.897993
0.927824	0.927824	1.000000	1.192289
0.904986	0.904986	1.000000	1.588929
-0.243442	0.243442	0.410162	54.823797
-0.049396	0.049396	0.691569	69.222358
$0.945916 \pm 0.013573j$	0.946013	0.968167	0.912333
$-0.044077 \pm 0.592773j$	0.594410	0.301504	27.459083
$-0.171180 \pm 0.680792j$	0.701983	0.191137	29.463780

TAB. 5.1 – Pôles et zéros de transmission du régulateur optimisé

la commande à modèle interne (placement des pôles du modèle comme les pôles de la boucle fermée). Les pôles du régulateur comportent deux pôles de l'intégrateur placés à 1, 4 pôles lents dont deux réels est 8 pôles complexes très rapides autour de 30Hz et plus. Le nombre plus important des pôles rapides permet d'accélérer la boucle fermée par rapport à la dynamique du modèle.

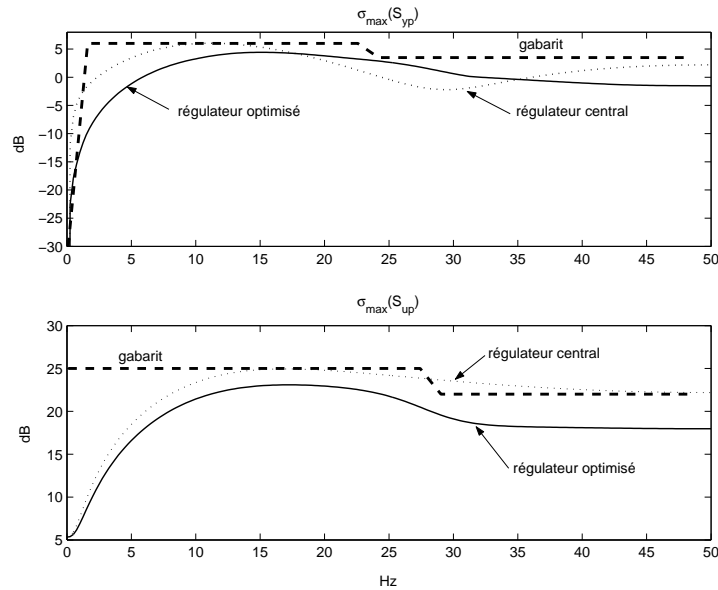


FIG. 5.6 – Les réponses fréquentielles de $\bar{\sigma}(S_{yp})$ et $\bar{\sigma}(S_{up})$ pour le régulateur central et pour le régulateur optimisé.

5.5.2 Dynamique verticale d'avion

L'autre exemple de la synthèse concerne la synthèse de régulateur robuste pour le modèle linéaire de la dynamique vertical d'avion. Le modèle et un régulateur H_{∞} vient de [Mac89]. La synthèse du régulateur par le calibrage de sensibilités et optimisation convexe est présentée dans l'Annexe B5.

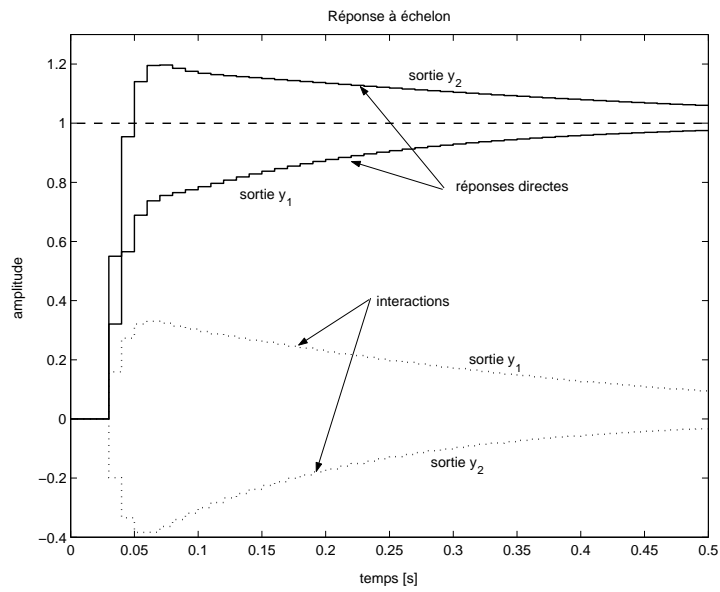


FIG. 5.7 – Les réponses à échelon du système bouclé avec le régulateur central.

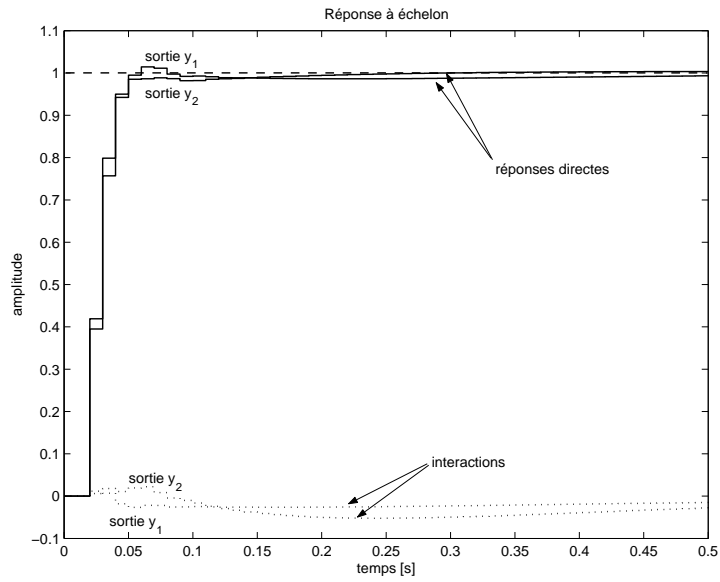


FIG. 5.8 – Les réponses à échelon du système bouclé avec le régulateur optimisé.

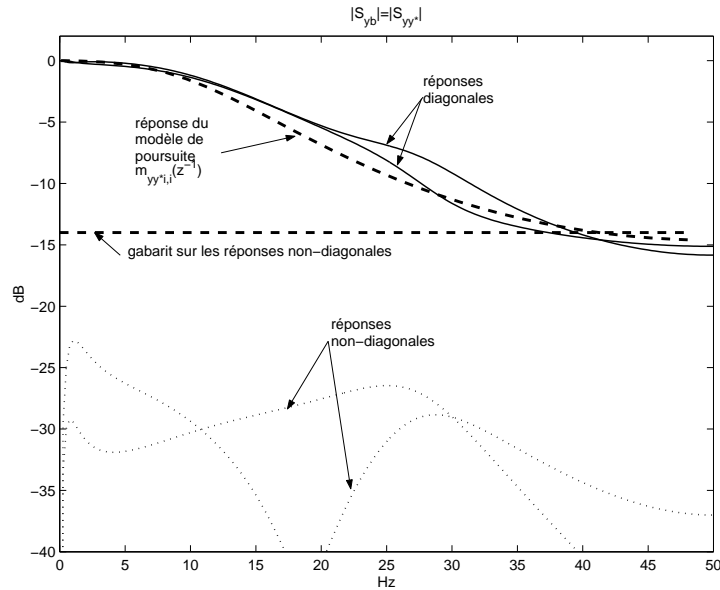


FIG. 5.9 – Les réponses fréquentielles de $|S_{yb}| = |S_{yy^*}|$ pour le régulateur optimisé.

5.6 Logiciel de synthèse RMC

Une application interactive nommée RMC (**R**obust **M**ultivariable **C**ontrol) à été développée pour la synthèse de régulateurs robustes multivariables par placement de pôles MIMO, le calibrage de sensibilités, et l'optimisation convexe dans le cadre de ce travail. L'application est une boîte à outils de MATLAB (développée pour la version 5.3) avec l'interface graphique d'utilisateur (GUI). Elle permet d'effectuer tous les étapes de la synthèse de régulateur robustes MIMO par la méthodologie décrite dans ce chapitre. En bref, l'application permet de :

- analyser le modèle échantillonné du procédé
- définir et modifier les spécifications de synthèse (gabarits, δ -région, partie fixe de régulateur)
- calculer le régulateur central K_0 par une méthode interactive
- fixer et ajuster facilement les paramètres de synthèse et les paramètres

d'optimisation

- calculer le régulateur complet par optimisation convexe
- vérifier les résultats - sensibilités, régulateurs, pôles, réponses temporels, etc. pour le modèle utilisé, et aussi pour des autres modèles éventuels.

La fenêtre principale de l'application est donnée dans la Fig.5.10. Pour la description détaillée de l'application voir l'Annexe B9 (User's Guide for RMC controller design tool).

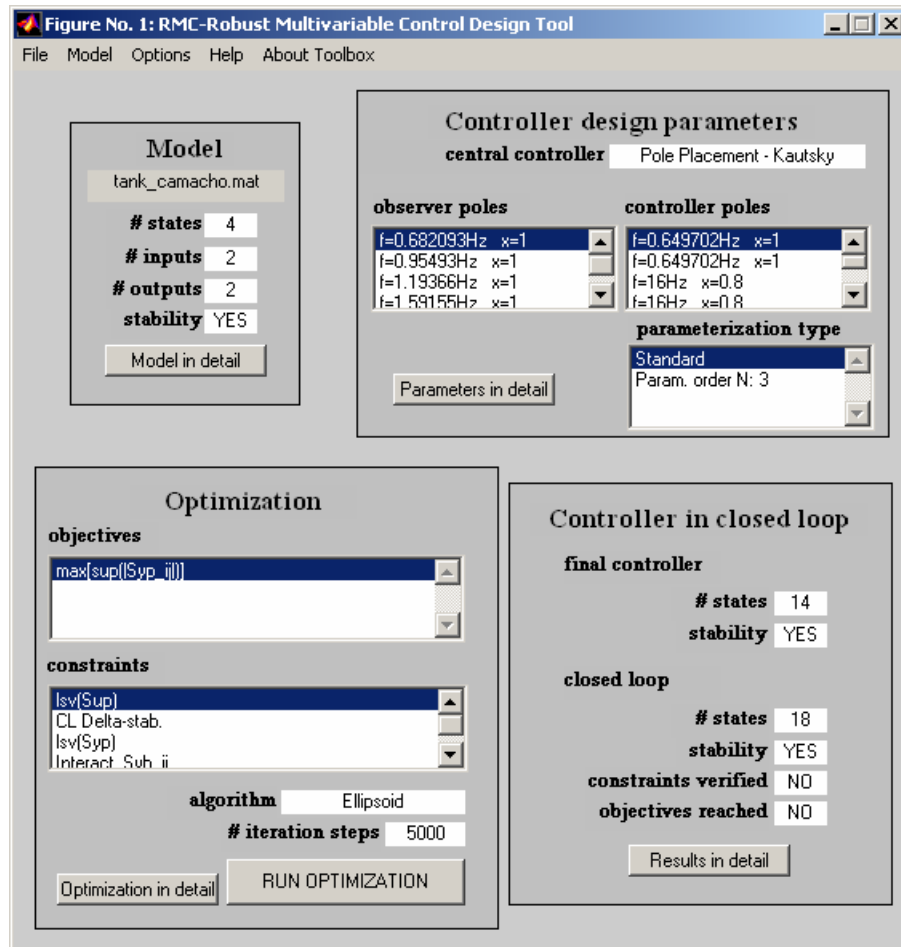


FIG. 5.10 – La fenêtre principale de l'application RMC

5.7 Notes et références

Références : La méthodologie monovariante sur laquelle est basée la procédure multivariante présentée est introduit dans [Lan98, LL99]. La théorie générale de l'optimisation convexe des régulateurs est présentée par [BB91]. Un chapitre de [Mac89] est aussi consacré à l'optimisation convexe. La (Q-)paramétrisation de "Youla-Kucera" est présentée dans un grand nombre des ouvrages comme [YJB76, Kuc79, BB91, Zho98, Mac89], etc. Le logiciel développé pour la synthèse est décrit en détail dans l'Annexe B9.

Fichiers et logiciels : La méthode développée de la synthèse de régulateurs par le calibrage de sensibilités avec l'optimisation convexe a été programmée comme la boîte à outils RMC (Robust Multivariable Control) pour Matlab. La boîte à outils RMC se trouve sur le CD de la thèse dans le répertoire `\Software\RMC_tool\`. Le guide d'utilisateur du logiciel est donné dans l'Annexe B9 et aussi dans le répertoire de la boîte à outils. D'autres m-fonctions programmées pour la commande est l'analyse des systèmes MIMO sont dans le répertoire `\M_functions\` (pour la description voir le fichier `Contents.m`). Les fichiers correspondants au exemple de la synthèse présenté dans la section 5.5.1 sont mises dans le répertoire `\Design_examples\sec_551_Bac_\`. Les fichiers du deuxième exemple (section 5.5.2) montré dans l'Annexe B5 se trouvent dans le dossier `\Design_examples\appB5_Convex_\`. Pour la description des fichiers des exemples lisez le fichier `_README.txt` dans le répertoire correspondant. Toutes les figures de graphes sont enregistrées sous format de Matlab dans le dossier `:\Figures\`. Les figures Fig. 5.2, Fig. 5.5, Fig. 5.6, Fig. 5.9, Fig. 5.7, et Fig. 5.8 correspondent respectivement aux fichiers `evol_phi.fig`, `sypsup_tank_pp.fig`, `sypsup_tank_oc2.fig`, `syb_tank_oc2.fig`, `timeresp_tank_pp.fig`, et `timeresp_tank_oc2.fig`.

References : *The monovariate methodology corresponding to the developed multivariable procedure is introduced in [Lan98, LL99]. A gene-*

ral theory of controller convex optimization is presented in [BB91]. One chapter of [Mac89] is also dedicated to convex optimization. The "Youla-Kucera" (Q-)parameterization is given in a lot of texts see for example [YJB76, Kuc79, BB91, Zho98, Mac89], etc. The developed software tool for controller design is described in detail in Appendix B9.

Files and software : The developed sensitivity shaping by convex optimization design method is programmed in this work as a Matlab toolbox named RMC (Robust Multivariable Control). The toolbox RMC is on thesis CD in the directory `\Software\RMC_tool\`. A corresponding user's guide is given in Appendix B9 and also in the toolbox directory. Other programmed m-functions for MIMO system control and analysis are in the subdirectory `\M_functions\` (for the m-functions description see `Contents.m`). The files corresponding to the presented design example of section 5.5.1 are placed in the directory `\Design_examples\sec_551_Bac_\`. The second example (section 5.5.2) presented in Appendix B5 has its files placed to the directory `\Design_examples\appB5_Convex_\`. For detailed files description see the file `_README.txt` in corresponding directory. All presented graphs are saved in Matlab format in the directory `:\Figures\`. The graphs in Fig. 5.2, Fig. 5.5, Fig. 5.6, Fig. 5.9, Fig. 5.7, and Fig. 5.8 correspond respectively to the files `evol_phi.fig`, `sypsup_tank_pp.fig`, `sypsup_tank_oc2.fig`, `syb_tank_oc2.fig`, `timeresp_tank_pp.fig`, and `timeresp_tank_oc2.fig`.

Chapitre 6

Synthèse de pré-compensateur par optimisation convexe

Le chapitre présente une procédure de synthèse de pré-compensateur par optimisation convexe. Le pré-compensateur est une matrice de transfert connectée devant la boucle fermée pour ajuster les performances en poursuite du système bouclé. Les fonctions critère convexes sont développées et une structure du pré-compensateur est proposée pour l'optimisation. La procédure de synthèse est donnée avec quelques commentaires.

This chapter presents a pre-compensator design procedure based on convex optimization. The pre-compensator is a transfer matrix connected in front of the closed loop in order to adjust some tracking performances of the closed loop. Some criterion functions are developed and a pre-compensator structure is proposed to optimize it. A design procedure is also given with some practical hints.

6.1 Régulateur à 2-degrés de liberté

Considérons la boucle fermée comme elle est présentée dans la Fig. 6.1, où G est un modèle échantillonné identifié, K est un régulateur numérique

de la boucle fermée (calculé dans les étapes précédentes) et T est la matrice de transfert numérique nommée *pré-compensateur* qui devrait garantir des performances désirées du système en poursuite.

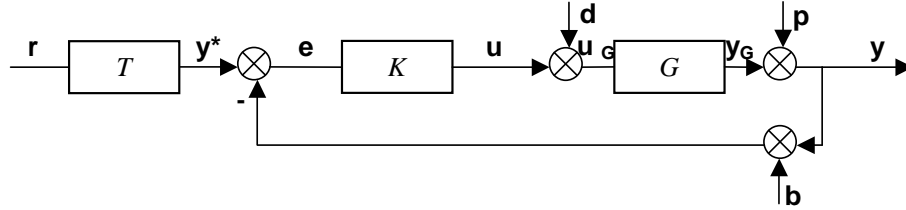


FIG. 6.1 – Configuration considérée de la boucle fermée

Les blocs T-K constituent un régulateur à 2-degrés de liberté. Ce régulateur permet de séparer les performances en boucle fermée des performances en poursuite. En d'autres termes, théoriquement il est possible d'avoir une autre dynamique (vitesse de réponse, amortissement) de la boucle fermée et une autre dynamique de l'ensemble. Cela n'est pas sans coût, le régulateur est plus complexe la partie T agrandit l'ordre de la loi de commande complète. En plus, plus les deux dynamiques sont complexes et différentes, plus la complexité du T augmente.

Dans la commande H_∞ les deux blocs K et T peuvent être optimisés simultanément [LKP93]. Dans l'approche présenté ici, la synthèse de chaque bloc est effectuée séparément, comme dans les cas monovariabiles [Lan98, Lan02]. Les chapitres précédents ont traités la synthèse du régulateur de boucle fermée K , dans la suite nous allons aborder la problématique de la synthèse du pré-compensateur T .

6.2 Structure du pré-compensateur

Supposons que le pré-compensateur a la forme suivante d'une matrice rationnelle quelconque :

$$T(z^{-1}) = d_T(z^{-1})N_T(z^{-1}) \quad (6.1)$$

où $d_T(z^{-1})$ est un polynôme contenant tous les pôles de $T(z^{-1})$ et $N_T(z^{-1})$ est la matrice polynôme. Contrairement à la paramétrisation de la matrice $Q(z^{-1})$ (5.7), (5.9), considérons que le dénominateur commun $d_T(z^{-1})$ soit fixe (déterminé selon certaines règles). Cela nous permettra d'obtenir des conditions nécessaires et suffisantes (non-conservatrices et non-déformé dans le domaine fréquentiel) pour l'optimisation de $N_T(z^{-1})$.

Par rapport au cas monovarié (voir le Chapitre 2), il manque dans la partie de la poursuite le modèle de la référence (dénommé $\frac{B_m}{A_m}$). En principe, ce modèle correspond au modèle de poursuite $M_t(z^{-1})$ utilisé dans l'optimisation (voir la section suivante).

Remarque 6.1 : Une structure du modèle $M_t(z^{-1})$ et du pré-compensateur $T(z^{-1})$ qui est proche au régulateur R-S-T de Chapitre 2 a la forme : $d_T(z^{-1}) = A_m(z^{-1})$ et

$$M_t(z^{-1}) = \text{diag} \left\{ \frac{B_m(z^{-1})}{A_m(z^{-1})} \right\} \quad (6.2)$$

La matrice $N_T(z^{-1})$ correspond au polynôme $T(z^{-1})B_m(z^{-1})$ où T est le polynôme de poursuite du régulateur R-S-T et B_m le numérateur du modèle de référence.

Pour optimiser le bloc $T(z^{-1})$ (la matrice polynôme $N_T(z^{-1})$), nous définissons une transformation du type $R^l \rightarrow H_{z_\infty}^{n_u \times n_y}$:

$$T(z^{-1}) = d_T(z^{-1})N_T(z^{-1}) \equiv Lt_{N_t}(X) : R^l \rightarrow H_{z_\infty}^{n_y \times n_y} \mid l = n_y n_y (N_q + 1) \quad (6.3)$$

où $X \in R^l$ est un vecteur d'optimisation. Le polynôme $d_T(z^{-1})$ est fixé, ainsi nous allons définir la structure de la matrice $N_T(z^{-1})$. La structure standard est utilisée (comme dans le cas du paramètre $Q(z^{-1})$ voir (5.7)) où chaque coefficient de chaque polynôme de la matrice $N_T(z^{-1})$ est un élément du vecteur X . La paramétrisation a la forme suivante :

$$\begin{aligned} X &= [x_0, x_1, x_2, \dots, x_l] \\ N_T(X) &= [n_{Tij}(z^{-1})] \mid i, j = 1, 2, \dots, n_y; \end{aligned}$$

$$\begin{aligned}
n_{Tij}(z^{-1}) &= x_k + x_{k+1}z^{-1} + \dots + x_{k+N_t}z^{-N_t} \\
k &= (i-1)n_y(N_t+1) + (j-1)(N_t+1), \\
&N_t + (i-1)n_y(N_t+1) + (j-1)(N_t+1) + 2, \dots, l
\end{aligned} \tag{6.4}$$

et N_t est l'ordre de la paramétrisation de la matrice $N_T(z^{-1})$ qui définit le nombre de zéros dans chaque polynôme de cette matrice.

Remarque 6.2 : Comme dans la paramétrisation du paramètre $Q(z^{-1})$, il est possible d'avoir dans la matrice $N_T(z^{-1})$ les polynômes des différents ordres sans perdre la convexité des fonctions critère développées dans les sections prochaines. Par exemple, les polynômes diagonaux peuvent avoir 3 zéros et les polynômes non-diagonaux juste une seule.

6.3 Spécifications de performance en poursuite

En général, les spécifications désirées de performance en poursuite sont déterminées dans le domaine temporel à partir d'une réponse à échelon. Elle peuvent être éventuellement converties dans le domaine fréquentiel par une simple fonction rationnelle du 2^e ordre. Les spécifications principales temporelles qui sont obtenues à partir d'une réponse à échelon¹ sont :

- *Temps de montée* - (rising time) Il existe plusieurs définitions. Dans notre cas nous considérons que c'est le temps entre l'instant d'application de l'échelon et l'instant lorsque le signal de la sortie atteint 90% la valeur finale donnée par le gain statique du système, tous cela moins le retard pur du système.

¹La réponse à échelon d'un système MIMO est donnée par n_u (nombre des entrées) réponses (une réponse = ensemble des signaux de la sortie) du système. Pour chaque réponse, l'échelon est appliqué à une autre entrée en maintenant les autres entrées constantes.

- *Dépassement maximal* - (maximum overshoot) Indique en % le dépassement maximal de la valeur de la sortie par rapport à la valeur finale.
- *Temps d'établissement* - (settling time) Généralement, il est défini comme le temps nécessaire pour que la valeurs de la sortie soit $\pm 5\%$ de la valeur finale
- *Découplage ou interaction* - (decoupling ou interaction) L'interaction détermine la valeur maximale de la sortie du canal où l'échelon n'a pas été appliqué.

Les spécifications en poursuite pour un système avec 2 entrées et 2 sorties et avec le gain statique égale à 1 (ainsi la valeur finale est 1) sont montrées dans la Fig. 6.2.

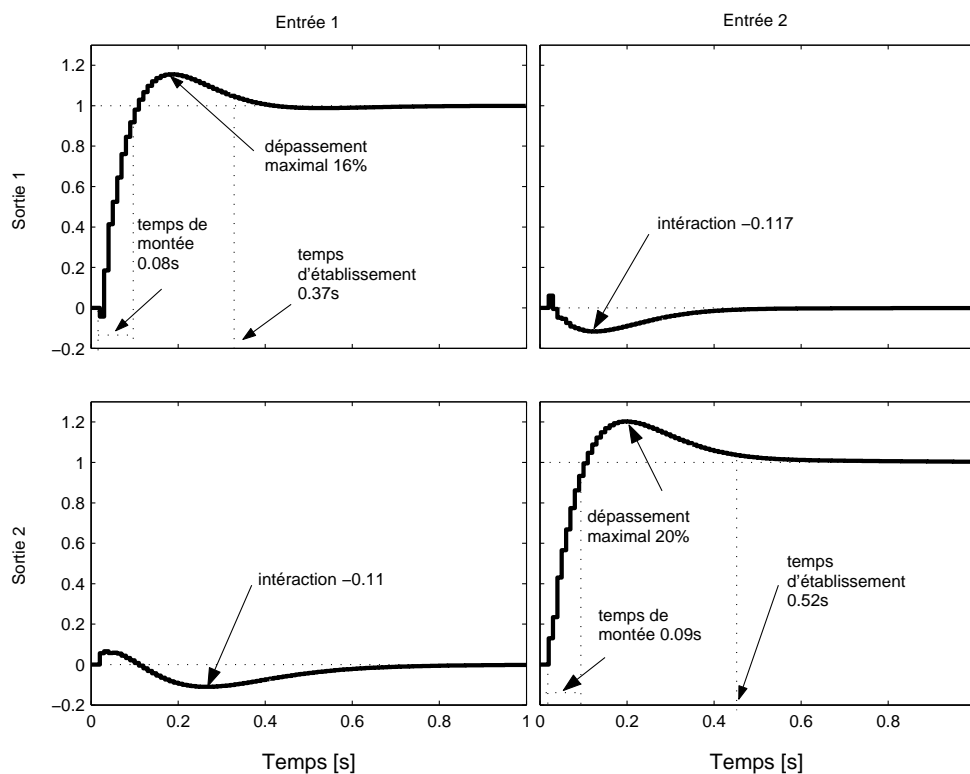


FIG. 6.2 – Spécifications en poursuite du système MIMO définies à partir de la réponse en échelon.

Pour exprimer les spécifications en poursuite nous utiliserons, comme dans la commande H_∞ [LKP93], un modèle de poursuite $M_t(z^{-1})$. Il s'agit d'une matrice de transfert qui reflète toutes les performances désirées. En général, $M_t(z^{-1})$ est une matrice de transfert diagonale constituée des fonctions de transfert simples :

$$M_t(z^{-1}) = \text{diag} \left\{ \begin{array}{ccc} \frac{N_1(z^{-1})}{D_1(z^{-1})} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{N_{n_y}(z^{-1})}{D_{n_y}(z^{-1})} \end{array} \right\} \quad (6.5)$$

où les fonctions $\frac{N_i(z^{-1})}{D_i(z^{-1})}$ correspondent aux spécifications de performance en poursuite définies pour chaque canal $i \mid i = 1, 2, \dots, n_y$. Les valeurs non-diagonales 0 signifient une interaction désirée nulle (découplage parfait).

6.4 Critères convexes de performances

Dans cette section plusieurs fonctions critère convexes sont développées pour optimiser le pré-compensateur $T(z^{-1})$. Les fonctions doivent interpréter toutes les performances possibles désirées en poursuite.

Pour toutes les fonctions critère développées ci-dessous il est supposé, que le régulateur de la boucle fermée $K(z^{-1})$ est déjà calculé. En conséquence, la boucle fermée est considérée fixée lors de l'optimisation, c'est-à-dire elle n'est pas ajustée par l'optimisation du pré-compensateur $T(z^{-1})$. Ainsi lors de l'optimisation il est possible d'utiliser des différents modèles du procédé sans toucher la convexité des fonctions critère. Cela conduit à la conclusion qu'il est possible d'optimiser le pré-compensateur simultanément pour les différents boucles fermée pour aborder des incertitudes en modèle.

6.4.1 Condition générale de poursuite

La condition générale à vérifier pour obtenir une performance en poursuite décrite par le modèle de poursuite $M_t(z^{-1})$ a la forme suivante :

$$\bar{\sigma} (S_{yy^*}(z^{-1})T(z^{-1}) - M_t(z^{-1})) \leq \gamma \mid \forall |z| = 1 \quad (6.6)$$

ou la forme avec $z^{-1} = e^{-j\omega_d}$

$$\bar{\sigma} \left(S_{yy^*}(e^{-j\omega_d})T(e^{-j\omega_d}) - M_t(e^{-j\omega_d}) \right) \leq \gamma \mid \forall \omega_d \in \langle 0, \pi \rangle \quad (6.7)$$

où le modèle de poursuite M_t représente les performances désirées du système en poursuite. La constante γ est la plus grande différence (dans le domaine fréquentiel) entre le comportement obtenu et le comportement désiré. Considérons la matrice T en forme (6.1). Si la constante γ est placée dans le terme à gauche, la condition (6.7) devient :

$$\bar{\sigma} \left(S_{yy^*}(e^{-j\omega_d})d_T^{-1}(e^{-j\omega_d})N_T(e^{-j\omega_d}) - M_t(e^{-j\omega_d}) \right) - \gamma \leq 0 \mid \forall \omega_d \in \langle 0, \pi \rangle \quad (6.8)$$

où le dénominateur commun $d_T(z^{-1})$ doit être fixé. Considérons la condition (6.8) et considérons $N_T(z^{-1})$ avec la structure (6.4). Une fonction critère correspondante peut avoir la forme suivante (la notation $(e^{-j\omega_d})$ a été supprimée) :

$$\phi_G(X) = \sup_{\omega_d} \left[\bar{\sigma} \left(S_{yy^*}d_T^{-1}N_T(X) - M_t \right) - \gamma \right] \quad (6.9)$$

Proposition 6.1 : *La fonction $\phi_G(X)$ de (6.9) est convexe par rapport au vecteur d'optimisation X .*

Preuve 6.1 : Une fonction $\phi(X) : R^l \rightarrow R$ est convexe si et seulement si [Roc82, SW70]

$$\phi(\lambda X_1 + (1 - \lambda)X_2) \leq \lambda\phi(X_1) + (1 - \lambda)\phi(X_2) \mid \forall \lambda \in \langle 0, 1 \rangle; \forall X_1, X_2 \in R^l$$

Dans notre cas la condition à vérifier a la forme générale suivante :

$$\begin{aligned} & \sup_{\omega_d} \left[\bar{\sigma} \left(S_{yy^*}d_T^{-1}N_T(\lambda X_1 + (1 - \lambda)X_2) - M_t \right) - \gamma \right] \leq \\ & \leq \lambda \sup_{\omega_d} \left[\bar{\sigma} \left(S_{yy^*}d_T^{-1}N_T(X_1) - M_t \right) - \gamma \right] + \\ & + (1 - \lambda) \sup_{\omega_d} \left[\bar{\sigma} \left(S_{yy^*}d_T^{-1}N_T(X_2) - M_t \right) - \gamma \right] \end{aligned} \quad (6.10)$$

Utilisant les propriétés de la valeur singulière et de la fonction $\sup[\cdot]$:

$$\bar{\sigma}(H_1 + H_2) \leq \bar{\sigma}(H_1) + \bar{\sigma}(H_2) \mid \forall H_1, H_2 \in C^{m \times n}$$

$$\begin{aligned}
\bar{\sigma}(\lambda H) &\leq \lambda \bar{\sigma}(H) \mid \forall H \in C^{m \times n}, \lambda \in \langle 0, 1 \rangle \\
\sup_x [f(x) + g(x)] &\leq \sup_x [f(x)] + \sup_x [g(x)] \\
\sup_x [\lambda f(x)] &= \lambda \sup_x [f(x)] \mid \lambda \in \langle 0, 1 \rangle
\end{aligned} \tag{6.11}$$

et sachant que pour $X_1, X_2, X \in R^l$ et $\lambda \in \langle 0, 1 \rangle$ les égalités suivantes sont vrai :

$$\begin{aligned}
N_T(X_1 + X_2) &= N_T(X_1) + N_T(X_2) \\
N_T(\lambda X) &= \lambda N_T(X)
\end{aligned} \tag{6.12}$$

développons la partie gauche de la condition de convexité (6.11) :

$$\begin{aligned}
&\sup_{\omega_d} [\bar{\sigma} (S_{yy^*} d_T^{-1} N_T(\lambda X_1 + (1 - \lambda) X_2) - M_t) - \gamma] = \\
&= \sup_{\omega_d} [\bar{\sigma} (\lambda S_{yy^*} d_T^{-1} N_T(X_1) + (1 - \lambda) S_{yy^*} d_T^{-1} N_T(X_2) - M_t) - \gamma] \leq \\
&\leq \sup_{\omega_d} [\bar{\sigma} (\lambda S_{yy^*} d_T^{-1} N_T(X_1) - \lambda M_t) - \lambda \gamma] + \\
&+ \sup_{\omega_d} [\bar{\sigma} ((1 - \lambda) S_{yy^*} d_T^{-1} N_T(X_2) - (1 - \lambda) M_t) - (1 - \lambda) \gamma] \leq \\
&\leq \lambda \sup_{\omega_d} [\bar{\sigma} (S_{yy^*} d_T^{-1} N_T(X_1) - M_t) - \gamma] + \\
&+ (1 - \lambda) \sup_{\omega_d} [\bar{\sigma} (S_{yy^*} d_T^{-1} N_T(X_2) - M_t) - \gamma]
\end{aligned}$$

et le terme résultant est égale à la partie droite de (6.11).

□

Remarque 6.3 : La constante γ est le seuil qui définit le niveau acceptable de différence entre la poursuite réelle et la poursuite désirée. Elle est importante si la fonction critère est utilisée comme la fonction de contrainte, autrement γ peut être zéro. Une relativement bonne similitude des poursuite désirée et poursuite réelle est obtenue pour $\gamma \leq 0.2$.

Remarque 6.4 : La condition (6.7) est utilisée dans l'approche H_∞ (dans le domaine temps-continu) pour garantir des performances en poursuite, surtout dans le cas de la synthèse de régulateurs à 2-degrés de liberté. Elle est

typiquement insérée parmi d'autres conditions sur les sensibilités pondérées. Le modèle de poursuite est généralement dans ce cas une matrice diagonale avec des fonctions de transfert du 1^{er} ou 2^e ordre sur la diagonale :

$$M_t(s) = \text{diag} \left\{ \begin{array}{ccc} \frac{g_1}{D_1(s)} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{g_{n_y}}{D_{n_y}(s)} \end{array} \right\} \quad (6.13)$$

En terme de spécification de poursuite, un modèle du type (6.13) signifie un découplage parfait (interaction nulle) et le temps de montée, le dépassement, et le temps d'établissement de canaux correspondent aux dynamiques des fonctions $\frac{g_i}{D_i(s)}$.

6.4.2 Critère pour une fonction de transfert

Pour optimiser la différence fréquentielle pour une fonction de transfert, nous allons multiplier la différence dans (6.8) à gauche et à droite par deux vecteurs-sélecteurs v_1 , v_2 comme dans le cas de la condition (5.34) et nous remplaçons la valeur singulière $\bar{\sigma}(\cdot)$ par la valeur absolue $|\cdot|$. La condition (6.8) devient :

$$|v_1^T (S_{yy^*}(e^{-j\omega_d})T(e^{-j\omega_d}) - M_t(e^{-j\omega_d})) v_2| - \gamma \leq 0 \quad \forall \omega_d \in \langle 0, \pi \rangle \quad (6.14)$$

et c'est une condition de différence entre une fonction de transfert de la matrice $S_{yy^*}(z^{-1})$ et la fonction correspondante du modèle de poursuite $M_t(z^{-1})$.

Proposons une fonction critère qui correspond à la condition (6.14) pour $N_T(z^{-1})$ avec la structure (6.4) en omettant la notation de $(e^{-j\omega_d})$:

$$\phi_{v_1 v_2 t}(X) = \sup_{\omega_d} [|v_1^T (S_{yy^*} d_T^{-1} N_T(X) - M_t) v_2| - \gamma] \quad \forall \omega_d \in \langle 0, \pi \rangle \quad (6.15)$$

qui est convexe par rapport au vecteur d'optimisation $X \in R^l$. Preuve de la convexité est similaire au cas de la condition générale de poursuite, voir aussi Remarque 5.6. Pour la constante γ voir Remarque 6.3.

6.4.3 Critère du gain statique

Pour avoir une poursuite correcte, il faut que le gain statique du système complet soit égale à 1. La matrice T est capable de modifier le gain statique du système, et pour cela nous développons une condition pour garantir un gain statique désiré. Le gain statique est le gain du système pour $\omega_d = 0$. Ainsi à partir de la condition (6.8) nous avons l'inégalité suivante :

$$\bar{\sigma}(S_{yy^*}(1)T(1) - M_t(1)) \leq \gamma \quad (6.16)$$

où généralement $M_t(1) = I$ pour avoir le découplage total dans l'état stationnaire et le gain statique égale à 1 pour chaque canal. La constante γ a une valeur suffisamment petite $\gamma < 0.05$ si la condition est considérée comme une contrainte, autrement elle peut être supposée $\gamma = 0$. La fonction critère développée à partir de la condition (6.16) et de la paramétrisation (6.4) de la matrice $T(z^{-1})$ est donnée pour $\omega_d = 0$ par :

$$\phi_g(X) = \bar{\sigma}_{\omega_d=0}(S_{yy^*}d_T^{-1}N_T(X) - M_t) - \gamma \quad (6.17)$$

La fonction critère (6.17) est en effet la condition générale de poursuite (6.8) évaluée pour une seule pulsation $\omega_d = 0$. La fonction $\phi_g(X)$ est convexe et la preuve est la même comme dans le cas de la fonction $\phi_G(X)$ (voir (6.9)), sauf que nous sommes dispensés d'évaluation du supremum de toutes pulsations ω_d .

6.4.4 Critère de robustesse

Dans la synthèse de régulateur à deux degrés de liberté par la commande H_∞ , la robustesse du pré-compensateur est traitée. Pour cela principalement deux sensibilités du système bouclé sont ajustées. La sensibilité $e(t) = S_{er}(z^{-1})r(t)$:

$$S_{er}(z^{-1}) = S_{ey^*}(z^{-1})T(z^{-1}) \equiv S_{yp}(z^{-1})T(z^{-1}) \quad (6.18)$$

qui est entre la référence et l'erreur, et la sensibilité $u(t) = S_{ur}(z^{-1})r(t)$:

$$S_{ur}(z^{-1}) = S_{uy^*}(z^{-1})T(z^{-1}) \equiv S_{up}(z^{-1})T(z^{-1}) \quad (6.19)$$

entre la référence et la commande. Remarquons que la troisième sensibilité possible qui prend en compte le pré-compensateur $T(z^{-1})$ est $y(t) = S_{yr}(z^{-1})r(t)$:

$$S_{yr}(z^{-1}) = S_{yy^*}(z^{-1})T(z^{-1}) \quad (6.20)$$

qui est la sensibilité utilisé dans les conditions de poursuite (6.8) et (6.14) pour garantir des performances en poursuite.

Les conditions de calibrage pour les sensibilités (6.18), (6.19) peuvent être définies comme suit :

$$\bar{\sigma}(\underbrace{S_{yp}(e^{-j\omega_d})T(e^{-j\omega_d})}_{S_{er}(e^{-j\omega_d})}) \leq F_{er}(\omega_d) \mid \forall \omega_d \in \langle 0, \pi \rangle \quad (6.21)$$

$$\bar{\sigma}(\underbrace{S_{ey^*}(e^{-j\omega_d})T(e^{-j\omega_d})}_{S_{ur}(e^{-j\omega_d})}) \leq F_{ur}(\omega_d) \mid \forall \omega_d \in \langle 0, \pi \rangle \quad (6.22)$$

avec les gabarits fréquentiels $F_{ur}(\omega_d)$, $F_{er}(\omega_d)$ imposés sur les valeurs singulières des réponses fréquentielles de sensibilités. La forme de ces gabarits dépend des performances désirées définies dans le domaine fréquentiel comme dans le calibrage de sensibilités en boucle fermée. Le choix de base est de fixer une borne droite et parallèle avec l'axe fréquentiel et utiliser la condition comme l'objectif². A partir des conditions (6.21), (6.22) et de la paramétrisation (6.4) nous développons les fonctions critère suivantes (la notation est simplifiée en omettant le terme $(e^{-j\omega_d})$) :

$$\phi_{er}(X) = \sup_{\omega_d} [\bar{\sigma}(S_{yp}d_T^{-1}N_T(X)) - F_{er}(\omega_d)] \mid \forall \omega_d \in \langle 0, \pi \rangle \quad (6.23)$$

$$\phi_{ur}(X) = \sup_{\omega_d} [\bar{\sigma}(S_{ey^*}d_T^{-1}N_T(X)) - F_{ur}(\omega_d)] \mid \forall \omega_d \in \langle 0, \pi \rangle \quad (6.24)$$

Proposition 6.2 : *Les deux fonctions critère $\phi_{er}(X)$ et $\phi_{ur}(X)$ dans (6.23), et (6.24) sont convexes par rapport au vecteur d'optimisation X .*

Preuve 6.2 : La preuve de convexité est le même comme la Preuve 6.1, il suffi de substituer $\gamma = F_{ij}(\omega_d)$ et de mettre $M_t(z^{-1}) = 0$.

²Dans ce cas il s'agit de la minimisation de norme H_∞ des sensibilités.

Remarque 6.5 : La minimisation des valeurs singulières de sensibilités $S_{er}(z^{-1})$, $S_{ur}(z^{-1})$ est une minimisation de la norme H_∞ de la matrice du pré-compensateur $\|T(z^{-1})\|_\infty$ mais pondérée par la matrice de BF et par le gabarit fréquentiel. Une des conséquences est aussi la réduction de la norme $\|X\|_\infty$ du vecteur d'optimisation, en d'autres termes les valeurs des éléments du vecteur X sont minimisées.

6.5 Synthèse de pré-compensateur - sommaire

Supposons un régulateur $K(z^{-1})$ de la boucle fermée satisfaisant des spécifications sur la robustesse pour le modèle échantillonné $G(z^{-1})$ du procédé (voir Fig.6.1). Considérons aussi des spécifications de performance en poursuite qui ne sont pas assurées par $K(z^{-1})$. L'objectif est de trouver une matrice de transfert $T(z^{-1})$ telle que les spécifications soit satisfaites. La procédure de la synthèse est la suivante :

1. *Transformation de spécifications* - Pour appliquer l'optimisation convexe, il faut transformer les spécifications de poursuite en fonctions critère d'objectif et de contrainte. Les éléments nécessaires à déterminer sont :
 - Modèle de poursuite $M_t(z^{-1})$ - Il exprime les performances désirées en poursuite. Généralement, la forme (6.5) est utilisée où les fonctions de transfert diagonales sont les fonctions du 1^{er} ou 2^e ordre.
 - Gabarits - Il est possible de spécifier les gabarits fréquentiels pour les fonctions critère (6.23), (6.24). En gros, cela va réduire le gain du pré-compensateur dans des zones de fréquences où le gabarit est bas.
 - Fonctions critère - Il faut définir quelles fonctions critère sont considérées pour l'optimisation. Nous possédons des fonctions suivantes :
 - Fonction critère générale $\phi_G(X)$ - Pour optimiser les performances en poursuite définies par le modèle de poursuite $M_t(z^{-1})$, il est possible d'utiliser la fonction du critère général (6.9), où l'interaction est optimisée ensemble avec la dynamique.

- Fonction critère d'une fonction de BF $\phi_{v1v2t}(X)$ - L'autre choix est d'appliquer un ensemble des fonctions critère d'un élément (6.15) permettant d'optimiser les fonctions diagonales et non-diagonales séparément. Pour les deux cas il faut spécifier la constante γ nécessaire surtout si la fonction critère est la contrainte.
 - Fonction critère du gain $\phi_g(X)$ - La fonction critère (6.17) peut être utilisée pour garantir un gain statique désiré (généralement c'est I_{ny}) où la constante γ donne la différence permise (nécessaire si la fonction est une des contraintes).
 - Fonctions critère de robustesse $\phi_{er}(X)$, $\phi_{ur}(X)$ - Les fonctions (6.23), (6.24) peuvent spécifier la robustesse désirée.
 - Fonctions d'objectif/contrainte - Il faut décider quelles fonctions critère sont considérées comme objectif et quelles comme la contrainte dans l'algorithme d'optimisation.
 - Coefficients de pondération - La fonction (5.69) est utilisée comme une fonction multi-critère, ainsi il faut spécifier les coefficients de pondération α_{ij} .
2. *Choix de paramètres d'optimisation* - Les paramètres d'optimisation sont des paramètres liés à l'algorithme d'optimisation et à la paramétrisation du pré-compensateur. Il s'agit de :
- Ordre de paramétrisation N_t - Il dépend de la complexité du problème, comme dans l'optimisation du régulateur en BF, et il détermine l'ordre de la matrice $T(z^{-1})$. En général, la valeur initiale est $N_t = 1$ et elle est augmentée si les résultats sont insatisfaisants.
 - Paramètres d'algorithme ellipsoïde - Les paramètres de l'algorithme d'optimisation ellipsoïde A , dx , centre initial, le critère d'arrêt, et le nombre maximal d'itération doivent être choisis.
3. *Optimisation* - L'optimisation est effectuée et les résultats sont analysés. Si les performances du système ne sont pas satisfaisantes il faut revenir au pas 1 ou 2 et modifier les spécifications selon l'analyse des résultats.

6.6 Notes et références

Références : La synthèse de pré-compensateur SISO par l'optimisation convexe est partiellement traitée dans [Lan98] pour le cas monovariabale. Dans [Lan02] elle est effectuée par l'analyse de la boucle fermée (voir le Chapitre 2). Les deux méthodes calculent séparément le régulateur de BF et le pré-compensateur. Dans le domaine multivariable, le pré-compensateur est typiquement calculé par la commande H_∞ . Dans ce cas, le pré-compensateur est optimisé simultanément avec le régulateur en boucle fermée avec le critère (6.6) [LKP93, PP01]. Une autre méthode de synthèse de pré-compensateur est la commande optimale [Yj85].

Fichiers et logiciels : Les m-fonctions programmées pour la commande est l'analyse des systèmes MIMO sont dans le répertoire `\M_functions\` (pour la description voir le fichier `Contents.m`). La figure Fig.6.2 est enregistrée sous format de Matlab dans le dossier `:\Figures\`. Le fichier correspondant est `track_specif.fig`.

References : *The SISO pre-compensator design is partially treated in [Lan98], in [Lan02] the design is obtained from closed loop analysis (see also Chapter 2). Both methods compute separately the closed-loop controller and pre-compensator. In the multivariable domain, the pre-compensator is typically computed by H_∞ control. In this case, the pre-compensator is optimized simultaneously with the closed-loop controller using the criterion (6.6) [LKP93, PP01]. Another pre-compensator design method is optimal control [Yj85].*

Files and software : *Some programmed m-functions for MIMO system control and analysis are in the subdirectory `\M_functions\` (for the m-functions description see `Contents.m`). The figure Fig. 6.2 is saved in Matlab format in the directory `:\Figures\`. The corresponding file is `track_specif.fig`.*

Chapitre 7

Réduction de régulateur par identification en boucle fermée

Dans ce chapitre une nouvelle procédure pour la réduction de régulateurs numériques SISO et MIMO est présentée. Elle est basée sur l'identification en boucle fermée et sur la pondération fréquentielle. L'idée principale vient de la méthode de réduction de régulateurs SISO par identification en boucle fermée [LKC01]. Il s'agit d'un algorithme d'identification en boucle fermée modifié qui permet d'identifier le régulateur réduit en boucle fermée à partir du modèle échantillonné du procédé, du régulateur complexe (nominal) et d'une séquence de signal d'excitation. Dans la technique développée ici, l'algorithme d'identification de sous-espaces [vOM96, Fav99] est employé pour identifier le régulateur réduit utilisant les mêmes éléments : le modèle, le régulateur, et la séquence d'excitation. En plus, d'autres algorithmes d'identification MIMO peuvent être utilisés à la place de la méthode de sous-espaces, car la technique développée utilise l'algorithme d'identification sans modification.

In this chapter a new simple SISO and MIMO controller reduction procedure is presented. It is based on closed-loop identification and on frequency weighting. The basic idea comes from SISO closed-loop based controller reduction technique [LKC01]. It is in fact modified recursive least squares identification

algorithm allowing to identify reduced controller in closed loop from discrete-time plant model, complex controller, and excitation signal sequence. In the developed technique, a sub-space identification algorithm [vOM96, Fav99] is employed to identify reduce controller from the same elements : model, controller, and excitation sequence. Moreover, other MIMO identification algorithms can be used instead of sub-space method, since the developed technique use the identification algorithm without any modification.

7.1 Identification en boucle fermée - rappel

Dans cette section nous rappelons deux procédures similaires de l'identification des modèles de procédé en boucle fermée. Basé sur ces procédures, nous allons développé une technique de réduction de régulateurs dans la section suivante.

Considérons un procédé continu et un régulateur numérique $K(z^{-1})$ connecté en boucle fermée comme dans la Fig.7.1. Les sensibilités considérées de cette boucle fermée sont : S_{yp} , S_{up} , S_{yb} , et S_{yd} et elles sont définies par (3.13)-(3.17). Pour obtenir un modèle échantillonné $G(z^{-1})$ du procédé, le procédé va être identifié en boucle fermée.

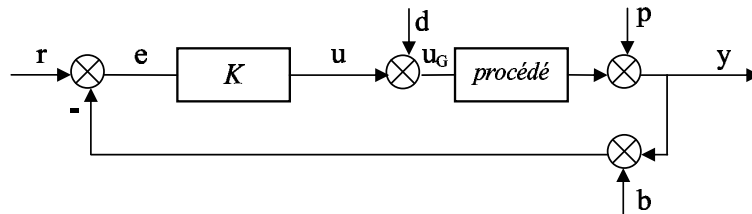


FIG. 7.1 – Boucle fermée considérée

La première procédure de l'identification en boucle fermée décrite ici est utilisée depuis longtemps. Elle peut être décrit par le schéma dans la Fig. 7.2 a) et elle comporte les étapes suivantes :

1. Connecter le régulateur K en boucle fermée de la même façon comme c'est montré dans la Fig. 7.1.

2. Ajouter un signal extérieur d'excitation persistante¹ $r(t)$ ou $d(t)$ (les signaux $b(t)$ et $p(t)$ ne sont pas considérés car ils ont le même effet comme $r(t)$). Le signal d'excitation est généralement considéré d'avoir une forme de SBPA qui a la caractéristique du bruit blanc.
3. Faire l'acquisition des données des signaux $u(t)$, et $y(t)$ (entrée et sortie du procédé).
4. Appliquer un algorithme d'identification en boucle ouverte sur les données acquises et identifier le nouveau modèle. Parmi d'autres nous pouvons utiliser les algorithmes suivants :
 - L'identification de sous-espace [vOM96, Fav99] (Sub-space identification) - elle appartient aux méthodes d'identification multivariable les plus efficaces. L'identification est effectuée par les opérations géométrique (projections oblique) sur les entrées et les sorties acquises. En plus, cette méthode permet de pondérer à droite et à gauche le modèle identifié.
 - L'identification par les moindres carrés récursifs (MCR) [Lan78] ou [Duo93] (recursive least-square identification) - elle est développée à partir de l'identification classique SISO, voir par exemple [Lan02]. Le carré de l'erreur entre la sortie réelle et la sortie prédite par le modèle identifié est minimisé dans cette méthode.

Si l'algorithme d'identification utilisé est le MCR multivariable [Lan78], le modèle identifié \hat{G} est ajusté selon l'erreur des sorties $\varepsilon_{yr}(t)$ défini, en gros, comme suit :

$$\varepsilon_{yr}(t) = (y(t) - \hat{y}(t)) = \left(P_r K (I + P_r K)^{-1} - \hat{G} K (I + P_r K)^{-1} \right) r(t) \quad (7.1)$$

où terme P_r remplace le procédé réel (la notation de l'opérateur z^{-1} est supprimée). A noter aussi qu'en réalité l'estimation $\hat{y}(t)$ est calculé à partir de $y(t-1)$, $y(t-2)$, ... et non à partir de $\hat{y}(t-1)$, $\hat{y}(t-2)$, ... Le critère

¹Pour la définition du signal d'excitation persistante voir par exemple [vOM96]. Ce signal permet d'exciter le procédé de telle façon que le signal de sortie contient suffisamment d'information pour décrire le procédé.

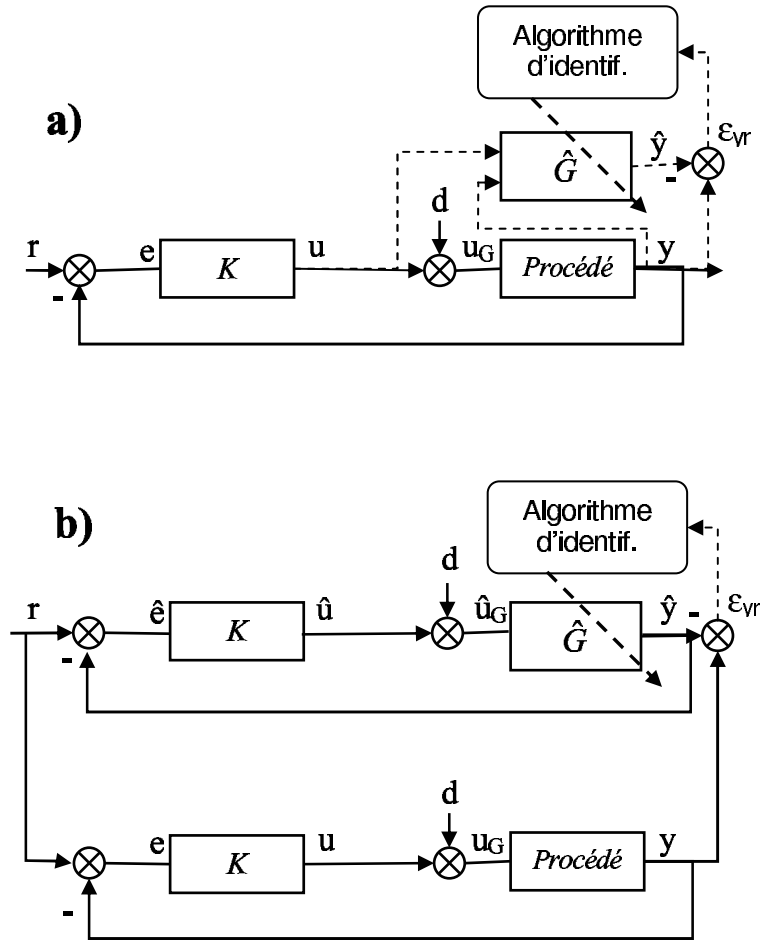


FIG. 7.2 – Deux types de l'identification en boucle fermée

d'identification $J(t)$ à minimiser a la forme suivante :

$$J(t) = \frac{1}{t} \sum_{i=1}^t |\varepsilon_{yr}(i)|^2 = \frac{1}{t} \sum_{i=1}^t |\varepsilon_{yr}(i)| |\varepsilon_{yr}^T(i)| \quad (7.2)$$

avec $J(t)$ minimisé dans chaque pas de l'identification en ajustant \hat{G} .

Remarque 7.1 : A noter que l'entrée du procédé $u_G(t)$ est un signal d'excitation du procédé qui est filtré par la boucle fermée, c'est-à-dire par les sensibilités de la boucle fermée.

L'autre méthode de l'identification en boucle fermée qui est plus efficace (en particulier en présence du bruit) a été développée pour les systèmes SISO [LK97]. Elle est basée sur l'algorithme d'identification de MCR modifié pour l'identification en boucle fermée. La Fig. 7.2 b) présente cette technique. Le principe d'identification est le même comme dans la méthode précédent, seule l'estimation de la sortie du modèle $\hat{y}(t)$ est calculée différemment. L'erreur des sorties devient :

$$\varepsilon_{yr}(t) = (y(t) - \hat{y}(t)) = \left(P_r K (I + P_r K)^{-1} - \hat{G} K (I + \hat{G} K)^{-1} \right) r(t) \quad (7.3)$$

Remarquons que cette erreur représente précisément l'écart entre la boucle fermée réelle et la boucle fermée en simulation. Nous n'utiliserons pas cette 2^e approche dans le cas multivariable.

7.2 Réduction de régulateurs multivariables

7.2.1 Objectif de réduction

Le but de la réduction est d'obtenir un régulateur \hat{K} qui est d'ordre réduit par rapport à l'ordre du régulateur nominal K . En plus le régulateur réduit \hat{K} doit garantir que les performances et la robustesse (qui sont les objectifs principales de la commande) de la boucle fermée dans la Fig. 7.1 ne soient pas détériorées. Les Critères variés peuvent être utilisés pour mesurer la différence entre les performances de la BF du régulateur K et du régulateur \hat{K} . Typiquement, l'écart entre les sensibilités du régulateur nominal et les sensibilités du régulateur réduit sont utilisés [AL89, WSL02, LKC01], etc. Or, les sensibilités reflètent bien les performances et la robustesse de la boucle fermée (voir le chapitre 3).

Supposons deux boucles fermée comme c'est montré dans la Fig. 7.3 : une avec le régulateur complexe nominal K , l'autre avec le régulateur d'ordre réduit \hat{K} . Le critère pour évaluer la différence entre les sensibilités de ces deux boucles peut être calculé comme l'erreur des signaux correspondants (comme dans le cas de l'identification en boucle fermée). Supposons d'évaluer

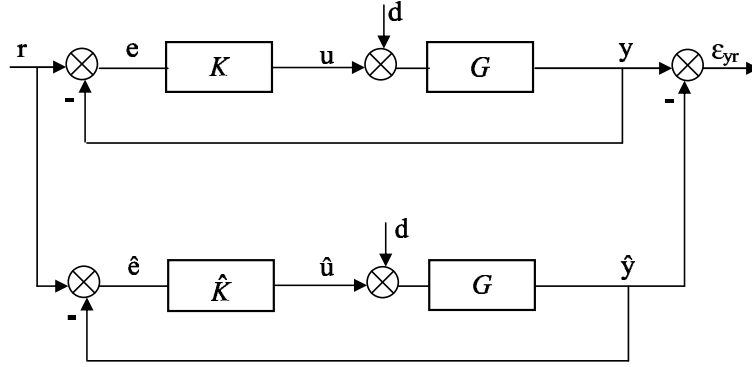


FIG. 7.3 – Schéma de l'erreur de sortie de la boucle fermée

la différence de la sensibilité S_{yb} , le critère va être l'erreur des sorties ε_{yr} décrit par :

$$\varepsilon_{yr}(t) = (y(t) - \hat{y}(t)) = \left(-S_{yb}(z^{-1}) + \hat{S}_{yb}(z^{-1}) \right) r(t) \quad (7.4)$$

avec $y(t) = -S_{yb}(z^{-1})r(t)$ (voir la Fig.7.4). Dans la relation (7.4), l'information sur la différence entre les deux boucles est portée par le terme $(-S_{yb}(z^{-1}) + \hat{S}_{yb}(z^{-1}))$ dont la norme peut être utilisée comme le critère. En conséquence, la relation pour obtenir un régulateur réduit optimale \hat{K}^* peut avoir, par exemple, la forme suivante (la notation de (z^{-1}) est supprimée) :

$$\begin{aligned} \hat{K}^* &= \arg \min_{\hat{K}} \|S_{yb} - \hat{S}_{yb}\|_2 = \\ &= \arg \min_{\hat{K}} \| -GK(I + GK)^{-1} + G\hat{K}(I + G\hat{K})^{-1} \|_2 \end{aligned} \quad (7.5)$$

ou utilisant l'erreur des sorties la forme suivante :

$$\hat{K}^* = \arg \min_{\hat{K}} \int_{-\infty}^{\infty} |\varepsilon_{yr}(t)|^2 dt \quad (7.6)$$

et d'après le Théorème de Parseval la relation (7.6) peut s'écrire dans le domaine fréquentiel comme suit :

$$\hat{K}^* = \arg \min_{\hat{K}} \frac{1}{2\pi} \int_{-\pi}^{\pi} \left| -S_{yb} + \hat{S}_{yb} \right|^2 \phi_r(j\omega) d\omega \quad (7.7)$$

où $\phi_r(j\omega)$ est la densité spectrale du signal d'excitation $r(t)$. Les relations (7.6) et (7.7) désignent la minimisation de l'énergie de l'erreur $\varepsilon_{yr}(t)$. Le terme (7.5) indique la même chose si la norme H_2 est utilisée.

De (7.5) on obtient :

$$\|S_{yb} - \hat{S}_{yb}\| \cong \left\| \underbrace{(I + GK)^{-1}}_{S_{yp}} G(\hat{K} - K) \underbrace{(I + GK)^{-1}}_{S_{yp}} \right\| \quad (7.8)$$

où l'élément $(\hat{K} - K)$ du deuxième ordre est négligé [AL89]. On observe qu'il s'agit d'une minimisation de la différence entre les deux régulateurs qui est pondérée dans le domaine fréquentiel.

Avec l'approximation (7.8), l'erreur des sorties ε_{yr} a la forme :

$$\varepsilon_{yr}(t) = (y(t) - \hat{y}(t)) \cong (S_{yp}G(\hat{K} - K)S_{yp})r(t) \quad (7.9)$$

Les expressions (7.8) et (7.9) nous indiquent qu'une minimisation de la différence pondérée de régulateurs correspond à la minimisation de l'écart des sensibilités de deux boucle fermée de Fig. 7.3. Cette idée va nous servir dans la section suivante pour construire une méthode de réduction.

Remarque 7.2 : La pondération de $(\hat{K} - K)$ est donnée par la sensibilité S_{yp} et par $S_{yp}G$. Ceci introduit une pondération plus forte de la différence $(\hat{K} - K)$ dans des zones de fréquences où la réponse $|S_{yp}|$ est grande c'est-à-dire dans les zones de fréquences où la robustesse est plus critique.

7.2.2 Réduction par identification en BF

Considérons la procédure de l'identification en boucle fermée décrite dans la section 7.1 par la Fig.7.2 a) mais appliquée au régulateur K à la place du procédé. Supposons en plus le procédé réel remplacé avec un modèle identifié G . La configuration obtenue est montrée dans la Fig.7.4 où l'erreur des commandes ε_{ur} a la forme :

$$\varepsilon_{ur}(t) = u(t) - \hat{u}(t) = KS_{yp}r(t) - \hat{K}S_{yp}r(t) = (K - \hat{K})S_{yp}r(t) \quad (7.10)$$

Si l'excitation $d(t)$ est utilisée à la place de $r(t)$, l'erreur de la commande va être :

$$\varepsilon_{ud}(t) = u(t) - \hat{u}(t) = KS_{yd}d(t) - \hat{K}S_{yd}d(t) = (K - \hat{K})S_{yd}d(t) \quad (7.11)$$

A partir de (7.10) et (7.11) nous observons que l'erreur de prédiction dépend de la différence $(K - \hat{K})$ pondérée par une sensibilité.

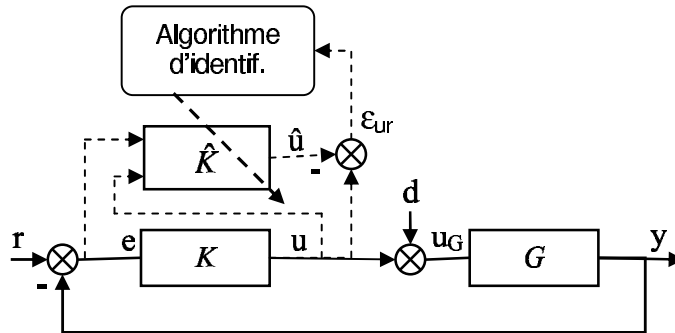


FIG. 7.4 – Schéma de l'identification en BF du régulateur

Dans le cas des systèmes SISO (où la commutation des sensibilités est possible) pour obtenir le critère (7.9) à partir de l'erreur (7.10), il suffit de filtrer le signal d'excitation $r(t)$ par la sensibilité $S_{yd} = GS_{yp}$. Le signal d'excitation a maintenant la forme $r(t) = S_{yd}(z^{-1})r_e(t)$ où $r_e(t)$ est un SBPA. L'erreur des commandes (7.10) devient :

$$\varepsilon_{ur}(t) = u(t) - \hat{u}(t) = (K - \hat{K}) S_{yp} S_{yd} r_e(t) \quad (7.12)$$

Dans le cas de l'excitation $d(t)$ la même opération peut être effectuée pour obtenir le même résultat, mais le signal d'excitation est filtré par la sensibilité S_{yp} : $d(t) = S_{yd}(z^{-1})d_e(t)$ et l'erreur va avoir la forme suivante :

$$\varepsilon_{ud}(t) = u(t) - \hat{u}(t) = (K - \hat{K}) S_{yd} S_{yp} d_e(t) \quad (7.13)$$

Les expressions (7.12) et (7.13) se rapproche de (7.9) à la commutativité près.

Le filtrage des signaux d'excitation est possible aussi dans le cas des systèmes MIMO, mais les erreurs résultantes ne correspondent pas à l'erreur (7.9). Néanmoins, le filtrage peut permettre d'obtenir des solutions satisfaisantes aussi pour les régulateurs MIMO. Or, l'autre point de vue est de dire que les sensibilités S_{yp} , S_{yd} sont des pondérations fréquentielles qui accentuent ou atténuent certaines régions fréquentielles de l'erreur $(K - \hat{K})$. Dans

ce cas, il est possible d'envisager de filtrer les signaux d'excitation par des sensibilités quelconques avec le but de créer une pondération appropriée.

Pour l'identification du régulateur multivariable nous allons utiliser la méthode de sous-espaces [vOM96]. Malgré que la méthode ne minimise pas directement le carré de l'erreur, comme le fait MCR multivariable [Lan78], les résultats obtenus pour la réduction sont nettement meilleurs par rapport à l'algorithme MCR.

7.2.3 Partie fixe de régulateur

Si le régulateur réduit \hat{K} doit contenir une partie fixe H_K (par exemple l'intégrateur), le régulateur nominal K et le régulateur réduit \hat{K} sont supposé d'avoir cette forme de matrices de transfert :

$$K(z^{-1}) = H_K(z^{-1})K_c(z^{-1}) \quad (7.14)$$

$$\hat{K}(z^{-1}) = H_K(z^{-1})\hat{K}_c(z^{-1}) \quad (7.15)$$

où la matrice $\hat{K}_c(z^{-1})$ doit être identifiée. Ensuite, nous allons définir le modèle augmenté du procédé :

$$G_a(z^{-1}) = G(z^{-1})H_K(z^{-1}) \quad (7.16)$$

Toutes les sensibilités considérées dans la procédure de réduction sont calculé avec le modèle augmenté (7.16) et avec la matrice de transfert $K_c(z^{-1})$, qui est la partie du régulateur à réduire.

Remarque 7.2 : Dans (7.14), le régulateur nominal est considéré de contenir la partie fixe imposée H_K . Une telle considération est cohérente, puisque si les spécifications de performance exigent la partie fixe, le régulateur nominal doit être calculé avec H_K .

7.3 Exemples de réduction de régulateurs

Dans cette section deux exemples d'utilisation de la nouvelle technique de réduction sont présentés. Le premier est un exemple SISO pratique lié

à un benchmark permettant de comparer la technique avec les meilleures méthodes de réduction utilisées dans le domaine monovarié sur une installation réelle (suspension active). Le deuxième exemple présente une réduction d'un régulateur multivariable où les performances du régulateur réduit sont mesurées par rapport au régulateur nominal.

7.3.1 Réduction du régulateur SISO de la suspension active

Description du Benchmark de la suspension active

Le système de la suspension active du benchmark [LKMP03] a permis de comparer des méthodes différentes pour la synthèse de régulateurs d'ordre réduit. L'installation de la suspension active est dans la Fig. 7.5, le schéma fonctionnel de la suspension est dans la Fig. 7.6.

Les spécifications du benchmark pour la synthèse du régulateur sont les suivantes [LKMP03] :

1. Les modules des S_{yp} et S_{up} sensibilités doivent être restreints dans le domaine fréquentiel par les gabarits montrés dans la Fig. 7.7.
2. Le numérateur du régulateur doit contenir le terme $(1 - z^{-1})$ de l'ouverture de boucle à la fréquence de Nyquist $\frac{1}{2T_e}$.
3. Les ordres n_R , n_S (nombres des zéros et des pôles) du régulateur doivent être les plus petits possible en vérifiant les performances désirées.

Le modèle ARMAX échantillonné identifié de la suspension est donné dans [LKMP03]. Les ordres du numérateur et du dénominateur sont respectivement $n_B = 16$ et $n_A = 14$, la période d'échantillonnage est $T_e = 0.00125$ s. Deux critères ΔS_{yp}^m et ΔS_{up}^m sont aussi définis dans [LKMP03] pour mesurer la satisfaction des spécifications. Les critères caractérisent le dépassement des gabarits des sensibilités pour le modèle identifié.

Dans le benchmark, plusieurs régulateurs complexes ont été calculés avec différentes méthodes et ensuite réduits par différents techniques de réduction



FIG. 7.5 – Installation de la suspension active.

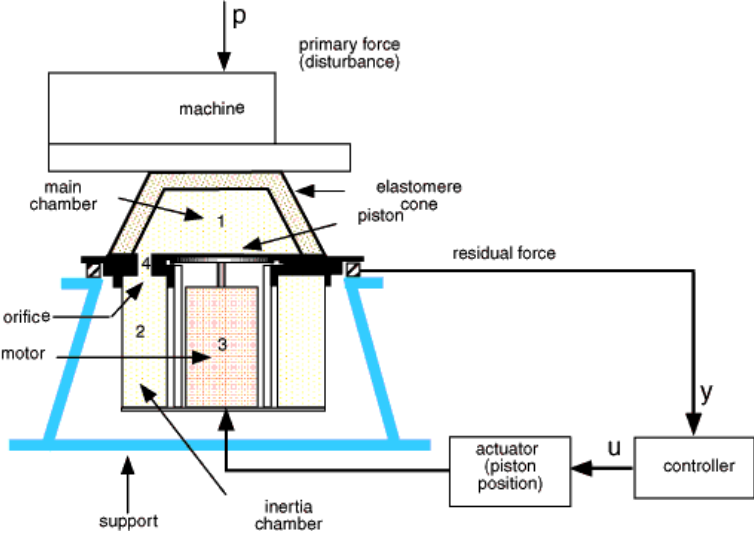


FIG. 7.6 – Schéma fonctionnel de la suspension active.

ε_{ur}	n_r	n_s	ΔS_{yp}^m	ΔS_{up}^m
reg.R.I.D.B.F.	4	4	4.22	0
$(K - \hat{K})S_{yp}S_{yd}r$	4	3	3.68	0
$(K - \hat{K})S_{yp}S_{yp}r$	4	3	6.36	0
$(K - \hat{K})S_{yp}S_{yp}r$	5	4	4.95	0

TAB. 7.1 – Régulateurs obtenus d’ordre réduit pour la suspension active (reg.R.I.D.B.F.= régulateur calculé par réduction d’identification direct en boucle fermée [Pro03]).

[IKH03]. Le régulateur nominal le plus simple qui a vérifié les spécifications a été calculé par placement de pôles avec calibrage de sensibilités [Pro03] (ou Annexe B3). Le régulateur nominal obtenu a eu les ordres du numérateur et du dénominateur $n_R = 18$, $n_S = 20$ et il satisfait à 100% les spécifications en simulation ($\Delta S_{yp}^m = 0$ et $\Delta S_{up}^m = 0$). Dans [Pro03], le régulateur nominal été réduit utilisant la technique de réduction d’identification directe en boucle fermée [LKC01] et le régulateur d’ordre réduit résultant a les ordres $n_R = 4$, $n_S = 4$ avec les critères des performances suivants : $\Delta S_{yp}^m = 4.22$ et $\Delta S_{up}^m = 0$ (voir reg.R.I.D.B.F. dans le Tab. 7.1).

Réduction du régulateur

Pour tester la technique de réduction développée, nous allons réduire le régulateur nominal avec les ordres $n_R = 18$, $n_S = 20$ calculé dans [Pro03]. Le signal d’excitation utilisé est un SBPA de $L = 511$ échantillons généré avec le registre de 9 bits et le diviseur de fréquence $d = 1$ [Lan02]. Le signal est ajouté à $r(t)$ et il est filtré soit par $S_{yp}(z^{-1})$, soit par $S_{yd}(z^{-1})$. La partie fixe du régulateur est une ouverture de boucle :

$$H_K = (1 - z^{-1})$$

qui est aussi exigée dans les spécifications.

L’algorithme d’identification de sous-espaces [vOM96] sans la pondération

de la projection oblique a été utilisé pour calculé le régulateur réduit. La méthode travaille en espace d'état et cela conduit dans le domaine SISO aux régulateurs avec les degrés du numérateur et du dénominateur qui sont égaux (avant de rajouter $(1 - z^{-1})$ au numérateur du régulateur). Les meilleurs régulateurs obtenus sont dans le Tab. 7.1. La Fig. 7.7 montre la comparaison des sensibilités du meilleur régulateur réduit obtenu (deuxième dans le tableau avec $\Delta S_{yp}^m = 3.68$) et du régulateur nominal. Par rapport aux régulateurs du benchmark calculés avec le même modèle [IKH03] et aux régulateurs donnés dans [Pro03], les correcteurs obtenus ici utilisant la nouvelle technique sont entièrement comparables aux meilleurs solutions obtenues.

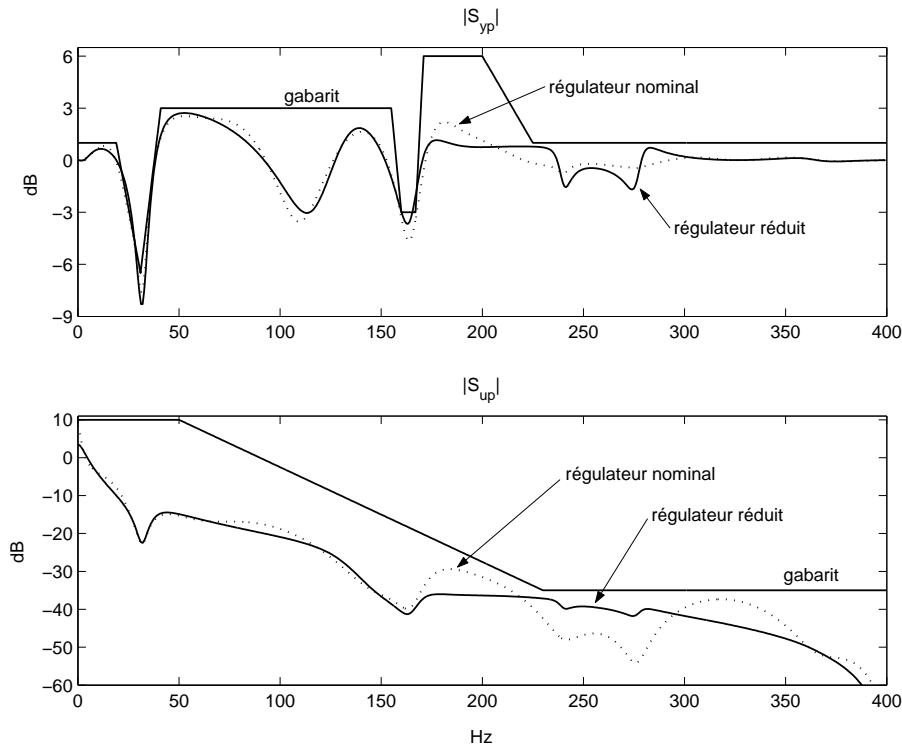


FIG. 7.7 – Fonctions de sensibilité pour le régulateur nominal (en pointillés) et pour le régulateur réduit (plein) avec les gabarits imposés.

7.3.2 Exemple de réduction d'un régulateur MIMO

Le modèle et le régulateur nominal

Le modèle MIMO considéré est un modèle linéaire temps-continu de la dynamique verticale d'un avion avec 3 entrées, 3 sorties et 5 états [Mac89]. Un régulateur continu de 17 états a été calculé par la commande H_∞ pour ce modèle dans [Mac89]. Quatre régulateurs numériques de différentes complexités (de 11, 14, 17, et 20 états) ont été calculés (voir Annexe B5) par calibrage de sensibilités et optimisation convexe. Ces régulateurs sont calculés en utilisant un modèle échantillonné obtenu à partir de la discrétisation du modèle continu avec la période d'échantillonnage $T_e = 0.05s$. Dans l'Annexe B5 les spécifications suivantes ont été imposés lors de la synthèse :

1. la dynamique désirée de la boucle fermée f_{BF} vers 1.6Hz (temps de montée correspondant 0.35s).
2. le dépassement maximal autour de 10%.
3. atténuation de hautes fréquences de bruit
4. restreindre l'interaction entre les canaux.
5. une commande acceptable (sans oscillations).
6. l'action intégrale dans chaque canal pour éliminer l'erreur statique.
7. minimiser les modules de la réponse fréquentielle de S_{yp} et S_{yb} pour obtenir la meilleure robustesse possible.

Les critères de performance et de robustesse pour les régulateurs obtenus dans l'Annexe B5 sont donnés dans la partie haute du Tab. 7.2. Un des régulateurs est calculé par le placement de pôles MIMO avec calibrage de sensibilités (nommé dans le tableau p.p.), les trois autres (dénommé c.o.) sont calculés par le calibrage de sensibilités avec l'optimisation convexe en utilisant le régulateur "p.p." comme le régulateur central.

Réduction du régulateur nominal

Nous avons décidé de réduire le régulateur le plus complexes "c.o.3" qui donne les meilleures performances. Le dépassement maximal et l'interaction sont satisfaisantes : 11% et 9% respectivement, et les maximums de $|S_{yp}|$ et de $|S_{up}|$ sont respectivement 2.59dB et 0.89dB.

Pour l'identification du régulateur réduit l'algorithme de sous-espaces de [vOM96] sera appliqué sans la pondération de la projection oblique. Un presque intégrateur multivariable est mis dans la partie fixe (comme dans [Mac89]) :

$$H_K = \text{diag}\{(1 - 0.9999z^{-1})^{-1}\}$$

Un ensemble de trois SBPA non-corrélées engendrées avec un registre de 9 bits et avec un diviseur de fréquence $d = 1$ a été utilisé comme le signal d'excitation.

Régula- teur	max. $ S_{yp} $ dB	max. $ S_{yb} $ dB	max. dépass. %	max. interaction %	# états -
p.p.	5.31	1.32	16	27	11
c.o.1	3.32	0.90	11	8.6	14
c.o.2	2.70	0.89	11	8.8	17
c.o.3	2.59	0.89	11	9.0	20
red.1	2.71	0.91	12	35	14
red.2	3.13	1.12	14	13	11
red.3	3.46	1.14	12	63	10
red.4	3.16	0.95	11	17	11

TAB. 7.2 – Comparaison des performances des différents régulateurs obtenus (p.p.=synthèse par placement de pôles, c.o.=synthèse par optimisation convexe, red.=régulateur réduit)

Les meilleurs régulateurs réduits obtenus sont "red.1", "red.2", "red.3",

et "red.4" (voir le Tab. 7.2). Les premiers trois régulateurs sont identifiés dans la configuration avec le signal d'excitation $r(t)$ filtré avec la sensibilité S_{yd} . Dans ce cas l'erreur des commandes a la forme :

$$\varepsilon_{ur}(t) = \left(K(z^{-1}) - \hat{K}(z^{-1}) \right) S_{yp}(z^{-1}) S_{yd}(z^{-1}) r(t)$$

Par contre, le régulateur "red.4" est réduit dans la configuration avec le signal d'excitation $d(t)$ filtré avec la sensibilité S_{yd} . Ainsi, l'erreur des commandes a la la forme suivante :

$$\varepsilon_{ud}(t) = \left(K(z^{-1}) - \hat{K}(z^{-1}) \right) S_{yd}(z^{-1}) S_{yd}(z^{-1}) d(t)$$

A noter que la performance la plus détériorée est l'interaction. Les autres critères de performance sont relativement préservés lors de la réduction et ils se dégradent continuellement avec la réduction de l'ordre du régulateur.

Il est évident à partir de Tab. 7.2, que le meilleur régulateur réduit est "red.2". Il a 11 états (9 états ont été supprimés) et ses performances sont très proche des performances désirées. Remarquons aussi que la complexité de ce régulateur est celle du régulateur calculé par placement de pôles (régulateur "p.p."), mais ses performances sont largement supérieures celle du régulateur "p.p." et proches des performances du régulateur nominal "c.o.3". La comparaison des valeurs singulières de S_{yp} et de S_{yb} pour le régulateur nominal "c.o.3" et pour le régulateur réduit "red.2" est donnée dans la Fig. 7.8, la comparaison de S_{up} et de S_{yd} est donnée dans la Fig. 7.9. A partir de ces deux figures nous observons qu'il n'y a pas des changements dramatiques des formes des réponses fréquentielles de ces sensibilités principales, c'est qui signifie en gros la conservation des performances. Aussi les maximums des réponses, qui indiquent approximativement la robustesse, sont juste légèrement augmentés. Les réponses à échelon de la boucle fermée pour les régulateurs "p.p.", "o.c.3", et "red.2" sont respectivement dans les figures Fig. 7.12, Fig. 7.10, et Fig. 7.11. Remarquons surtout l'interaction de 27% dans le cas du régulateur "p.p." qui est bien réduite pour les deux autres régulateurs.

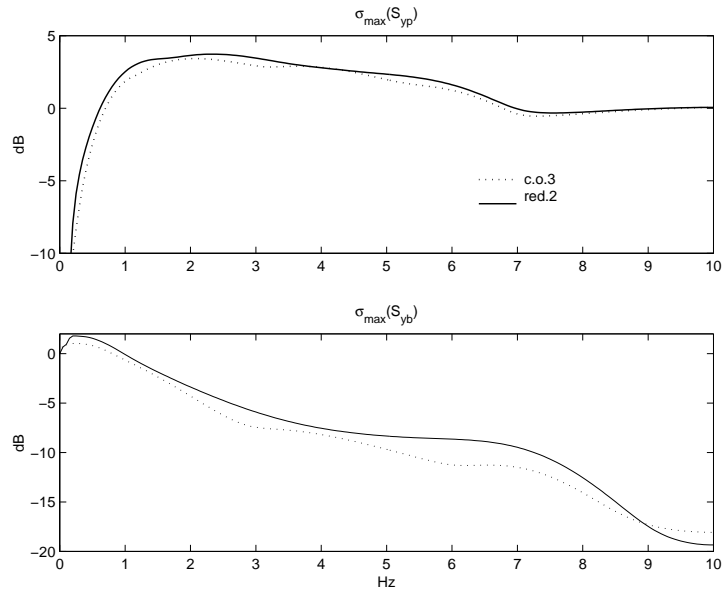


FIG. 7.8 – Les plus grandes valeurs singulières de S_{yp} et de S_{yb} pour "c.o.3" (en pointillé) et pour "red.2" (plein).

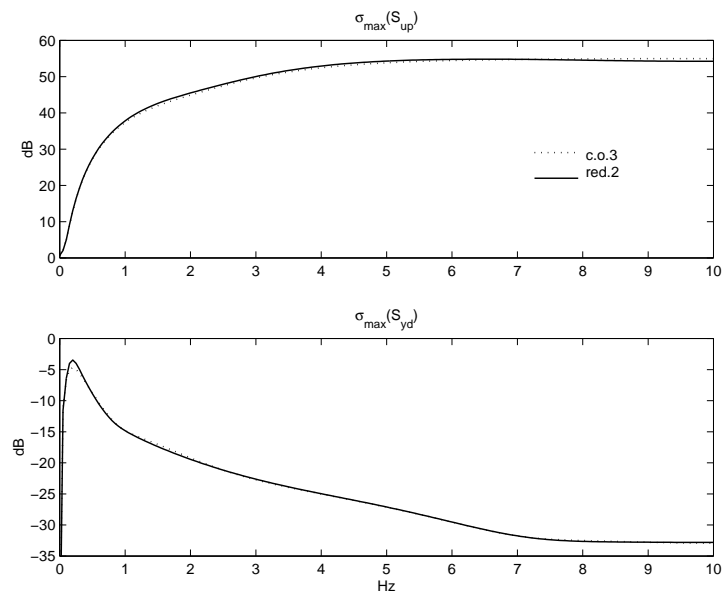


FIG. 7.9 – Les plus grandes valeurs singulières de S_{up} et de S_{yd} pour "c.o.3" (en pointillé) et pour "red.2" (plein).

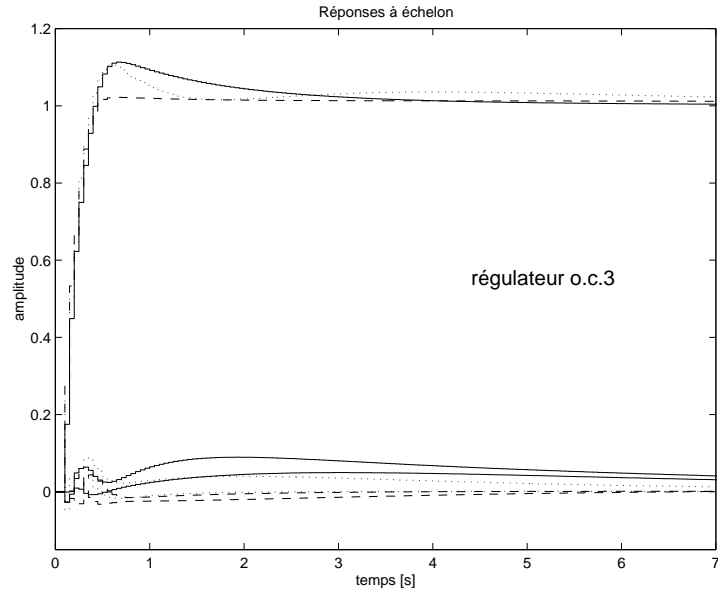


FIG. 7.10 – Réponses à échelon de la boucle fermée avec le régulateur "o.c.3" (sortie 1 - plein, sortie 2 - en tirets, sortie 3 - en pointillés).

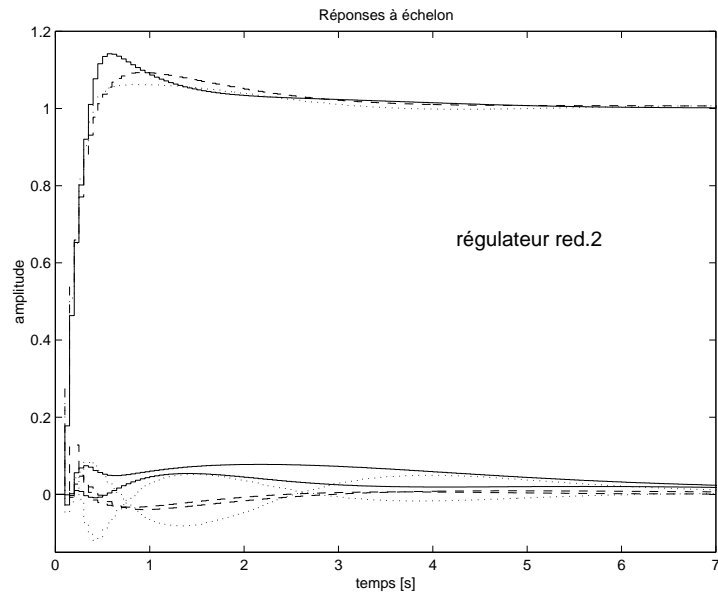


FIG. 7.11 – Réponses à échelon de la boucle fermée avec le régulateur "red2" (sortie 1 - plein, sortie 2 - en tirets, sortie 3 - en pointillés).

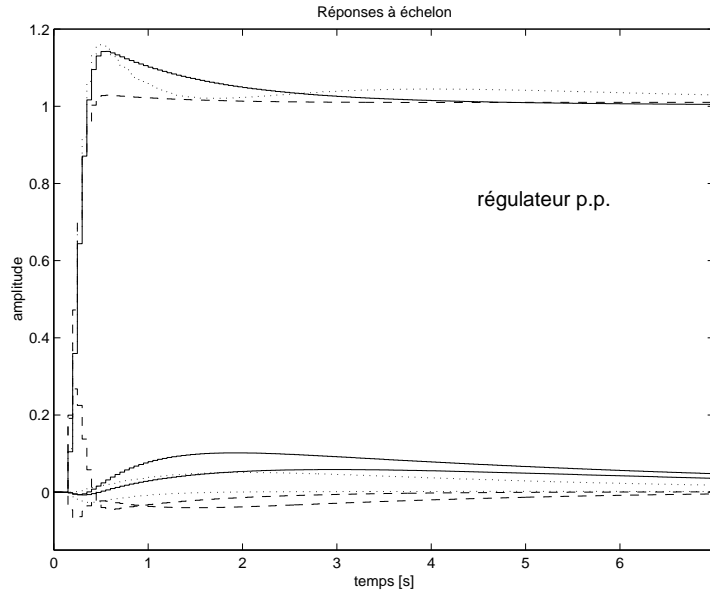


FIG. 7.12 – Réponses à échelon de la boucle fermée avec le régulateur "p.p." (sortie 1 - plein, sortie 2 - en tirets, sortie 3 - en pointillés).

7.4 Notes et références

Références : Une vue générale sur la réduction de régulateurs est donnée par exemple dans [AL89]. La réduction de régulateurs par l'identification en boucle fermée est introduite dans [LKC01], voir aussi [Lan02], avec une application pratique dans [CL03, Pro03] ([Pro03] correspond à l'annexe B3). L'extension MIMO de cette approche décrite dans ce chapitre est présentée avec les deux exemples d'application dans l'Annexe B6. Des différentes méthodes de réduction qui essaient de conserver les performances de la boucle fermée sont développées comme : "weighted balanced order reduction" [Zho98] ou [WSL02], réduction basé sur la factorisation coprime [LAL90], etc.

Fichiers et logiciels : Tous les fichiers concernant les deux exemples de la synthèse (includ les scripts .m utilisés pour la réduction) se trouve sur le CD de thèse dans le répertoire : `\\Design_exemples \appB6_Multivariable\`. Les fichier de l'exemple SISO sont dans le dossier `\SISO_exemple\`, les fichier

de l'exemple MIMO sont dans le dossier `\MIMO_example\`. Pour la description des fichiers lisez le fichier `_README.txt` dans le dossier correspondant. Toutes les figures de graphes sont enregistrées sous format de Matlab dans le dossier `:\Figures\`. Les figures Fig. 7.7, Fig. 7.8, Fig. 7.9, Fig. 7.10, Fig. 7.11, et Fig. 7.12 correspondent respectivement aux fichiers `sypsup_susp.fig`, `sypsyb_macired.fig`, `supsyd_macired.fig`, `stepresp_oc3.fig`, `stepresp_red2.fig`, et `stepresp_pp.fig`.

References : *A general overview of controller reduction topic is given in [AL89]. The controller reduction by closed-loop identification is introduced in [LKC01], see also [Lan02], with one practical application in [CL03, Pro03] ([Pro03] corresponds to Appendix B3). The MIMO extension of this approach described in this chapter is presented with two practical applications in Appendix B6. Several other reduction methods which try to conserve closed loop performances have been developed such as : weighted balanced order reduction [Zho98] or [WSL02], coprime factorization based reduction [LAL90], etc.*

Files and software : *All files concerning two presented design examples (including .m scripts used for reduction) can be found in the directory `:\Design_examples \appB6_Multivariable\`. Files of SISO example are in the subdirectory `\SISO_example\`, files of MIMO example are in the subdirectory `\MIMO_example\`. Read the file `_README.txt` in a corresponding directory for the files description. All presented graphs are saved in Matlab format in the directory `:\Figures\`. The Fig. 7.7, Fig. 7.8, Fig. 7.9, Fig. 7.10, Fig. 7.11, and Fig. 7.12 corresponds respectively to the files `sypsup_susp.fig`, `sypsyb_macired.fig`, `supsyd_macired.fig`, `stepresp_oc3.fig`, `stepresp_red2.fig`, and `stepresp_pp.fig`.*

Chapitre 8

Conclusion et perspectives

8.1 Conclusion

Dans ce travail plusieurs thèmes concernant les procédures de synthèse de régulateurs numériques linéaires robustes ont été abordés. Un chapitre a été consacré aux systèmes monovariables (SISO) où une contribution importante a été apportée à la synthèse de régulateurs robustes par placement de pôles et calibrage de sensibilités. Le reste du texte a traité des systèmes multivariables (MIMO), une méthodologie complète de la synthèse de régulateurs robustes multivariables a été développée. Cette méthodologie comporte les trois étapes principales : 1) la synthèse du régulateur de la boucle fermée, 2) la synthèse du pré-compensateur, 3) la réduction du régulateur par identification en boucle fermée. La première étape comporte elle même deux sous-étapes : a) synthèse du régulateur central par placement de pôles ou une autre technique, b) optimisation convexe. Dans les paragraphes suivants nous allons souligner le sens et la contribution de chaque partie de ce travail.

- Dans le chapitre 2 qui concerne les systèmes SISO, les deux procédures de synthèse de régulateurs robustes ont été rappelées : le placement de pôles avec calibrage de sensibilités et le calibrage de sensibilités par optimisation convexe. Les deux procédures ont servi comme base pour le développement d'une méthodologie correspondante MIMO. En plus, le

chapitre présente les améliorations significatives apportées par ce travail à la procédure de placement de pôles. Il s'agit de : Premièrement, un nouveau outil très efficace de calibrage - filtres bande-étroite - qui a été développé. L'Annexe B8 est une analyse complète de ce nouveau outil. Deuxièmement, l'approche interactive introduite dans la procédure de synthèse qui simplifie la synthèse. Troisièmement, un logiciel (boite à outils) pour la synthèse par placement de pôles et calibrage de sensibilités a été programmé dans le cadre de ce travail. Ce logiciel reflète toutes les améliorations apportées et il fait la procédure très convenable pour la pratique d'ingénierie. Finalement le logiciel et les améliorations ont été testés sur les exemples pratiques présentés dans les Annexes B1, B2, et B3. Grâce à ces trois innovations le placement de pôles avec calibrage de sensibilités est devenu une méthode de synthèse de régulateurs SISO robustes extrêmement efficace et puissante. Les régulateurs sont facile à calculer et en même temps très performants.

- Le chapitre 3 introduit la notation nécessaire et la théorie fondamentale de la synthèse de commande multivariable par calibrage de sensibilités. Les principes du calibrage pour garantir la performance et la robustesse sont aussi précisées. Ils résultent des expériences faites sur des différents exemples de synthèse et ils permettent de mieux comprendre les relations entre la performance et la robustesse de la boucle fermée en terme des caractéristiques fréquentielles des sensibilités. Ces relations jouent le rôle crucial dans la synthèse par calibrage de sensibilités mais aussi dans la commande H_∞ .
- Dans le chapitre 4 la théorie de placement de pôles multivariable est rappelée. La structure utilisée du régulateur est montrée et trois algorithmes différents d'optimisation pour placer les pôles de la boucle fermée sont présentés et testés. Ensuite, des nouvelles règles de placement de pôles pour calibrer les sensibilités sont développés et illustrés sur des exemples. La méthodologie est utilisée dans la suite pour calculer le régulateur central qui est une partie du régulateur d'optimisation

convexe.

- Le chapitre 5 est la partie principale de ce travail. Il décrit la procédure développée de synthèse de régulateurs numériques robustes MIMO par calibrage de sensibilités et optimisation convexe. La structure du régulateur, les fonctions convexes de critère et les étapes de synthèse sont présentées. Une comparaison avec la méthode existante (commande H_∞) est aussi effectuée. La procédure développée est ensuite illustrée sur des exemples. Finalement, un logiciel (boîte à outils) interactif a été programmé pour mettre en oeuvre la procédure développée. En plus, le logiciel intègre aussi la méthode de placement de pôles multivariable avec calibrage de sensibilités décrite dans le Chapitre 4. La procédure développée de synthèse donne de résultats satisfaisants et comparables avec d'autres méthodes comme la commande H_∞ ou la commande prédictive (voir les exemples de synthèse). Ceci avec l'avantage claire d'une prise en compte plus facile des spécifications et de l'utilisation du modèle échantillonné de procédé c'est qui permet d'obtenir directement un régulateur numérique. Néanmoins, les régulateurs obtenus sont souvent de complexité élevée. Pour cela il est souhaitable d'utiliser cette procédure de synthèse ensemble avec une technique de réduction de régulateurs.
- Le chapitre 6 concerne la théorie de la synthèse du pré-compensateur par l'optimisation convexe. La structure du pré-compensateur est établie, les fonctions convexes de critère sont définies, et les étapes de conception sont décrites dans ce chapitre.
- Finalement, le chapitre 7 présente une nouvelle technique de réduction de régulateurs MIMO. La technique est basée sur l'identification en boucle fermée et elle exploite un algorithme d'identification en boucle ouverte déjà existant : identification en sous-espaces. L'avantage de la technique de réduction développée est qu'elle est facile à utiliser et ainsi appropriée pour la pratique d'ingénierie. Deux exemples d'utilisation sont présentés : Un exemple pratique SISO qui comporte la comparaison

du résultat avec d'autres méthodes de réduction et testé expérimentalement sur une suspension active. L'autre exemple est multivariable et il examine la conservation des performances de la boucle fermée.

8.2 Perspectives

L'objectif principal de ce travail, c'est-à-dire de développer et de réaliser une méthodologie de synthèse de régulateurs numériques robustes MIMO par calibrage de sensibilités et optimisation convexe a été atteinte. Néanmoins, il est possible d'envisager de poursuivre le développement des méthodes et des algorithmes présentés. Ceci concerne notamment les problèmes suivants :

- Pour la synthèse du régulateur central qui est un observateur avec le retour d'état estimé, seule la méthode de placement de pôles par optimisation de vecteurs propres a été effectivement appliquée. D'autres méthodes pour le calcul de régulateur central peuvent être envisagées comme placement de pôles et de vecteurs propres, LQG, ou la commande prédictive. La seule contrainte pour ces méthodes c'est que le régulateur obtenu doit avoir possibilité d'être décrit comme un observateur avec le retour d'état estimé.
- Une autre structure du régulateur central pourrait être considérée. Actuellement, l'observateur estime les états de la partie fixe qui est connue. Il serait intéressant de voir si un observateur réduit peut être appliqué dans la paramétrisation utilisée.
- Les dérivés des fonctions critère pourrait être développés pour obtenir les fonctions de gradient. Elle permettrions d'accélérer radicalement l'optimisation de fonctions critère.
- La paramétrisation du pré-compensateur ne permet pas, pour l'instant, d'optimiser son dénominateur. Une autre structure de pré-compensateur pourrait être développé afin de rendre l'optimisation plus efficace.
- La méthodologie de synthèse de régulateur à deux degrés de liberté proposée dans ce travail considère l'optimisation de régulateur de la

boucle fermée séparée de l'optimisation du pré-compensateur. Dans le futur, une optimisation simultanée du régulateur de la boucle fermée et du pré-compensateur pourrait être envisagée, comme dans certains approches de la commande H_∞ .

- Par le manque de temps, la méthode de synthèse du pré-compensateur n'a pas été testée. Il faudrait programmer la méthode et réaliser les tests en simulation pour savoir les avantages et les désavantages et pour obtenir les règles et les conseils pratiques d'application.
- La technique proposée de réduction de régulateurs MIMO utilise les algorithmes d'identification en boucle ouverte. Cependant, le développement des algorithmes spécifiques d'identification en boucle fermée peut être envisagé comme c'est le cas pour les systèmes SISO. Ceci permettrait d'obtenir sans filtrage un effet de pondération fréquentielle proche de l'optimum théorique désiré.

Bibliographie

- [Ada99] Adaptech. *User's guide for OPTREG - Automated design of robust digital controllers, version 1.1 for Matlab 4.2*. Adaptech, 1999.
- [AL89] B.D.O. Anderson and Y. Liu. Controller reduction : Concepts and approaches. *IEEE Transactions on Automatic Control*, 34(8) :802–812, 1989.
- [BB91] S.P. Boyd and C.H. Barratt. *Linear controller design - Limits of performance*. Prentice Hall, New Jersey, 1991.
- [CB95] E.F. Camacho and C. Bordons. *Model Predictive Control in the Process Industry*. Springer-Verlag London, 1995.
- [CL03] A. Constantinescu and I.D. Landau. Controller order reduction by identification in closed loop applied to a benchmark problem. *European Journal of Control*, 9(1), 2003.
- [DF99] G. Duc and S. Font. *Commande H-inf et mu-analyse*. Hermes, Paris, 1999.
- [Duo93] H.N. Duong. *Identification Structurel et Parametrique des Systèmes Linéaire Monovariabiles et Multivariabiles*. PhD thesis, Institut Nationale Polytechnique de Grenoble, 1993.
- [Fav99] W. Favoreel. *Subspace Identification and Control of Linear and Bilinear Systems*. PhD thesis, Katholieke Universiteit Leuven - Faculteit Toegepaste Wetenschappen Arenbergkasteel, B-3001 Heverlee, Belgium, 1999.

- [Gan66] F.G. Gantmacher. *Theorie des matrices*. Dunod, 1966.
- [GL95] M. Green and D.J.M. Limbeer. *Linear robust control*. Prentice-Hall, New Jersey, 1995.
- [IKH03] I.D.Landau, A. Karimi, and H. Hjalmarsson, editors. *International Workshop on Design and optimisation of restricted complexity controllers*. Laboratoire d'Automatique de Grenoble, January 2003.
- [Kai80] T. Kailath. *Linear Systems*. Prentice-Hall, New Jersey, 1980.
- [KND85] J. Kautsky, N.K. Nichols, and P. Van Doren. Robust pole assignment in linear state feedback. *Int. J. Control*, 41(5) :1129–1155, 1985.
- [KNR98] Kolmanovskii, Niculescu, and Richard. On the lyapunov-krasovskii functionals for stability analysis of linear delay systems. *International Journal of Control*, 1998.
- [Kuc79] V. Kucera. *Discret Linear Control : The polynomial equation approach*. Wiley, New York, 1979.
- [Kuc91] V. Kucera. *Analysis an Design of Discret Linear Control Systems*. Academia, Prague and Prentice Hall International, London, 1991.
- [LAL90] Y. Liu, B.D.O. Anderson, and U. Ly. Coprime factorization controller reduction with bezout identity induced frequency weighting. *Automatica*, 26(2) :233–249, 1990.
- [Lan78] A. Gauthier I.D. Landau. On the recursive identification of multi-input multi-output system. *Automatica*, 18(5) :609–614, 1978.
- [Lan98] J. Langer. *Synthèse de régulateurs numériques robustes - Application aux structures souples*. PhD thesis, Institut National Polytechnique de Grenoble, 1998.
- [Lan02] I.D. Landau. *Commande des systèmes*. Hermes, Paris, 2002.
- [LC99] J. Langer and A. Constantinescu. Pole placement design using convex optimization criteria for the flexible transmission benchmark. *European Journal of Contr.*, 5 :193–207, 1999.

- [Lim91] D.J.N. Limbeer. The specification and purpose of a controller design case study. *Proceedings of the 30th Conference on Decision and Control, Brighton, England, 1991*.
- [LK97] I.D. Landau and A. Karimi. Recursive algorithms for identification in closed loop : a unified approach and evaluation. *Automatica*, 33(8) :1499–1523, 1997.
- [LK98] I.D. Landau and A. Karimi. Robust digital control using pole placement with sensitivity function shaping method. *Int. J. Robust and Nonlin. Cont.*, 8 :191–210, 1998.
- [LKC01] I.D. Landau, A. Karimi, and A. Constantinescu. Direct controller order reduction by identification in closed loop. *Automatica*, 37(11) :1689–1702, 2001.
- [LKMP03] I.D. Landau, A. Karimi, L. Miskovic, and H. Prochazka. Control of an active suspension system as a benchmark for design and optimisation of restricted complexity controllers. *European Journal of Control*, 9(1), 2003.
- [LKP93] D.J.N. Limbeer, E.M. Kasenally, and J.D. Perkins. On the design of robust two degree of freedom controllers. *Automatica*, 29(1) :157–168, 1993.
- [LL99] J. Langer and I.D. Landau. Combined pole placement/sensitivity function shaping method using convex optimisation criteria. *Automatica*, 35 :1111–1120, 1999.
- [LLRB96] I.D. Landau, J. Langer, D. Rey, and J. Barnier. Robust control of 360 flexible arm using the combined pole placement/sensitivity function shaping method. *IEEE Trans. Control Systems Tchnol.*, pages 369–383, 1996.
- [Mac89] J.M. Maciejowski. *Multivariable Feedback Design*. Addison-Wesley, 1989.
- [Mag02] J.F. Magni. *Robust modal control*. Kluwer Academic/Plenum Publishers, New York, 2002.

- [MZ89] M. Morari and E. Zafiriou. *Robust Process Control*. Prentice Hall, New Jersey, 1989.
- [PL02] H. Prochazka and I.D. Landau. Logiciel pour l'enseignement et le calcul du placement de ples robuste. *Conf. Int. Francophone d'Automatique Nantes*, pages 694–698, 7 2002.
- [PL03] H. Prochazka and I.D. Landau. Pole placement with sensitivity function shaping using 2nd order digital notch filters. *Automatica*, 39(6) :1103–1107, 2003.
- [PP01] E. Prempain and I. Postlethwaite. Feedforward control : a full-information approach. *Automatica*, 37(1) :17–28, 2001.
- [Pro03] H. Prochazka. Comments on the paper "controller order reduction by identification in closed-loop applied to a benchmark problem". *European Journal of Control*, 9(5), 2003.
- [RM94] A. Rantzer and A. Megretski. A convex parameterization of robustly stabilizing controllers. *IEEE Trans. on Automatic Cont.*, 39(9) :1802–1808, 1994.
- [Roc82] R.T. Rockafellar. *Convex Analysis*. Princeton Univ. Press, 1982.
- [SSj94] K.M. Sobel, E.Y. Shapiro, and A.N. Andry jr. Eigenstructure assignment. *Int. J. Control*, 59(1) :13–37, 1994.
- [SW70] J. Stoier and C. Witzgall. *Convexity and Optimisation in Finite Dimensions*. Springer-Verlag, 1970.
- [TY96] A.L. Tits and Y. Yang. Globally convergent algorithms for robust pole assignment by state feedback. *IEEE Trans. Autom. Control*, 41(10) :1432–1452, 1996.
- [Var00] A. Varga. Robust pole assignment via sylvester equation based state feedback parametrization. *Proc. IEEE International Symposium on Computer Aided Control System Design, CACSD'2000, Anchorage, Alaska*, 2000.

- [vOM96] P. van Overschee and B. De Moor. *Subspace identification for linear systems : Theory, implementation, applications*. Kluwer Academic Publishers, 1996.
- [Wol74] W.A. Wolovich. *Linear multivariable systems*. Springer-Verlag, 1974.
- [WSL02] G. Wang, V. Sreeram, and W.Q. Liu. Balanced performance preserving controller reduction. *Systems and Control Letters*, 46 :99–110, 2002.
- [Yj85] D.C. Youla and J.J. Bongiorno jr. A feedback theory of two-degree-of-freedom optimal wiener-hopf design. *IEEE Transactions on Automatic Control*, AC-30(7) :652–662, 1985.
- [YJB76] D.C. Youla, H.A. Jabri, and J. Bongiorno. Modern wiener-hopf design of optimal controllers. *IEEE Trans. on Automatic Control*, AC-21 :319–338, 1976.
- [Zho98] K. Zhou. *Essentials of Robust Control*. Prentice Hall, New Jersey, 1998.

Annexe A

Concepts de base

A.1 Systèmes linéaires

A.1.1 Décomposition aux valeurs singulières

La plus grande valeur singulière $\bar{\sigma}(M)$ d'une matrice $M \in C^{m \times n}$ est calculée à partir de la décomposition aux valeurs singulières (singular value decomposition) suivante :

$$M = U \begin{bmatrix} \Sigma & 0 \\ 0 & 0 \end{bmatrix} V$$

où

$$\Sigma = \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sigma_p \end{bmatrix}$$

avec $\bar{\sigma}(M) = \sigma_1$, $p = \min(m, n)$, et avec des matrices unitaires $U \in C^{m \times m}$, $V \in C^{n \times n}$. Au sens géométrique, les valeurs singulières σ_i sont précisément les longueurs des semis-axes d'un hyperelipsoïde E défini par :

$$E = \{y : y = Mx, x \in C^n, \|x\| = 1\}$$

Pour les détails voir [Zho98, GL95] etc.

A.1.2 Zéros de transmission

Considérons une matrice de transfert quelconque $M_q(z^{-1}) = M(z)$ de la dimension $m \times n$. Il est possible de trouver deux matrices unimodulaires $U_1(z)$, $U_2(z)$ des polynômes¹ telles que la matrice $M(z)$ peut s'écrire sous la forme suivante :

$$M(z) = U_1(z)H(z)U_2(z) \quad (\text{A.1})$$

où la matrice de transfert $H(z)$ avec la dimension $m \times n$ a la forme :

$$H(z) = \begin{bmatrix} \frac{\epsilon_1(z)}{\psi_1(z)} & 0 & \dots & 0 \\ 0 & \ddots & & \\ \vdots & & \frac{\epsilon_r(z)}{\psi_r(z)} & \vdots \\ 0 & \dots & & 0 \end{bmatrix} \quad (\text{A.2})$$

où r est le rank (normal) de la matrice $M(z)$.

Les zéros de transmission de la matrice $M(z)$ sont les racines des polynômes $\epsilon_i(z)$ [Kai80].

A.1.3 Conditions de robustesse

Les conditions de robustesse qui concernent les incertitudes sont développées à partir du théorème du petit gain (Small gain theorem), voir par exemple [Zho98, DF99]. Par contre les marges de robustesse et les conditions éventuelles liées aux marges qui sont critères de stabilité de la boucle fermée sont typiquement développées à partir du lieu de Nyquist de la boucle fermée² (voir par exemple [Lan02]). Ici nous allons rappeler le théorème du petit gain et des conditions de robustesse pour l'incertitude non-structurée.

¹Matrice dont le déterminant est égale à une constante indépendante de l'opérateur z : $\det U(z) = cst$. En conséquence, l'inverse d'une matrice unimodulaire des polynômes est aussi une matrice des polynômes.

²Le lieu de Nyquist d'une boucle fermée est la réponse fréquentielle du système en boucle ouverte dessinée dans le plan complexe.

Théorème du petit gain

Théorème : Supposons une matrice de transfert stable $M(z^{-1})$ et une constante $\gamma > 0$. Après, le système montré dans la Fig. A.1 est stable pour toutes matrices de transfert d'incertitude $\Delta(z^{-1})$ avec

- $\|\Delta(z^{-1})\|_\infty \leq \frac{1}{\gamma}$ si et seulement si $\|M(z^{-1})\|_\infty < \gamma$
- $\|\Delta(z^{-1})\|_\infty < \frac{1}{\gamma}$ si et seulement si $\|M(z^{-1})\|_\infty \leq \gamma$

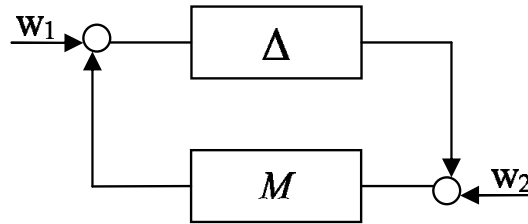


FIG. A.1 – La boucle de stabilité du théorème du petit gain.

Incertainité non-structurée additive

Supposons un modèle nominal $G(z^{-1})$ du procédé et un modèle incertain $G_\Delta(z^{-1}) = (G(z^{-1}) + \Delta(z^{-1}))$ où $\Delta(z^{-1})$ est une matrice de transfert qui représente l'incertitude additive.

Théorème : Un régulateur $K(z^{-1})$ stabilise la boucle fermée avec le modèle $G_\Delta(z^{-1})$ si les conditions suivantes sont satisfaites [GL95] :

1. La boucle fermée nominale (le modèle $G(z^{-1})$ avec le régulateur $K(z^{-1})$) est stable.
2. La matrice $\Delta(z^{-1})$ est telle que le modèle $G(z^{-1})$ et le modèle $G_\Delta(z^{-1})$ ont le même nombre des pôles instables.
3. La plus grande valeur singulière de la sensibilité $S_{up}(z^{-1})$ est :

$$\bar{\sigma}(S_{up}(z^{-1})) \leq \frac{1}{\bar{\sigma}(\Delta(z^{-1}))} \quad \forall \omega_d \quad (\text{A.3})$$

Incertitude non-structuré multiplicative de sortie

Supposons un modèle nominal $G(z^{-1})$ du procédé et un modèle incertain $G_{\Delta}(z^{-1}) = (I + \Delta(z^{-1}))G(z^{-1})$ où $\Delta(z^{-1})$ est une matrice de transfert qui représente l'incertitude multiplicative de sortie.

Théorème : *Un régulateur $K(z^{-1})$ stabilise la boucle fermée avec le modèle $G_{\Delta}(z^{-1})$ si les conditions suivantes sont satisfaites [GL95] :*

1. *La boucle fermée nominal (le modèle $G(z^{-1})$ avec le régulateur $K(z^{-1})$) est stable.*
2. *La matrice $\Delta(z^{-1})$ est telle que le modèle $G(z^{-1})$ et le modèle $G_{\Delta}(z^{-1})$ ont le même nombre des pôles instables.*
3. *La plus grand valeur singulière de la sensibilité $S_{yb}(z^{-1})$ (ou $S_{yy^*}(z^{-1})$) est :*

$$\bar{\sigma}(S_{yb}(z^{-1})) \leq \frac{1}{\bar{\sigma}(\Delta(z^{-1}))} \quad | \quad \forall \omega_d \quad (\text{A.4})$$

A.2 Convexité

A.2.1 Fonction convexe

Définition : Une fonction $\phi(X) : R^N \rightarrow R$ est convexe si et seulement si [Roc82, SW70]

$$\begin{aligned} \phi(\lambda X_1 + (1 - \lambda)X_2) &\leq \lambda\phi(X_1) + (1 - \lambda)\phi(X_2) \\ &\quad \forall \lambda \in \langle 0, 1 \rangle \\ &\quad \forall X_1, X_2 \in R^l \end{aligned}$$

A.2.2 Condition convexe

Définition : Une condition (inégalité)

$$f(X) < g(X) \quad (\text{A.5})$$

avec le vecteur réel $X \in R^N$ et avec les fonctions quelconques $f(X), g(X) \in R^N \rightarrow R$ va être nommée *convexe par rapport au vecteur X* si et seulement si la fonction :

$$h(X) : h(X) = f(X) - g(X) \quad (\text{A.6})$$

est convexe par rapport au vecteur X .

A.3 Représentations d'état de matrices de sensibilité

Supposons une configuration de la boucle fermé qui est sur la Fig.A.2, où K est un régulateur multivariables et G est un modèle multivariables. T est le pré-compensateur qui n'influence pas les sensibilités de la boucle fermée. Supposons que le modèle et le régulateur ont les représentations d'état suivantes :

$$\begin{aligned} x_G(t+1) &= Ax_G(t) + Bu_G(t) \\ y_G(t) &= Cx_G(t) \end{aligned} \quad (\text{A.7})$$

$$\begin{aligned} x_K(t+1) &= A_K x_K(t) + B_K e(t) \\ u(t) &= C_K x_K(t) + D_K e(t) \end{aligned} \quad (\text{A.8})$$

avec $A \in R^{n_{x_G} \times n_{x_G}}$, $B \in R^{n_{x_G} \times n_u}$, $C \in R^{n_y \times n_{x_G}}$, et $A_K \in R^{n_{x_K} \times n_{x_K}}$, $B_K \in R^{n_{x_K} \times n_y}$, $C_K \in R^{n_u \times n_{x_K}}$, $D_K \in R^{n_u \times n_y}$. Les variables n_u , n_y , n_{x_G} , n_{x_K} représentent les nombres des entrées, des sorties, des états du modèle G et des états du régulateur K .

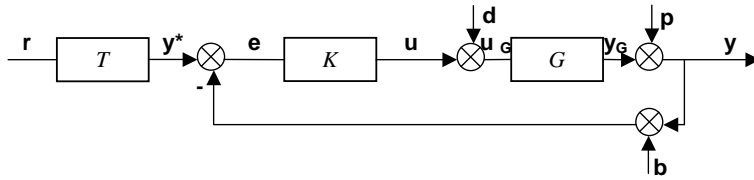


FIG. A.2 – Configuration considérée de la boucle fermée

Les matrices de sensibilité ont les représentations d'état suivantes :

Matrice de sensibilité S_{yp} : perturbation de sortie - sortie

$$S_{yp}(z^{-1}) = \frac{y(z^{-1})}{p(z^{-1})} = (I + G(z^{-1})K(z^{-1}))^{-1}$$

$$A_{yp} = \begin{bmatrix} A - BD_K C & BC_K \\ -B_K C & A_K \end{bmatrix}$$

$$B_{yp} = \begin{bmatrix} -BD_K \\ -B_K \end{bmatrix}$$

$$C_{yp} = \begin{bmatrix} C & 0 \end{bmatrix}$$

$$D_{yp} = I_{n_y \times n_y}$$

Matrice de sensibilité S_{up} : perturbation de sortie - commande

$$S_{up}(z^{-1}) = \frac{u(z^{-1})}{p(z^{-1})} = -K(z^{-1})(I + G(z^{-1})K(z^{-1}))^{-1}$$

$$A_{up} = A_{yp}$$

$$B_{up} = B_{yp}$$

$$C_{up} = \begin{bmatrix} -D_K C_G & C_K \end{bmatrix}$$

$$D_{up} = -D_K$$

Matrice de sensibilité S_{yb} : bruit de mesure - sortie

$$S_{yb}(z^{-1}) = \frac{y(z^{-1})}{b(z^{-1})} = -G(z^{-1})K(z^{-1})(I + G(z^{-1})K(z^{-1}))^{-1}$$

$$A_{yb} = A_{yp}$$

$$B_{yb} = B_{yp}$$

$$C_{yb} = C_{yp}$$

$$D_{yb} = 0$$

Matrice de sensibilité S_{yd} : perturbation de commande - sortie

$$S_{yd}(z^{-1}) = \frac{y(z^{-1})}{d(z^{-1})} = G(z^{-1})(I + K(z^{-1})G(z^{-1}))^{-1}$$

$$\begin{aligned} A_{yd} &= A_{yp} \\ B_{yd} &= \begin{bmatrix} B \\ 0 \end{bmatrix} \\ C_{yd} &= C_{yp} \\ D_{yb} &= 0 \end{aligned}$$

Dans le cas de retour de sorties positif (avec $e(t) = -y^*(t) + b(t) + y(t)$), les matrices sont les même seulement tous les signes "-" sont remplacés par "+".

Annexe B

Publications et rapports

- B1. H.Prochazka and I.D.Landau. Logiciel pour l'enseignement et le calcul du placement de pôles robuste. Conférence International Francophone d'Automatique, Nantes, pages 694-698, 2002
- B2. H.Prochazka and I.D.Landau. Pole placement with sensitivity function shaping using 2nd order digital notch filters. Automatica, 39(6), pages 1103-1107,2003
- B3. H. Prochazka. Comments on the paper "Controller order reduction by identification in closed-loop applied to a benchmark". European Journal of Control, 9(5), 2003
- B4. H.Prochazka and I.D.Landau. Multivariable robust controller design by pole placement and sensitivity shaping, Internal Rapport of LAG, 2002
- B5. H.Prochazka and I.D.Landau. Convex optimisation for robust multi-variable controller design as alternative of H_∞ optimisation. submitted to European Journal of Control, 2003
- B6. H.Prochazka and I.D.Landau. Multivariable controller reduction by closed-loop identification, 2003.
- B7. H.Prochazka. User's Guide for Pole-Placement Master, 2003.
- B8. H.Prochazka. Synthèse du filtre discret bande-étroite 2pôles/2zéros - application au calibrage des fonctions de sensibilité

B9. H.Prochazka. User's Guide for MV controller design tool, 2003.

Fichiers : Les fichiers correspondants aux exemples présentés dans les articles B1, B2, B3, B4, B5, et B6 se trouve sur le CD de thèse dans les répertoires suivants

```
\Design_examples\appB1_Logiciel_\
\Design_examples\appB2_Pole_\
\Design_examples\appB3_Comments_\
\Design_examples\appB4_Multivariable_\
\Design_examples\appB5_Convex_\
\Design_examples\appB6_Multivariable_\
```

respectivement. Pour la description des fichiers des exemples lisez le fichier `_README.txt` dans le répertoire correspondant. Tous les fichiers des boites à outils pour Matlab PPMaster et RMC tool présentés dans les guides d'utilisateurs B7 et B8 sont placés respectivement dans les répertoires suivantes :

```
\Software\PPMaster\
\Software\RMC_tool\
```

Files : *All files corresponding to the examples presented in the articles B1, B2, B3, B4, B5, and B6 can be found on the thesis CD in the following directories*

```
\Design_examples\appB1_Logiciel_\
\Design_examples\appB2_Pole_\
\Design_examples\appB3_Comments_\
\Design_examples\appB4_Multivariable_\
\Design_examples\appB5_Convex_\
\Design_examples\appB6_Multivariable_\
```

respectively. For detailed files description see the file `_README.txt` in corresponding directory. All files for Matlab toolboxes presented in the user's guides B7 and B8 are placed respectively to the following directories :

```
\Software\PPMaster\
\Software\RMC_tool\
```

Logiciel pour l'enseignement et le calcul du placement de pôles robuste

Hynek Procházka, Ioan Doré Landau*

1^{er} octobre 2003

Conférence Internationale Francophone d'Automatique, Nantes 2002

Résumé

Ce document présente une nouvelle méthode de synthèse des régulateurs numériques avec calibrage des fonctions de sensibilité et un logiciel associé. Le logiciel reflète les améliorations actuelles de la méthode de synthèse du régulateur. Les améliorations ont radicalement simplifiées la procédure de synthèse. Le logiciel développé peut servir comme un outil moderne d'apprentissage. La capacité du logiciel est démontrée sur un exemple réel de contrôle d'un bras souple.

Mots-clé : logiciel, placement de pôles, fonctions de sensibilité, robustesse

1 Introduction

Le placement de pôles robuste est couramment utilisé pour la synthèse des régulateurs linéaires SISO (single input, single output). La robustesse est dans ce cas traitée par le calibrage des fonctions de sensibilité : Les fonctions de sensibilité du système bouclé sont ajustées pour qu'elles rentrent dans des gabarits imposés lesquels nous garantissent la robustesse (voir [1, 2, 3]). Ceci s'obtient par un ajustement itératif des pôles de la boucle fermée et des parties fixes du régulateur.

Actuellement, un nouvel outil d'ajustement a été développé. L'amélioration de la méthode est basée sur

l'observation qu'on peut ajuster simultanément les parties fixes du régulateur et les pôles auxiliaires de la boucle fermée car cet ensemble apparaît en cascade comme un atténuateur bande étroite de la fonction de sensibilité. Une fois déterminée l'atténuation nécessaire à introduire dans une certaine zone de fréquence, la valeur des pôles additionnels de la boucle fermée et des zéros (pôles) des parties fixes additionnelles du régulateur résulte de la synthèse d'un filtre bande étroite 2zéros/2pôles.

des pôles et de partie fixe résulte d'une synthèse de filtre 2zéros/2pôles bande étroite ("stop bande" filtre).

La synthèse par le placement de pôles avec calibrage des fonctions de sensibilité est un processus itératif (consistant des modifications des paramètres et des vérifications des spécifications). L'informatique nous offre une solution pour ce type des tâches : Les logiciels interactifs avec une interface utilisateur appropriée. Ces logiciels sont des programmes tournant dans une boucle infinie d'attente. Le programme attend les actions d'utilisateur et quand l'utilisateur agit, il lance une routine de réaction. Lorsque la routine est finie le programme revient dans la boucle. Cette approche interactive peut apporter des améliorations significatives aussi au calcul des régulateurs.

Nous avons développé un logiciel interactif pour la méthode de placement de pôles robuste avec des résultats très satisfaisants. L'interaction simplifie et accélère la synthèse et ainsi même un utilisateur connaissant seulement quelques règles principales peut obtenir un bon régulateur assez rapidement. Par ailleurs, l'interaction met en évidence (par visualisation) la logique des changements des propriétés du système bouclé produit par des différents ajustements. L'apport du logiciel est donc aussi éducatif et explicatif. L'efficacité du logiciel est démontrée sur un exemple connu du bras souple (le même modèle est utilisé dans [3] et [4]).

Le texte est organisé comme suit : La section 2 intro-

*Laboratoire d'Automatique de Grenoble (CNRS-INPG-UJF), ENSIEG, BP. 46 38402 St. Martin d'Hères, FRANCE; e-mail : hynek.prochazka@lag.ensieg.inpg.fr, landau@lag.ensieg.inpg.fr

duit la nouvelle méthode de synthèse par placement de pôles avec calibrage des fonctions de sensibilité. Dans la section 3 le logiciel et son fonctionnement est présenté, ensuite un exemple d'utilisation du logiciel est traité dans la section 4. Finalement, la section 5 donne quelques conclusions.

2 Placement de pôles avec calibrage des fonctions de sensibilité

2.1 Principe de placement de pôles robuste

Une des configurations standard du placement de pôles utilisant le régulateur polynômial R-S est présentée sur la Fig.1. La fonction de transfert G du modèle a la forme suivante :

$$G(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})} = \frac{z^{-d}(b_1 z^{-1} + \dots + b_{n_B} z^{-n_B})}{1 + a_1 z^{-1} + \dots + a_{n_A} z^{-n_A}} \quad (1)$$

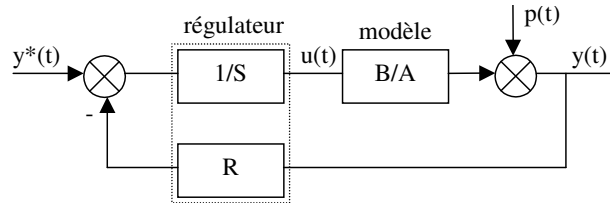


Fig. 1 : Schéma de la boucle fermée.

Le régulateur R-S est décrit par la fonction de transfert :

$$\frac{R(z^{-1})}{S(z^{-1})} = \frac{R_0(z^{-1})H_R(z^{-1})}{S_0(z^{-1})H_S(z^{-1})} \quad (2)$$

où H_R et H_S décrivent les parties fixes du régulateur (imposées pour garantir les performances spécifiées et la robustesse). R_0 et S_0 sont les solutions de l'équation de Bezout (pôles de la boucle fermée) :

$$A(z^{-1})S_0(z^{-1})H_S(z^{-1}) + B(z^{-1})R_0(z^{-1})H_R(z^{-1}) = P(z^{-1}) = P_D(z^{-1})P_F(z^{-1}) \quad (3)$$

où P définit les pôles fixés désirés de la boucle fermée partagés dans deux parties : P_D définit les pôles dominants (les plus lents) déterminant la dynamique essen-

tielle du système et P_F définit les pôles auxiliaires assurant les autres propriétés du système bouclé.

Pour garantir la robustesse et aussi les autres propriétés du régulateur, on impose des bornes (gabarits) sur les fonctions de sensibilité (fonction de transfert entre la perturbation et le signal à analyser). En général, dans le placement de pôles robuste les deux fonctions de sensibilité suivantes sont prises en compte [1] :

– fonction de sensibilité perturbation-sortie $y(z^{-1}) = S_{yp}(z^{-1})p(z^{-1})$:

$$S_{yp}(z^{-1}) = \frac{A(z^{-1})S_0(z^{-1})H_S(z^{-1})}{P(z^{-1})} \quad (4)$$

– fonction de sensibilité perturbation-entrée $u(z^{-1}) = S_{up}(z^{-1})p(z^{-1})$:

$$S_{up}(z^{-1}) = \frac{-A(z^{-1})R_0(z^{-1})H_R(z^{-1})}{P(z^{-1})} \quad (5)$$

L'élément nouveau dans la procédure de synthèse est l'introduction d'une nouvelle paramétrisation des fonctions de sensibilité. Pour réaliser l'ajustement nous avons proposé la paramétrisation suivante : $H_R = H'_R \gamma_R$, $H_S = H'_S \gamma_S$, $P_F = P'_F \delta_R$ ou $P_F = P''_F \delta_S$. Par cette notation nous obtenons :

$$|S_{yp}(z^{-1})| = \left| \frac{A(z^{-1})S_0(z^{-1})H'_S(z^{-1})}{P''_F(z^{-1})} \right| \left| \frac{\gamma_S(z^{-1})}{\delta_S(z^{-1})} \right| \quad (6)$$

$$|S_{up}(z^{-1})| = \left| \frac{A(z^{-1})R_0(z^{-1})H'_R(z^{-1})}{P'_F(z^{-1})} \right| \left| \frac{\gamma_R(z^{-1})}{\delta_R(z^{-1})} \right| \quad (7)$$

où les filtres $F_{yp}(z^{-1}) = \frac{\gamma_S(z^{-1})}{\delta_S(z^{-1})}$ et $F_{up}(z^{-1}) = \frac{\gamma_R(z^{-1})}{\delta_R(z^{-1})}$ contiennent plusieurs filtres 2zéros/2pôles faciles à ajuster. Il s'agit des filtres "stop bande" avec 2zéros/2pôles pour lesquels nous avons développé un algorithme de calcul. Dans cet algorithme la fréquence du "stop bande" f_t , l'atténuation du "stop bande" M_t , la période d'échantillonnage T_e , et l'amortissement du dénominateur ζ_{den} doivent être spécifiées et ensuite le filtre est calculé : $\{f_t, M_t, T_e, \zeta_{den}\} \rightarrow F_{stop-bande}(z^{-1})$. L'avantage de ce filtre c'est qu'il atténue dans la zone des fréquences du "stop bande" et son influence sur les autres zones est très faible. Ainsi en l'utilisant, il est possible d'éliminer séparément plusieurs dépassements du gabarit d'une fonction de sensibilité.

Les polynômes H'_R et H'_S représentent les parties fixes déterminées en fonction des spécifications (une demande d'action intégrale exige un intégrateur rajouté dans H'_S ou pour l'ouverture de la boucle à la fréquence $\frac{1}{2T_s}$ il faut

mettre un pôle réel à -1 dans H_R) et des contraintes de robustesse. Aussi pour le choix des pôles auxiliaires dans P'_F (P''_F), ils existent quelques règles de base [1]. Par exemple 1 pôle réel multiple atténue les hautes fréquences de $|S_{up}|$ (une des spécifications typiques).

2.2 Synthèse du régulateur robuste par placement de pôles

La synthèse par placement de pôles avec calibrage des fonctions de sensibilité est un processus itératif et en général il comporte les étapes suivantes :

- 1) Détermination des performances désirées pour le système en boucle fermée et formulation de ces performances en termes des spécifications de la synthèse. Comme spécifications de synthèse sont considérées : Les gabarits sur $|S_{yp}|$, $|S_{up}|$, la fréquence naturelle, l'amortissement minimal des pôles dominants (ils sont liés aux spécifications de la dynamique en temps : Temps de montée, temps de rejet, dépassements), etc.
- 2) Ajustement des paramètres de synthèse d'après les spécifications définies. Les paramètres de synthèse sont : H_S , H_R , P_D , P_F , F_{yp} , F_{up} .
- 3) Calcul du régulateur par placement de pôles à partir de l'équation (3) et son application sur le modèle.
- 4) Vérification des spécifications de synthèse. Si elles ne sont pas satisfaites, revenir au pas 2).

3 Logiciel pour placement de pôles robuste

Le terme d'interaction signifie dans le domaine de l'informatique, que le logiciel attend l'action de l'utilisateur, et ensuite réagit sur cette action. Dans notre cas, l'approche interactive est utilisée pour la mise en oeuvre du placement de pôles avec calibrage des fonctions de sensibilité. Les modifications (ajustements) des paramètres de synthèse (P_D , H_S , ...) sont considérés comme les actions de l'utilisateur et la ré-affichage des propriétés recalculées d'après les nouveaux paramètres est la réaction du logiciel.

Le logiciel est réalisé dans l'environnement de *Matlab* 5.3 en utilisant les boîtes à outils : De l'interface utilisateur, de langage de programmation, de traitement de polynômes, etc. Lors de la programmation nous avons profité de l'approche "orientée objet" qui est favorisée dans les nouvelles versions de *Matlab*.

Nom	Description	Formes prévues
P_D	pôles dominants	i) paire des pôles complexes définie par f_n, ζ
P_{Fi}	pôles auxiliaires	i) paires des pôles complexes définie par f_n, ζ ii) pôle multiple $(1 - \alpha z^{-1})^N$
H_{Ri}	partie fixe de R	i) zéro simple $(1 - \alpha z^{-1})$ ii) paire des zéros complexes définie par f_n, ζ
H_{Si}	partie fixe de S	i) pôle simple $(1 - \alpha z^{-1})$ ii) paire des zéros complexes définie par f_n, ζ
F_{ypi}	2zeros/2pôles filtre sur S_{yp}	i) fonctions de transfert déterminée par M_t, f_t, ζ_{den}
F_{upi}	2zeros/2pôles filtre sur S_{up}	i) fonctions de transfert déterminée par M_t, f_t, ζ_{den}

Tab. 1 : Paramètres accessibles de synthèse et ses valeurs à ajuster (f_n est la fréquence naturelle et ζ est l'amortissement d'une paire complexe).

Dans le programme on a l'accès direct aux paramètres ajustable (à travers de "sliders") de synthèse et parallèlement on visualise les propriétés principales du système bouclé (par les "axis") où les spécifications de synthèse peuvent être vérifiées. Les paramètres de synthèse accessibles dans le logiciel sont donnés dans le Tableau 1 et les propriétés visualisées pour la vérification des spécifications sont présentées dans le Tableau 2. La fenêtre principale du logiciel est montrée dans la Fig. 2.

4 Exemple d'utilisation - Bras souple

Le modèle considéré est le même que celui considéré dans [3], [4]. C'est un modèle du bras souple de 360° sans charge. Il possède des 3 modes vibratoires et il a l'ordre 6 (l'ordre du numérateur $n_B = 6$, du dénominateur $n_A = 6$). La réponse fréquentielle du modèle est présentée dans la Fig. 3, la carte des pôles et des zéros est montrée dans la Fig. 4. La période d'échantillonnage est $T_e = 0.05s$.

4.1 Spécifications de la synthèse

Les spécifications de synthèse pour le bras sont les suivantes :

Nom	Description	Param. influant la propriété
$ S_{yp} $	v.abs. de la fonction de sens. perturb.-sortie + gabarit	P_D, P_F H_S, F_{yp}
$ S_{up} $	v.abs. de la fonction de sens. perturb.-entrée + gabarit	P_D, P_F H_R, F_{up}
P	pôles de boucle fermée	P_D, P_F F_{yp}, F_{up}

Tab. 2 : Les propriétés visualisées par le logiciel

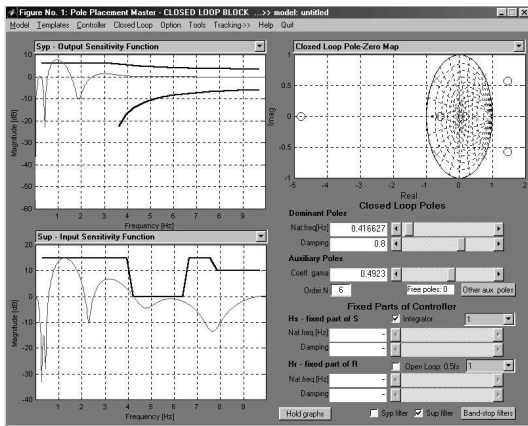


Fig. 2 : Fenêtre principale du logiciel

S1) *dynamique de BF - pôles dominants* : Ne pas avoir les pôles dominants de la BF plus lents, que ceux du modèle, amortissement de ces pôles $\zeta_D = 0.8$.

S2) *dynamique de BF - pôles auxiliaires* : Ne pas exciter les modes vibratoires hautes fréquences du bras, c'est à dire de travailler en boucle ouverte aux fréquences de ces modes vibratoires.

S3) *erreur statique* : Ne pas avoir une erreur statique.

S4) *gabarits sur $|S_{yp}|$* : Avoir la marge de module $\Delta M \geq 0.5$ et la marge de retard $\Delta \tau \geq T_e$ (voir le gabarit sur $|S_{yp}|$ dans la Fig. 5).

S5) *gabarits sur $|S_{up}|$* : 15dB en bas fréquences, 0dB de 4.5 à 6.5Hz (pour éliminer l'action du régulateur dans la zone où le gain du procédé est faible) et 10dB de 8 à 10Hz (voir le gabarit sur $|S_{up}|$ dans la Fig. 5)

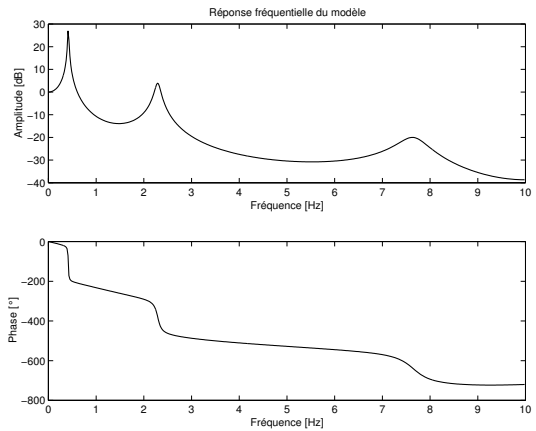


Fig. 3 : Réponse fréquentielle du modèle de bras souple

Initialement, on fixe les paramètres de synthèse d'après les spécifications désirées. Les spécifications S1, S2 et S3 seront pris en compte dans les paramètres de la synthèse comme suit :

P_D pôles dominants de BF (S1) : même fréquence naturelle $f_D = 0.4166\text{Hz}$ que celle des pôles dominants du modèle mais avec l'amortissement désiré $\zeta_D = 0.8$

P_{F1} pôles auxiliaires 1 (S2) : 2 paires des pôles complexes aux fréquences $f_{F1} = 2.2922\text{Hz}$, $f_{F2} = 7.6580\text{Hz}$ et avec les amortissements $\zeta_{F1} = 0.02491$, $\zeta_{F2} = 0.03824$ correspondant au 2^e et 3^e modes vibratoires du modèle.

H'_S une des parties fixes de S (S3) : 1 pôle $(1 - z^{-1}) \equiv$ l'intégrateur dans S.

4.2 Calcul du régulateur - ajustement des paramètres

Par la détermination initiale des paramètres de synthèse, les spécifications S1, S2 et S3 sont garanties. Il reste de vérifier les gabarits sur $|S_{yp}|$ et $|S_{up}|$ (spécifications S4 et S5). Les courbes des $|S_{yp}|$, $|S_{up}|$ pour les paramètres de synthèse initiaux sont dans la Fig. 5 (courbes en pointillé), et elles dépassent largement les gabarits imposés surtout en hautes fréquences.

Pour éliminer les dépassements on applique un filtre passe-bas en forme d'un pôle auxiliaire multiple $P_{F2} = (1 - \alpha z^{-1})^N$ de boucle fermée où $0.05 \leq \alpha \leq 0.5$. En ce qui concerne du nombre des pôles N on a décidé d'utiliser tous

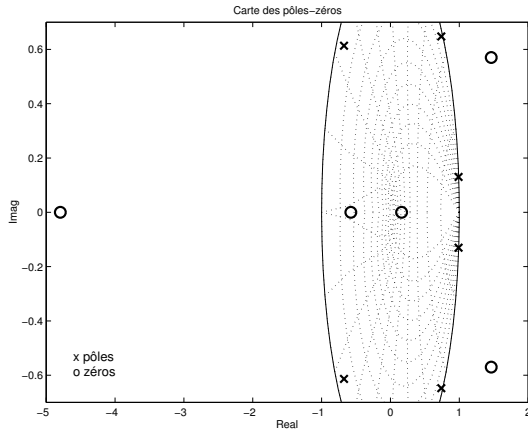


Fig. 4 : Carte des pôles et des zéros du modèle de bras souple

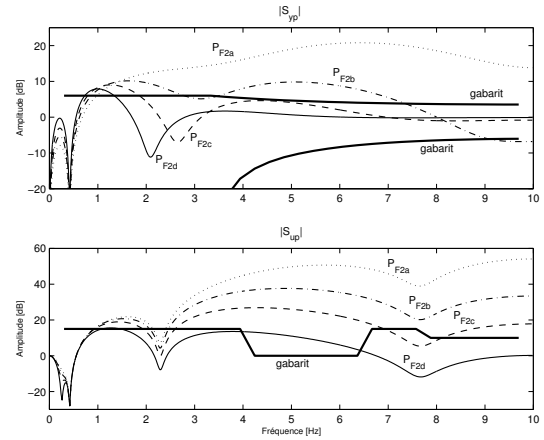


Fig. 5. Fonctions de sensibilité lors de l'itération du pôle auxiliaire multiple P_{F2}

les pôles "libres" pour obtenir une pente d'atténuation du filtre la plus raide : $N = 6$. Ensuite, la valeur appropriée de α est déterminée par une itération utilisant le logiciel. Un exemple des pas de l'itération est montré dans la Fig. 5. Les valeurs de α pour ces pas sont les suivantes :

- P_{F2a} : $\alpha_a = 0$ (sans pôle aux.), courbe en pointillé
- P_{F2b} : $\alpha_b = 0.20$, courbe en tiret-pointillé
- P_{F2c} : $\alpha_c = 0.35$, courbe en tiret
- P_{F2d} : $\alpha_d = 0.49$, courbe solide

Nous avons choisi comme la valeur optimale $\alpha_d = 0.49$. Le pôle auxiliaire multiple introduit dans P_F est ainsi : $P_{F2} = (1 - 0.49z^{-1})^6$.

En examinant les courbes de $|S_{yp}|$ et de $|S_{up}|$, on observe que $|S_{yp}|$ dépasse le gabarit imposé de $+2dB$ autour de $1Hz$ et $|S_{up}|$ dépasse de $+14dB$ autour $4.5Hz$. Nous éliminerons ces dépassements par deux filtres de 2zéros/2pôles de "stop bande" F_{yp1} et F_{up1} avec les propriétés suivantes :

Les choix de M_t sont déterminés par l'atténuation désirée. Le filtre F_{yp1} est placé à peu près à $1Hz$ (atténuation effective $-3dB$) et F_{up1} autour de $4.5Hz$ avec l'atténuation effective de $-16dB$. Les caractéristiques des filtres calculés sont présentées dans le Tab. 3. Pour les obtenir il faut itérer avec chaque filtre comme dans le cas du pôle auxiliaire P_{F2} . La Fig. 6 présente un exemple de telle itération pour le filtre F_{up1} .

Les valeurs des paramètres du filtre F_{up1} pour les pas de l'itération sont les suivantes :

- F_{up1a} : sans filtre, courbe en pointillé

F_{yp1} : filtre sur S_{yp}

Param. : $M_t = -5.7dB$, $f_t = 1Hz$, $\zeta_{den} = 0.806$

γ_{S1} : $f_0 = 1.00298Hz$, $\zeta = 0.42357$

δ_{S1} : $f_0 = 1.01076Hz$, $\zeta = 0.80118$

F_{up1} : filtre sur S_{up}

Param. : $M_t = -16.4dB$, $f_t = 4.7Hz$, $\zeta_{den} = 0.9$

γ_{R1} : $f_0 = 4.71747Hz$, $\zeta = 0.09269$

δ_{R1} : $f_0 = 6.38469Hz$, $\zeta = 0.73909$

Tab. 3 : Les paramètres des filtres F_{up1} , F_{yp1}

- F_{up1b} : $f_{tb} = 4.5Hz$, $M_{tb} = -14dB$, $\zeta_b = 0.906$, courbe en tiret-pointillé

- F_{up1c} : $f_{tc} = 4.5Hz$, $M_{tc} = -16.4dB$, $\zeta_c = 0.906$, courbe en tiret

- F_{up1d} : $f_{td} = 4.7Hz$, $M_{td} = -16.4dB$, $\zeta_d = 0.906$, courbe solide

Après les deux itérations des filtres F_{yp1} et F_{up1} , les spécifications S4 et S5 sont bien vérifiées, c'est à dire les fonctions de sensibilités rentrent dans les gabarit imposés (voir la Fig. 7). Le régulateur final est d'ordre 10 ($n_R = 10$, $n_S = 10$) et il rassure toutes les performances désirées exprimées par les spécifications S1 - S5.

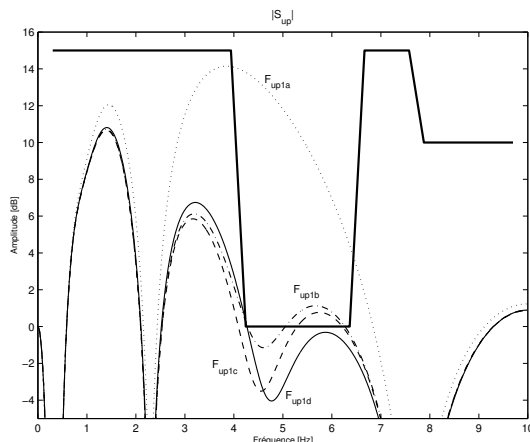


Fig. 6. Fonction de sensibilité $|S_{up}|$ lors de l'itération du filtre F_{up1}

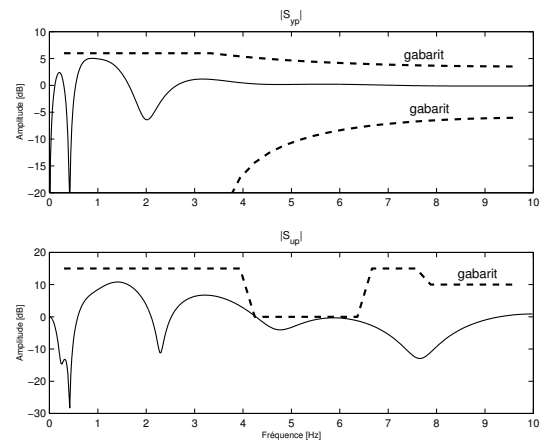


Fig. 7. Les fonctions de sensibilité pour le régulateur final

5 Conclusion

Une amélioration significative de la méthode de synthèse par placement de pôles avec calibrage des fonctions de sensibilité a été proposée dans cette communication. Cette amélioration considère l'ajustement simultané des parties fixes du régulateur et les pôles auxiliaires comme une cascade des filtres "stop bande" de type 2zéros/2pôles.

Cette communication présente aussi un logiciel interactif qui met en oeuvre cette méthodologie. Grâce à lui la synthèse est commode et le temps de synthèse est diminué.

Les potentialités de la nouvelle méthode et du logiciel de mise en oeuvre ont été illustrées sur un exemple du bras souple. Les résultats obtenus sont entièrement comparable avec ceux obtenus par les autres approches dans [4, 3] mais l'obtention de résultat est beaucoup plus simple.

Références

- [1] Landau, I.D. *Identification et commande des systèmes*. Hermès, Paris, 1993
- [2] I. D. Landau, and A. Karimi, « Robust digital control using pole placement with sensitivity shaping method, » *Int.J. Robust Nonlinear Control*, vol. 4, pp. 191–210, 1998

- [3] I. D. Landau, and J. Langer, and D. Rey, and J. Barnier, « Robust control of a 360 flexible arm using the combined pole placement/sensitivity function shaping method, » *IEEE Trans. Control Systems Technol*, vol. 4, pp. 369–383, 1996
- [4] J. Langer, and I.D. Landau, « Combined pole placement/sensitivity function shaping method using convex optimisation criteria, » *Automatica*, vol. 35, pp. 1111–1120

Pole placement with sensitivity function shaping using 2nd order digital notch filters

Hynek Procházka, Ioan Doré Landau*

October 1, 2003

Automatica, 39(6), pages 1103-1107, 2003

Abstract

This technical note deals with the design of robust digital controllers using pole placement with sensitivity shaping by means of the use of digital notch (band-stop) filters. The use of this type of filters drastically simplifies the effective shaping of the sensitivity functions and the resulting procedure competes with the procedure based on the use of Q-parameterization combined with convex optimization [Langer and Landau(1999)]. The feasibility of the technique is illustrated by its application to the control of a very flexible arm.

Keywords: *digital notch filter; robust control; pole placement; sensitivity function shaping*

1 Introduction

Pole placement with sensitivity function shaping is a convenient procedure for robust control design of SISO systems. The specifications, such as robust stability or closed loop performances, are set in terms of the dominant poles of the closed loop, the fixed part of the controller resulting from design specifications (integrator, opening of the loop etc.) and desired templates on the sensitivity functions (output sensitivity function and input sensitivity function are generally considered).

In general, the nominal design using Bezout equation for the given dominant poles does not allow to

verify the constraint on the sensitivity functions. Introduction of an additional fixed part in the controller (which appears as zeros in the sensitivity functions) and additional closed loop auxiliary poles (which appear as poles of the sensitivity functions) allows to shape the sensitivity functions. The first method based on an iterative adjustment of these elements has been described and applied in [Landau et al.(1996)], [Landau and Karimi(1998)]. A significant improvement has been achieved by using Q-parameterization combined with convex optimization for finding automatically the additional fixed controller parts and auxiliary poles to be added [Langer and Landau(1999)]. This method is now currently used for applications but it requires the use of an elaborate software related to the use of the convex optimization.

Our concern is to show in this note that an elementary and efficient approach for design of pole placement with sensitivity function shaping can be developed. The origin of this approach has its roots in an observation made on many designs using Q-parameterization combined with the convex optimization approach. The resulting sensitivity functions obtained by this design technique often feature as factors 2nd order notch filters (2poles/2zeros band-stop filters with limited attenuation). Therefore, instead of tuning alternatively fixed parts in the controllers (zeros of sensitivity functions) and auxiliary poles we propose an elementary design approach based on direct tuning of digital notch filters, which will give *in fine* the fixed parts to be introduced in the controller and the additional closed loop poles. The resulting performances and complexity of the controller match those obtained by Q-parameterization combined with convex optimization.

The technical note is organized as follows: In section 2 we re-write the basic equations for pole placement in order to emphasize the presence of notch filters and describe the design approach. In section 3 we briefly present the

*Laboratoire d'Automatique de Grenoble (CNRS-INPG-UJF) - ENSIEG - Domaine Universitaire - BP. 46, 38402 St. Martin d'Hères - Cedex, FRANCE, e-mail: hynek.prochazka@lag.ensieg.inpg.fr, landau@lag.ensieg.inpg.fr

design used for the band-stop filters. Section 4 illustrates the application of the design method for the control of a 360° flexible arm.

2 Pole placement with sensitivity function shaping using band-stop filters

A standard digital pole placement configuration using a polynomial controller (denoted R-S) is shown in Fig.1. The plant model $G(z^{-1})$ is of the form:

$$G(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})} = \frac{z^{-d}(b_1 z^{-1} + \dots + b_{n_B} z^{-n_B})}{1 + a_1 z^{-1} + \dots + a_{n_A} z^{-n_A}} \quad (1)$$

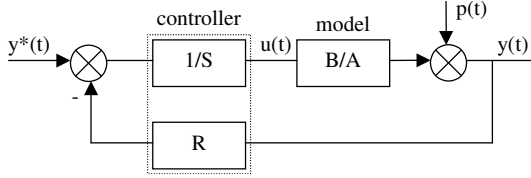


Figure 1: Considered closed loop configuration

The R-S controller has the transfer function:

$$\frac{R(z^{-1})}{S(z^{-1})} = \frac{R_0(z^{-1})H_R(z^{-1})}{S_0(z^{-1})H_S(z^{-1})} \quad (2)$$

where $H_R(z^{-1})$ and $H_S(z^{-1})$ denote the fixed parts of the controller (either imposed by the design or introduced in order to shape the sensitivity functions). $R_0(z^{-1})$ and $S_0(z^{-1})$ are solutions of the Bezout equation (poles of the closed loop):

$$A(z^{-1})S_0(z^{-1})H_S(z^{-1}) + B(z^{-1})R_0(z^{-1})H_R(z^{-1}) = P(z^{-1}) = P_D(z^{-1})P_F(z^{-1}) \quad (3)$$

where $P(z^{-1})$ represents the desired closed loop poles, $P_D(z^{-1})$ defines the dominant poles (specified) and $P_F(z^{-1})$ defines the auxiliary poles (which in part can be specified by design specifications and the remaining part is introduced in order to shape the sensitivity function).

The output sensitivity function is given by:

$$S_{yp}(z^{-1}) = \frac{A(z^{-1})S_0(z^{-1})H_S(z^{-1})}{P(z^{-1})} \quad (4)$$

The input sensitivity function is given by:

$$S_{up}(z^{-1}) = \frac{-A(z^{-1})R_0(z^{-1})H_R(z^{-1})}{P(z^{-1})} \quad (5)$$

We can now introduce the following parameterization:

$$\begin{aligned} H_R(z^{-1}) &= H'_R(z^{-1})\gamma_R(z^{-1}) \\ H_S(z^{-1}) &= H'_S(z^{-1})\gamma_S(z^{-1}) \\ P_F(z^{-1}) &= P'_F(z^{-1})\delta_R(z^{-1}) \\ P(z^{-1}) &= P''_F(z^{-1})\delta_S(z^{-1}) \end{aligned}$$

With these notations we get:

$$|S_{yp}(z^{-1})| = \left| \frac{A(z^{-1})S_0(z^{-1})H'_S(z^{-1})}{P_D P''_F(z^{-1})} \right| \left| \frac{\gamma_S(z^{-1})}{\delta_S(z^{-1})} \right| \quad (6)$$

$$|S_{up}(z^{-1})| = \left| \frac{A(z^{-1})R_0(z^{-1})H'_R(z^{-1})}{P_D P'_F(z^{-1})} \right| \left| \frac{\gamma_R(z^{-1})}{\delta_R(z^{-1})} \right| \quad (7)$$

where the filters $F_{yp}(z^{-1}) = \frac{\gamma_S(z^{-1})}{\delta_S(z^{-1})}$ and $F_{up}(z^{-1}) = \frac{\gamma_R(z^{-1})}{\delta_R(z^{-1})}$ consist of several 2nd order notch filters (2zeros/2poles band-stop filters with limited attenuation) simultaneously tuned. The tuning means in fact searching for appropriate frequency characteristics of $|F_{yp}(z^{-1})|$ and $|F_{up}(z^{-1})|$. Specifically in our case, we are interested in frequency band-stop with limited attenuation characteristics and thus the tuning concerns the frequency of band-stop, its bandwidth and the maximum attenuation in the band-stop frequency.

The rationale of this idea is the fact that we would like to lower the modulus of the sensitivity function around a certain frequency but also we would like that the "water bed effect" (in case of S_{yp}) be localized in the neighborhood of this frequency region. In other words, no influence on any frequency region far from the central band-stop frequency is expected.

Remark: Adding one 2nd order notch filter will increase the controller complexity by 2 (two additional poles and two additional zeros appear in the final controller). The reason is that the notch filter numerators $\gamma_R(z^{-1})$, $\gamma_S(z^{-1})$ are introduced to the controller fixed parts $H_R(z^{-1})$, $H_S(z^{-1})$, which in consequence rises the order of $R(z^{-1})$ and $S(z^{-1})$.

The design methodology for the shaping of the sensitivity function in the frequency domain can be summarized as follows (for the case of S_{yp}):

- 1) Design a notch filter $\frac{\gamma_S(z^{-1})}{\delta_S(z^{-1})}$ which is able to introduce the desired attenuation in the pre-specified frequency region. We determine first the frequency

zone where the sensitivity function need to be attenuated (band-stop frequency and bandwidth determination) and we determine the required attenuation in this zone. The obtained values then serve as the desired notch filter properties.

2) Solve the Bezout equation:

$$\begin{aligned} A(z^{-1})S_0(z^{-1})H'_S(z^{-1})\gamma_S(z^{-1}) + \\ + B(z^{-1})R_0(z^{-1})H_R(z^{-1}) = \\ P_D(z^{-1})P'_F(z^{-1})\delta_S(z^{-1}) \end{aligned} \quad (8)$$

3) Apply the controller. If the results are satisfactory, move to next frequency region requiring reduction of the sensitivity function. If not re-tune the characteristics of the notch filter.

For the shaping of the input sensitivity function the procedure is similar, but equation (8) is replaced by:

$$\begin{aligned} A(z^{-1})S_0(z^{-1})H_S(z^{-1}) + \\ + B(z^{-1})R_0(z^{-1})H'_R(z^{-1})\gamma_R(z^{-1}) = \\ = P_D(z^{-1})P'_F(z^{-1})\delta_R(z^{-1}) \end{aligned} \quad (9)$$

One needs therefore to give a method for the design of the 2nd order notch filter once its characteristics have been defined.

3 Design of 2nd order digital notch filter

A number of digital filter design methods used in digital signal processing (see for example [Oppenheim and Schafer(1975)] or [Taylor et al.(1998)]) require a design of an analog band-stop filter followed by an analog to digital conversion using bilinear transformation. Here, we will apply the same approach.

2nd order analog notch (band-stop) filter: We are interested in band-stop filters centered around a frequency ω_0 and with limited band-stop attenuation (frequency response shape of such filter is in Fig.2). The transfer function is in this case:

$$F(s) = \frac{s^2 + 2\zeta_{num}\omega_0 s + \omega_0^2}{s^2 + 2\zeta_{den}\omega_0 s + \omega_0^2} \quad (10)$$

with $\zeta_{num}, \zeta_{den} \in (0, 1)$ and $\omega_0 \in (0, \frac{\pi}{T_s})$ which implies 2 complex conjugate poles and zeros. The sampling time of the controlled system T_s is considered.

An attenuation around ω_0 is obtained for $\zeta_{num} < \zeta_{den}$. The minimum of $|F(j\omega)|$ is denoted M_{tlin} and its expression is given by:

$$M_{tlin} = |F(j\omega)|_{min} = |F(j\omega_0)| = \frac{\zeta_{num}}{\zeta_{den}} \quad (11)$$

Bilinear Transformation (BiTr) is used for obtaining the equivalent digital filter. The reason of using bilinear transformation is that it preserves to large extent the analog frequency response. Hence, the digital filter frequency response shape and its continuous time equivalent are very similar but frequency warping has to be taken into account. For more details see [Oppenheim and Schafer(1975)]. The used BiTr is:

$$s = \frac{2}{T_s} \frac{1 - z^{-1}}{1 + z^{-1}} \quad (12)$$

By applying the BiTr to the analog band-stop filter (10) one gets the following digital filter:

$$F(z^{-1}) = \frac{b_{z0} + b_{z1}z^{-1} + b_{z2}z^{-2}}{a_{z0} + a_{z1}z^{-1} + a_{z2}z^{-2}} = \frac{\gamma(z^{-1})}{\delta(z^{-1})} \quad (13)$$

where the coefficients a_i and b_i are given by:

$$\begin{aligned} b_{z0} &= \frac{4}{T_s} \left(\frac{1}{T_s} + \zeta_{num}\omega_0 \right) + \omega_0^2 \\ b_{z1} &= 2 \left(\omega_0^2 - \frac{4}{T_s^2} \right) \\ b_{z2} &= \frac{4}{T_s} \left(\frac{1}{T_s} - \zeta_{num}\omega_0 \right) + \omega_0^2 \\ a_{z0} &= \frac{4}{T_s} \left(\frac{1}{T_s} + \zeta_{den}\omega_0 \right) + \omega_0^2 \\ a_{z1} &= 2 \left(\omega_0^2 - \frac{4}{T_s^2} \right) \\ a_{z2} &= \frac{4}{T_s} \left(\frac{1}{T_s} - \zeta_{den}\omega_0 \right) + \omega_0^2 \end{aligned} \quad (14)$$

The poles and zeros (its continuous time equivalents) natural frequencies and dampings for the digital filter (13) are close to the values of the analog filter equivalent (10) in the low band-stop frequencies (from 0 to $0.2/T_s$). With the band-stop frequency behind $0.25/T_s$ Hz, the values start to differ seriously. To see the effect examine the example in section 4.

It follows from the bilinear transformation properties (stability region is mapped onto stability region) that considering the values of the filter (10) as indicated ($\zeta_{num}, \zeta_{den} \in (0, 1)$ and $\omega_0 \in (0, \frac{\pi}{T_s})$) the filter poles/zeros remain stable in z-plane. As a consequence,

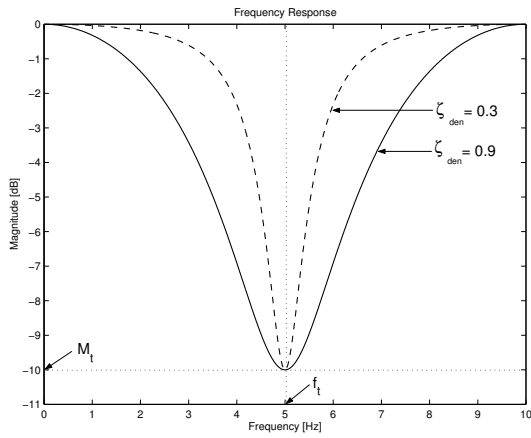


Figure 2: Digital filter design specifications

in order to obtain a stable digital filter it suffices to have the values of desired digital filter properties within the corresponding ranges (desired band-stop frequency from 0 to $\frac{1}{2T_s}$, desired denominator attenuation from 0 to 1) - see also next paragraph. The zeros stability is important in the case of S_{yp} shaping, where the filter zeros become the poles of controller.

The effective design of 2nd order digital notch filter can be summarized as follows:

- *Determination of the desired digital filter properties.*

M_t	desired band-stop attenuation, $M_t < 0$	[dB]
f_t	band-stop frequency (under condition: $0 < f_t < \frac{1}{2T_s}$)	[Hz]
T_s	sampling time	[sec]
ζ_{den}	denominator damping of the analog filter. The damping should be $\zeta_{den} \in \langle 0.3, 1 \rangle$ for our purposes. A damping close to 0.3 gives a narrow banded, filter close to 1 gives a wide band filter. The lowest value 0.3 assures a reasonable damping for the auxiliary closed loop poles.	

The properties of the filter are illustrated in Fig.2.

- *Analog filter design.* The parameters determining the analog filter are evaluated as follows:

- analog band-stop frequency (frequency warping): $\omega_0 = \frac{2}{T_s} \tan(\pi T_s f_t)$, where $\omega_0 \in (0, \infty)$ and $f_t \in (0, \frac{1}{2T_s})$.

- attenuation (conversion to linear scale):

$$M_{tlin} = 10^{\frac{M_t}{20}}$$

- damping of the numerator: $\zeta_{num} = \zeta_{den} M_{tlin}$

From the known values $\omega_0 = \omega_{num} = \omega_{den}$, ζ_{num} , ζ_{den} , we can now construct the analog filter (10).

- *Conversion from analog to digital transfer function.* The digital form (13) of our analog notch (band-stop) filter (10) is computed by employing the BiTr (12). Hence, for evaluation of the digital filter coefficients we utilize the equations (14).

Remark: For $f_t \leq \frac{0.15}{T_s}$ one can design with enough precision the filter directly in discrete domain. The reason is that as we already mentioned above, for low band-stop frequencies the digital filter pole/zero natural frequencies and dampings are close to the corresponding values of analog filter (10). Hence we can estimate the values of the natural frequencies and dampings for digital filter (13) directly from desired filter properties: f_t , M_t , and ζ_{den} . The estimation is as follows: $\omega_{0dnum} = \omega_{0dden} \approx 2\pi f_t$, $\zeta_{dden} \approx \zeta_{den}$, $\zeta_{dnum} = 10^{\frac{M_t}{20}} \zeta_{dden}$. The variables ω_{0dnum} , ω_{0dden} are the natural frequencies and ζ_{dnum} , ζ_{dden} are the dampings of the estimated digital filter numerator and denominator roots. Unfortunately, it is quite lengthy and difficult to show from the transformation formulae for ζ and ω_n (using (13), (14), and $z = e^{sT_s}$) that this is indeed true for low frequencies.

4 Example of application - model of 360° flexible arm

The considered model is the same as in the papers [Landau et al.(1996)], and [Langer and Landau(1999)]. It is a model of a 360° flexible arm without load with 3 vibration modes. The orders of its numerator and denominator are $n_B = 6$ and $n_A = 6$ respectively. The sampling time is 0.05 second. The used controller design specifications can be found in [Landau et al.(1996)], [Langer and Landau(1999)]. Briefly, the templates on the sensitivity functions S_{yp} , S_{up} are the same, an integrator is implemented in the controller and poles of the model's higher vibration modes are introduced as fixed auxiliary closed loop poles. More detailed explanations are given in the next subsection.

4.1 Initial controller design

To compute a nominal controller (nominal means here a controller without 2zeros/2poles band-stop filters) by pole placement the specifications are set as follows:

- *fixed dominant poles*: at the same frequency $f_d = 0.4166\text{Hz}$ as the model's slowest vibration mode but well damped, $\zeta_d = 0.8$ instead of model's damping $\zeta = 0.01833$.
- *fixed auxiliary poles - part 1*: 2 pairs of complex poles at the frequencies $f_{a1} = 2.2922\text{Hz}$, $f_{a2} = 7.6580\text{Hz}$ and with the damping $\zeta_{a1} = 0.02491$, $\zeta_{a2} = 0.03824$ corresponding to the 2nd and 3rd vibration mode of the model (the same choice as in [Langer and Landau(1999)]). The reason is that these vibration modes are beyond the closed loop system bandwidth and the arm can operate as in open loop.
- *fixed auxiliary poles - part 2*: 1 multiple real pole $(1 - 0.4923z^{-1})^6$. The pole performs attenuation of the input sensitivity function in high frequency region beyond $0.25/T_s$ Hz. In fact it restrains the controller action in higher frequencies.
- *integrator*: 1 pole $(1 - z^{-1})$ in the fixed part H_S of the controller denominator S to eliminate steady state errors.
- *templates on S_{yp}* : modulus margin (6dB) and delay margin (T_s) resulting in the template shown in Fig.3. This standard template on S_{yp} guarantees sufficient robust stability of closed loop system with phase margin $> 30^\circ$, gain margin ≥ 2 , and modulus margin ≥ 0.5 .
- *templates on S_{up}* : bound is set to 15dB in low frequencies, to 0dB from 4.5Hz to 6.5Hz (in this zone there are some uncertainties on the model) and to 10dB from 8Hz to 10Hz (to reduce the stress on the actuator). The same template was used in [Langer and Landau(1999)].

The resultant nominal controller has the orders $n_R = 6$, $n_S = 6$. The sensitivity functions are shown in Fig.3 (dotted curves). Both sensitivity functions are above the templates in some frequency region (S_{yp} in low frequencies and S_{up} in middle frequencies).

4.2 Filter design - final controller

To attenuate the sensitivity functions, 2 notch (band-stop) filters are to be designed. For S_{yp} from Fig.3 it results that we need an attenuation of approximately -5dB

around the frequency of 1Hz and for S_{up} we need an attenuation of about -16dB around 4.5Hz. After one iteration the following resulting final filters are introduced (sampling time is $T_s = 0.05\text{s}$):

$$\underline{F_{yp1}(z^{-1}) - \text{filter on } S_{yp}}$$

Design specifications:

$$M_t = -5.7\text{dB}, f_t = 1\text{Hz}, \zeta_{den} = 0.806$$

Digital filter numerator:

$$f_0 = 1.00298\text{Hz}, \zeta = 0.42357$$

Digital filter denominator:

$$f_0 = 1.01076\text{Hz}, \zeta = 0.80118$$

$$\underline{F_{up1}(z^{-1}) - \text{filter on } S_{up}}$$

Design specifications:

$$M_t = -16.4\text{dB}, f_t = 4.7\text{Hz}, \zeta_{den} = 0.906$$

Digital filter numerator:

$$f_0 = 4.71747\text{Hz}, \zeta = 0.0926902$$

Digital filter denominator:

$$f_0 = 6.38469\text{Hz}, \zeta = 0.739087$$

A controller with the orders $n_R = 10$, $n_S = 10$ is obtained by applying these 2 filters. The resulting sensitivity functions are shown on the Fig.3 (solid curves). Both imposed templates are verified.

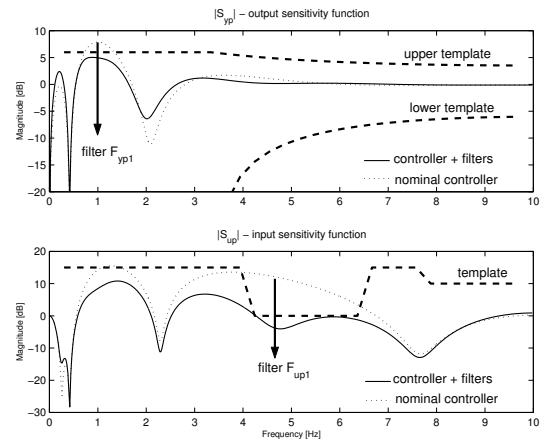


Figure 3: Resulting sensitivity functions for the nominal and the final controller

5 Conclusions

This technical note offers an elementary approach for the design of pole placement with shaping of the sensitivity functions by using 2nd order notch (band-stop) filters.

The numerator corresponds to fixed parts to be introduced in the controller and the denominator corresponds to additional closed loop poles. The efficiency of the tool was demonstrated on one example of a 360° flexible arm. The obtained results are entirely comparable (the robustness specifications are verified and controller complexity is similar: 10 instead of 8) with the results achieved by Q-parameterization combined with convex optimization [Langer and Landau(1999)] on the same example.

References

- [Landau and Karimi(1998)] Landau, I.D., and Karimi, A. (1998) Robust digital control using pole placement with sensitivity shaping method. *Int.J. Robust Nonlinear Control*, 4, 191-210
- [Landau et al.(1996)] Landau, I.D., and Langer, J., and Rey, D., and Barnier, J. (1996) Robust control of a 360 flexible arm using the combined pole placement/sensitivity function shaping method. *IEEE Trans. Control Systems Technol.*, 4(4), 369-383
- [Langer and Landau(1999)] Langer, J., and Landau, I.D. (1999) Combined pole placement/sensitivity function shaping method using convex optimisation criteria. *Automatica*, 35, 1111-1120
- [Oppenheim and Schafer(1975)] Oppenheim, A.V., and Schafer, R.W. (1975) Digital Signal Processing. *Prentice-Hall, New Jersey*
- [Taylor et al.(1998)] Taylor, S.A., and the Antena Group Inc, and Melott, J. (1998) Hands on digital signal processing. *McGraw-Hill*

Comments on the paper "Controller order reduction by identification in closed-loop applied to a benchmark problem"

Hynek Procházka*

January 15, 2004

European Journal of Control, 9(5), 2003

Abstract

This note makes some remarks and comments on the paper [1] concerning a reduced-complexity controller design for an active suspension benchmark [6]. Mainly, a more straightforward and more efficient initial controller design procedure is proposed, which provides simple controllers with excellent performances satisfying completely all design specifications. Then, a similar but simpler reduction technique is applied (without any intermediary model reduction) with very satisfactory results.

Keywords: *active suspension benchmark; controller design; controller reduction; pole placement with sensitivity shaping*

1 Introduction

The recently presented benchmark problem of active suspension control by reduced complexity controllers [6] allowed to compare different approaches of robust low-order single-input single-output (SISO) controller design. Several design methods and reduction techniques were applied [2] including H_∞ , convex optimisation, non-parametric design method and modal reduction approach. One of the best controller was obtained using convex optimisation design procedure combined with closed-loop identification based reduction [1]. In [1], the design proceeds as follows: After model identification in open loop and model reduction an initial high complexity controller,

which does not fully satisfy the design specifications in simulation, is designed using convex optimisation design technique [7]. Then, a new model is identified in closed loop [3] and the obtained model is reduced. With the reduced model another initial high-complexity controller is designed by convex optimisation. Finally, this controller is reduced applying closed-loop identification based reduction technique [5].

In this note a simpler alternative to the approach of [1] is proposed avoiding intermediary model reduction and providing simpler and high-performance initial controllers. An initial complex controller is designed using improved pole placement with sensitivity shaping [4, 9] and implemented in the toolbox "ppmaster" for MATLAB[®] [8]. This method gives quite simple initial controllers with excellent performances satisfying completely all benchmark specifications in simulation. Then, closed-loop (CL) identification based reduction [5] is applied on this controller. In next step, closed-loop identification [3] is carried out with the initial controller and another relatively simple initial controller is designed and then reduced using the closed-loop identified model.

A standard closed loop configuration with a digital polynomial controller (denoted R/S) and a plant model $G(z^{-1})$ is considered (as it is shown in Fig.2 of [1]). The plant model $G(z^{-1})$ has the form:

$$G(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})} = \frac{z^{-d}(b_1 z^{-1} \dots + b_{n_B} z^{-n_B})}{1 + a_1 z^{-1} \dots + a_{n_A} z^{-n_A}} \quad (1)$$

and the R-S controller has the transfer function:

$$\frac{R(z^{-1})}{S(z^{-1})} = \frac{r_0 + r_1 z^{-1} \dots + r_{n_r} z^{-n_r}}{1 + s_1 z^{-1} \dots + s_{n_s} z^{-n_s}} \quad (2)$$

where the variables n_r and n_s denote respectively the order of controller numerator R and the order of controller denominator S . Hence, the number of coefficients to be

*Laboratoire d'Automatique de Grenoble (CNRS-INPG-UJF), ENSIEG; BP. 46, 38402 St. Martin d'Hères, FRANCE; e-mail: hynek.prochazka@lag.ensieg.inpg.fr; Tel: +33 4 76 82 63 87; Fax: +33 4 76 82 63 88

computed (denoted also controller complexity) is $n_r + n_s + 1$.

The considered sensitivity functions are the output sensitivity function $S_{yp} = \frac{y}{p}$ and the input sensitivity function $S_{up} = \frac{u}{p}$ (see [6]).

This note is structured as follows: Section 2 introduces briefly the active suspension benchmark, and presents the proposed controller design technique and obtained results, Section 3 gives some conclusions.

2 Active suspension reduced controller design

2.1 Model and design specifications

Briefly, the benchmark controller specifications are as follows (for more details see [6, 2]):

- Modulus of input sensitivity S_{up} and output sensitivity S_{yp} has to be bounded in frequency domain as it shows Fig.1 solid thick curves.
- The controller numerator R in (2) has to incorporate the term $(1 - z^{-1})$ opening the loop at the Nyquist frequency ($\frac{1}{2T_s}$).
- Controller orders n_r, n_s should be as low as possible but satisfying desired performance.

The discrete-time ARMAX model of secondary path given in [6], denoted here ARMAXOL, is used for design of the first controller. It has the orders $n_A = 14, n_B = 16$ and $d = 0$, the sampling time is $T_s = 0.00125s$. It contains 2 differentiators (zeros at 1) and its complex conjugated poles are given in Tab.1. Moreover, it has also 2 real poles one at 0.956 and another one at -0.218 (pole with no real physical correspondence, but without influence on performance, since controller is in open-loop at $1/2T_s$).

2.2 Initial Controller design with ARMAXOL

First of all, a controller satisfying all benchmark specifications on the model ARMAXOL is designed. The controller design procedure consists of the following steps:

Step 1: Initial pole-placement and loop opening. Initially, in order to obtain a smooth and low S_{yp} sensitivity, all the model poles are placed as poles of the closed loop. In addition, one zero at -1 is introduced to the fixed part of the controller numerator

mode	f_n [Hz]	ζ [-]
1	31.9	0.078
2	164	0.033
3	176	0.189
4	240	0.014
5	277	0.015
6	364	0.034

Table 1: Complex poles of the model ARMAXOL (f_n is natural frequency, ζ is damping)

$H_R(z^{-1}) = (1 + z^{-1})$ to assure the benchmark specification concerning the opening of the loop at $\frac{1}{2T_s}$ Hz. The resulting sensitivities are shown in Fig.1 -dotted curves.

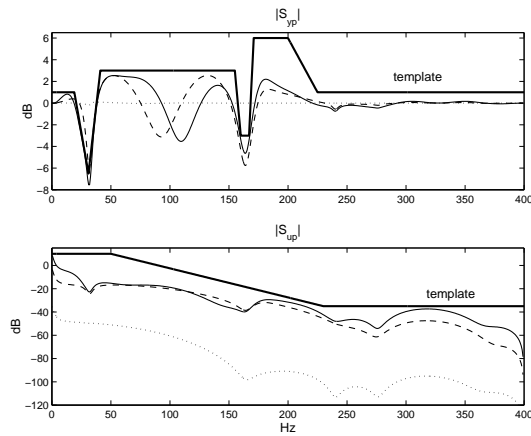


Figure 1: S_{yp} and S_{up} for different stages of controller design for the model ARMAXOL.

Step 2: Adding real CL auxiliary poles. To attenuate more low frequencies, one real poles is placed as one auxiliary closed loop pole at 0.99.

Step 3: CL pole adjustment. Some of the placed closed loop poles are to be modified in a design trial using the software package [8]. The objective of these trial is to enter the treated sensitivity functions into templates specified in the benchmark. The closed loop poles were modified as follows:

mode 1: $f_n = 31.9 \rightarrow 32.5\text{Hz}$ $\zeta = 0.078 \rightarrow 0.2$
mode 2: $f_n = 164 \rightarrow 168\text{Hz}$ $\zeta = 0.033 \rightarrow 0.1$
mode 3: $f_n = 176\text{Hz}$ $\zeta = 0.189 \rightarrow 0.08$

mode 4: $f_n = 240\text{Hz}$ $\zeta = 0.014 \rightarrow 0.015$

Resulting sensitivities are shown in Fig.1 -dashed curves.

Step 4: Notch filters on S_{yp} . The exceed of S_{yp} around 25Hz is eliminated by the use of two notch filters applied on S_{yp} (see [9]). The first one (F_{yp1}) attenuates the sensitivity at 26.5Hz, the other one (F_{yp2}) is necessary to reduce a sensitivity amplification around 60Hz caused by the application of the first filter. The filter properties are band-stop frequencies $f_{t1} = 26.5\text{Hz}$, $f_{t2} = 64\text{Hz}$, attenuation $M_{t1} = -5.1\text{dB}$, $M_{t2} = -7\text{dB}$, and denominator damping $\zeta_{d1} = 0.45$, $\zeta_{d2} = 0.93$.

The resulting sensitivity functions are shown in Fig.1 - solid curves. The obtained controller has the order of numerator $n_r = 18$ and the order of denominator $n_s = 20$ (simpler than the one designed in [1], which has $n_r = 21$, $n_s = 24$). The two performance criterions ΔS_{yp}^m and ΔS_{up}^m based on identified model are zero for this controller, hence in simulation the controller satisfies completely the benchmark specifications.

Remark 3.1: The obtained controller is the simplest which completely satisfies the benchmark specifications in simulation with ARMAXOL (see the results of other participants in [2]). Consequently, this initial simplicity together with the good performance may have a positive influence on controller reduction results. Also, it is not necessary to reduce the nominal model in order to obtain a controller with a reasonable complexity and good performances (a procedure applied in [1]).

Remark 3.2: A controller with the same low complexity can be designed using the convex optimisation technique [7]. In this case, we took advantage of the previous pole placement design trial using the controller of the step 3 as a central controller for Q-parametrisation. The convex optimisation then find automatically an approximation of two notch filters designed in the step 4, which also allow to satisfy the constraints.

2.3 Controller reduction

The used excitation signal is a PRBS generated with 9-bit shift register and with the frequency divisor $d = 4$. The obtained set of the best reduced controllers with their resulting criterions computed from the ARMAXOL model are presented in Tab.2.

In real-time experiments, the most satisfactory results were obtained from the controller with $n_r = n_s = 4$. The benchmark criterion was in its case $\Delta S_{yp} = 10.3$, $\Delta S_{up} = 0$, which is very close to the best obtained real-time benchmark results 9.2, 0 [2, 1].

n_r	n_s	ΔS_{yp}^m	ΔS_{up}^m
4	6	2.82	0
4	5	3.14	0
4	4	4.22	0
3	4	4.93	0

Table 2: Reduced controllers based on ARMAXOL model

2.4 Closed-loop model identification and new controller design

As in [1], the closed-loop model identification is applied. The controller used for the closed-loop identification is the initial complex controller designed by pole placement. The obtained model ARMAXCL is then used for further controller design. The initial controller satisfying completely the benchmark sensitivity constraints on the closed-loop identified model is designed as in the Section 2.2. Then, the controller is reduced. The performance criterions in simulation (denoted S_{yp}^m and S_{up}^m) of the reduced-complexity controllers are given in Tab.3.

n_r	n_s	ΔS_{yp}^m	ΔS_{up}^m	S_{yp}^r	S_{up}^r
5	6	1.42	0	4.48	0
4	4	2.42	0	10.47	0
3	3	2.73	0	10.2	0

Table 3: Reduced controllers based on ARMAXCL model.

Three obtained restricted-complexity controllers were tested in real time, the resulting criterions are also given in Tab.3 (denoted by S_{yp}^r and S_{up}^r). The two controllers with the degrees 3 and 4 have equivalent performances, thus it is preferable to hold the simplest one $n_r = n_s = 3$. This controller has better real-time performance criterion than the controller of SUPAERO (with the same complexity $n_r + n_s + 1 = 7$) and also than the controller of CERT (with higher complexity $n_r + n_s + 1 = 9$) - see [2]. The real time experiment compared with the simulation for the controller with $n_r = n_s = 3$ is shown in Fig.2.

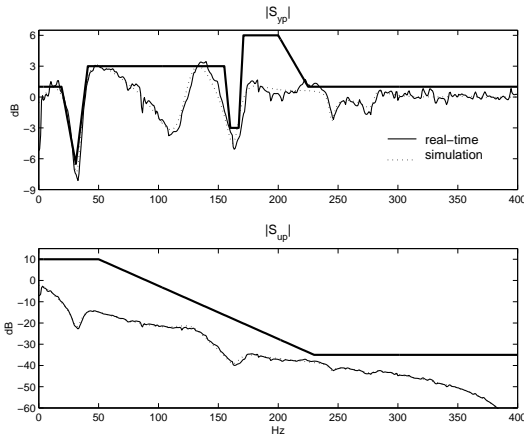


Figure 2: S_{yp} and S_{up} in simulation and in real-time for the reduced controller ($n_r = n_s = 3$) designed from ARMAXCL.

3 Conclusions

This note shows that a controller of reasonable complexity and of excellent performances can be designed directly from a high-order identified model using improved pole placement with sensitivity shaping design. Then, using controller reduction technique based on closed-loop identification one obtains a set of high-performance low-complexity controllers. Conversely to [1] and thanks to the initial controller simplicity we avoid any model reduction, which makes controller design trial simpler. Comparing with the benchmark results [2], the proposed controllers belong to the best benchmark control solutions.

References

- [1] A. Constantinescu and I.D. Landau. Controller order reduction by identification in closed loop applied to a benchmark problem. *European Journal of Control*, (1), 2003.
- [2] I.D.Landau, A. Karimi, and H. Hjalmarsson, editors. *International Workshop on Design and optimisation of restricted complexity controllers*. Laboratoire d'Automatique de Grenoble, January 2003.
- [3] I.D. Landau and A. Karimi. Recursive algorithms for identification in closed loop: a unified approach and evaluation. *Automatica*, 33(8):1499–1523, 1997.
- [4] I.D. Landau and A. Karimi. Robust digital control using pole placement with sensitivity function shaping method. *Int. J. Robust and Nonlin. Cont.*, 8:191–210, 1998.
- [5] I.D. Landau, A. Karimi, and A. Constantinescu. Direct controller order reduction by identification in closed loop. *Automatica*, 37(11):1689–1702, 2001.
- [6] I.D. Landau, A. Karimi, L. Miskovic, and H. Prochazka. Control of an active suspension system as a benchmark for design and optimisation of restricted complexity controllers. *European Journal of Control*, (1), 2003.
- [7] J. Langer and I.D. Landau. Combined pole placement/sensitivity function shaping method using convex optimization criteria. *Automatica*, pages 1111–1120, 1999.
- [8] H. Prochazka and I.D. Landau. Logiciel pour l'enseignement et le calcul du placement de pôles robuste. *Conf. Int. Francophone d'Automatique Nante*, pages 694–698, 7 2002.
- [9] H. Prochazka and I.D. Landau. Pole placement with sensitivity function shaping using 2nd order digital notch filters. *Automatica*, 39(6), 2003.

Multivariable robust controller design by pole placement and sensitivity shaping

H. Procházka, I. D. Landau*

October 1, 2003

Internal Rapport, LAG

[4, 8]. In practice, the method was successfully applied on different SISO complex systems [5, 9].

Abstract

This note review the problem of multivariable pole assignment in relation with the SISO design method of combined pole placement with sensitivity function shaping [4]. It provides several recommendations and tools simplifying choice of closed loop pole placements and shed light on its influence with regard to closed loop sensitivity. Finally, the proposed design technique is illustrated on one MIMO controller design example.

Keywords: *multivariable robust control; pole placement; sensitivity shaping; eigenstructure assignment*

1 INTRODUCTION

In single-input, single-output (SISO) controller design domain, methods based on pole placement are extensively used. The idea behind these methods is to choose desired closed loop (CL) poles and then, using an appropriate algorithm, compute a controller providing desired closed loop system dynamic. This paper concerns mainly the combined pole placement with sensitivity function shaping [4, 3, 8]. In this robust digital controller design technique, chosen sensitivity functions are shaped by shifting the closed-loop poles, to obtain required robustness and performances. Obviously, such approach demands a good knowledge of pole effects on sensitivity functions. Hence a number of tools and recommendations were developed

The pole placement or more precisely eigenstructure assignment is a method commonly used also for multivariable (multi-input, multi-output or MIMO) controller design. An essential domain of application is aircraft controller design [10, 7]. In contrast to the SISO version, MIMO eigenstructure assignment does not imply a unique solution for the controller and different algorithms were developed [2, 11, 12, 7, 10] to deal with this problem.

In this note, the procedure of combined pole placement with sensitivity function shaping [4, 8] is extended to multivariable domain. Particularly, the objective is to enhance the pole selection in multivariable pole-assignment design trial. The controller is supposed to have a standard form of observer with estimated state feedback (see for example [1]). Three different optimisation algorithms of eigenstructure assignment are tested in this paper [2, 11, 12]. Finally, a list of recommendations concerning the closed-loop pole locations is given and a related basic theory is reviewed. The proposed design technique is illustrated on one example of MIMO controller design problem.

The paper is organized as follows: First section reviews the theory related to the proposed design technique. Next section gives the list of rules and recommendation for pole placement and a short analysis of three tested pole placement optimisation algorithms. Section 4 presents the procedure of controller design step by step. In Section 5 one MIMO example demonstrates the design and finally the last section provides some conclusions.

*Laboratoire d'Automatique de Grenoble, (CNRS-INPG-UJF), ENSIEG, BP. 46, 38402 St. Martin d'Hères, FRANCE, hynek.prochazka@lag.ensieg.inpg.fr, Tel: +33 4 76 82 62 44; Fax: +33 4 76 82 63 88

2 MULTIVARIABLE POLE PLACEMENT AND SENSITIVITY SHAPING - A REVIEW

2.1 Multivariable control problem

A linear discrete time-invariant multi-input multi-output (MIMO) model $P(z)$ of a plant is considered. The model is supposed to have the following state space representation form:

$$\begin{aligned} x_p(t+1) &= Ax_p(t) + Bu_p(t) \\ y_p(t) &= Cx_p(t) + Du_p(t) \end{aligned} \quad (1)$$

with $A \in R^{n_x \times n_x}$, $B \in R^{n_x \times n_u}$, $C \in R^{n_y \times n_x}$, $D \in R^{n_y \times n_u}$. The matrices A , B , C are some constant state space matrices and $u_p(t)$, $y_p(t)$ and $x_p(t)$ are any vectors of plant inputs, outputs, and states respectively. The variables n_u , n_y , and n_x represent the number of plant inputs, outputs and states respectively. The matrix D is in the case of digitalized continuous time plants zero, hence we suppose $D = 0$. The time variable t is a discrete time: $t = kT_e$, $k = 0, 1, 2, \dots$ with a sampling time T_e . A corresponding model transfer function matrix is then:

$$P(z) = C(zI - A)^{-1}B \quad (2)$$

where z is frequency operator $z = e^{j\omega_d}$ with $\omega_d \in (0, \pi)$ a normalized discrete-time frequency $\omega_d = \omega T_e \mid \omega \in (0, \frac{2\pi}{T_e})$.

The considered closed loop system configuration to achieve any desired control performances is in Fig. 1. The block K is an appropriate MIMO controller described either in state space representation with A_K , B_K , C_K , D_K , or in the form of transfer matrix $K(z)$.

2.2 Sensitivities

Generally, output disturbance $p(t)$, measuring noise $b(t)$ and command disturbance $d(t)$ are expected to act on the system (see Fig. 1). Influence of these undesirable signals onto closed-loop output $y(t)$ and command $u(t)$ can be expressed via sensitivity functions $S_{ij}(z)$ (sensitivity matrices in MIMO case). These functions (matrices) play a crucial role in the robustness and nominal performance analysis of the CL system. Essentially, the following sensitivities are used in [3]:

- $S_{yp}(z)$: Sensitivity matrix *output disturbance - output*

$$S_{yp}(z) = \frac{y(z)}{p(z)} = (I + P(z)K(z))^{-1} \quad (3)$$

- $S_{up}(z)$: Sensitivity matrix *output disturbance - command*

$$S_{up}(z) = \frac{u(z)}{p(z)} = -K(z)(I + P(z)K(z))^{-1} \quad (4)$$

- S_{yb} : Sensitivity matrix *measurement noise - output*

$$S_{yb}(z) = \frac{y(z)}{b(z)} = -P(z)K(z)(I + P(z)K(z))^{-1} \quad (5)$$

To analyze MIMO system sensitivity, a largest singular value evolution $\bar{\sigma}(S_{ij}(e^{j\omega_d}))$ of a transfer matrix $S_{ij}(z)$ is often used instead of frequency response matrix $|S_{ij}(e^{j\omega_d})|$ (the case of H_∞ controller design method [13, 6], etc.). In order to assure closed-loop performances, we impose superior frequency-domain bounds (templates) $F_{ij}(\omega_d)$ on the sensitivity singular values $\bar{\sigma}(S_{ij}(e^{j\omega_d}))$. A general condition has following form:

$$\bar{\sigma}(S_{ij}(e^{j\omega_d})) \leq F_{ij}(\omega_d) \mid \forall \omega_d \quad (6)$$

Remark: H_∞ -optimisation design technique proceeds in the same way only the templates are replaced with weighting filters. The template can be interpreted as the inverses of weighting filter largest singular value in H_∞ . The same approach is applied in SISO pole placement with sensitivity function shaping as well. Only the largest singular value of sensitivity matrix is replaced with frequency response module of sensitivity function.

2.3 Sensitivity bounds for performances

To obtain the most typical closed-loop performances, we decided to impose bounds on sensitivity matrices. Some advices for the most typical performance specifications are listed below.

- CL robust stability - $\sup_{\omega_d} [\bar{\sigma}(S_{yp}(e^{j\omega_d}))]$ as small as possible [6]. In SISO case, a template limiting $|S_{yp}(e^{j\omega_d})| < 6\text{dB}$ in low and $< 3\text{dB}$ in high frequencies guarantees sufficient modulus margin, phase margin, and delay margin [3]. We consider a similar template shape in MIMO domain.

- Tracking - The tracking properties are the closed loop step response rising time and overshoot (defined for direct CL branches) and decoupling. All the properties can be approximately characterized by $|S_{yp}(e^{j\omega_d})|$ frequency response. The overshoot in one channel corresponds to the low frequency maximum of corresponding $|S_{yp}(e^{j\omega_d})|$ diagonal frequency response (0%, 10% overshoot requires about 0dB, 0.9dB maximum respectively). The rising time corresponds (similarly to SISO systems) to the intersection frequency of $|S_{yp}(e^{j\omega_d})|$ diagonal response with 0dB. This property is roughly observable also from $\bar{\sigma}(S_{yp}(e^{j\omega_d}))$ evolution. Finally decoupling is determined by non-diagonal maximums of $|S_{yb}(e^{j\omega_d})|$ frequency response.
- Measuring noise high frequency attenuation (roll-off) - Generally a ramp with -20dB (-40dB) per decade is required for $\bar{\sigma}(S_{yb}(e^{j\omega_d}))$. The same effect is obtained with $\bar{\sigma}(S_{yp}(e^{j\omega_d})) \rightarrow 0$ dB in high frequencies.
- Controller gain (actuator effort) - In CL, it can be expressed by $\bar{\sigma}(S_{up}(e^{j\omega_d}))$. If a lower controller gain is required in a certain frequency region (classically high frequency controller gain), one has to attenuate $\bar{\sigma}(S_{up}(e^{j\omega_d}))$ at this region.

2.4 Multivariable controller structure

Suppose that the final controller $K(z)$ contains a pre-fixed block named $H_K(z)$, which is any stable transfer matrix. Then, the final controller transfer matrix is as follows:

$$K(z) = H_K(z)K_0(z) \quad (7)$$

where $K_0(z)$ is a controller performing CL pole placement. The considered pole placement controller structure is well known MIMO controller form utilized for pole placement or LQG and presented for example in [1]. It consists of state observer (named also Luenberg observer) and estimated state feedback with observer matrix L and state feedback matrix F (see Fig. 2). The controller complexity, i.e. the number of controller states is equal to the number of model states + two times the number of pre-fixed block states ($\deg(K) = \deg(P) + 2\deg(H_K)$).

The vector signals $y_o(t)$ and $e_o(t)$ of Fig.2 are respectively estimated output and estimation error. The constant matrices A_H , B_H , C_H , and D_H are the state-space representation (SSR) matrices of the cascade connection $H(z) = P(z)H_K(z)$ (pre-fixed transfer matrix and plant

model) with the state vector $x_H = [x_{H_K}, x_p]^T$. The following SSR describes $H(z)$ and $H_K(z)$:

$$\begin{aligned} x_H(t+1) &= A_H x_H(t) + B_H u_K(t) \\ y(t) &= C_H x_H(t) \\ x_{H_K}(t+1) &= A_{H_K} x_{H_K}(t) + B_{H_K} u_K(t) \\ u(t) &= C_{H_K} x_{H_K}(t) + D_{H_K} u_K(t) \end{aligned} \quad (8)$$

The resulting pole placement controller $K_0(z)$ has this well known [1] SSR:

$$\begin{aligned} \hat{x}(t+1) &= (A_H - B_H F - LC_H) \hat{x}(t) + Ly(t) \\ u_K(t) &= -F \hat{x}(t) \end{aligned} \quad (9)$$

where $\hat{x}(t)$ is the estimation of the state vector $x(t)$.

The constant matrices L and F determine placements of closed loop poles, where eigenvalues of matrices $(A_H - LC_H)$ and $(A_H - B_H F)$ define observer and estimated state feedback closed loop poles respectively.

Unfortunately, there is not a unique solution for L and F if only desired closed-loop poles are fixed. To obtain one solution, it is necessary to fix additionally to closed-loop poles also closed-loop eigenvectors. Two approaches are currently used to place the eigenvectors. The first one defines some criterion (guaranteeing robust placement - a small shifting of model poles causes a small changes in CL pole locations, low controller gain, etc.) and, using an optimization technique, search for a criterion minimum optimizing the CL eigenvectors. Several methods based on these idea were developed [2, 11, 12]. The second approach restrains the degrees of freedom in controller solution by defining all or some parts of desired CL eigenvectors. The modal control [7] or the methods presented in [10] are typical examples of this approach. Only the first approach is considered in this text.

3 RECOMMENDATIONS AND TOOLS FOR MIMO POLE PLACEMENT

3.1 Pole-placement for desired performances

Applying the methods proposed in [2, 11, 12] for pole placement solution, the resulting controller depends only on an appropriate choice of desired closed loop poles. Correspondingly to SISO case [3] we developed a set of useful

rules for placing the poles correctly according to some desired control specifications. Below a number of rules for robust MIMO pole placement with sensitivity shaping are proposed.

Robustness - Generally, the best robustness is obtained if the model poles (if stable and well damped) and its eigenvectors are fixed as closed loop poles and eigenvectors [7]. The eigenvectors are assigned by optimizing algorithms in our case and its solutions are generally satisfying. If the eigenvectors are well assigned, the resulted controller contains the placed model poles in transmission zeros (similarly to SISO pole placement [3]). In H_∞ mixed-sensitivity design the resulting controller has such structure. The robust control is obtained in our procedure as follows:

- Put all stable model's poles as **observer** desired CL poles, eventually as **state-feedback** poles too (mainly the slow ones). If some poles has an undesirable low damping, modify the damping to an appropriate value. This is mostly the case of low frequency poles, in high frequencies the poles can remain low damped, if they are far beyond the closed-loop bandwidth. For the case of unstable model poles see controller gain section, they are also important for robustness.
- Put stable low frequency model transmission zeros into **observer** desired CL poles, if necessary. These zeros appear in controller poles and they eliminate the model transmission zero influence.

Remark: Such approach is called in SISO case [3] - (partial) internal model (reference) control and it is used if a particularly good robustness is required. Anyway, in our case the resulted controller does not contain exact model poles and zeros (in contrast to SISO case or H_∞ control), but the resulting controller pole/zero locations are close to these values.

Controller gain - There are two general requirements imposed on controller gain: It should be as low as possible - energy saving (high gain = high power consumption). It should be low enough in high frequencies - this avoids actuator solicitation with high frequency oscillations. Following tools are available:

- Put all unstable model poles into desired **observer** CL poles with the same frequency and with positive damping corresponding to the absolute value

of model unstable pole's damping. Such choice together with putting all stable model's pole to desired CL poles lead to minimum energy assignment [7].

- add one or more real multiple poles into desired **state feedback** (or **observer**) CL poles generally with multiplicity equal to n_y (number of outputs). The position of the poles depends on desired high frequency attenuation. In fact, the pole represent a low-pass filter added to CL dynamic, where the filter bandwidth is determined by the pole position. The tool is taken from SISO version of the design method [4].

CL dynamic (bandwidth) - by fixing the observer poles for robustness we have to take into account its slow modes, if a closed-loop acceleration with respect to plant is required. These slow poles introduce some slow dynamics to CL response. Nevertheless, one can guarantees a desired dominant CL dynamic (bandwidth) by:

- Fixing n_y complex conjugated pairs of poles with a frequency of desired CL bandwidth (or higher if there is an important slow dynamic in observer poles). The damping should corresponds to desired closed-loop time response (reasonable values are between 0.75 and 0.99).
- Verifying that all fixed CL real multiple poles are beyond CL bandwidth (faster than desired dominant dynamic)
- Accelerating (mostly the slowest) fixed observer CL poles going together with maintaining the defined CL observer pole structure. This step leads automatically to worse CL robustness - higher overshoot (S_{yb}), worse robust stability (S_{yp}).

High frequency attenuation (roll-off) - to obtain a desired measurement noise attenuation in high frequencies it is necessary to introduce some real multiple poles to desired **state feedback** or **observer** CL poles. The same tool is also used for limiting controller gain. Higher multiplicity of introduced poles assures higher attenuation.

Remark: In discrete-time systems the term "high frequency" signifies the frequency range from $0.25/T_e$ to $0.5/T_e$. Any higher frequencies are supposed to be totally eliminated by an appropriate anti-aliasing filter. This is one of the important advantages of discrete-time control.

Interaction (Decoupling) - The question of decoupling is still opened in considered MIMO pole placement method (method of optimizing the eigenvectors). The other method (heuristic placement of eigenvectors), on the contrary, proposes some suitable eigenvector choices to limit interaction.

3.2 Pole-placement algorithms

Three algorithm of eigenvector optimisation were tested and their properties are presented below.

Method of Kautsky [2] - The algorithm gives solutions which can differ seriously with only small modifications (changes of 10^{-6}) in desired CL pole placement, mainly if CL poles are placed as model poles (the method is ill-conditioned for robust control placement). Thus, to obtain a good sub-optimal solution - with a good sensitivity responses - it is necessary to iterate around known pole placements (hence it is necessary to know approximately pole placements). The algorithm is very fast, its computing time is in fractions of seconds, and therefore it is suitable for the necessary iterations. We used the realization of this algorithm programed in Matlab Control Toolbox - the function "place".

Method of Tits [11] - Using this algorithm, the obtained sensitivity responses are close to the best possible solutions obtained by the method [2]. Moreover, the algorithm is well conditioned even for robust pole placement. So, there is no need of the iteration necessary in the case of Kautsky method. Nevertheless, the Kautsky method can give slightly better results with an iteration effort. The improved algorithm of Tits is quite fast (only slightly slower then the one of Kautsky). To test the algorithm we applied the function for Matlab "robplace" programed by Tits.

Method of Varga [12] - It is well-conditioned even for robust pole placement technique, but quite slow (optimization takes several seconds). Although, the algorithm has its properties similar to the properties of the Tits method, proposed solutions can differ seriously. The tested function for Matlab "sylvplace" was developed by the author of the algorithm.

The method of Tits is an improvement of the method of Kautsky and the algorithm of Varga is based on Sylvester equation resolution. All presented methods are based on

the same principle. They optimize closed-loop eigenvectors according to some criterion. This criterion assures that the resulting set of placed CL eigenvectors provides maximal robustness of pole placement (small changes in model pole placements produce only small modifications in CL pole placements), minimal norm of the feedback matrix, etc.

4 CONTROLLER DESIGN PROCEDURE

The procedure of robust controller design is very similar to the one proposed in [4] for SISO systems and it consists of the following steps:

STEP 1: Choose the sensitivity matrices to be shaped and define corresponding templates according to closed-loop performance specifications. Generally, we consider S_{yp} for robust stability, S_{up} for controller gain restriction, and S_{yb} sensitivity matrix to assure desired tracking properties as it was explained previously.

STEP 2: Specify the pre-fixed block H_K , if necessary, to satisfy some particular specifications. One of the most typical example is the introduction of an integrator for steady-state error elimination.

STEP 3: Fix the locations of CL poles, using the proposed rules, in order to keep the chosen largest singular values below the templates.

STEP 4: Evaluate the matrices F and L of the controller $K_0(z)$ in (9) and reconstruct the controller $K(z)$ from (7). Next, evaluate frequency responses of sensitivity matrices.

STEP 5: Verify resulting sensitivity matrix responses. If some results are not satisfactory, go to STEP 2 or STEP 3.

5 EXAMPLE OF APPLICATION

5.1 Model and performance specifications

The used multivariable model is a continuous time linearized model of vertical-plane dynamics of an aircraft (see [6]). The model has 3 inputs, 3 outputs, and 5 states and the following 5 poles: 2 pairs of complex poles at 0.029194Hz, 0.205589Hz with the corresponding damping of 0.0959353 and 0.60387, and one real pole at the origin (a pure integrator).

In [6], a H_∞ continuous-time controller is designed for this model. Using the results from the continuous time design of [6], we constructed the following closed-loop specifications:

- desired closed loop dynamic f_{CL} of about 1.6Hz (corresponding rising time is about 0.35s) with a good attenuation > 0.75 and maximum overshoot about 10%.
- integral action in each loop in order to eliminate steady-state errors.
- restrain if possible interactions between loops.
- attenuate well high frequencies (-20dB and more) in order to eliminate an influence of high frequency disturbances and noise.
- minimize the sensitivity functions S_{yp} and S_{yb} in order to obtain the best possible robustness.
- command has to be admissible.

To apply our controller design methodology we need a discrete time equivalent of the continuous time plant model. The chosen sampling frequency is $f_e \cong 12f_{CL}$, precisely $f_e = 20\text{Hz}$ and thus sampling time is $T_e = 0.05\text{s}$. For a continuous to discrete time transformation the command "c2d" of Matlab, and particularly zero-order hold on the model inputs method, was used.

5.2 Controller Design

1) To satisfy the design specifications, two templates are constructed. The first one is the template on largest singular value of S_{yp} (in Fig. 3 traced with crosses) and it guarantees a relatively good robustness. The second template is imposed on S_{yb} sensitivity matrix responses (Fig. 5 in crosses). Satisfying this template assures closed-loop bandwidth, maximal overshoot (1dB in low frequencies

correspond to approx. 12%), and a good roll-off (see previous analysis of sensitivity functions).

2) A MIMO almost-integrator (with 3 poles at 0.9999) is introduced to the pre-fixed block H_K (similarly to [6]).

3) The model poles are placed as the observer closed-loop poles with modified damping and with the pure integrator pole slightly accelerated. Three pairs of state feedback poles fixed to 2.2Hz should provide required CL dynamic. Although, the desired closed-loop bandwidth is 1.6Hz, we choose the higher frequency of 2.2Hz in order to compensate slow observer CL poles. Two non-specified state feedback poles and three non-specified observer poles were put close to the origin (on the circle with a radius 10^{-5}) and so their influence can be neglected. Fixed closed-loop poles are placed as follows:

- Observer poles - two complex pairs at 0.029194Hz and 0.205589Hz with corresponding damping 0.9 and 0.9; one real pole at 0.999.
- State feedback poles - three complex pairs at 2.2Hz with the damping 0.8.

Next, the observer and state feedback matrices are computed. Resulting sensitivity responses are shown in the Fig. 3, 4, (dotted lines) and 5a. The characteristics of $\bar{\sigma}(S_{yp})$ and $\bar{\sigma}(S_{yb})$ are quite satisfactory, but $|S_{yb}|$ is not acceptable in middle and high frequencies - Fig. 5a.

4) One real multiple pole is added to observer CL poles at 0.5 and with the multiplicity 3 (number of outputs, also maximal admissible multiplicity for the used pole placement algorithms). The corresponding resulting sensitivity responses are in Fig. 3, 4, (dashed lines) and 5b. The responses are satisfactory except for $|S_{yb}|$ responses, where one non-diagonal frequency response is not attenuated sufficiently in high frequencies.

5) As we mentioned, it is difficult to treat a decoupling only by the placement of CL poles. Nevertheless, the problem is in high frequencies and so we apply an additional real multiple pole attenuating high frequencies. The pole is set to 0.25 with the multiplicity 2 to state feedback CL poles and thus we fixed all possible closed-loop poles. The resulting sensitivity responses are in the Fig. 3, 4, (solid fine lines) and 5c.

6) The results are quite satisfactory, only the closed-loop dynamic is slightly slower than required. We accelerate the closed loop by shifting the slowest CL poles (the real pole in 0.999 and the complex pair in 0.029Hz) to higher frequencies. The real pole is set to 0.96 and the complex pole to 0.07Hz. The final sensitivity responses are in Fig. 3, 4, (solid thick lines) and 5d. The closed-loop dynamic is accelerated this time but, as was expected, the overshoot is higher and the robustness is worse.

The pole placement algorithm of Tits was used in the controller design, but very similar controller with similar sensitivity responses was also designed using the method of Kautsky and Varga. In the case of Kautsky method, it was necessary to iterate around desired CL pole positions, but better results for decoupling are obtained. The controller has 11 states.

6 CONCLUSIONS

In this note, a set of useful tools for multivariable pole-assignment controller design technique was proposed based on combined SISO pole placement with sensitivity function shaping. Several rules for placement of closed-loop poles and some tools for sensitivity shaping are proposed, or reviewed, which allow to satisfy some of typical closed-loop performance specifications. While the proposed methodology can certainly not compete with H_∞ design technique on complex examples (the controller complexity is fixed and closed-loop eigenvector optimisation does not provide the best solutions), the technique is indeed a feasible approach in many cases and it gives a clear insight of how to tune closed-loop dynamics. Finally, the design procedure was demonstrated on one non-trivial multivariable example.

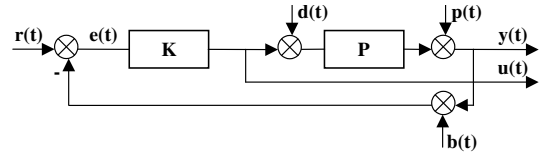


Figure 1: Considered closed loop configuration

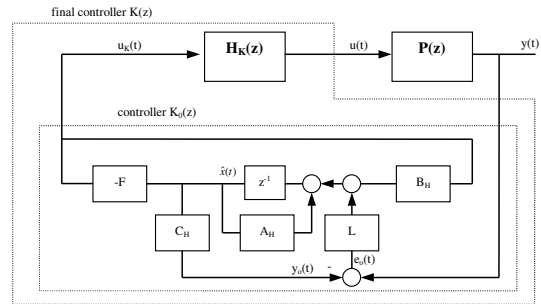


Figure 2: Controller structure

References

- [1] Kailath T. *Linear Systems*. Prentice-Hall: New Jersey, 1980.
- [2] Kautsky J, Nichols NK, Van Doren P. Robust pole assignment in linear state feedback. *International Journal of Control* 1991; **41**(5):1129–1155.
- [3] Landau ID. *Commande de Systèmes*. Hermès: Paris, 2002.
- [4] Landau ID, Karimi A. Robust digital control using pole placement with sensitivity function shaping method. *International Journal of Robust and Nonlinear Control* 1998; **8**:191–210.
- [5] Landau ID, Langer J, Rey D, Barnier J. Robust control of 360 flexible arm using the combined pole placement/sensitivity function shaping method. *IEEE Transactions on Control Systems Technology* 1996; **4**(4):369–383.
- [6] Maciejowski JM. *Multivariable Feedback Design*. Addison-Wesley, 1989.
- [7] Magni JF. *Robust Modal Control*. Kluwer Academic/Plenum Publishers: New York, 2002.
- [8] Procházka H, Landau ID. Pole placement with sensitivity function shaping using 2nd order digital notch filters. *Automatica* 2003; **39**(6):1103–1107.
- [9] Procházka H. Comments on the paper "Controller order reduction by identification in closed-loop applied to a benchmark". *to appear in European Journal of Control* 2003; **9**.
- [10] Sobel KM, Shapiro EY, Andry AN jr. Eigenstructure assignment. *International Journal of Control* 1994; **59**(1):13–37.
- [11] Tits AL, Yang Y. Globally convergent algorithms for robust pole assignment by state feedback. *IEEE Transactions on Automatic Control* 1996; **41**(10):1432–1452.
- [12] Varga A. Robust pole assignment via Sylvester equation based state feedback parametrization. *IEEE International Symposium on Computer Aided Control System Design, CACSD'2000, Anchorage, Alaska* 2000;
- [13] Zhou K. *Essentials of Robust Control*. Prentice-Hall: New Jersey, 1998.

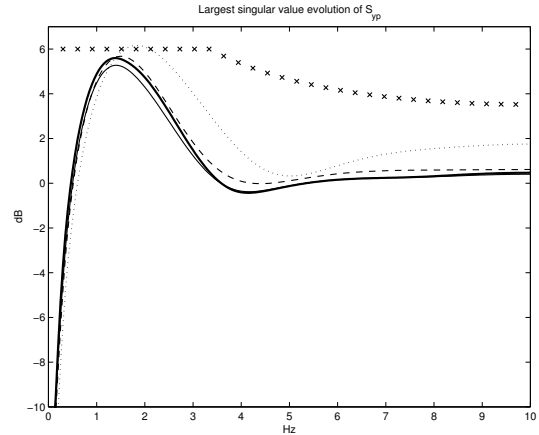


Figure 3: The largest singular value evolution of S_{yp} for the controllers of design steps: 3(dotted), 4(dashed), 5(solid thin), 6(solid thick)

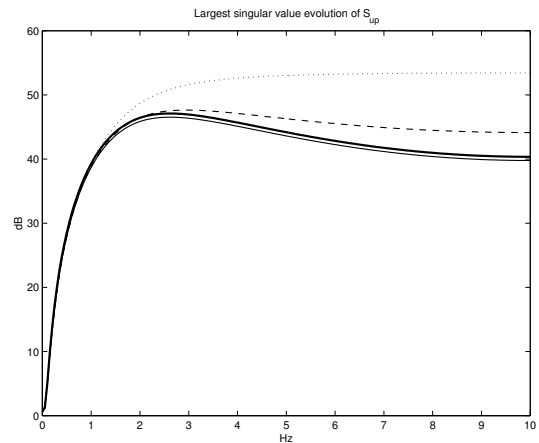


Figure 4: The largest singular value evolution of S_{up} for controllers of design steps: 3(dotted), 4(dashed), 5(solid thin), 6(solid thick)

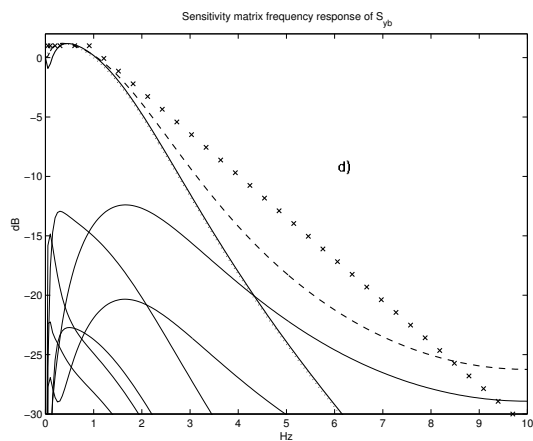
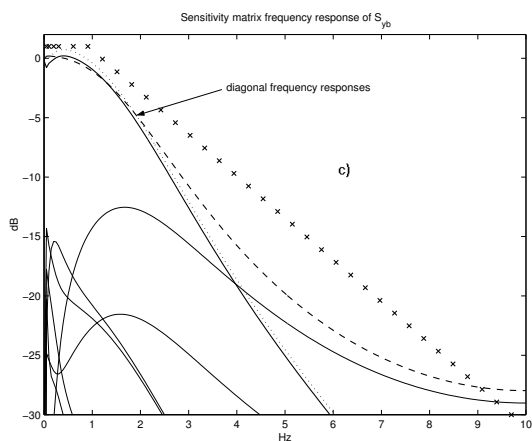
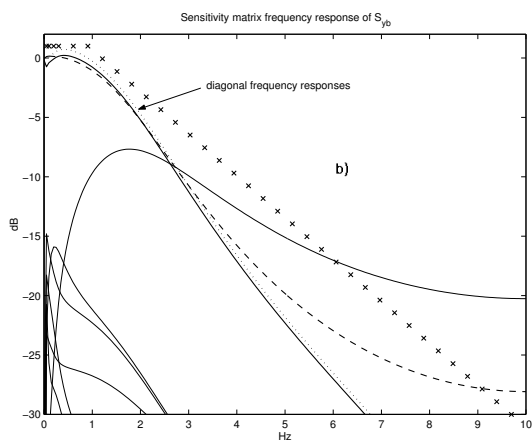
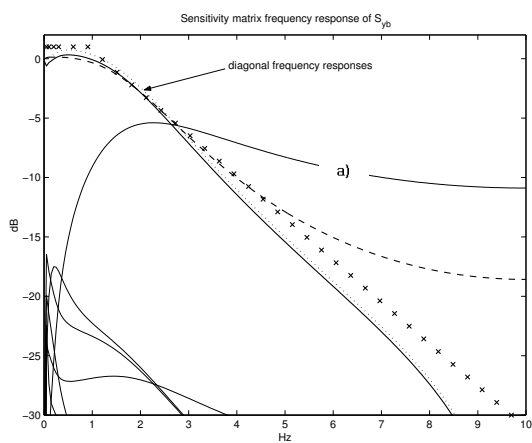


Figure 5: $|S_{yb}|$ frequency characteristics ($|S_{yb11}|$ dotted, $|S_{yb22}|$ dashed, $|S_{yb33}|$ solid, template in crosses) for controllers of the step 4(a), 5(b), 6(c), and 7(d).

Convex optimisation for robust multivariable controller design as alternative of H_∞ optimisation

Hynek Procházka*, Ioan Doré Landau†

October 1, 2003

Laboratoire d'Automatique de Grenoble (CNRS-INPG-UJF), ENSIEG; BP. 46, 38402 St. Martin d'Hères, FRANCE

Abstract

The paper introduces a new robust controller design methodology for multi-input multi-output digital controllers based on frequency domain robustness analysis. It provides also a link between H_∞ -optimisation design and the new technique. The proposed methodology extends a design approach existing in single-input single-output domain and it is based on Youla-Kucera parameterization, pole placement and sensitivity function shaping by convex optimisation. The procedure is illustrated on one multivariable example and solutions are compared to H_∞ -optimisation results.

Keywords: *multivariable controller design, robustness, pole placement, sensitivity shaping, convex optimisation.*

1 Introduction

The pole placement with sensitivity function shaping by convex optimisation is a discrete-time single-input single-output (SISO or monovariable) controller design method introduced in [7]. It is based on previously developed robust controller design methodology of pole placement with sensitivity function shaping presented in [6, 5] and improved in [12]. An automated procedure for shaping the sensitivity functions using convex optimisation and the celebrated Youla-Kucera parameterization [18] has been developed in [7]. Efficiency of the design procedure have been proved on several practical examples [9, 8, 2].

The pole placement, or more precisely the eigenstructure assignment, is a method used also for multi-input multi-output (MIMO or multivariable) controller synthesis. An essential domain of application is aircraft controller design

*e-mail: hynek.prochazka@lag.ensieg.inpg.fr; Tel:+33 4 76 82 63 87; Fax:+33 4 76 82 63 88

†e-mail: landau@lag.ensieg.inpg.fr; Tel:+33 4 76 82 63 91; Fax:+33 4 76 82 63 88

[14, 11]. In MIMO eigenstructure assignment, fixing desired CL poles does not imply a unique solution, in addition closed-loop eigenvectors have to be fixed. To deal with the eigenvector assignment, several approaches have been developed. Some optimize closed-loop eigenvectors according to some criterions [3, 16, 17], while the others propose some rules for the assignment [14, 11]. Only the first approach will be used in this note.

This paper extends the method of pole placement with sensitivity shaping by convex optimisation [7, 9] into multivariable domain. Similarly to [9], the developed methodology employs the "Youla-Kucera" parameterization of all stabilizing controllers. A parameterized controller consists of a fixed-size central controller, which has to stabilize plant model, connected with a free parameter transfer matrix Q . It has been shown [1], that a number of closed-loop specifications on sensitivity matrices are convex with respect to the matrix Q . An adapted multivariable eigenstructure assignment design technique, which corresponds to SISO pole placement with sensitivity function shaping technique [6], is used for central controller evaluation.

Similarly to well-known H_∞ -optimisation (see for example [19, 10]), the proposed methodology deals with system robustness via singular value evolution of sensitivity matrices. The H_∞ -optimisation minimizes H_∞ -norm maximum of chosen weighted sensitivity matrices. The sensitivity shaping approach presented here defines a set of templates and optimize controller to keep the corresponding singular values of sensitivity matrices below. Conversely to H_∞ -optimisation, the sensitivities are treated separately and it makes this procedure much more flexible. In addition, use of templates is more convenient in practice, since it represents directly frequency domain specification and there is no need of template rational expression. Detailed comparison of the presented procedure with H_∞ -optimisation can also be found in this paper. One design example from [10] illustrates the developed design procedure and allows to compare solutions obtained with our procedure and with H_∞ -optimisation.

The paper is organized as follows: Section 2 introduces necessary basic MIMO theory utilized in methodology. In section 3, controller and its parameterization are presented, section 4 explains convex optimisation implementation in controller design. Section 5 summarize the complete controller design procedure and the Section 6 provides the comparison of the developed procedure with H_∞ -optimisation. In Section 7 we apply the developed methodology on one example of MIMO system. Finally section 8 gives some conclusions.

2 Theoretical background

2.1 Multi-input multi-output linear robust control problem

A linear discrete-time multi-input multi-output time-invariant (MIMO) model $G(z)$ of a plant is considered. The model is supposed to have the following state

space representation form:

$$\begin{aligned}x_p(t+1) &= Ax_p(t) + Bu_p(t) \\y_p(t) &= Cx_p(t) + Du_p(t)\end{aligned}\quad (1)$$

with $A \in R^{n_x \times n_x}$, $B \in R^{n_x \times n_u}$, $C \in R^{n_y \times n_x}$, $D \in R^{n_y \times n_u}$. The matrices A , B , C are some constant state space matrices and $u_p(t)$, $y_p(t)$ and $x_p(t)$ are any vectors of plant inputs, outputs, and states respectively. The variables n_u , n_y , and n_x represent the number of plant inputs, outputs and states respectively. In the case of digitalized continuous time plants (considered in here), the matrix D is zero. The time variable t is discrete time: $t = kT_e$, $k = 0, 1, 2, \dots$ with a sampling time T_e . A corresponding model transfer function matrix is then:

$$G(z) = C(zI - A)^{-1}B \quad (2)$$

The considered closed loop system configuration to achieve any desired closed-loop performance is shown in Fig. 1. The block K is an appropriate MIMO controller described either in state space representation with A_K , B_K , C_K , D_K , where the matrix D_K is non-zero, or by a transfer matrix $K(z)$.

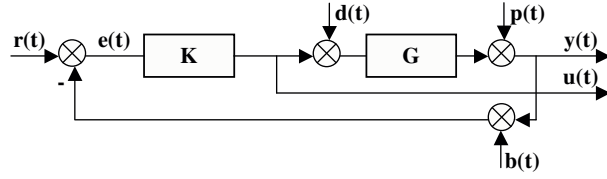


Figure 1: Considered closed loop configuration

2.2 Robustness and performance by sensitivity shaping

Generally, output disturbance $p(t)$, measuring noise $b(t)$ and command disturbance $d(t)$ are expected to act on the system (see Fig. 1). Influence of these disturbances onto CL output $y(t)$ and command $u(t)$ is expressed in the form of sensitivity functions. Typically, the following sensitivity matrices are usually treated:

- $S_{yp}(z)$: Sensitivity matrix output disturbance - output

$$S_{yp}(z) = \frac{y(z)}{p(z)} = (I + G(z)K(z))^{-1} \quad (3)$$

- $S_{up}(z)$: Sensitivity matrix output disturbance - command

$$S_{up}(z) = \frac{u(z)}{p(z)} = -K(z)(I + K(z)G(z))^{-1} \quad (4)$$

- S_{yb} : Sensitivity matrix measuring noise - output

$$S_{yb}(z) = \frac{y(z)}{b(z)} = -G(z)K(z)(I + G(z)K(z))^{-1} \quad (5)$$

Imposing bounds on these sensitivity function matrices one can match a majority of control specifications, especially the ones related to robustness. Restrictions on $S_{yp}(z)$ and $S_{yb}(z)$ can guarantee closed-loop robust stability, and certain bounds on S_{yb} adjust closed loop tracking properties. The $S_{up}(z)$ matrix allows to assess the stress on the actuator in the frequency domain, in other words adjusting this sensitivity mainly the controller gain is shaped.

A typical robustness condition in H_∞ -optimisation approach for one sensitivity matrix $S_{ij}(z)$ has the following general form:

$$\|W_j(z)S_{ij}(z)W_i(z)\|_\infty \leq \gamma \quad (6)$$

where γ is an appropriate constant assuring satisfactory performances and $\|\cdot\|_\infty$ is the norm of H_∞ . The indices i, j denote any considered output ($y(t), u(t)$) - input ($p(t), b(t), d(t)$) signals. The $W_i(z)$ is a weighting transfer matrix and its inverse represents a template imposed on the sensitivity matrix. The weighting transfer matrix $W_j(z)$ can be interpreted as a disturbance model for the corresponding input signal j .

Replacing the norm $\|\cdot\|_\infty$ by its expression with the largest singular value $\bar{\sigma}(\cdot)$ and substituting z with $z = e^{j\omega_d}$, where $\omega_d \in \langle 0, \pi \rangle$ is normalized discrete frequency:

$$\omega_d \doteq \omega T_e \mid \omega \in \langle 0, \frac{\pi}{T_e} \rangle \quad (7)$$

the condition (6) becomes:

$$\bar{\sigma}(W_j(e^{j\omega_d})S_{ij}(e^{j\omega_d})W_i(e^{j\omega_d})) \leq \gamma \mid \forall \omega_d \quad (8)$$

To implement the sensitivity function shaping [7], a template $F_{ij}(\omega_d)$ is imposed on the largest singular value evolution $\bar{\sigma}(S_{ij}(e^{j\omega_d}))$. This template replaces the weighting matrix $W_i(e^{j\omega_d})$ (the template corresponds to the singular value evolution of the weighting matrix inverse). Next, to simplify expressions suppose $W_j(e^{j\omega_d}) = I$ and we obtain:

$$\bar{\sigma}(S_{ij}(e^{j\omega_d})) \leq F_{ij}(\omega_d) \mid \forall \omega_d \quad (9)$$

where the developed sensitivity shaping condition (9) is non-convex with respect to the controller matrix $K(z)$.

3 Multivariable controller for convex optimisation

3.1 Controller parameterization

The used controller parameterization is the "Youla-Kucera" (named also Q-) parameterization presented for example by [4, 1, 19, 10]. In brief, the Q-

parameterization is defined as follows [10]: Suppose any model transfer function matrix $G(z)$ stabilized in closed loop by a central controller $K_0(z)$. Both, the model and the central controller can be described using the following coprime factorization:

$$\begin{aligned} G(z) &= N(z)M^{-1}(z) = \tilde{M}(z)^{-1}\tilde{N}(z) \\ K_0(z) &= U_0(z)V_0^{-1}(z) = \tilde{V}_0^{-1}(z)\tilde{U}_0(z) \end{aligned}$$

where $U_0(z)$, $\tilde{U}_0(z)$, $V_0(z)$, $\tilde{V}_0(z)$, $M(z)$, $\tilde{M}(z)$, $N(z)$, $\tilde{N}(z)$ are stable transfer matrices. Then, any controller $K_c(z)$ stabilizing closed loop (of $K_c(z)$ and $G(z)$) can be written in the following fractional form:

$$K_c(z) = (U_0(z) + M(z)Q(z))(V_0 - N(z)Q(z))^{-1} \quad (10)$$

where $Q(z)$ is any realizable stable transfer matrix. Thus, to implement Q-parameterization in a convex optimisation an appropriate structure of the transfer matrix $Q(z)$ is needed.

The parameterization has the following essential properties:

- Any closed loop sensitivity matrix can be written in the following form:

$$S_{ij}(z) = T_1(z) + T_2(z)Q(z)T_3(z) \quad (11)$$

where the matrix $Q(z)$ is a free parameter to optimize and $T_1(z)$, $T_2(z)$, $T_3(z)$ are some transfer matrices depending on $S_{ij}(z)$. If (11) is taken into account, the sensitivity shaping condition (9) becomes:

$$\bar{\sigma}(T_1(e^{j\omega_d}) + T_2(e^{j\omega_d})Q(e^{j\omega_d})T_3(e^{j\omega_d})) \leq F_{ij}(\omega_d) \quad \forall \omega_d \quad (12)$$

which is a convex condition with respect to the transfer matrix $Q(z)$.

- The central controller $K_0(z)$ can be described as a state observer with estimated state feedback [1, 19] (see subsection on controller realization).
- Poles of $Q(z)$ are also closed-loop poles. Thus, to make sensitivity shaping really effective these poles should be optimized.

3.2 $Q(z)$ matrix structure

To parameterize $Q(z)$, we will use the following form of any transfer matrix:

$$Q(z) = d^{-1}(z)N(z) \quad (13)$$

where $N(z)$ is a polynomial matrix of the same dimension as $Q(z)$ and $d(z)$ is a polynomial containing all poles of $Q(z)$. The convenience of this form is explored in Section 4 concerning convex optimisation.

It remains to determine a structure of the polynomial $d(z)$ and of the polynomial matrix $N(z)$. The simplest solution preserving convexity (see Section 4)

is to consider every coefficient of the polynomial matrix $N(z)$ and of the polynomial $d(z)$ as one element of an optimized vector X . Subsequently, the vector X , the numerator polynomial matrix $N(z)$, and the denominator $d(z)$ have the following forms:

$$\begin{aligned}
X &= [x_0, x_1, x_2, \dots, x_l] \mid l = n_u \times n_y \times (N_q + 1) + N_q \\
d_{N_q}(X) &= x_0 z^{N_q} + x_1 z^{N_q-1} + \dots + x_{N_q} \\
N_{N_q}(X) &= [n_{ij}(z)] \mid i = 1, 2, \dots, n_u; j = 1, 2, \dots, n_y \\
n_{ij}(z) &= x_k z^{N_q} + x_{k+1} z^{N_q-1} + \dots + x_{k+N_q} \\
k &= N_q + (i-1)n_y(N_q + 1) + (j-1)(N_q + 1) + 1, \\
&\quad N_q + (i-1)n_y(N_q + 1) + (j-1)(N_q + 1) + 2, \dots, l
\end{aligned} \tag{14}$$

where N_q is a parameterization order and it corresponds to the orders of all polynomial elements in $N(z)$ and also to the order of the denominator $d(z)$.

3.3 Controller structure

Suppose the controller $K_c(z)$ parameterized as above. Moreover, suppose that the final controller contains a pre-fixed block named $H_K(z)$, which is any stable transfer matrix (for example MIMO integrator). Then, the final controller $K(z)$ is defined as follows:

$$K(z) = H_K(z)K_c(z) = H_K(z)(U_0(z) + M(z)Q(z))(V_0(z) - N(z)Q(z))^{-1} \tag{15}$$

It is obvious, that the optimized controller $K_c(z)$ is constructed from the model $G(z)$, the central controller $K_0(z)$ and the optimized block $Q(z)$. It was shown (see for example [19, 1]), that the optimized controller $K_c(z)$ can be represented as an observer with state feedback matrix, where the block $Q(z)$ is connected between the output prediction error $e_0(t)$ and the state feedback command $u_K(t)$. The corresponding diagram is shown in Fig. 2.

To construct final (complete) controller $K(z)$, we first define a system $H(z)$ of the cascade connection $H(z) = G(z)H_K(z)$ with the state vector $x_H = [x_{H_K}, x_p]^T$. The state space representation (SSR) of $H(z)$, $H_K(z)$ and $Q(z)$ are as follows:

$$\begin{aligned}
x_H(t+1) &= A_H x_H(t) + B_H u(t) \\
y(t) &= C_H x_H(t) \\
x_{H_K}(t+1) &= A_{H_K} x_{H_K}(t) + B_{H_K} u_o(t) \\
u(t) &= C_{H_K} x_{H_K}(t) + D_{H_K} u_o(t) \\
x_Q(t+1) &= A_Q x_Q(t) + B_Q e_o(t) \\
y_Q(t) &= C_Q x_Q(t) + D_Q e_o(t)
\end{aligned} \tag{16}$$

where A_i, B_i, C_i, D_i ($i \equiv (H, H_K, Q)$) are some constant state space matrices and $x_i(t)$, are corresponding states. The signal $y_Q(t)$ is an output vector of the

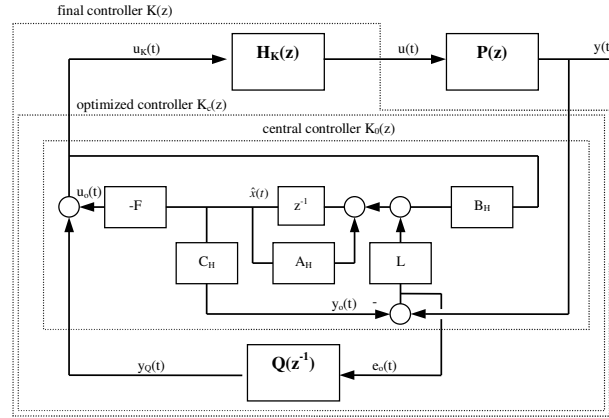


Figure 2: Controller structure

matrix $Q(z)$, and it is added to the estimated state feedback vector $u_o(t)$ - see Fig. 2.

The state observer has the following form:

$$\begin{aligned}\hat{x}(t+1) &= A_H \hat{x}(t) + L e_o(t) + B_H u_K(t) \\ y_o(t) &= C_H \hat{x}(t)\end{aligned}\quad (17)$$

where $\hat{x}(t)$ is an estimation of the state vector x_H . The estimated state feedback vector $u_K(t)$ and the output estimation error vector $e_o(t)$ are defined as follows:

$$\begin{aligned}u_K(t) &= -F \hat{x}(t) + y_Q(t) \\ e_o(t) &= y(t) - y_o(t)\end{aligned}\quad (18)$$

The central controller $K_0(z)$ includes the state space matrices of the cascade connection $H(z)$ together with the observer matrix L - forming observer (17), and the estimated state feedback matrix F - estimated state feedback part. The matrices L and F determine assignment of one part of closed loop poles (the rest of poles are in $Q(z)$), where eigenvalues of $(A_H - LC_H)$ and $(A_H - B_H F)$ matrices define respectively observer and estimated state feedback closed loop poles. Two major methods exist for the computation of the central controller: eigenstructure assignment and linear quadratic gaussian -LQG method (kalman filter with optimal state feedback). In this paper, we consider only assignment method, and particularly the algorithms of CL eigenvector optimisation [3, 16, 17], but LQG technique can be applied as well. The algorithms of multivariable pole assignment define some criterion (guaranteeing robust placement - a small shift of model poles causes only small changes in CL pole locations, low controller gain, etc.) and then search by an optimisation technique for a criterion minimum (not necessarily global) by optimizing the closed-loop eigenvectors.

The relations (18) and the state space representation of $Q(z)$ together with the equation (17) give the state space representation of the optimized controller

$K_c(z)$ with the state vector $x_c(t) = [\hat{x}(t), x_Q(t)]^T$:

$$\begin{aligned} x_c(t+1) &= A_c x_c(t) + B_c y(t) \\ u_K(t) &= C_c x_c(t) + D_c u_K(t) \end{aligned} \quad (19)$$

with A_c , B_c , C_c and D_c controller state space matrices. The final controller $K(z)$ is then a cascade connection of the optimized controller $K_c(z)$ and the pre-fixed transfer matrix block $H_K(z)$ as it shows (15) and Fig. 2.

4 Design specifications as convex criteria

Convex optimisation can be applied only if all required specifications on robustness and on closed-loop performances are expressed via a convex criterion function. In this section, convex criterion functions will be developed for these two design specifications:

- *Sensitivity shaping* - Based on (9), templates are imposed on the largest singular value of sensitivity matrix. These templates imposed on different closed-loop sensitivities can guarantee a major of closed-loop specifications concerning robustness and performance.
- *Closed-loop δ -stability* - fixed region imposed on closed loop pole locations. It allows to assure eventual specifications concerning closed-loop dynamic - pole natural frequency and damping.

4.1 Sensitivity shaping criteria

Applying (13) the sensitivity shaping condition (12) becomes:

$$\bar{\sigma}(T_1(e^{j\omega_d}) + T_2(e^{j\omega_d})d^{-1}(e^{j\omega_d})N(e^{j\omega_d})T_3(e^{j\omega_d})) \leq F_{ij}(\omega_d) \quad \forall \omega_d \quad (20)$$

All the terms in the inequality (20) are depending on the frequency ω_d . We will simplify the inequality expression by omitting the notation of frequency dependence $e^{j\omega_d}$. In addition, we multiply the inequality with $|d(e^{j\omega_d})|$ and put the template $F_{ij}(\omega_d)$ on the left side. The robustness condition is now:

$$\bar{\sigma}(dT_1 + T_2NT_3) - |d|F_{ij} \leq 0 \quad \forall \omega_d \quad (21)$$

The inequality (21) is non-convex yet, because of the element $-|d|F_{ij}$. Replacing $|d|$ with $Re\{d\}$ it becomes convex (see Appendix A for convexity proof of corresponding criterion function (24)) and we have:

$$\bar{\sigma}(T_1d + T_2NT_3) - Re\{d\}F_{ij} \leq 0 \quad \forall \omega_d \quad (22)$$

Particular disadvantage of the last operation is that the new sensitivity shaping condition 22 is more conservative than 21. On the other hand, stability of the polynomial $d(z)$, and consequently the closed loop stability is assured with

the new condition, since verifying the condition (22) the polynomial $d(z)$ fulfills the condition: $Re\{d(e^{j\omega_d})\} > 0, \forall \omega_d$, which means that $d(z)$ is strictly positive real and this is sufficient condition of polynomial stability.

Similar condition can be developed for every element (transfer function) of any sensitivity matrix S_{ij} . In the condition (20), the largest singular value $\bar{\sigma}(\cdot)$ is replaced with modulus $|\cdot|$ and hence we obtain a matrix of frequency response modulus. Then, multiplying both sides of sensitivity matrix with two vectors-selectors v_1 and v_2 , we can choose one element of that matrix. The two vectors v_1, v_2 have all elements except one null. The one non-null element is equal to one and its position determines which transfer function is chosen. The developed one-element sensitivity shaping condition has the following form:

$$|v_1^T(T_1 d + T_2 N T_3)v_2| - Re\{d\}F_{ij} \leq 0 \quad \forall \omega_d \quad (23)$$

Define now criterion functions ϕ_{ij} and $\phi_{ijv_1v_2}$ suitable for optimisation of the conditions (22) and (23) using the developed form (14) of $d(z)$ and $N(z)$:

$$\phi_{ij}(X) = \sup_{\omega_d} [\bar{\sigma}(T_1 d_{N_q}(X) + T_2 N_{N_q}(X) T_3) - Re\{d_{N_q}(X)\}F_{ij}] \quad (24)$$

$$\phi_{ijv_1v_2}(X) = \sup_{\omega_d} [|v_1^T(T_1 d_{N_q}(X) + T_2 N_{N_q}(X) T_3)v_2| - Re\{d_{N_q}(X)\}F_{ij}] \quad (25)$$

where the criterion functions are convex with respect to the optimized vector X . The appendix A demonstrates convexity of (24), the convexity proof for (25) is almost identical.

To evaluate the criterion, a frequency point grid is defined and the function $\phi_{ij}(X)$ is computed at these points. The maximum value from the obtained results will approximate the real criterion value.

4.2 Closed-loop δ -stability criterion

The poles of the observer and of the state feedback are fixed before the optimisation during central controller calculation, generally according to some desired performances. It remains to restrict the optimized poles of $d(z)$. First, recall that verification of the inequality (22) guarantees stability of $d(z)$, because the condition (with $z = e^{j\omega_d}$):

$$-Re\{d(z)\} < 0 \quad \forall \omega_d \quad (26)$$

is fulfilled and this is a sufficient condition of stability.

Based on the condition (26) [7, 9] have introduced (for SISO case) a condition of δ -stability by defining bounds in the form of a circle with its center on real axis. The modified condition of δ -stability is defined as follows:

$$-Re\{\bar{d}(z)\} < 0 \quad \forall \omega_d \quad (27)$$

where

$$\bar{d}(z) = d(\bar{z})|_{\bar{z}=rz+c} \quad (28)$$

The variables r and c are respectively a radius of circle and a circle center position on real axis.

The criterion function ϕ_δ for the inequality (27) has the following form:

$$\phi_\delta(X) = \sup_{\omega_d} [-Re\{\bar{d}_{N_q}(X)\}] \quad (29)$$

with $\bar{d}_{N_q}(X)$ denoting $d(z)$ parameterized by (14). The criterion function ϕ_δ is convex and the proposed evaluation of this function is equivalent to that one provided in the case of sensitivity function shaping criterion.

4.3 Multi-criterion function and optimisation algorithm

The function $\max(\cdot)$ is used to assembly several criterion functions. Consequently, the multi-criterion convex function ϕ_{mc} is defined as follows:

$$\phi_{mc}(X) = \max_k [\alpha_k \phi_k(X)] \quad (30)$$

where the index k denote appropriate criterion function ($k = \{\delta, yp, ypv_1v_2, up, \dots\}$) and α_k is weighting coefficient ($\alpha_k \in R$) giving desired weight to this function.

The optimisation algorithm is a standard ellipsoid algorithm described in [1] (the same was used by [7, 9]). The algorithm is gradient based and it allows also optimisation with constrains. Using optimisation with constrains, we can distinguish an objective function $\phi_{OBJ}(X)$ and a constraint function $\phi_{CST}(X)$. Both of this criterion functions have to be convex and, in our case, they have the general form (30). The optimisation algorithm proceeds as follows:

$$\begin{array}{ll} \text{if } (\phi_{CST}(X) < 0) & \rightarrow \text{ minimizing } \phi_{OBJ}(X) \\ \text{else} & \rightarrow \text{ minimizing } \phi_{CST}(X) \end{array}$$

5 Controller design procedure

The complete MIMO controller design procedure consists of following steps:

- Transform desired closed-loop specifications to design specifications compatible with the proposed procedure. As it was shown in previous sections, basically these two design specifications can be used:
 - Templates on sensitivity matrices. The templates should express desired robustness and performances (robust stability, disturbance rejection in specific frequencies, etc.)
 - δ -stability region for optimized poles. Using the criterion function (24) as the optimisation objective, the stability region is automatically the entire unit circle (because of the element $Re\{d_{N_q}(X)\}$). To restrain more the region for optimized CL poles, one has to set the function (29) as the constraint function of optimisation. The radius r and the center c from (28) determine the desired region-circle in complex plain for optimized CL pole locations.

Finally, it has to be decided which design specifications will be considered as constraints and which will form optimisation objective.

- Select the pre-fixed block H_K from (15). Some of required performances can be directly assured by incorporating a fixed filter into the controller. This block often contains the internal model of a disturbance or a filter opening the loop at certain frequencies. The most common example is an integrator to eliminate static error.
- Central controller $K_0(z)$ design. Closed-loop poles of observer and estimated state feedback have to be fixed and accordingly the matrices L , F are computed. The central controller is then constructed from (17) and (18) as observer with estimated state feedback. This step is important for final controller complexity and performance. To obtain a simple high-performance final controller, the central controller should reflect desired closed-loop performance and robustness specifications. These several principal rules are proposed to deal with the choice of desired closed-loop poles:
 - use the plant model dynamics (eventually with appropriately attenuated vibration modes) to define the observer poles together with some real multiple auxiliary pole to provide corresponding roll-off.
 - to specify the estimated state feedback poles use a complex pair of poles with a damping > 0.75 , at the frequency corresponding to a desired dominant closed loop dynamics and with the multiplicity n_y -number of outputs (maximal multiplicity allowed by assignment algorithms).

Remark: The matrices L and F were computed by the algorithm [3, 16] or [17]. In the case of [3], sometimes only a small pole modification produces an important changes in solutions (the problem is ill-conditioned), and thus some iteration effort is needed. As we already mentioned, the controller can also be computed using LQG method.

- Set optimisation parameters. The criterion function weights α_i from (30) providing an appropriate scaling has to be chosen and the parameterization order N_q of the matrix Q is set. In general, one starts with the order $N_q = 1$ and rises the order if the results are not satisfactory.
- Run optimisation and verify obtained solution. If some obtained results are not satisfactory, one has to modify some of the optimisation parameters, to re-adjust the central controller or to change some design specifications.

6 Sensitivity shaping by convex optimisation versus H_∞ optimisation

Suppose a general controller design problem solved by H_∞ -optimisation as it is shown in Fig.3. In sensitivity shaping by convex optimisation (S.S.C.O.) design framework the configuration is the same only output weighting filters W_y and W_u are removed and replaced with templates on corresponding sensitivities. The input weighting filters W_d , W_p , W_b are considered purely as disturbance models. Finally, the weighting W_H corresponds to H_K in our framework, thus there is no need of gain selection for this block.

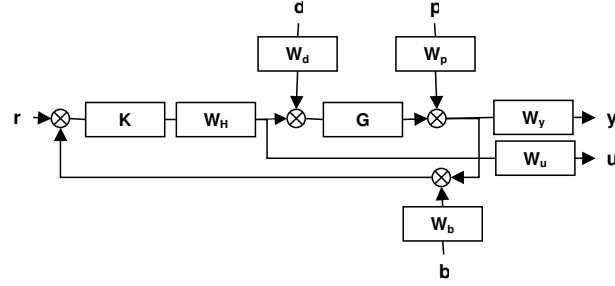


Figure 3: General H_∞ -optimisation problem configuration

In the case of H_∞ -optimisation the minimized criterion function can be viewed as follows:

$$\phi_{H_\infty} = \left\| \begin{array}{ccc} W_y S_{yd} W_d & W_y S_{yp} W_p & W_y S_{yb} W_b \\ W_u S_{ud} W_d & W_u S_{up} W_p & W_u S_{ub} W_b \end{array} \right\|_\infty - \gamma \quad (31)$$

with the constant γ appropriately small. On the other hand, for S.S.C.O. design we have two (constraint and objective) criterion functions:

$$\phi_{OBJ}(X) = \max_{ij} [\alpha_{ij} \phi_{ij}(X)] \quad (32)$$

$$\phi_{CST}(X) = \max_{ij} [\alpha_{ij} \phi_{ij}(X)] \quad (33)$$

where the index i corresponds to inputs d , p , b and the index j to outputs u , y . According to (24), the function $\phi_{ij}(X)$ has the following general form:

$$\phi_{ij}(X) = |d_{N_Q}| \bar{\sigma}(W_i S_{ij}) - \text{Re}\{d_{N_Q}\} F_{ij} \quad (34)$$

where F_{ij} is a bound on S_{ij} . Additionally, we can add to constraint or objective function a criterion function (25) dealing with any element of any sensitivity. Comparing (31) with (32), (33), and (34), one notes that:

- The selection of the bound F_{ij} is similar to the choice of weighting matrices in H_{infy} , but it is direct and transparent. Also a rational expression of

the bound is not necessary, one needs only a set of points in the frequency domain.

- The choice of weighting coefficients α_{ij} (important mainly for objective function) corresponds to gain selection for weighting matrices. In both cases the aim is to create an appropriate balance between different sensitivity shaping criterions.
- In S.S.C.O. design, every sensitivity can have its own bound (template) and thus it can be adjusted totally independently and any combination of bounded sensitivities is possible. H_∞ -optimisation is less flexible from this point of view.
- With (25) the frequency bounds can be specified on transfer function elements of any sensitivity matrix S_{ij} while this is not possible in H_∞ -optimisation. For example, to impose decoupling, it suffices to fix a set of low-modulus bounds on non-diagonal transfer function elements of S_{yb} matrix.
- The criterion functions $\phi_{ij}(X)$ are more conservative than H_∞ -optimisation criterion function.
- The controller complexity can be increased by rising the parameterization order N_q , if design specifications are difficult to satisfy. In H_∞ -optimisation, the weighting filters have to be modified in such case.
- The parametrization of S.S.C.O. needs a central controller, and so before applying the optimisation one has to design this controller. The complexity of the central controller is $\deg(K_0) = \deg(G) + 2\deg(H_K)$ and it corresponds to H_∞ -optimized controller designed so that the sum of degrees for all weighting filters gives $\sum_k \deg(W_k) = 2\deg(H_K)$. Because in our procedure the block H_K contains generally only an integrator, the value is relatively small $2\deg(H_K) = 2n_u$.
- Comparing approximately H_∞ and S.S.C.O. design, the S.S.C.O. design method replaces the iterative and quite heuristic weighting filter selection by automated convex optimisation, but first a simple controller has to be designed. Obviously, such approach is useful only for more complex problems, where the simple controller can not satisfy design specifications.
- The S.S.C.O. use directly a discrete-time representation of the plant. This allows to use standard identified discrete-time models and the resulting controller is in discrete-time as well. One avoids the controller discretization leading in general to high sampling frequency and therefore to a high-performance hardware requirements.

7 Multivariable controller design example

7.1 Model and closed-loop specifications

The used MIMO model is a continuous-time linearized model of vertical-plane dynamics of an aircraft. The model has 3 inputs, 3 outputs, and 5 states and it was taken from [10], where H_∞ -optimisation based continuous-time controller is designed for this model. The model has 5 poles, two pairs of complex poles at 0.029194Hz, 0.205589Hz with the corresponding damping of 0.0959353 and of 0.60387, and one real pole at the origin -pure integrator.

Using the results from the continuous time design of [10], we constructed the following control design objectives:

1. desired closed loop dynamic f_{CL} of about 1.6Hz (corresponding rising time is about 0.35s)
2. maximal overshoot about 10%.
3. to attenuate high-frequency noise disturbance
4. to restrain, if possible, interactions between loops
5. the controller output signal has to be acceptable (without any important oscillations).
6. an integral action in each loop in order to eliminate steady state errors
7. to minimize the sensitivity S_{yp} to obtain robustness as good as possible.

To apply our controller design methodology a discrete-time equivalent of the continuous-time plant model is necessary. Choice of the sampling time T_e depends on model dynamics and on desired closed loop dynamics. Since the model poles are between 0Hz and 0.206Hz and the desired closed loop dynamics is $f_{CL} = 1.6\text{Hz}$, the sampling frequency is set to $f_e \cong 12f_{CL}$, precisely $f_e = 20\text{Hz}$ ($T_e = 0.05\text{s}$). For continuous to discrete-time transformation, command "c2d" of MATLAB[®], and particularly zero-order hold on the model inputs method, was used. The resulting 5-state discrete time model has the poles at the same frequencies and with the same damping as its continuous time equivalent.

7.2 Controller design

Design specifications. The desired closed-loop specifications 1-7 are transformed to design specifications suitable for optimisation as follows:

- *Specification 1* - A template for $\bar{\sigma}(S_{yp})$ shown in Fig. 4 by crosses is fixed as a constraint. The low-frequency ramp with the minimum of -12dB guarantees an appropriate closed-loop step response (rise time about 0.35s). The 6dB-limit in the other region assures primitive robustness.

- *Specification 2 and 3* - A template for $\bar{\sigma}(S_{yb})$ shown in Fig. 5 by crosses is imposed as a constraint. In low frequencies, the response is limited to 2dB - corresponding maximal overshoot is about 25%, but largest singular value maximum is generally higher than frequency characteristic maximum. In high frequencies, $\bar{\sigma}(S_{yb})$ is considered to lie below the ramp with the minimal value -18dB assuring sufficient noise attenuation.
- *Specification 4* - The required low interaction is restricted by a set of bounds imposed on non-diagonal transfer function characteristics of $|S_{yp}|$. All the bounds are set to -19dB (see Fig.6) corresponding approximatively to the interaction of 0.11.
- *Specification 5* - Controller gain can be restricted by a bound imposed to frequency characteristic of S_{up} . Since an appropriate shape of such bound is unknown in our case, the 55dB-limit for $\bar{\sigma}(S_{up})$ is chosen as a constraint according to central controller sensitivity response (maximum about 50dB). The central controller is designed in next two steps, and so this bound is set later when the central controller is done.
- *Specification 6* - An integral action is implemented via pre-fixed block H_K specified in next step.
- *Specification 7* - A set of templates (at 3dB) for every transfer function frequency characteristic of $|S_{yp}|$ is imposed as an objective.

Note, that the desired specifications 1-6 are defined as optimisation constraints and only specification 7 is our objective. This configuration drastically simplifies choice of weighting coefficients α_k as it will be shown.

Pre-fixed part $H_K(z)$. In order to verify specification no.6, a transfer matrix of 3 almost-integrators (with a pole placed to $z = 0.9999$), one in each loop, is put to pre-fixed block $H_K(z)$.

Design of central controller. In our design problem we decided to use partial internal model control (a part of model dynamics is reassigned to closed loop) with the sub-optimal closed-loop pole assignment shown in Tab.1.

The modified model poles (the oscillating poles are sufficiently damped and the real pole is accelerated to accelerate CL response) are introduced to the observer poles. A sufficient high frequency attenuation is provided by two multiple real poles: $(z - 0.498)^3$ in observer CL poles, and $(z - 0.1)^3$ in estimated feedback CL poles. The frequency of state feedback CL poles is higher than the desired CL dynamics $f_{CL} = 1.6\text{Hz}$. In this way we try to reduce the influence of the slow dynamics in the observer necessary for a good robustness. The applied pole-placement algorithm is [3] (Matlab function "place"). The obtained frequency characteristics are given in Fig.4, Fig.5 in dotted lines.

pole type	frequency[Hz]/ position	damping/ multiplicity
o.p.		
complex pair	0.029194Hz	0.967
complex pair	0.205589Hz	0.93
real	0.96	1
real	0.498	3
e.s.f.p.		
3 complex pairs	2.8Hz	0.8
real	0.1	2

Table 1: Closed-loop poles assigned with central controller (o.p. -observer poles; e.s.f.p. -estimated state feedback poles)

Optimisation parameters. The weighting coefficients α_k (playing the role of scaling factors) defined in (30) has to be chosen. Their values are all set to 1 except the interaction weighting coefficients $\alpha_{ybv_1v_2}$ for non-diagonal frequency characteristics of S_{yb} , which is set to 10. Such a simple choice is possible thanks to selected constraint-objective configuration - several constraints and one objective. The value 10 in weighting coefficient set is not necessary. It only accentuates importance of interaction with respect to the rest of constraint criterion functions.

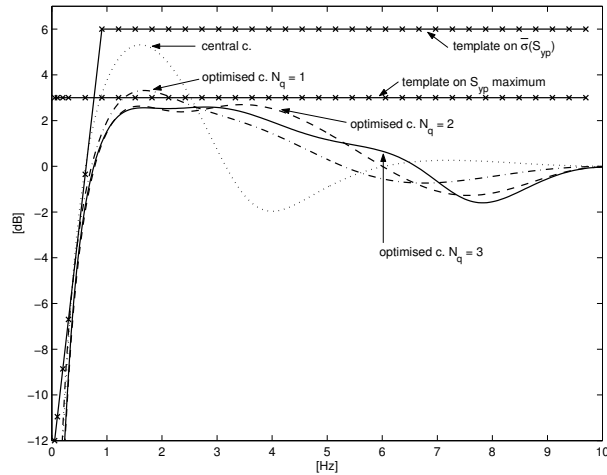


Figure 4: Frequency characteristics of $|S_{yp}(e^{j\omega_d})|$ maximal response for central controller, and the optimized controllers.

Optimization no.1. The optimisation is proceed with the parameterization order $N_q = 1$ and hence controller has 14 states. The resulting frequency characteristics are shown in Fig.4, Fig.5 (dash-dotted lines). The maximum of $|S_{yp}|$ characteristics (Fig.4) is about 3.3dB and all imposed constraints are verified.

Optimization no.2. The parameterization order is increased to $N_q = 2$, which produce a controller with 17 states. The resulting sensitivity characteristics are given in Fig.4, Fig.5 (dashed lines). Visibly, the maximum of $|S_{yp}|$ was reduced to 2.7dB respecting all constraints.

Optimisation no.3. The parameterization order is set to $N_q = 3$ with corresponding controller of 20 states. The sensitivity characteristics are given in Fig.4, Fig.5 (solid lines). Fig.6 shows frequency characteristics of $|S_{yb}|$ illustrating obtained closed-loop interaction, and tracking performances and noise attenuation in each channel.

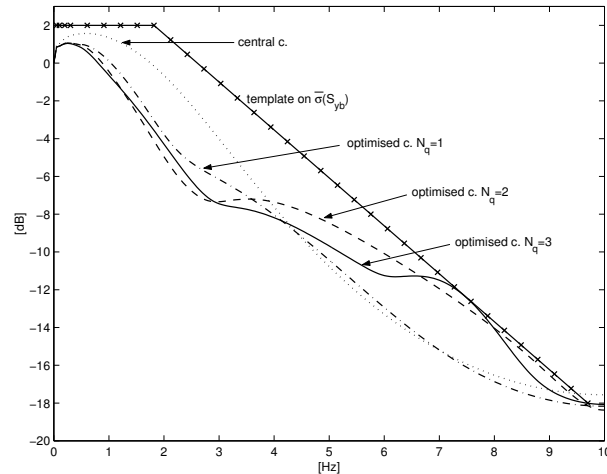


Figure 5: Evolution of $\bar{\sigma}(S_{yb}(e^{j\omega_d}))$ for the central controller, and the optimised controllers.

Finally, closed loop step responses, which demonstrate a disturbance rejection as well as an interaction, are shown in Fig.7. These time responses are simulated in Simulink for MATLAB by applying the 17-state digital controller (designed with $N_q = 2$) on the continuous-time model. The obtained rising time is the one required - about 0.35s or less. The maximum overshoot is 11.4%, and the most important interaction is about 8.9%. Compared to the H_∞ -optimisation results attained in [10], we have equivalent or better decoupling, similar rising time, lower overshoot and the same controller degree. In Tab.2, the maximums of $|S_{yb}|$ and $|S_{yp}|$ characteristics and controller complexities are shown for designed controllers and for H_∞ -optimised controller.

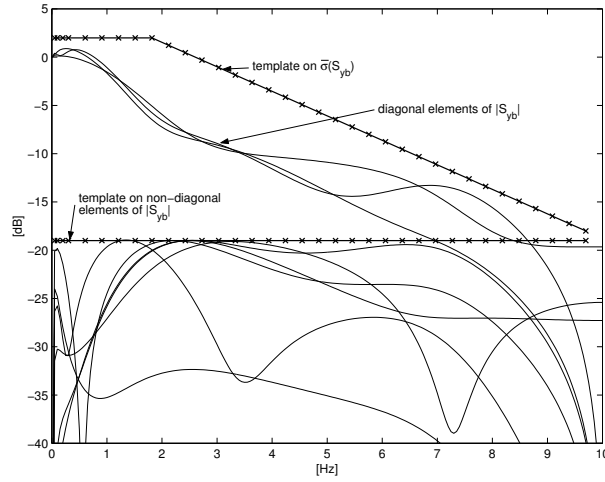


Figure 6: Frequency characteristic of $|S_{yp}(e^{j\omega_d})|$ for the the optimised controller with $N_q = 3$.

8 Conclusion

The paper presents a new procedure of MIMO robust controller design, which extends [9] to multivariable domain. Based on "Youla-Kucera" parameterization, sensitivity shaping approach and multivariable pole assignment the procedure is shown to be an alternative for H_∞ -optimisation design. It proves some important advantages, but it suffers with some inconveniences as well. The procedure allows to treat easily, transparently and with high flexibility the closed-loop design specifications, the design procedure is partially automated and obtained controllers are in discrete-time and therefore directly implementable. Moreover, the procedure allows a trade-off between performance and controller complexity. On the other hand, a simple central controller is necessary before the optimisation can be applied and the used criterion functions are conservative. Compared on one design example, both procedures give controllers of the same complexity and with similar performances.

Controller	$ S_{yp} _{max}$ dB	$ S_{yb} _{max}$ dB	controller states
central	5.31	1.32	11
convex opt.1	3.32	0.90	14
convex opt.2	2.70	0.89	17
convex opt.3	2.59	0.89	20
H_∞ opt.	2.69	1.8	17

Table 2: Comparison of $|S_{yp}|$ and $|S_{yb}|$ maximums and controller complexities

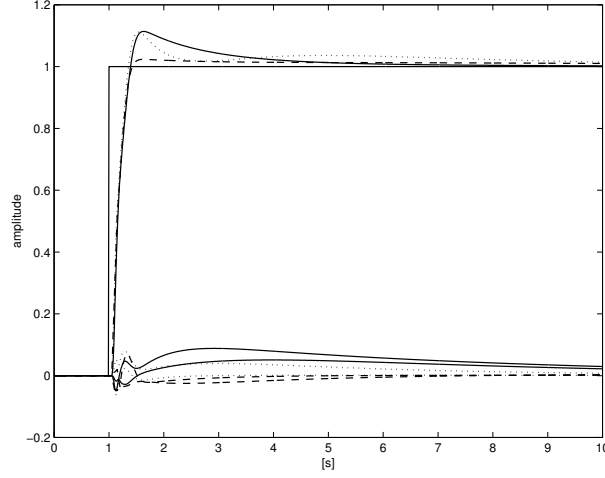


Figure 7: Closed loop step responses of output 1 (solid), output 2 (dashed), and output 3 (dotted)

A Proof of convexity for developed criterion function

Any convex function $\phi(X) : R^l \rightarrow R$ has to verify following condition [13, 15]:

$$\phi(\lambda X_1 + (1 - \lambda)X_2) \leq \lambda\phi(X_1) + (1 - \lambda)\phi(X_2) \mid \forall \lambda \in \langle 0, 1 \rangle; \forall X_1, X_2 \in R^l$$

For the criterion function (24) the convexity condition becomes:

$$\begin{aligned} & \sup_{\omega_d} [\bar{\sigma}(T_1 d_{N_q}(\lambda X_1 + (1 - \lambda)X_2) + T_2 N_{N_q}(\lambda X_1 + (1 - \lambda)X_2) T_3) - \\ & - \operatorname{Re}\{d_{N_q}(\lambda X_1 + (1 - \lambda)X_2)\} F_{ij}] \leq \\ & \leq \lambda \sup_{\omega_d} [\bar{\sigma}(T_1 d_{N_q}(X_1) + T_2 N_{N_q}(X_1) T_3) - \operatorname{Re}\{d_{N_q}(X_1)\} F_{ij}] + \\ & + (1 - \lambda) \sup_{\omega_d} [\bar{\sigma}(T_1 d_{N_q}(X_2) + T_2 N_{N_q}(X_2) T_3) - \operatorname{Re}\{d_{N_q}(X_2)\} F_{ij}] \quad (35) \end{aligned}$$

Using the properties of the H_∞ norm and of the function $\sup[\cdot]$:

$$\begin{aligned} \bar{\sigma}(H_1 + H_2) & \leq \bar{\sigma}(H_1) + \bar{\sigma}(H_2) \mid \forall H_1, H_2 \in C^{m \times n} \\ \bar{\sigma}(\lambda H) & \leq \lambda \bar{\sigma}(H) \mid \forall H \in C^{m \times n}, \lambda \in \langle 0, 1 \rangle \\ \sup_x [f(x) + g(x)] & \leq \sup_x [f(x)] + \sup_x [g(x)] \end{aligned} \quad (36)$$

and the following properties:

$$\begin{aligned} d_{N_q}(X_1 + X_2) & = d_{N_q}(X_1) + d_{N_q}(X_2) \\ N_{N_q}(X_1 + X_2) & = N_{N_q}(X_1) + N_{N_q}(X_2) \\ d_{N_q}(\lambda X) & = \lambda d_{N_q}(X) \\ N_{N_q}(\lambda X) & = \lambda N_{N_q}(X) \end{aligned} \quad (37)$$

for $X_1, X_2, X \in R^l$ and $\lambda \in \langle 0, 1 \rangle$. We develop the left side of the convexity condition (35):

$$\begin{aligned}
& \sup_{\omega_d} [\bar{\sigma}(T_1 d_{N_q}(\lambda X_1 + (1 - \lambda)X_2) + T_2 N_{N_q}(\lambda X_1 + (1 - \lambda)X_2)T_3) \\
& \quad - Re\{d_{N_q}(\lambda X_1 + (1 - \lambda)X_2)\}F_{ij}] = \\
= & \sup_{\omega_d} [\bar{\sigma}(T_1 d_{N_q}(\lambda X_1) + T_1 d_{N_q}((1 - \lambda)X_2) + T_2 N_{N_q}((1 - \lambda)X_2)T_3) + T_2 N_{N_q}(\lambda X_1)T_3 \\
& \quad - Re\{d_{N_q}(\lambda X_1) + d_{N_q}((1 - \lambda)X_2)\}F_{ij}] = \\
= & \sup_{\omega_d} [\bar{\sigma}(\lambda T_1 d_{N_q}(X_1) + \lambda T_2 N_{N_q}(X_1)T_3 + (1 - \lambda)T_1 d_{N_q}(X_2) + (1 - \lambda)T_2 N_{N_q}(X_2)T_3) \\
& \quad - Re\{\lambda d_{N_q}(X_1) + (1 - \lambda)d_{N_q}(X_2)\}F_{ij}] \leq \\
\leq & \sup_{\omega_d} [\lambda \bar{\sigma}(T_1 d_{N_q}(X_1) + T_2 N_{N_q}(X_1)T_3) + (1 - \lambda)\bar{\sigma}(T_1 d_{N_q}(X_2) + T_2 N_{N_q}(X_2)T_3) \\
& \quad - \lambda Re\{d_{N_q}(X_1)\}F_{ij} - (1 - \lambda)Re\{d_{N_q}(X_2)\}F_{ij}] \leq \\
\leq & \sup_{\omega_d} [\lambda \bar{\sigma}(T_1 d_{N_q}(X_1) + T_2 N_{N_q}(X_1)T_3) - \lambda Re\{d_{N_q}(X_1)\}F_{ij}] + \\
& \sup_{\omega_d} [(1 - \lambda)\bar{\sigma}(T_1 d_{N_q}(X_2) + T_2 N_{N_q}(X_2)T_3) - (1 - \lambda)Re\{d_{N_q}(X_2)\}F_{ij}] = \\
= & \lambda \sup_{\omega_d} [\bar{\sigma}(T_1 d_{N_q}(X_1) + T_2 N_{N_q}(X_1)T_3) - Re\{d_{N_q}(X_1)\}F_{ij}] + \\
& (1 - \lambda) \sup_{\omega_d} [\bar{\sigma}(T_1 d_{N_q}(X_2) + T_2 N_{N_q}(X_2)T_3) - Re\{d_{N_q}(X_2)\}F_{ij}]
\end{aligned}$$

and obviously this term is equal to the right side of (35).

References

- [1] S.P. Boyd and C.H. Barratt. *Linear controller design - Limits of performance*. Prentice Hall, New Jersey, 1991.
- [2] A. Constantinescu and I.D. Landau. Controller order reduction by identification in closed loop applied to a benchmark problem. *European Journal of Control*, (1), 2003.
- [3] J. Kautsky, N.K. Nichols, and P. Van Doren. Robust pole assignment in linear state feedback. *Int. J. Control*, 41(5):1129–1155, 1985.
- [4] V. Kucera. *Discret Linear Control: The polynomial equation approach*. Wiley, New York, 1979.
- [5] I.D. Landau. *Commande des systemes*. Hermes, Paris, 2002.
- [6] I.D. Landau and A. Karimi. Robust digital control using pole placement with sensitivity function shaping method. *Int. J. Robust and Nonlin. Cont.*, 8:191–210, 1998.
- [7] J. Langer. *Synthese de regulateurs numeriques robustes - Application aux structures souples*. These, Lab. d'Automatique de Grenoble, 1998.

- [8] J. Langer and A. Constantinescu. Pole placement design using convex optimization criteria for the flexible transmission benchmark. *European Journal of Contr.*, 5:193–207, 1999.
- [9] J. Langer and I.D. Landau. Combined pole placement/sensitivity function shaping method using convex optimization criteria. *Automatica*, pages 1111–1120, 1999.
- [10] J.M. Maciejowski. *Multivariable Feedback Design*. Addison-Wesley, 1989.
- [11] J.F. Magni. *Robust modal control*. Kluwer Academic/Plenum Publishers, New York, 2002.
- [12] H. Prochazka and I.D. Landau. Pole placement with sensitivity function shaping using 2nd order digital notch filters. *Automatica*, 39(6):1103–1107, 2003.
- [13] R.T. Rockafellar. *Convex Analysis*. Princeton Univ. Press, 1982.
- [14] K.M. Sobel, E.Y. Shapiro, and A.N. Andry jr. Eigenstructure assignment. *Int. J. Control*, 59(1):13–37, 1994.
- [15] J. Stoier and C. Witzgall. *Convexity and Optimisation in Finite Dimensions*. Springer-Verlag, 1970.
- [16] A.L. Tits and Y. Yang. Globally convergent algorithms for robust pole assignment by state feedback. *IEEE Trans. Autom. Control*, 41(10):1432–1452, 1996.
- [17] A. Varga. Robust pole assignment via sylvester equation based state feedback parametrization. *Proc. IEEE International Symposium on Computer Aided Control System Design, CACSD'2000, Anchorage, Alaska*, 2000.
- [18] D.C. Youla, H.A. Jabri, and J. Bongiorno. Modern wiener-hopf design of optimal controllers. *IEEE Trans. on Automatic Control*, AC-21:319–338, 1976.
- [19] K. Zhou. *Essentials of Robust Control*. Prentice Hall, New Jersey, 1998.

Multivariable controller reduction by closed-loop identification

Hynek Procházka*, Ioan Doré Landau†

January 15, 2004

Abstract

A simple procedure for controller reduction of both single-input, single-output and multi-input, multi-output digital controllers is presented. The procedure is based on basic closed-loop identification and it requires only discrete-time plant model and any open-loop identification algorithm. The capability of the proposed technique is illustrated on one SISO practical and one MIMO academical example.

1 Introduction

It is well known, that sophisticated advanced controller design techniques such as H_∞ -optimization, LQG, or convex optimization leads often to complex solutions. An implementation of a complex controller is difficult and expensive, additionally there is often a doubt about its reliability. Therefore it is not surprising that an important effort was focused on reduced-order controller design in latest years. An early works outlining the principal reduced-complexity controller design approaches is [AL89]. Currently often used approach is the direct controller reduction technique. In this approach a complex high-performance controller is supposed to be designed using some already developed advanced design technique. This controller is a priori considered to satisfy all imposed closed-loop performance specifications. Then, the controller is reduced by a reduction technique, which has one main objective to preserve at large the obtained closed-loop performances. Consequently, the reduction technique is efficient only if the closed loop is taken into account in some way.

There is several types of reduction techniques conserving closed-loop performances. Among others one can mention the most common method based on

*Laboratoire d'Automatique de Grenoble (CNRS-INPG-UJF), ENSIEG; BP. 46, 38402 St. Martin d'Hères, FRANCE; e-mail: hynek.prochazka@lag.ensieg.inpg.fr, Tel: +334 76 82 62 44; Fax: +33 4 76 82 63 88

†Laboratoire d'Automatique de Grenoble (CNRS-INPG-UJF), ENSIEG; BP. 46, 38402 St. Martin d'Hères, FRANCE; e-mail: landau@lag.ensieg.inpg.fr; Tel: +33 4 76 82 63 91; Fax: +33 4 76 82 63 88

weighted balanced-order reduction (see for instance [WSL02]), or coprime factor reduction [AL89, LAL90], or the recently developed single-input, single-output digital controller reduction by identification in closed loop [LKC01]. The lastly mentioned technique reduces controllers with a modified algorithm of closed-loop identification [LK97]. Roughly, we can say that instead of a plant model a reduced controller is identified in a closed loop. We explore this idea in this note to develop a simple multivariable reduction technique.

For controller reduction we suppose to dispose with a discrete-time plant model and of course with a complex controller to be reduced satisfying some desired closed-loop specifications. Then, using a basic closed-loop identification procedure we identify a reduced-order controller connected in closed loop with the model. The identification procedure can be summarized as follows: Firstly, an appropriate external excitation is added to the closed loop. Next, controller inputs and outputs are collected, and applied afterward to a standard open-loop identification algorithm. To accentuate more the closed-loop impact, the external excitation is additionally weighted before application. The used advanced identification algorithm is the subspace identification method [vOM96] giving very satisfactory results. Any other multivariable algorithm can be used (as for instance [Lan78]) and obviously better identification algorithms give better solutions. The reduction procedure is illustrated on two examples, one practical single-input, single-output, the other academical and multivariable.

The note is organized as follows: Section 2 introduce some necessary notation. In Section 3, the principal idea of controller order reduction is developed. Then, Section 4 presents the two examples of application and Section 5 gives some conclusions.

2 Notations

2.1 Closed loop

Consider the closed-loop multivariable system shown in Fig. 1, where G is either a real plant or a plant discrete-time model and K is a digital controller. The model and the controller are given either by state space representations:

$$\begin{aligned} x_G(t+1) &= A_G x_G(t) + B_G u_G(t) \\ y_G(t) &= C_G x_G(t) \end{aligned} \quad (1)$$

$$\begin{aligned} x_K(t+1) &= A_K x_K(t) + B_K e(t) \\ u(t) &= C_K x_K(t) + D_K e(t) \end{aligned} \quad (2)$$

or by the transfer matrices:

$$\begin{aligned} G(z^{-1}) &= C_G(zI + A_G)^{-1} B_G \\ K(z^{-1}) &= C_K(zI + A_K)^{-1} B_K + D_K \end{aligned} \quad (3)$$

where the time vectors are defined as follows: $x_G(t) \in R^{n_{x_G}}$, $x_K(t) \in R^{n_{x_K}}$, $y_G(t) \in R^{n_y}$, and $u(t) \in R^{n_u}$ and the state space matrices have corresponding

dimensions. The operator z^{-1} represents either the frequency operator $z^{-1} = e^{-j\omega T_s}$ with T_s the sampling time, or the time delay operator $z^{-1}u(t) = u(t-1)$.

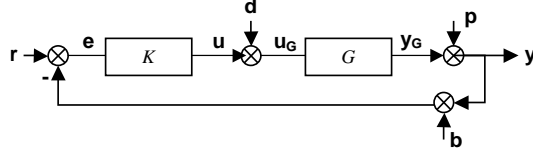


Figure 1: Considered closed loop configuration

Now the sensitivity matrices (sensitivities) of the closed loop with the model and the controller are defined as follows:

- $S_{yp}(z^{-1}) = \frac{y(z^{-1})}{p(z^{-1})}$: Output disturbance - output sensitivity

$$S_{yp}(z^{-1}) = (I + G(z^{-1})K(z^{-1}))^{-1} \quad (4)$$

- $S_{up}(z^{-1}) = \frac{u(z^{-1})}{p(z^{-1})}$: Output disturbance - command sensitivity

$$S_{up}(z^{-1}) = -K(z^{-1})(I + K(z^{-1})G(z^{-1}))^{-1} \quad (5)$$

- $S_{yb} = \frac{y(z^{-1})}{b(z^{-1})}$: Noise - output sensitivity

$$S_{yb}(z^{-1}) = -G(z^{-1})K(z^{-1})(I + G(z^{-1})K(z^{-1}))^{-1} \quad (6)$$

- $S_{yd} = \frac{y(z^{-1})}{d(z^{-1})}$: Input disturbance - output sensitivity

$$S_{yd}(z^{-1}) = G(z^{-1})(I + G(z^{-1})K(z^{-1}))^{-1} \quad (7)$$

2.2 Model identification in closed loop

In this subsection we recall two procedures for model identification in closed loop. Based on these procedures, a new controller reduction technique will be developed in next section.

The first closed loop identification procedure is well known for a long time. It can be illustrated by the diagram in the Fig. 2 a) and it consists of following steps:

1. Connect the controller K with the real plant to the closed loop as it is shown in Fig. 1.
2. Add to the loop a persistently exciting external signal $r(t)$ or $d(t)$. The signals $b(t)$ and p are not considered, since they have the same effect as the signal $r(t)$. A pseudo random binary sequence is generally used as the external excitation.

3. Make data acquisition and collect the $u(t)$ and $y(t)$ data signals.
4. Apply an open-loop identification algorithm onto collected data sequences and hence identified a new model. Among others following multivariable algorithms can be used:
 - Sub-space identification [vOM96, Fav99] - It is one of the most efficient multivariable identification algorithms. The identification is provided by geometric operations (oblique projections) over the input/output data sequences. Moreover, it allows to weight from the left and from the right identified model.
 - Recursive least-square (RLS) identification [Lan78, Duo93]- The origin of this algorithm is in classical SISO identification (see for example [Lan02]). A square output error is minimized, where the error is difference between real output and estimated model output.

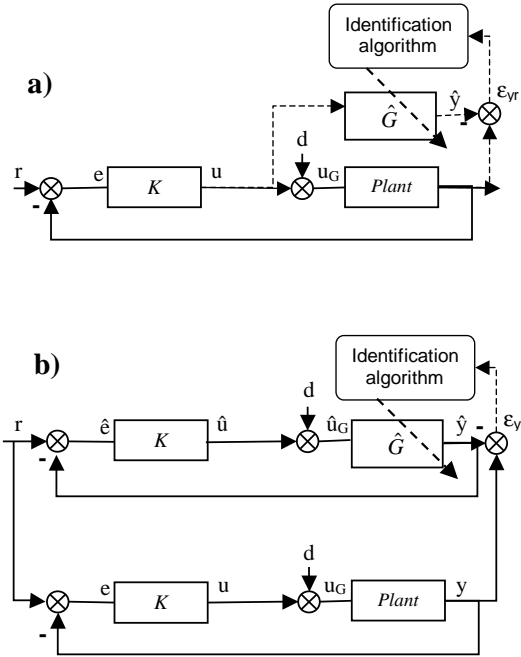


Figure 2: Two closed loop model identification procedures.

If the RLS identification algorithm is applied, roughly the plant model is identified by minimization of the following output error:

$$\epsilon_{yr}(t) = (y(t) - \hat{y}(t)) = \left(P_l K (I + P_l K)^{-1} - \hat{G} K (I + P_l K)^{-1} \right) r(t) \quad (8)$$

where P_l is the real plant, \hat{G} is identified model, and $\hat{y}(t)$ is output estimation. Note, that $\hat{y}(t)$ is in reality computed from $y(t-1)$, $y(t-2)$, ... and not from $\hat{y}(t-1)$, $\hat{y}(t-2)$, ... The minimized criterion has the form:

$$J(t) = \frac{1}{t} \sum_{i=1}^t |\varepsilon_{yr}(i)|^2 = \frac{1}{t} \sum_{i=1}^t |\varepsilon_{yr}(i)| |\varepsilon_{yr}^T(i)| \quad (9)$$

with $J(t)$ minimized in each identification step by adjusting \hat{G} .

Remark: Notice, that the plant input $u_G(t)$ is in fact a model external excitation signal filtered by a corresponding closed loop sensitivity. This can be viewed as a frequency weighted identification, where the weighting filters are provided by closed-loop sensitivities.

Other, more efficient (mainly in presence of noise), closed loop identification method was developed for SISO systems [LK97]. It is based on RLS identification algorithm modified for identification in closed loop. The identification procedure is similar as for the previous method, only the output estimation $\hat{y}(t)$ is different. The output error becomes:

$$\varepsilon_{yr}(t) = (y(t) - \hat{y}(t)) = \left(P_r K (I + P_r K)^{-1} - \hat{G} K (I + \hat{G} K)^{-1} \right) r(t) \quad (10)$$

Notice that the error represent exactly the discrepancy between real and simulated closed loop. We will not use this method.

3 Multivariable controller reduction

3.1 Objective criterions

The goal of controller reduction is to obtain a reduced order controller \hat{K} of less complexity than an original (nominal) controller K . In addition, the reduced controller \hat{K} has to preserve at large extent closed loop performance and robustness (which are the real control objectives) of the nominal controller. A number of criterions can be used to measure the difference between the reduced controller \hat{K} and the nominal controller K closed loop performance and robustness. Criterions to measure sensitivity discrepancy are typically applied [AL89, WSL02, LKC01] etc.

Consider two closed loops: one with the nominal controller K and the other with reduced controller \hat{K} . A criterion to measure a sensitivity discrepancy can be an error of some two corresponding signals, as it shows Fig. 4. Suppose that S_{yb} sensitivity variation is measured, then the criterion is output error ε_{yr} defined as:

$$\varepsilon_{yr}(t) = (y(t) - \hat{y}(t)) = \left(-S_{yb}(z^{-1}) + \hat{S}_{yb}(z^{-1}) \right) r(t) \quad (11)$$

where $y(t) = -S_{yb}(z^{-1})r(t)$. In (11) the information on sensitivity discrepancy is contained in the term $(-S_{yb}(z^{-1}) + \hat{S}_{yb}(z^{-1}))$. Hence, H_2 norm of this term

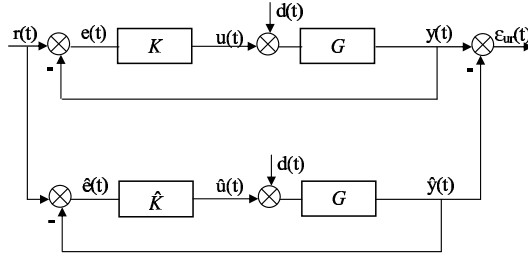


Figure 3: Output error configuration

can be provided an appropriate criterion as well. A corresponding expression to obtain an optimal reduced controller \hat{K}^* can have, for example, the following form (the z^{-1} notation is omitted):

$$\begin{aligned} \hat{K}^* &= \arg \min_{\hat{K}} \|S_{yb} - \hat{S}_{yb}\|_{\infty} = \\ &= \arg \min_{\hat{K}} \left\| -GK(I + GK)^{-1} + G\hat{K}(I + G\hat{K})^{-1} \right\|_{\infty} \end{aligned} \quad (12)$$

or using the output error the following form:

$$\hat{K}^* = \arg \min_{\hat{K}} \int_{-\infty}^{\infty} |\varepsilon_{yr}(t)|^2 dt \quad (13)$$

and according to Parseval's theorem the expression (13) can be written in frequency domain as follows:

$$\hat{K}^* = \arg \min_{\hat{K}} \frac{1}{2\pi} \int_{-\pi}^{\pi} \left| -S_{yb} + \hat{S}_{yb} \right|^2 \phi_r(j\omega) d\omega \quad (14)$$

where $\phi_r(j\omega)$ is spectral density of an excitation signal $r(t)$. The equations (13), (14) represent energy minimization of the error $\varepsilon_{yr}(t)$.

From (12) we have:

$$\|S_{yb} - \hat{S}_{yb}\| \cong \left\| \underbrace{(I + GK)^{-1}}_{S_{yp}} G(\hat{K} - K) \underbrace{(I + G\hat{K})^{-1}}_{S_{yp}} \right\| \quad (15)$$

where the term $(\hat{K} - K)$ of the second order is omitted [AL89]. We observe that it is a controller discrepancy minimization weighted in frequency domain.

Using the approximation (15), the output error has the following form:

$$\varepsilon_{yr}(t) = (y(t) - \hat{y}(t)) \cong (S_{yp}G(\hat{K} - K)S_{yp})r(t) \quad (16)$$

The expressions (15) and (16) indicate that the sensitivity discrepancy minimization can be approximated by a minimization of weighted controller difference $(\hat{K} - K)$. This idea will serve us later to build a controller reduction technique.

Remark: The weighting of the term $(\hat{K} - K)$ is made by S_{yp} et $S_{yp}G$ sensitivities. It accentuates the frequency zones of the term $(\hat{K} - K)$, where $|S_{yp}|$ frequency characteristic is higher, hence where the system robustness is more critical.

3.2 Reduction by identification in closed loop

Consider closed loop identification procedure described in section 2.2 but applied on the controller K . Additionally, suppose the real plant replaced with the plant model G . The obtained configuration is in the Fig. 4, where the input error ε_{ur} has the forme:

$$\varepsilon_{ur}(t) = u(t) - \hat{u}(t) = KS_{yp}r(t) - \hat{K}S_{yp}r(t) = (K - \hat{K})S_{yp}r(t) \quad (17)$$

If the excitation signal $d(t)$ is added instead, the input error is :

$$\varepsilon_{ud}(t) = u(t) - \hat{u}(t) = KS_{yd}d(t) - \hat{K}S_{yd}d(t) = (K - \hat{K})S_{yd}d(t) \quad (18)$$

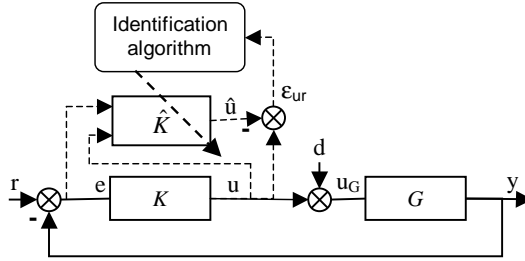


Figure 4: Schéma de l'identification en BF du régulateur

From (17) and (18) we observe that the identification error to be minimized depends on the term $(K - \hat{K})$ weighted with a sensitivity.

In the case of SISO systems (where a commutation of sensitivities is possible), to obtain the error (16) from (17), it suffices to filter the excitation signal $r(t)$ with $S_{yd} = GS_{yp}$ sensitivity. The excitation signal become $r(t) = S_{yd}(z^{-1})r_e(t)$ where $r_e(t)$ is a PRBS. The corresponding input error is now:

$$\varepsilon_{ur}(t) = u(t) - \hat{u}(t) = (K - \hat{K})S_{yp}S_{yd}r_e(t) \quad (19)$$

If $d(t)$ is used as an excitation signal, the same result can be obtained by filtering the excitation with S_{yp} : $d(t) = S_{yd}(z^{-1})d_e(t)$ with the following form of the input error:

$$\varepsilon_{ud}(t) = u(t) - \hat{u}(t) = (K - \hat{K})S_{yd}S_{yp}d_e(t) \quad (20)$$

The expressions (19) and (20), approach (16).

The excitation signal filtering is possible also in MIMO domain, but the resulting errors do not correspond to the error (16). Nevertheless, the filtering allows to provide a satisfactory result even in MIMO system domain. The other point of view is that S_{yp} and S_{yd} are some frequency weights performing amplification or attenuation of certain frequency regions for the controller error term ($K - \hat{K}$). In such case, any sensitivity can be assumed as a filter of excitation signal, in order to realize an appropriate weighting.

Only sub-space algorithm [vOM96] will be used for identification of reduced controllers. It does not directly minimize the square of signal error as the RLS does, but the obtained results are superior with respect to RLS algorithm.

3.3 Controller fixed part

If the reduced-order identified controller $\hat{K}(z^{-1})$ has to contain a fixed part $H_K(z^{-1})$ (for instance an integrator), the full-order controller and the reduced controller have the following forms:

$$K(z^{-1}) = H_K(z^{-1})K'(z^{-1}) \quad (21)$$

$$\hat{K}(z^{-1}) = H_K(z^{-1})\hat{K}'(z^{-1}) \quad (22)$$

where the matrix $\hat{K}'(z^{-1})$ has to be identified. Next, we define an augmented plant model:

$$G_a(z^{-1}) = G(z^{-1})H_K(z^{-1}) \quad (23)$$

and the sensitivities (4) - (7), used for data acquisition and for weighting, are computed from the augmented plant (23) and the matrix $K'(z^{-1})$.

Remark: In (21), it is considered that the nominal controller contains the imposed fixed part $H_K(z^{-1})$. Such requirement is coherent, since if any closed-loop specification demands a fixed part, the complex nominal controller is supposed to be designed with it.

4 Controller reduction examples

In this section two application examples of the proposed controller reduction techniques are presented. The first one is a practical SISO example of a benchmark allowing to compare the proposed technique with best reduction methods used in SISO domain and to test it on real installation. The second example is multivariable and it analyzes performance conservation of the developed reduction technique.

4.1 SISO controller reduction for Active suspension

4.1.1 Active suspension benchmark description

The active suspension system of the benchmark [LKMP03] allows to compare different reduced-complexity controller design methods (for complete bench-

mark description see [LKMP03]). Briefly, the benchmark controller specifications are as follows [LKMP03]:

- Modulus of input sensitivity S_{up} and output sensitivity S_{yp} has to be bounded in frequency domain as it shows Fig.5 solid gray curves.
- The controller numerator has to incorporate the term $(1 - z^{-1})$ opening the loop at the Nyquist frequency $(\frac{1}{2T_s})$.
- Controller orders n_r, n_s should be as low as possible but satisfying desired performance.

The discrete-time identified ARMAX model of the plant is given in [LKMP03] with the orders of numerator and denominator $n_A = 14, n_B = 16$, the sampling time is $T_s = 0.00125s$. Two performance criterions ΔS_{yp}^m and ΔS_{up}^m are also defined in [LKMP03] to measure how the specification on sensitivities are satisfied on the identified model.

Several high-complexity controllers were designed with different methods for the model given in [LKMP03], followed by a reduction technique (see [IKH03, Pro03]). One of the simplest high-complexity controllers with excellent performances is the one designed by pole-placement with sensitivity shaping [LK98, PL03b] presented in [Pro03]. This controller has the orders of numerator and denominator $n_r = 18, n_s = 20$ and it fully satisfies the desired closed-loop specifications ($\Delta S_{yp}^m = 0, \Delta S_{up}^m = 0$). In [Pro03], the controller is reduced using direct closed-loop identification reduction technique [LKC01] and the resulted reduced-complexity controller has the orders of numerator and denominator $n_r = 4, n_s = 4$ with the corresponding performance criterions $\Delta S_{yp}^m = 4.22, \Delta S_{up}^m = 0$.

4.1.2 Controller reduction

In order to test the proposed reduction technique, we will reduce the high-complexity SISO controller designed in [Pro03]. The added excitation signal is a PRBS of $L = 511$ samples, generated with 9-bit shift register and frequency divisor $d = 1$. The signal is added to $r(t)$ or to $d(t)$ and it was filtered either by $S_{yp}(z^{-1})$ or by $S_{yd}(z^{-1})$. The fixed controller part

$$H_K = (1 - z^{-1})$$

required in design specification was imposed as well.

The subspace identification technique [vOM96] without oblique projection weighting was used for reduced-controller identification. This technique is state-space based and thus the number of identified object states has to be specified, which in SISO case leads to the equal degrees of numerator and denominator.

The best resulted reduced-order controller performance criterions are presented in Tab.1. The controller numerator order is always increased by one, this is caused by the imposed part H_R . The Fig.5 shows the comparison of the original sensitivities (with high-complexity controller) - dotted black curve and

ε_u	n_r	n_s	ΔS_{yp}^m	ΔS_{up}^m
$(K - \hat{K})S_{yp}S_{yd}r$	4	3	3.68	0
$(K - \hat{K})S_{yp}S_{yp}r$	4	3	6.36	0
$(K - \hat{K})S_{yp}S_{yp}r$	5	4	4.95	0

Table 1: Resulting reduced-order active suspension controllers

the one of the best obtained reduced controller (with the performance criterion $\Delta S_{yp}^m = 3.68$) - solid black curve. Compared to the other reduced controllers [IKH03, Pro03] designed with the same model, the proposed solutions are entirely comparable with the best given results.

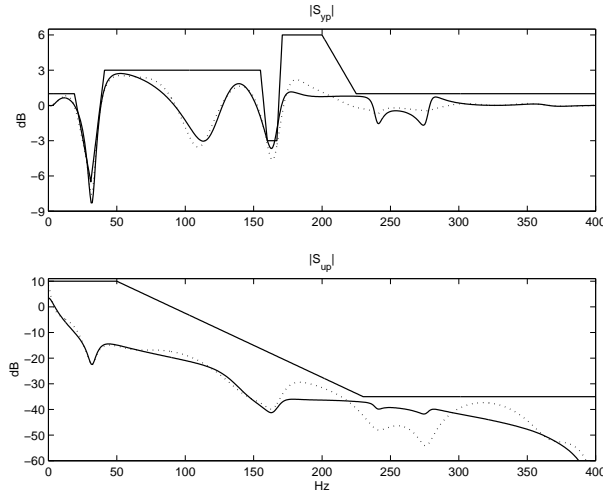


Figure 5: Sensitivity functions for high-complexity controllers (dotted) and for the reduced order controller (solid). The grey curves represent the imposed bounds.

4.2 MIMO controller reduction for Aircraft dynamic

4.2.1 Aircraft dynamic model

The MIMO model is a continuous-time linearized model of vertical-plane dynamics of an aircraft with 3 inputs, 3 outputs, and 5 states [Mac89]. One H_∞ -optimisation based continuous-time controller (with 17 states) is designed in [Mac89] for this model. Four other discrete-time controllers of different complexities (with 11, 14, 17, and 20 states) based on pole placement and convex optimisation are designed in [PL03a] using a discrete-time equivalent of the model ($T_e = 0.05s$).

In [PL03a] the following closed-loop specifications has been imposed on the controller:

1. desired closed loop dynamic f_{CL} of about 1.6Hz (corresponding rising time is about 0.35s)
2. maximal overshoot about 10%.
3. to attenuate high-frequency noise disturbance
4. to restrain, if possible, interactions between loops
5. the controller output signal has to be acceptable (without any important oscillations).
6. an integral action in each loop in order to eliminate steady state errors
7. to minimize the sensitivity S_{yp} to obtain robustness as good as possible.

The performance criterions for controllers obtained in [PL03a] are given in the upper part of Tab.2. All controllers have satisfactory rise time (less then required 0.35s). One controller is computed by multivariable pole-placement with sensitivity shaping (denoted in the table by "p.p."), the other three controllers of different complexities (denoted "c.o.") are designed with convex optimisation technique.

4.2.2 Controller reduction

We decided to reduce the most complex controllers "c.o.3", which provides the best closed-loop performance. The overshoot and the interaction are satisfactory (11% and 9% respectively) and the maximums of $|S_{yp}|$, $|S_{yb}|$ characteristics representing the measure of controller robustness have the lowest values 2.59dB and 0.89dB.

Similarly to the SISO example, for reduced-controller identification, we use the subspace identification technique [vOM96] without oblique projection weighting. A multivariable almost-integrator is considered as a fixed imposed part:

$$H_K = \text{diag}\{(1 - 0.9999z^{-1})^{-1}\}$$

since this block was used also in controller design to satisfy one of the closed-loop specifications. A set of three uncorrelated PRBS generated with 9-bit shift register and with the frequency divisor $d = 1$ is used as the excitation signal.

The best obtained reduced-order controllers are "red.1", "red.2", "red.3", and "red.4" (see Tab.2). The first three controllers ("red.1", "red.2", and "red.3") are identified in the configuration where $r(t)$ is the external excitation signal additionally filtered with $S_{yd}(z^{-1})$. Hence, the identification error has the form:

$$\varepsilon_u^r(t) = \left(K(z^{-1}) - \hat{K}(z^{-1}) \right) S_{yp}(z^{-1}) S_{yd}(z^{-1}) r(t)$$

Con- tro- ller -	max. $ S_{yp} $ dB	max. $ S_{yb} $ dB	max. over- shoot %	max. inter- action %	# sta- tes -
p.p.	5.31	1.32	16	27	11
c.o.1	3.32	0.90	11	8.6	14
c.o.2	2.70	0.89	11	8.8	17
c.o.3	2.59	0.89	11	9.0	20
red.1	2.71	0.91	12	35	14
red.2	3.13	1.12	14	13	11
red.3	3.46	1.14	12	63	10
red.4	3.16	0.95	11	17	11

Table 2: Performance comparison of different controllers (p.p.=pole-placement design, c.o.=convex optimisation design, red.=reduced controller)

While the controller "red.4" is reduced in the configuration where the external excitation is the signal $d(t)$ filtered additionally with $S_{yd}(z^{-1})$. The corresponding identification error is:

$$\varepsilon_u^d(t) = \left(K(z^{-1}) - \hat{K}(z^{-1}) \right) S_{yd}(z^{-1}) S_{yd}(z^{-1}) d(t)$$

Remark that the performance property which suffers the most from the controller reduction is the interaction. The other performance criterions (maximum of S_{yp} and of S_{yb} , overshoot) are relatively preserved and they deteriorate persistently with the decreasing controller order.

It is obvious from Tab.2, that the best reduced controller is "red.2". It has 11 states (9 states was removed) and its closed-loop performances are very close to the desired ones. Note, that the controller complexity is the same as for the pole-placement designed controller "p.p.", but its performances are largely superior, and close to the performances of the original high-complexity controller. A comparison of S_{yp} and S_{yb} largest singular values is given in Fig.6, the comparison of S_{up} and S_{yd} sensitivity singular values is shown in Fig. 7. Observing the two figures, we note that sensitivity frequency responses do not change significantly the shape, which roughly indicates performance conservation. Also, the sensitivity maximums are only slightly higher in the case of reduced controller and it represents a good robustness preservation.

5 Conclusions

In this note we propose a simple and efficient procedure for both SISO and MIMO digital controller reduction preserving closed-loop performance and robustness of original high-complexity controllers. The procedure only requires

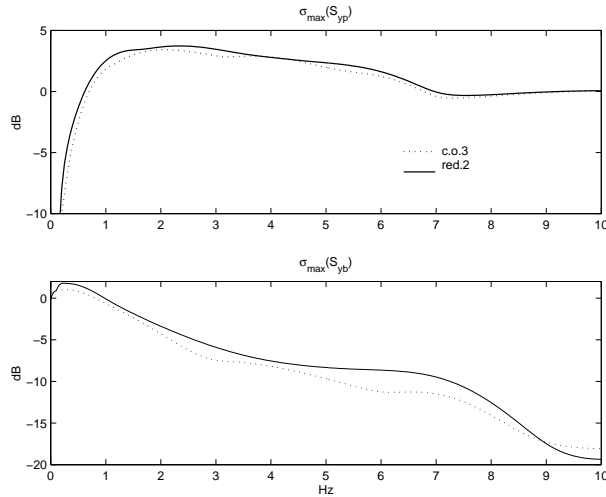


Figure 6: S_{yy} and S_{yb} largest singular values for original controller "c.o.3" (dotted) and for the reduced order controller "red.2" (solid).

an identification algorithm and a discrete-time plant model. Two presented examples proved efficiency of the proposed technique.

References

- [AL89] B.D.O. Anderson and Y. Liu. Controller reduction: Concepts and approaches. *IEEE Transactions on Automatic Control*, 34(8):802–812, 1989.
- [Duo93] H.N. Duong. *Identification Structurel et Parametrique des Systèmes Linéaire Monovariabiles et Multivariabiles*. PhD thesis, Institut Nationale Polytechnique de Grenoble, 1993.
- [Fav99] W. Favoreel. *Subspace Identification and Control of Linear and Bilinear Systems*. PhD thesis, Katholieke Universiteit Leuven - Faculteit Toegepaste Wetenschappen Arenbergkasteel, B-3001 Heverlee, Belgium, 1999.
- [IKH03] I.D.Landau, A. Karimi, and H. Hjalmarsson, editors. *International Workshop on Design and optimisation of restricted complexity controllers*. Laboratoire d'Automatique de Grenoble, January 2003.
- [LAL90] Y. Liu, B.D.O. Anderson, and U. Ly. Coprime factorization controller reduction with bezout identity induced frequency weighting. *Automatica*, 26(2):233–249, 1990.

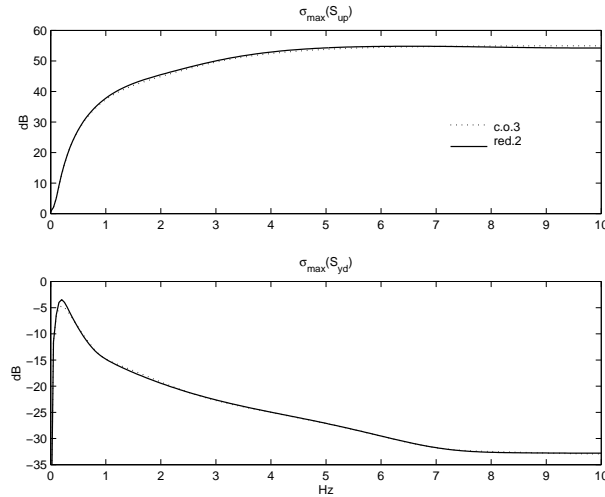


Figure 7: S_{up} and S_{yd} largest singular values for original controller "c.o.3" (dotted) and for the reduced order controller "red.2" (solid).

- [Lan78] A. Gauthier I.D. Landau. On the recursive identification of multi-input multi-output system. *Automatica*, 18(5):609–614, 1978.
- [Lan02] I.D. Landau. *Commande des systèmes*. Hermes, Paris, 2002.
- [LK97] I.D. Landau and A. Karimi. Recursive algorithms for identification in closed loop: a unified approach and evaluation. *Automatica*, 33(8):1499–1523, 1997.
- [LK98] I.D. Landau and A. Karimi. Robust digital control using pole placement with sensitivity function shaping method. *Int. J. Robust and Nonlin. Cont.*, 8:191–210, 1998.
- [LKC01] I.D. Landau, A. Karimi, and A. Constantinescu. Direct controller order reduction by identification in closed loop. *Automatica*, 37(11):1689–1702, 2001.
- [LKMP03] I.D. Landau, A. Karimi, L. Miskovic, and H. Prochazka. Control of an active suspension system as a benchmark for design and optimisation of restricted complexity controllers. *European Journal of Control*, 9(1), 2003.
- [Mac89] J.M. Maciejowski. *Multivariable Feedback Design*. Addison-Wesley, 1989.
- [PL03a] H. Prochazka and I.D. Landau. Convex optimisation for robust multivariable controller design as alternative of h_{∞} optimisation. *submitted to European Journal of Control*, 2003.

- [PL03b] H. Prochazka and I.D. Landau. Pole placement with sensitivity function shaping using 2nd order digital notch filters. *Automatica*, 39(6):1103–1107, 2003.
- [Pro03] H. Prochazka. Comments on the paper "controller order reduction by identification in closed-loop applied to a benchmark problem". *European Journal of Control*, 9(5), 2003.
- [vOM96] P. van Overschee and B. De Moor. *Subspace identification for linear systems: Theory, implementation, applications*. Kluwer Academic Publishers, 1996.
- [WSL02] G. Wang, V. Sreeram, and W.Q. Liu. Balanced performance preserving controller reduction. *Systems and Control Letters*, 46:99–110, 2002.

User's Guide for Pole Placement Master

MATLAB[®] graphical toolbox for robust digital controller design

Hynek Procházka

October 1, 2003

Contents

1	About Pole Placement Master	2
2	Basic theory	3
2.1	Notation	3
2.2	Controller design procedure	5
3	How to use the application	7
3.1	Software requirements	7
3.2	Getting started	7
3.3	Description of the toolbox	8
3.3.1	Principal closed-loop controller design window	8
3.3.2	Tracking part design window	17
4	Bibliographical indications	19

Chapter 1

About Pole Placement Master

The Pole Placement Master (briefly PPMaster) is a toolbox programmed in MATLAB[®] 5.3 environment¹ (it works also under MATLAB 6.0 and higher). It is user friendly application with graphical user interface (GUI) for design of digital SISO (single-input, single-output) robust controller. The implemented controller design procedure is combined pole placement with sensitivity shaping design technique presented in [2] and improved in [6].

The toolbox was developed in Laboratoire d'Automatique de Grenoble as a part of doctoral thesis "H. Prochazka, Synthèse de régulateurs multivariables en utilisant le placement de pôles avec calibrage des fonctions de sensibilité par optimisation convexe, Institut National Polytechnique de Grenoble". The principal reference for the toolbox is [5].

¹MATLAB is a registered trademark of The MathWorks, Inc.

Chapter 2

Basic theory

2.1 Notation

A standard digital pole placement configuration using a polynomial controller (denoted R-S) is shown in Fig.2.1. The plant model $G(z^{-1})$ is of the form:

$$G(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})} = \frac{z^{-d}(b_1 z^{-1} + \dots + b_{n_B} z^{-n_B})}{1 + a_1 z^{-1} + \dots + a_{n_A} z^{-n_A}} \quad (2.1)$$

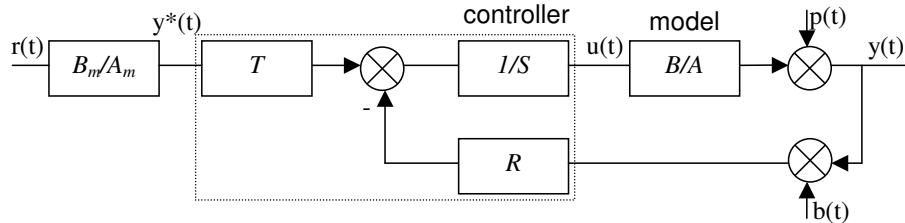


Figure 2.1: Considered closed loop configuration

The R-S part of the controller has the transfer function:

$$\frac{R(z^{-1})}{S(z^{-1})} = \frac{R_0(z^{-1})H_R(z^{-1})}{S_0(z^{-1})H_S(z^{-1})} \quad (2.2)$$

where $H_R(z^{-1})$ and $H_S(z^{-1})$ denote the fixed parts of the controller (either imposed by the design or introduced in order to shape the sensitivity functions). $R_0(z^{-1})$ and $S_0(z^{-1})$ are solutions of the Bezout equation (poles of the closed loop):

$$A(z^{-1})S_0(z^{-1})H_S(z^{-1}) + B(z^{-1})R_0(z^{-1})H_R(z^{-1}) = \underbrace{P_D(z^{-1})P_F(z^{-1})}_{=P(z^{-1})} \quad (2.3)$$

where $P(z^{-1})$ represents the desired closed loop poles, $P_D(z^{-1})$ defines the dominant poles (specified) and $P_F(z^{-1})$ defines the auxiliary poles (which in part can be specified by design specifications and the remaining part is introduced in order to shape the sensitivity function).

Sensitivity function shaping is one of the way how to assure controller and closed-loop performances, since the sensitivity functions are one of the crucial indicator of these performances. The considered sensitivity functions are:

The output sensitivity function:

$$S_{yp}(z^{-1}) = \frac{A(z^{-1})S_0(z^{-1})H_S(z^{-1})}{P(z^{-1})} \quad (2.4)$$

The input sensitivity function:

$$S_{up}(z^{-1}) = \frac{-A(z^{-1})R_0(z^{-1})H_R(z^{-1})}{P(z^{-1})} \quad (2.5)$$

The complementary sensitivity function:

$$S_{yb}(z^{-1}) = \frac{B(z^{-1})R_0(z^{-1})H_R(z^{-1})}{P(z^{-1})} \quad (2.6)$$

where S_{yp} is shaped to obtain a sufficient closed-loop robust stability, the shaping of S_{up} allows to limit controller gain and hence actuator effort and S_{yb} shaping help to limit noise sensitivity of the closed loop and it serves to fix a desired closed-loop tracking performance. More details can be found in [1, 2].

We can now introduce the following parameterization:

$$\begin{aligned} H_R(z^{-1}) &= H'_R(z^{-1})\gamma_R(z^{-1}) \\ H_S(z^{-1}) &= H'_S(z^{-1})\gamma_S(z^{-1}) \\ P_F(z^{-1}) &= P'_F(z^{-1})\delta_R(z^{-1}) \\ P_F(z^{-1}) &= P''_F(z^{-1})\delta_S(z^{-1}) \end{aligned}$$

With these notations we get:

$$|S_{yp}(z^{-1})| = \left| \frac{A(z^{-1})S_0(z^{-1})H'_S(z^{-1})}{P_D P''_F(z^{-1})} \right| \left| \frac{\gamma_S(z^{-1})}{\delta_S(z^{-1})} \right| \quad (2.7)$$

$$|S_{up}(z^{-1})| = \left| \frac{A(z^{-1})R_0(z^{-1})H'_R(z^{-1})}{P_D P'_F(z^{-1})} \right| \left| \frac{\gamma_R(z^{-1})}{\delta_R(z^{-1})} \right| \quad (2.8)$$

where the filters $F_{yp}(z^{-1}) = \frac{\gamma_S(z^{-1})}{\delta_S(z^{-1})}$ and $F_{up}(z^{-1}) = \frac{\gamma_R(z^{-1})}{\delta_R(z^{-1})}$ consist of several 2nd order notch filters (2zeros/2poles band-stop filters with limited attenuation) simultaneously tuned. The tuning means in fact searching for appropriate frequency characteristics of $|F_{yp}(z^{-1})|$ and $|F_{up}(z^{-1})|$. Specifically in our case, we are interested in frequency band-stop with limited attenuation characteristics and thus the tuning concerns the frequency of band-stop, its bandwidth and the maximum attenuation in the band-stop frequency.

The tracking part $T(z^{-1})$ of the controller is used to compensate the closed loop dynamic in such way that the entire system transfer function (from $r(t)$ to $y(t)$) has the dynamic of the reference model $\frac{B_m}{A_m}$. The polynomial $T(z^{-1})$ is considered to have three basic forms:

- T contains all closed-loop poles given by the polynomial $P = AS + BR$ and its static gain is adjusted so the static gain of the transfer function from $y * (t)$ to $y(t)$ is 1. Hence,

$$T(z^{-1}) = \frac{P(z^{-1})}{B(1)} \quad (2.9)$$

- T contains dominant closed-loop poles given by the polynomial P_D and its static gain is adjusted so the static gain of the transfer function from $y * (t)$ to $y(t)$ is 1. Hence,

$$T(z^{-1}) = \frac{P_D(z^{-1})P_F(1)}{B(1)} \quad (2.10)$$

- T is a gain with the value $T = \frac{P(1)}{B(1)}$.

The reference model $\frac{B_m}{A_m}$ is considered to be either 2nd order transfer function with dynamics defined by natural frequency and damping, or two same 2nd order transfer functions connected in cascade.

2.2 Controller design procedure

Suppose to dispose with a digital model G of the plant to be controlled. The controller design consists of the following steps:

1. *Design specifications* - Determine desired closed loop and tracking performances. The closed loop performances, such as robust stability, disturbance rejection, etc., has to be expressed by some templates imposed on sensitivity functions. The tracking properties include rise time, maximal overshoot, or settling time.

2. *Closed-loop part R-S design* - The sensitivity functions are shaped to satisfy design specifications (to enter the frequency responses to imposed templates). As we can see from the previous section, one disposes with the following design parameters:

- P_D polynomial of desired dominant (the slowest) closed loop poles
- P'_F/P''_F polynomial of desired auxiliary closed loop poles
- H_R fixed part of the controller numerator
- H_S fixed part of controller denominator
- F_{yp} 2nd order digital notch filters on S_{yp}
- F_{up} 2nd order digital notch filters on S_{up}

which allow us to shape appropriately the sensitivity functions S_{yp} , S_{up} , S_{yb} .

3. *Tracking part design* - If the tracking properties are not satisfied by closed loop controller part *R-S*, the tracking part has to be designed. One has to choose an appropriate structure of T and to design the reference model $\frac{B_m}{A_m}$ corresponding to the desired tracking performances. For reference model adjusting the natural frequency and damping of the reference model denominator is modified.

Chapter 3

How to use the application

3.1 Software requirements

The PPMaster is MATLAB toolbox, and so it can work only in the MATLAB environment. Specifically, the toolbox was programmed for MATLAB version 5.3, but it works also under version 6.0 or version 6.5.

Moreover, the PPMaster requires the standard MATLAB toolboxes and Signal Processing toolbox (it uses only the function `freqz` for discrete-time frequency responses).

3.2 Getting started

1. To run Pole-Placement Master (`ppmaster`):
 - Run MATLAB version 5.3 or higher.
 - Change the MATLAB current directory on the deirectory with m-files / p-files of `ppmaster` (using command `cd` or Path Browser) or add this directory to the `PATH` variable (for example using `path-browser`).
 - Type "`ppmaster`" to MATLAB command line
2. To load a model and start design you go to menu bar:
Model→Load model→Matlab format / WinPim format
3. To load a whole design workspace (two examples are in the directory "Examples") go to menu bar:
Model→Current Parameter Set →Load Parameters and model

4. To save a whole design workspace go to menu bar:
Model→Current Parameter Set→Save Parameters and model

For any problems contact: landau@lag.ensieg.inpg.fr.

3.3 Description of the toolbox

3.3.1 Principal closed-loop controller design window

Launching the command "ppmaster", the principle window for closed-loop controller (R/S) design will be opened, see Fig.3.1.

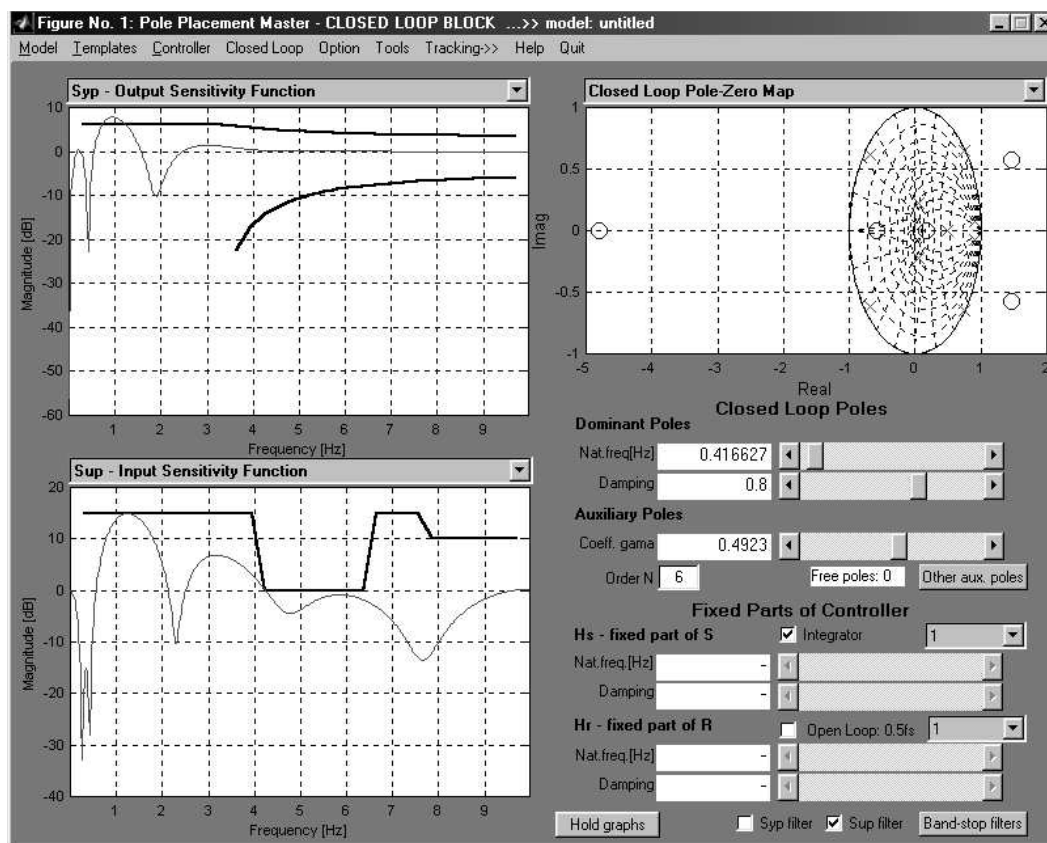


Figure 3.1: Principal window for closed-loop design

Menu bar

The menu bar of the principal window has following options (see Fig.3.1 at the top):

- *Model* - allows to load, save, edit and make analysis of the current digital plant model. The sub-menu contains:

- Load model - for loading the digital plant model saved either in WinPim (extension .mod), or in MATLAB format .mat (variable A is a vector of denominator coefficients, B is a vector of numerator coefficients, Ts is sampling time).

Example: Consider to have a digital plant model as follows:

$$G(z^{-1}) = \frac{z^{-1}(b_1z^{-1} + b_2z^{-2})}{1 + a_1z^{-1} + a_2z^{-2} + a_3z^{-3}} = \frac{B(z^{-1})}{A(z^{-1})} \quad (3.1)$$

with a sampling time $T_s = 0.05s$. The corresponding MATLAB variables will have the following format: $A = [1 \ a_1 \ a_2 \ a_3]$; $B = [0 \ 0 \ b_1 \ b_2]$; $Ts = 0.05$;

- Save model - allows to save actual model either in WinPim, or in MATLAB format.
 - Edit model - opens a small dialog for editing the coefficients of the model, the pure delay and the sampling time.
 - Sampling time - allows to modify actual sampling time.
 - Analysis of model - open sub-window for transfer-function analysis. For details on this window see bellow.
 - Current Parameter Set - allows to save and load whole actual workspace of PPMaster. The loading is possible, either with model, or without model. **This is the most efficient way how to store actual design stage!**
- *Templates* - allows to load, save and edit templates for S_{yp} and S_{up} sensitivity functions. The option "Edit templates" opens a sub-window where some basic shapes can be created. For loading/saving the extension .tpl is considered, but the format of the file is standard MATLAB .mat format. The file with templates contains vectors: $Wsyp$ - vector of S_{yp} upper template magnitude points in linear scale, $Wsyp2$ - vector of S_{yp} lower template magnitude points in linear scale, $Wsup$ - vector of S_{up} upper template magnitude points in linear scale, fn - vector of normalized frequency points for all templates (the values are from 0 to 0.5).
 - *Controller* - This option has following sub-options:

- Save controller - for saving the final controller either in WinReg format or in MATLABfig:ppmwin version 5/version 4 .mat format. The saved variables are vectors R, S, T, Am, Bm of corresponding polynomial coefficients in increasing power of z^{-k} (see Load model example).
 - Display controller - opens transfer-function analysis sub-window for analysis of controller transfer function $R(z^{-1})/S(z^{-1})$. For details on this sub-window see below.
 - Stability Window - opens small sub-window displaying controller pole locations and indicating the controller stability. This sub-window is refreshed after every design parameter modification and it can be opened throughout entire controller design.
- *Closed loop* - opens transfer-function analysis sub-window for analysis of closed-loop transfer function $B(z^{-1})R(z^{-1})/P(z^{-1})$. For details on this sub-window see below.
 - *Option* - Two sub-options allows to modify application colors and axes scales.
 - *Tools* - contains some useful tools, these are:
 - Point Cursor - activating this tool (it is activated by default), when the left mouse-button is pressed and the mouse is positioned on some graph, the mouse actual position (X-Y scale values) on the graph is displayed.
 - Zoom - activates zoom capability of the mouse.
 - Figure - opens a dialog allowing to select which graphs will be plotted, when the selection is done, an independent figure is plotted with selected graphs.
 - *Tracking*– \gg - closes the closed-loop design principal window and opens the tracking part design principal window. In this window the tracking part T of the controller can be designed. For details see below.
 - *Help* - The sub-option "Theory" opens a sub-window with theoretical background concerning the controller design. The sub-option "About application" gives some general information about the toolbox.
 - *Quit* - close the application.

Graphs

The principal window contains three graphs (in left and right upper corners and in left lower corner) allowing to display the principal closed-loop properties, these are:

- $|S_{yp}(z^{-1})|$ output sensitivity modulus with corresponding imposed upper and lower template.
- $|S_{up}(z^{-1})|$ input sensitivity modulus with corresponding imposed upper template.
- $|S_{yb}(z^{-1})|$ complementary sensitivity modulus with corresponding imposed upper template.
- $P(z^{-1})$ closed-loop poles.
- $S(z^{-1})$ controller poles.
- step disturbance rejection (output signal evolution, when a step is added to $p(t)$).

The property, which is displayed on a graph, can be chosen using popup selector situated above the graph.

Design parameter adjusting tools

Adjusting slider/button object - The right lower part of the window contains several sliders and buttons making accessible all the design parameters. Every numerical parameter is adjustable either by slider, or by a button displaying current parameter value. Pressing the button, a small edit-window allows to modify from the keyboard the actual value.

Description of tools - The right lower part consists of several sections, which have following significance:

- Section "Dominant Poles" - contains two sliders/buttons to modify natural frequency (Nat.freq[Hz]) and damping (Damping) of the dominant closed-loop pole pair P_D .
- Section "Auxiliary Poles" - contains one slider/button allowing to set desired value of one real multiple pole. The pole's multiplicity is displayed below and it can be modified using '+/-' buttons. Aside, the value denoted "Free poles" indicates how many poles can be assigned

yet. Finally, the button "Other aux. poles" opens sub-window which allows to assign more complicated auxiliary pole structures (complex pairs, other real poles). For sub-window description see below.

- Section "Hs - fixed part of S" - contains one check-box allowing to introduce one integrator as a part of the fixed part H_S . Next, in pop-up selection object, one can choose the form of the rest of the fixed part. The possibilities include:
 - 1 - no fixed parts except eventual integrator.
 - 1st order - one real pole. Its position on the real axe is adjustable form the lower slider/button denoted for this case "Coeff".
 - 2nd order - complex pair of poles with adjustable natural frequency and damping using two slider/buttons denoted "Nat.freq.[Hz]" and "Damping" respectively.
- Section "Hr - fixed part of R" - contains one check-box allowing to introduce one zero at -1 (opening of the loop at $1/2T_s$ Hz) as a part of the fixed part H_R . Next, in pop-up selection object, one can choose the form of the rest of the fixed part. The possibilities include:
 - 1 - no fixed parts except eventual loop opening.
 - 1st order - one real zero. Its position on the real axe is adjustable form the lower slider/button denoted for this case "Coeff".
 - 2nd order - complex pair of poles with adjustable natural frequency and damping using two slider/buttons denoted "Nat.freq.[Hz]" and "Damping" respectively.
- Section at the bottom - contains two buttons: "Hold graphs" and "Band-stop filters" and two check-boxes: "Syp filter" and "Sup filter".
 - "Hold graph" button allows to freeze actual cures, they are displayed in grey and dashed. The freezed curves are constantly displayed together with new actual curves. Once the "Hold graphs" button pressed it changed to "Release graphs" button allowing to erase the freezed curves.
 - "Band-stop filters" button opens a new sub-window for design of 2nd order digital notch filters F_{yp} , F_{up} . For detailed description of this sub-window see bellow.
 - The check-boxes "Syp filter" and "Sup filter" signal, if the designed sets of 2nd order notch filters (band-stop filters) F_{yp} , F_{up} will be used for controller design or not.

Transfer-function analysis sub-window

The transfer-function analysis sub-window (see Fig.3.2) is opened when plant model, controller or closed-loop transfer-function analysis is demanded (accessible from menu bar). The following graphical objects are displayed in this window:

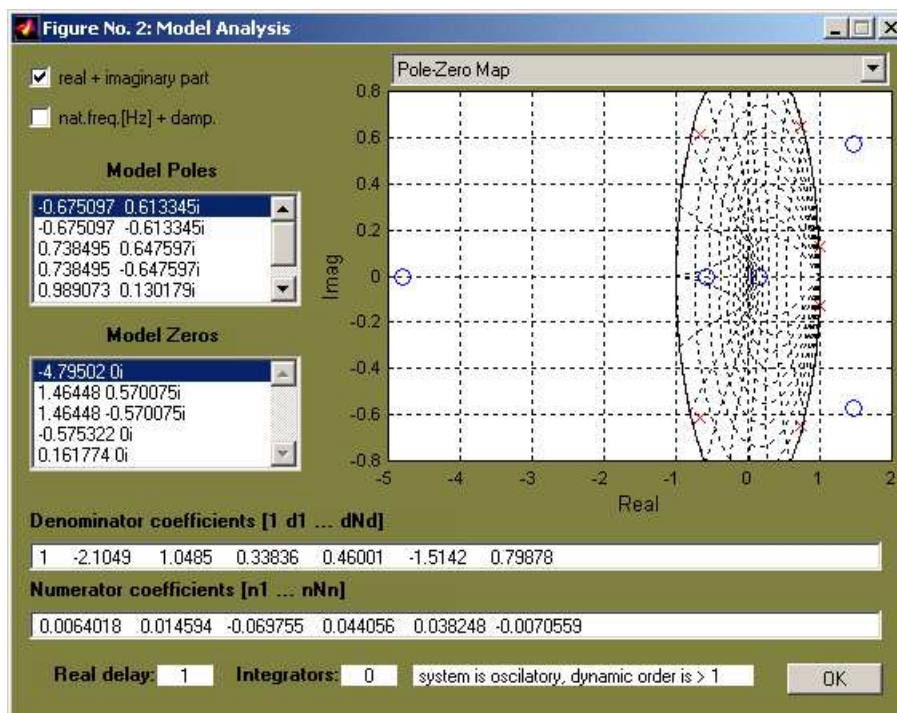


Figure 3.2: Transfer function analysis window

- *Graph* - allows to display pole-zero map, frequency response amplitude and phase. The choice of what is displayed is made by popup selector placed above the graph.
- *Check-boxes* - "real + imaginary part" and "nat.freq.[Hz] + damp." switch the display format for poles and zeros listed below these check-boxes. Either the real and the imaginary parts of the poles/zeros, or the natural frequencies and dampings are displayed.
- *Lists* - as mentioned, they list the transfer function poles and zeros in one of the possible format.
- *Edit lines* - contain transfer function denominator and numerator coefficients.

- *Real delay / Integrators* - give real delay of the transfer function (the delay in number of samples including zero-order holder delay) and the number of integrators. aside these numbers, some comment on the system dynamic is displayed.

Auxiliary closed-loop poles sub-window

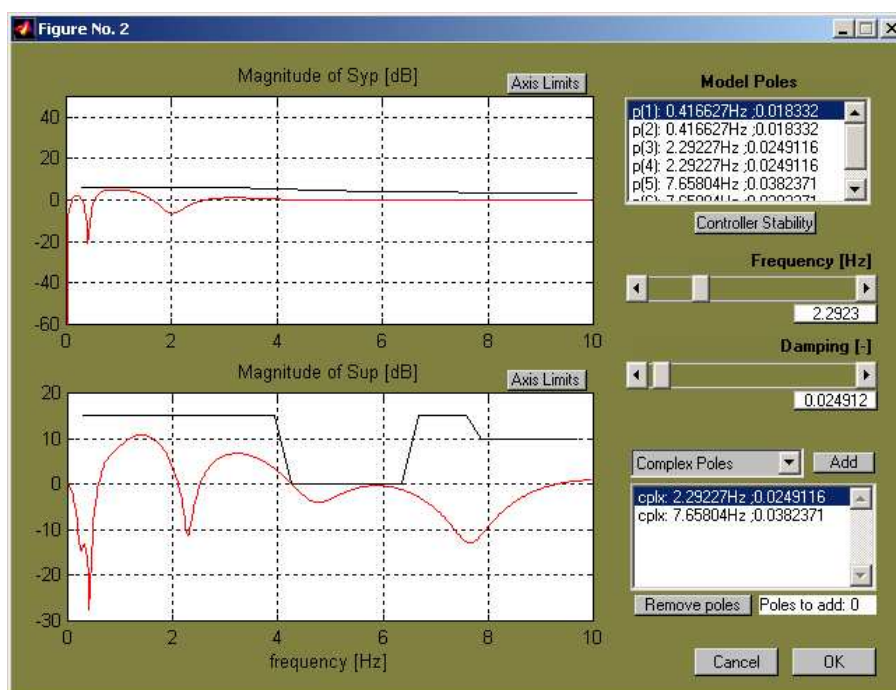


Figure 3.3: Auxiliary closed-loop poles selection window

This window is used to set any desired auxiliary closed-loop pole's structure, thus it modifies the design parameter P'_F/P''_F . The window contains following graphical objects:

- *Graphs* - two graphs displays S_{yp} and S_{up} sensitivity function characteristics for observing the effects of the placed auxiliary poles onto the closed loop. The buttons "Axis Limits" above every graph serve for adjusting the scales limits. They open small dialog displaying the actual limits. To modify the limits, one has to change the displayed values and to close the dialog.
- *Model Poles List* - the listing object in the upper right corner list all model's poles in the format of natural frequency and damping. The

cursor in this list marks one pole. When a new pole is added to auxiliary poles in this window, it has by default the same position as the pole marked in model poles, but only if the same kind of poles (complex or real pole) are marked and added.

- *List of auxiliary poles* - The list of all desired auxiliary poles (except the real multiple pole adjustable from the principal window) is at the bottom on the right. On the contrary to the model pole list, a complex pair of poles is denoted only by one item named "cplx:" with the corresponding natural frequency and damping. Also a multiple real pole is denoted only by one item named "real:" and with pole's position followed with pole's multiplicity.
- *Add button and popup* - To add a pole one has to use "Add" button. The popup selector on the left of this button allows to choose what type of pole will be added (real multiple pole or complex pair).
- *Remove poles button* - This button remove auxiliary pole/s marked by the cursor in the auxiliary pole list.
- *Poles to add value* - this value signify how many poles can be assigned yet (number of free non-assigned poles).
- *Controller Stability button* - the same as in the bar menu option - "Controller→Stability Window".
- *Slider/buttons* - The sliders and buttons with the actual values allow us to modify the properties of the auxiliary pole/s marked by cursor in auxiliary pole list. If a complex pair of poles is marked, one can change natural frequency and damping, if a real multiple pole is marked, the pole's position on real axe and its multiplicity can be modified.

Band-stop filter design sub-window

The band-stop filter window serve for design of 2nd order notch filters F_{yp} and F_{up} of pole-placement shaping. The number of filter is not limited, by one filter rise the controller complexity by 2. The graphical objects in the window has following utility:

- *Graphs* - two large graphs on the left displays S_{yp} and S_{up} sensitivity function characteristics for observing the effects of the filters onto the closed loop. Any added filter is marked by one dashed line describing his actual position. The buttons "Axis Limits" above every graph serve

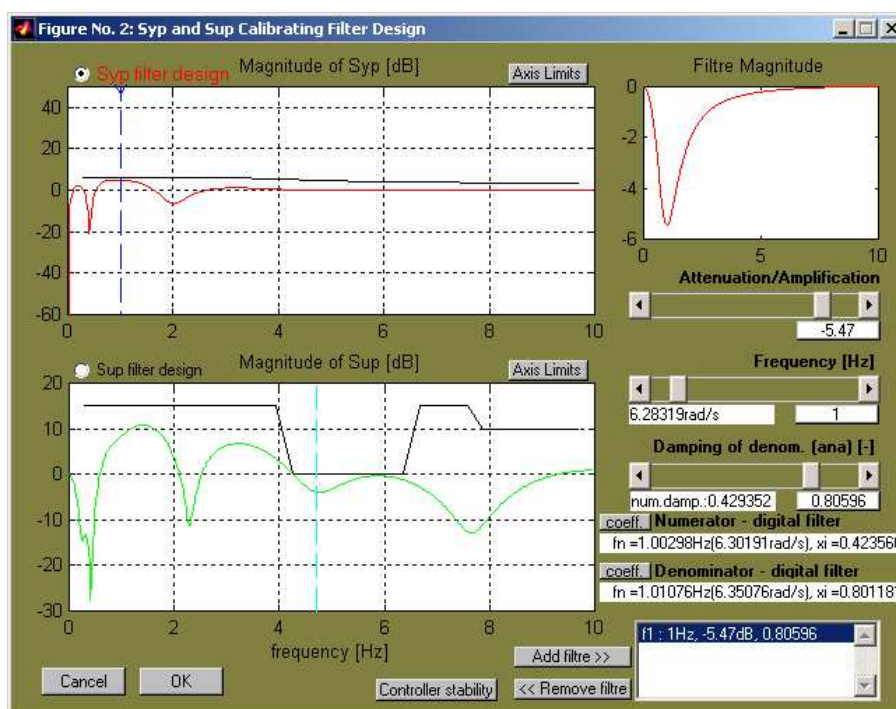


Figure 3.4: Band-stop filter design window

for adjusting the scales limits. They open small dialog displaying the actual limits. To modify the limits, one has to change the displayed values and to close the dialog.

One small graph on the right display frequency response of actually modified 2nd order notch (band-stop) filter.

- *Radio buttons "Syp filter design", "Sup filter desig"* - they determine what filters are designed, if F_{yp} for S_{yp} sensitivity function or F_{up} for S_{up} sensitivity function.
- *Slider/buttons* - The sliders and the buttons displaying actual value of each slider allow to modify the three design properties of actual notch filter. The properties are :
 - M_t - band-stop attenuation/amplification.
 - f_t - band-stop frequency.
 - ζ_d - analog equivalent denominator damping (adjust band-stop width).

- *Numerator and Denominator - digital filter* - The white text lines entitled "Numerator - digital filter" and "Denominator - digital filter" display respectively the natural frequency (in Hertz and in rad/s) and damping of digital filter numerator and denominator.
- *"coeff" button* - It opens a small dialog with one edit line containing the coefficients of the actual digital filter numerator/denominator.
- *List of filters* - The listing object at the bottom lists all applied filters F_{yp} / F_{up} . The cursor marks the actually modified filter. This filter is also marged in the corresponding graph by two triangles.
- *Add/Remove filter button* - The button "Add filter" adds one filter F_{yp} or F_{up} to the list. The button "Remove filter" removes the filter marked by the cursor in the filter list.
- *Controller Stability button* - the same as in the bar menu option - "Controller→Stability Window".

3.3.2 Tracking part design window

The tracking part design window enable to design the tracking part T of the controller and the tracking model B_m/A_m . These graphical objects are used:

- *Graphs* - The three graphs can display tracking (time domain) properties of the closed loop system. These properties are accessible:
 - System output $y(t)$ step response.
 - Desired trajectory $y * (t)$ step response.
 - Command $u(t)$ step response.
- *Upper popup selector* - It is used for choosing either one or two tracking models connected in cascade.
- *Lower popup selector* - This popup determine the form of the tracking part T . There are three possibilities:
 - "All CL poles - Pd.Pf/B*(1)" - $T(z^{-1})$ is a polynomial with adjusted static gain containing all closed loop poles.
 - "dominant CL poles - Pd.Pf(1)/B*(1) - $T(z^{-1})$ is a polynomial with adjusted static gain containing one complex pair of roots corresponding to closed loop dominant poles defined by P_D .

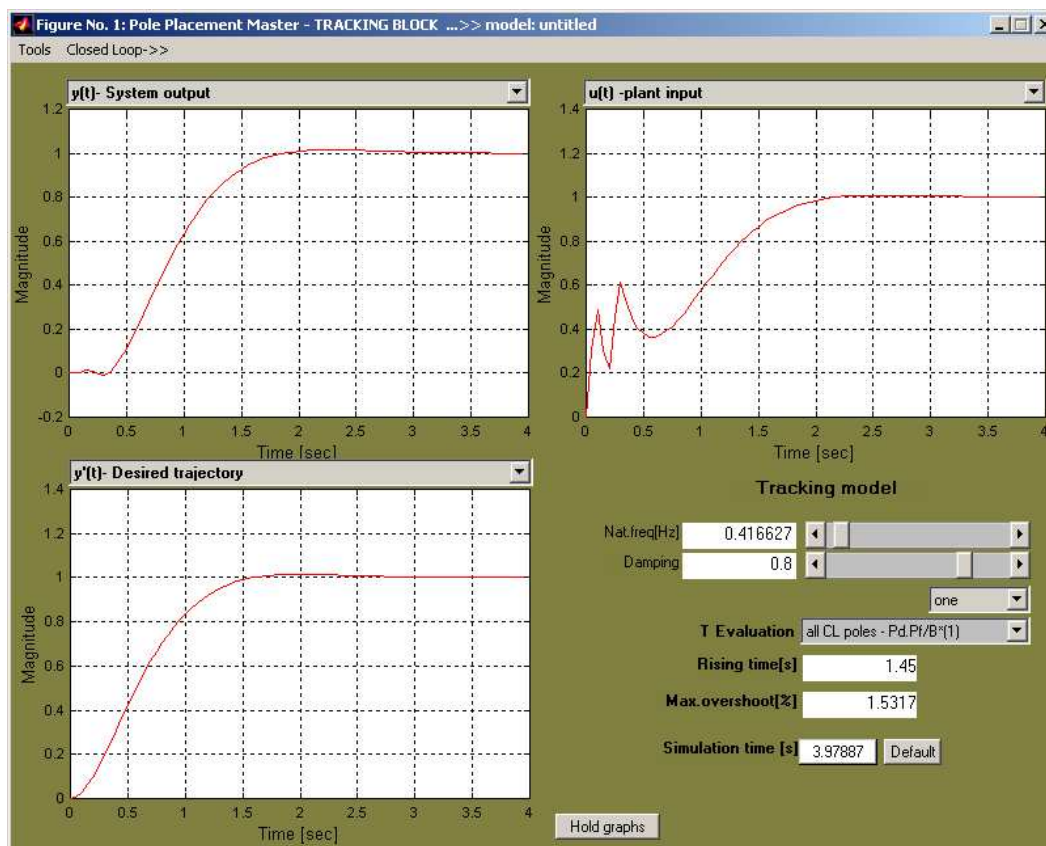


Figure 3.5: Tracking part design window

- "Gain - $P(1)/B(1)$ " - T is a constant with an appropriate value setting static gain of the whole system equal to one.
- *Sliders* - the two sliders modify actual tracking model properties, specifically the natural frequency and the damping. The current values are displayed on the left.
- *Rising time value* - It gives the system rising time (step response time to reach 90% of the reference final value) in seconds.
- *Max.overshoot value* - It represents the overshoot over the final reference value of step response in %.
- *Simulation time button* - On this button, the actual simulation time of displayed responses is showed. Pressing the button, the simulation time can be modified. The "Default" button compute a default sampling time according to system dynamics.

Chapter 4

Bibliographical indications

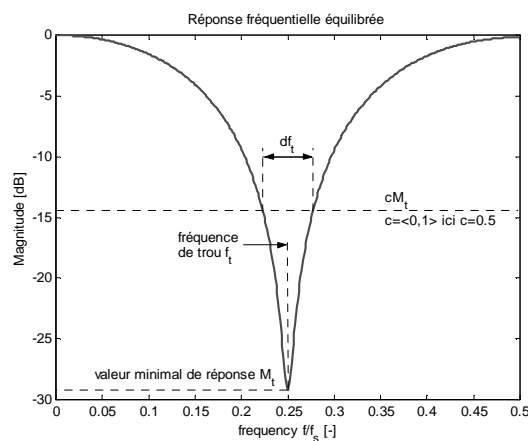
An integral description of the theory related to the pole placement with sensitivity shaping can be found in [1]. The method was introduced in [2] and improved in [6]. The software toolbox "PPMaster" was presented in [5]. Some practical examples of robust controller design can be found in [3, 6, 4, 1].

Bibliography

- [1] I.D. Landau. *Commande des systèmes*. Hermes, Paris, 2002.
- [2] I.D. Landau and A. Karimi. Robust digital control using pole placement with sensitivity function shaping method. *Int. J. Robust and Nonlin. Cont.*, 8:191–210, 1998.
- [3] I.D. Landau, J. Langer, D. Rey, and J. Barnier. Robust control of 360 flexible arm using the combined pole placement/sensitivity function shaping method. *IEEE Trans. Control Systems Technol.*, pages 369–383, 1996.
- [4] H. Prochazka. Comments on the paper ”controller order reduction by identification in closed-loop applied to a benchmark”. *accepted to European Journal of Control*, 2003.
- [5] H. Prochazka and I.D. Landau. Logiciel pour l’enseignement et le calcul du placement de pôles robuste. *Conf. Int. Francophone d’Automatique Nante*, pages 694–698, 7 2002.
- [6] H. Prochazka and I.D. Landau. Pole placement with sensitivity function shaping using 2nd order digital notch filters. *Automatica*, 39(6):1103–1107, 2003.

Synthèse du filtre discret bande étroite 2 pôles / 2 zéros – application au calibrage des fonctions de sensibilité

Rapporte



1. Filtre 2pôles/2zéros – théorie

1.1. Filtre analogique

Le filtre analogique 2pôles/2zéros a cette forme générale :

$$F(s) = \frac{s^2 + 2\xi_{num} \omega_{num} s + \omega_{num}^2}{s^2 + 2\xi_{den} \omega_{den} s + \omega_{den}^2} \quad (1)$$

Pour le cas où $\omega_{num} = \omega_{den} = \omega_0$ on a la réponse fréquentielle de $|F(j\omega)|$ avec un extremum qui est placé à la fréquence ω_0 (voir figure). La valeur de l'extremum peut être calculé à partir de l'équation (1) en mettant $s = j\omega$ et $\omega_{num} = \omega_{den} = \omega_0$. On a donc :

$$F(j\omega) = \frac{(j\omega)^2 + 2\xi_{num} \omega_{num} j\omega + \omega_{num}^2}{(j\omega)^2 + 2\xi_{den} \omega_{den} j\omega + \omega_{den}^2} \quad (2)$$

où

$$|F(j\omega)| = \left| \frac{(j\omega)^2 + 2\xi_{num} \omega_{num} j\omega + \omega_{num}^2}{(j\omega)^2 + 2\xi_{den} \omega_{den} j\omega + \omega_{den}^2} \right| = \frac{\sqrt{(\omega_{num}^2 - \omega^2)^2 + (2\xi_{num} \omega_{num} \omega)^2}}{\sqrt{(\omega_{den}^2 - \omega^2)^2 + (2\xi_{den} \omega_{den} \omega)^2}}$$

et si $\omega_{num} = \omega_{den} = \omega_0$ on va obtenir :

$$|F(j\omega_{nat})| = \frac{2\xi_{num} \omega_0^2}{2\xi_{den} \omega_0^2} = \frac{\xi_{num}}{\xi_{den}} \quad (3)$$

ce qui est la valeur de l'extremum de la réponse fréquentielle du filtre. A noter, qu'on a possibilité de créer soit maximum, soit minimum.

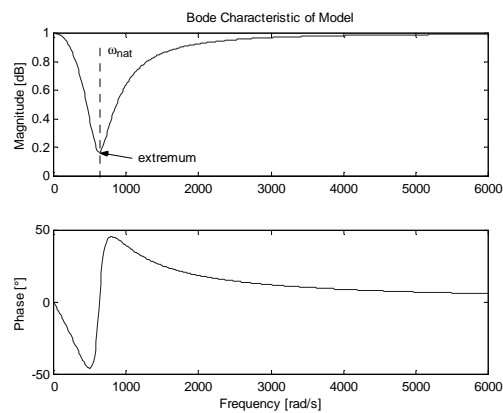


Figure 1 : La réponse du filtre analogique 2pôles/2zéros avec la même fréquence naturelle dans numérateur et dénominateur

1.2. Transformation bilinéaires

Le filtre analogique calculé peut être converti à un filtre discret par la transformation bilinéaire. Cette transformation garde les propriétés du filtre dans le domaine fréquentiel ce qui nous convient. Mais il faut faire attention sur le « wrapping » des fréquences. Or, la transformation convertit les fréquences comme suite : $\omega \in \langle 0, \infty \rangle \rightarrow \omega_{disc} \in \langle 0, \pi \rangle$.

La transformation est défini par l'équation suivante :

$$s = \frac{2}{T_e} \frac{1 - z^{-1}}{1 + z^{-1}} \quad (4)$$

où T_e est la période d'échantillonnage du filtre discret.

A partir de l'équation (4) et des relations $z^{-1} = e^{-j\omega_{disc}}$ et $s = j\omega$ on peut déterminer l'équation entre la fréquence analogique ω la fréquence discret ω_{disc} . Il s'agit des équations suivantes :

$$\omega = \frac{2}{T_e} \tan\left(\frac{\omega_{disc}}{2}\right) \quad (5)$$

$$\omega_{disc} = 2 \arctan\left(\frac{\omega T_e}{2}\right) \quad (6)$$

Quand on applique la transformation (4) dans l'équation (1), après des certaines opérations mathématiques élémentaires, on obtient le filtre discret de la forme suivantes :

$$F(z) = \frac{b_{z0} + b_{z1}z^{-1} + b_{z2}z^{-2}}{a_{z0} + a_{z1}z^{-1} + a_{z2}z^{-2}} = \gamma \frac{1 + \beta_1 z^{-1} + \beta_2 z^{-2}}{1 + \alpha_1 z^{-1} + \alpha_2 z^{-2}} \quad (7)$$

où les coefficients ont la forme suivant :

$$\begin{aligned} b_{z0} &= \frac{4}{T_e^2} + 4 \frac{\xi_{num} \omega_0}{T_e} + \omega_0^2 \\ b_{z1} &= 2\omega_0^2 - \frac{8}{T_e^2} \\ b_{z2} &= \frac{4}{T_e^2} - 4 \frac{\xi_{num} \omega_0}{T_e} + \omega_0^2 \\ a_{z0} &= \frac{4}{T_e^2} + \frac{4\xi_{den} \omega_0}{T_e} + \omega_0^2 \\ a_{z1} &= 2\omega_0^2 - \frac{8}{T_e^2} \\ a_{z2} &= \frac{4}{T_e^2} - \frac{4\xi_{den} \omega_0}{T_e} + \omega_0^2 \end{aligned} \quad (8)$$

$$\begin{aligned}
\gamma &= \frac{b_{z0}}{a_{z0}} \\
\beta_1 &= \frac{b_{z1}}{b_{z0}}; \beta_2 = \frac{b_{z2}}{b_{z0}} \\
\alpha_1 &= \frac{a_{z1}}{a_{z0}}; \alpha_2 = \frac{a_{z2}}{a_{z0}}
\end{aligned} \tag{9}$$

Maintenant il suffit de exprimer le numérateur et le dénominateur en la fréquence naturelle ω_0 et l'amortissement ξ . Pour le faire on va d'abord calculer les racines du numérateur et du dénominateur. Les expressions sont les suivants :

$$\begin{aligned}
z_{n1,2} &= \frac{-\beta_1 \pm j\sqrt{4\beta_2 - \beta_1^2}}{2} = A_n e^{j\varphi_n} \\
z_{d1,2} &= \frac{-\alpha_1 \pm j\sqrt{4\alpha_2 - \alpha_1^2}}{2} = A_d e^{j\varphi_d}
\end{aligned} \tag{10}$$

Puis à partir du tableau de conversion entre la forme continue (fréquence naturelle + amortissement) et la forme discret (fonction de transfert), on peut déduire pour les racines des polynômes discret la relation suivante :

$$z_{1,2} = e^{-\xi_{disc}\omega_{0disc}T_e} e^{\pm j\omega_{0disc}\sqrt{1-\xi_{disc}^2}T_e} \tag{11}$$

Quand on fait égal la valeur absolu et l'angle, les équations qui nous donnent les valeurs de la fréquence naturelle discret et de l'amortissement discret sont les suivantes :

$$\begin{aligned}
\omega_{0num} &= \sqrt{\frac{\varphi_n^2 + \ln^2 A_n}{T_e^2}} \\
\xi_{numd} &= -\frac{\ln A_n}{\omega_{0num}T_e} \\
\omega_{0den} &= \sqrt{\frac{\varphi_d^2 + \ln^2 A_d}{T_e^2}} \\
\xi_{dend} &= -\frac{\ln A_d}{\omega_{0den}T_e}
\end{aligned} \tag{12}$$

1.3. Synthèse du filtre discret – sommaire

La synthèse du filtre discret 2pôles/2zéros « bandstop » a ces pas :

1. A déterminer
 - M_t ... atténuation/amplification désiré [dB] (pour l'atténuation $M_t < 0$)
 - f_t ... fréquence de « bandstop » [Hz] (il faut que $f_t \leq \frac{1}{2T_e}$)

- T_e ... période d'échantillonnage du filtre discret
- ξ_d ... amortissement de dénominateur du filtre analogique (certaines règles peuvent être établis pour son choix)

2. Calcul des paramètres du filtre analogique

- fréquence de « bandstop » pour le filtre discret :

$$\omega_{disc} = 2\pi T_e f_t, \omega_{disc} \in \langle 0, \pi \rangle$$

- fréquence de « bandstop » pour le filtre analogique :

$$\omega_{tan} = \frac{2}{T_e} \tan\left(\frac{\omega_{disc}}{2}\right), \omega_{tan} \in \langle 0, \infty \rangle$$

- atténuation – échelle linéaire $M_{lin} = 10^{\frac{M_t}{20}}$
- amortissement de numérateur $\xi_{num} = \xi_{den} M_{lin}$
- construction du filtre analogique avec des paramètres $\omega_{num}, \omega_{den}, \xi_{num}, \xi_{den}$ – équation (1)

3. Transformation bilinéaire

- Conversion de l'équation (1) dans la forme (7) avec des coefficients (8).
- Calcule des fréquences naturelles et des amortissements pour le numérateur et le dénominateur en utilisant les équations (9), (10) et (12).

2. Exemples d'utilisation

2.1. Modèle de bras souple d'ordre 6

Le modèle est pris de livre. Il a les paramètres suivantes :

Modèle : A= 1.0 -2.19358 1.85893 -1.00016 1.12598 -1.43643 0.726861
 B= 0.05723 -0.164075 0.256 -0.13623 0.02946
 d=1
 Te=0.07s

Pôles

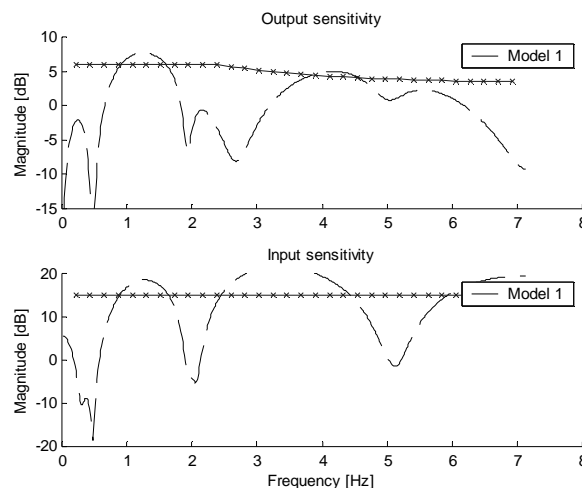
Eigenvalue	Magnitude	Damping	Freq.(rad/s)	Freq.(Hz)
0.971210-0.207125i	0.993051	0.033170	3.003326	0.477994
0.971210+0.207125i	0.993051	0.033170	3.003326	0.477994
0.645353-0.733382i	0.976898	0.027515	12.135435	1.931414
0.645353+0.733382i	0.976898	0.027515	12.135435	1.931414
-0.519773-0.708645i	0.878830	0.058513	31.534558	5.018881
-0.519773+0.708645i	0.878830	0.058513	31.534558	5.018881

Zéros

Eigenvalue	Magnitude	Damping	Freq.(rad/s)	Freq.(Hz)
0.361361-0.240022i	0.433811	0.818447	14.577186	2.320031
0.361361+0.240022i	0.433811	0.818447	14.577186	2.320031
1.072110-1.259323i	1.653878	-0.502553	14.301905	2.276219
1.072110+1.259323i	1.653878	-0.502553	14.301905	2.276219

On démarre avec le régulateur initial, lequel a les paramètres suivantes :

Régulateur initial : Pdom=0.477Hz, 0.8
 Hs :intégrateur
 Paux : 2modes oscillants du modèle avec amortissement 0.1
 $(1-0.3q^{-1})^6$



Normalement on ouvrirait la boucle à $0.5f_c$ en mettant le zéro $(1+q^{-1})$ dans H_r . Cette ouverture enlèverait deuxième dépassement et enverrait S_{up} vers $-\infty$ à la fréquence $0.5f_c$.

2.1.1. Calcul des filtres par la méthodologie analog-discret

On impose les spécifications suivantes : atténuation -2dB à 1.3Hz et -3dB à 4.6Hz . Pour la méthodologie de calcul du filtre discret on a déterminé les paramètres et calculé les filtres suivantes:

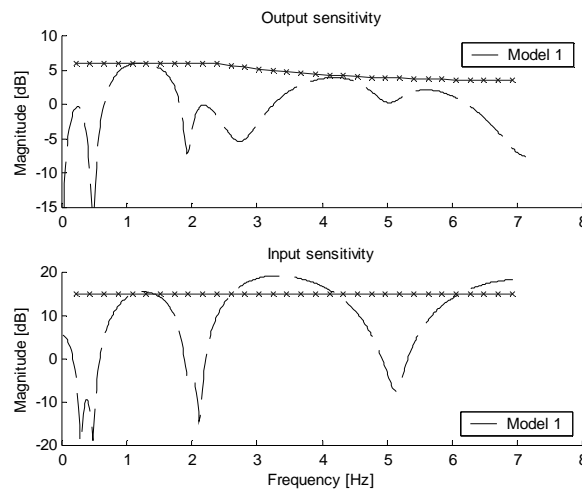
Paramètres : $f_t=1.3$; $M_t=-4$; $\xi_{den}=0.6$; $T_e=0.05s$

Filtre1 : $f_{num}=1.305012$ Hz = 8.1996 rads $^{-1}$, $\xi_{num}=0.369525$
 $f_{den}=1.312823$ Hz = 8.2487 rads $^{-1}$, $\xi_{den}=0.589084$

Paramètres : $f_t=4.2$; $M_t=-2$; $\xi_{den}=0.6$; $T_e=0.05s$

Filtre2 : $f_{num}=4.393155$ Hz = 27.6030 rads $^{-1}$, $\xi_{num}=0.361825$
 $f_{den}=4.534495$ Hz = 28.4911 rads $^{-1}$, $\xi_{den}=0.466249$

Les fonctions de sensibilité sont les suivantes :

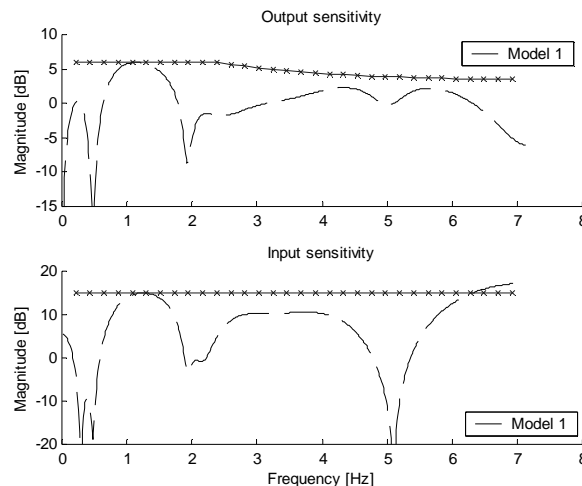


Pour présenter l'efficacité de cette approche on va encore baisser Sup à la fréquence 3.3Hz, où elle pour l'instant monte jusqu'à 20dB. On va donc essayer d'appliquer un filtre avec atténuation -8 dB. Les paramètres de la synthèse de ce filtre et le filtre résultant et les suivantes :

Paramètres : $f_t=3.3$; $M_t=-8$; $\xi_{den}=0.6$; $T_e=0.05s$

Filtre : $f_{num}=3.325688$ Hz = 20.8959 rads $^{-1}$, $\xi_{num}=0.199631$
 $f_{den}=3.485953$ Hz = 21.9029 rads $^{-1}$, $\xi_{den}=0.521830$

Le filtre calculé est cette fois appliqué de la façon différente. Le numérateur est introduit dans la partie fixe de $R - H_r$. Le dénominateur est introduit comme dans les cas précédents dans les pôles auxiliaires fixes de la boucle fermée de placement de pôles. Les fonctions de sensibilité ont avec le nouveau régulateur la forme suivantes :



3. Influence sur les fonctions de sensibilité – comparaison avec la paramétrisation convexe

On considère surtout ces 2 fonctions de sensibilité :

$$S_{yp} = \frac{AS}{AS + BR} = \frac{AS}{P_{BF}} \quad (13)$$

et

$$S_{up} = \frac{BR}{AS + BR} = \frac{BR}{P_{BF}} \quad (14)$$

où B et A sont le numérateur et le dénominateur du modèle respectivement et R, S sont le numérateur et dénominateur du régulateur connecté en boucle fermée. On va étudier seulement l'influence sur la fonction Syp, les relations pour la deuxième sont pareilles. Le filtre 2pôles/2zéros est supposé de la forme générale suivante :

$$F_{zp} = \frac{\gamma}{\delta} \quad (15)$$

Pour une équation de Bezout générale on a ces relations :

$$AY + BX = P$$

$$\deg(A) = n_A$$

$$\deg(B) = n_B$$

$$\deg(P) = n_P = n_A + n_B - 1$$

$$\deg(Y) = n_Y = n_B - 1$$

$$\deg(X) = n_X = n_A - 1$$

où X et Y sont les polynômes inconnus à calculer et A, B, P sont les polynômes fixés.

3.1. Influence du filtre 2zéros/2pôles en placement de pôles

Le régulateur est paramétrisé comme suit :

$$K = \frac{R}{S} = \frac{H_R R_0}{H_S S_0} \quad (16)$$

où H_R et H_S sont les parties fixes et R₀, S₀ les parties calculées par placement de pôles en résolvant l'équation de Bezout :

$$AH_S S_0 + BH_R R_0 = P_{BF} \quad (17)$$

Maintenant, le filtre F_{zp} est appliqué dans le placement de pôles de la façon suivante :

$$\begin{aligned}H_S &= \gamma \\P_f &= \delta \\P_{BF} &= P_x P_f\end{aligned}$$

La fonction de sensibilité devient :

$$S_{yp} = \frac{AS}{AS + BR} = \frac{AS_0 H_S}{P_{BF}} = \frac{AS_0}{P_x} \cdot \frac{\gamma}{\delta} \quad (18)$$

et pour le module on a la relation suivante :

$$|S_{yp}| = \left| \frac{AS_0}{P_x} \right| \cdot \left| \frac{\gamma}{\delta} \right| \quad (19)$$

Ainsi, le module du filtre simplement multiplie le module de la fonction de sensibilité S_{yp} de la boucle fermée sans filtre.

Pour la fonctions de sensibilité S_{up} l'effet est le même, mais le filtre doit être appliqué comme suit :

$$\begin{aligned}H_R &= \gamma \\P_f &= \delta \\P_{BF} &= P_x P_f\end{aligned}$$

3.2. Paramétrisation convexe – comparaison avec le filtre 2pôles/2zéros

La paramétrisation convexe du régulateur a la forme suivante :

$$K = \frac{R}{S} = \frac{H_R (R_0 + AH_S Q)}{H_S (S_0 - BH_R Q)} = \frac{H_R (R_0 \alpha + AH_S \beta)}{H_S (S_0 \alpha - BH_R \beta)} \Big|_{Q=\frac{\beta}{\alpha}} \quad (20)$$

où H_R , H_S sont les partie fixe, α , β sont les polynomiaux à optimiser et R_0 , S_0 sont calculés à partir de l'équation de Bezout :

$$AH_S S_0 + BH_R R_0 = P_D \quad (21)$$

Les pôles de la boucle fermée ont cette forme :

$$AS + BR = P_D \alpha = P_{BF} \quad (22)$$

L'approximation de Ritz utilisé dans [4] pour α et β a la forme suivante :

$$\alpha(x_a) = 1 + \sum_{k=1}^N x_{ak} \alpha_k$$

$$\beta(x_b) = x_{b0} + \sum_{k=1}^N x_{bk} \beta_k$$

$$\alpha_k = \beta_k = \left(\frac{z_0 - z^{-1}}{1 - z_0 z^{-1}} \right)^k$$

où N est l'ordre de paramétrisation. Les paramètres α et β sont ici les fonctions de transfert avec le dénominateur commun, qui s'élimine dans la fonction de sensibilité. A noter, que les ordres des numérateur (et aussi des dénominateur) sont même et sont égaux à N. C'est à dire, avec l'approximation de Ritz utilisé dans [4], on a pour les ordres de α et β la relation suivante :

$$n_\alpha = n_\beta = N$$

Avec ce paramétrisation convexe la fonction de sensibilité devient :

$$S_{yp} = \frac{AS}{AS + BR} = \frac{AH_S(S_0\alpha - BH_R\beta)}{P_D\alpha} \quad (23)$$

Pour la fonctions de sensibilité (23) on demande la forme suivante :

$$S_{yp} = \frac{AH_S(S_0\alpha - BH_R\beta)}{P_D\alpha} = \frac{AS_{fix}}{P_x} \cdot \frac{\gamma}{\delta} \quad (24)$$

Les questions sont :

1. Quelles sont les valeurs γ et δ exprimer en fonction de α et β :

$$\gamma = f_1(\alpha, \beta)?$$

$$\delta = f_2(\alpha, \beta)?$$

En fait on cherche quel est le filtre connecter en cascade avec la fonction de sensibilité qui est optimisé par l'optimisation convexe.

2. Ou bien, quelles valeurs doivent avoir α et β pour que γ et δ aient la forme connu du filtre 2pôles/2zéros :

$$\alpha = f_3(\gamma, \delta)?$$

$$\beta = f_4(\gamma, \delta)?$$

On cherche la forme de α et β pour qu'on ait un filtre 2pôles/2zéros connecter avec la fonction de sensibilité.

3.2.1. Le filtre sur S_{yp} déterminé par paramétrisation convexe – Question 1

Les inconnu à déterminés sont pour l'instant γ , δ , P_x et S_{fix} de l'équation (24).

En supposant certains zéros z_i communs des polynomiaux S_0 et BH_R on définit ce polynomial :

$$Z_c = \sum_i (1 - z_i z^{-1}) \quad (25)$$

La fonction de sensibilité (23) devient :

$$S_{yp} = \frac{AH_S Z_c (S'_0 \alpha - B'H'_R \beta)}{P_x \alpha} = \frac{AH_S Z_c}{P_x} \cdot \frac{(S'_0 \alpha - B'H'_R \beta)}{\alpha} \quad (26)$$

où S'_0 et $B'H'_R$ sont les polynômes sans les zéros communs. On peut distinguer, comme dans le cas du filtre avec placement de pôles, deux parties de la fonction de sensibilité. Une partie qui n'est pas touchée par le filtre formé de α et β et l'autre partie qui est notre filtre. Mais cette fois-ci le filtre a la structure plus compliquée. En comparant les équations (24) et (26) on a pour les inconnus la solution suivante :

$$\begin{aligned} S_{fix} &= H_S Z_c \\ P_x &= P_D \\ \gamma &= S'_0 \alpha - B'H'_R \beta \\ \delta &= \alpha \end{aligned} \quad (27)$$

si on suppose un filtre 2pôles/2zéros, les ordres des polynômes sont les suivants :

$$\begin{aligned} \deg(\delta) &= n_\delta = 2 \\ \deg(\alpha) &= n_\alpha = n_\delta \\ \deg(\gamma) &= n_\gamma = 2 \\ \deg(S'_0 \alpha + B'H'_R \beta) &= n_\gamma \end{aligned} \quad (28)$$

évidemment, pour assurer ces relations, les conditions suivantes doivent être remplies :

$$\begin{aligned} \deg(S'_0) &= n_{S'_0} = 0 \\ \deg(B'H'_R \beta) &= n_{B'} + n_{H'_R} + n_\beta \leq 2 \end{aligned} \quad (29)$$

Pour les ordres du modèle et du régulateur on a ces relations (voir [4]):

$$\begin{aligned} n_{S_0} &= n_B + n_{H_R} - 1 \\ n_{R_0} &= n_A + n_{H_S} - 1 \end{aligned} \quad (30)$$

d'où et de l'équation (26) on peut déduire la relation suivante :

$$n_{S'_0} = n_{B'} + n_{H'_R} - 1 \quad (31)$$

Maintenant, les relations (29) et (31) nous donnent la condition suivante :

$$n_\beta \leq 1 \quad (32)$$

Evidement pour créer un filtre 2pôles/2zéros on a besoin α d'ordre 2 et β d'ordre 1. Ce qui n'est pas vérifié avec l'approximation de Ritz utilisé (la $n_\alpha=n_\beta$) mais l'approximation peut être théoriquement modifiée pour vérifier si elle trouve le même filtre 2pôles/2zéros comme par le calcul du chapitre 1.

3.2.2. La forme des paramètres α et β pour le filtre 2pôles/2zéros – Question 2

Les inconnu sont cette fois ci α et β et on cherche de l'exprimer en utilisant les autres polynomiaux connu entre lesquels on suppose que γ et δ ont la forme du filtre 2pôles/2zéros.

Approche par $S_{\gamma\beta}$

En prenant l'équation (24) on peut écrire :

$$\begin{aligned} S_{fix}\gamma &= H_S(S_0\alpha - BH_R\beta) \\ \delta &= \alpha \end{aligned} \quad (33)$$

A partir de (33) on voit que α est résolu, donc il manque de trouver l'expression pour β . Pour le faire on réécrit (33) de la façon suivante :

$$S_{fix}\gamma + H_S BH_R\beta = H_S S_0\delta \quad (34)$$

et c'est une équation de Bezout où notre inconnu sont β et S_{fix} . En résolvant cette équation on a la solution. Le problème c'est que en regardant l'équation (24) on connaît la forme du filtre $\frac{\gamma}{\delta}$ seulement

dans le cas où on connaît aussi la partie fixe $\frac{AS_{fix}}{P_x}$. Autrement dit, il faut connaître la fonction

laquelle on veut modifier par le filtre pour qu'on puisse ce filtre calculer. Mais dans l'équation (34) on suppose S_{fix} (une partie de la fonction à modifier) comme inconnu, c'est à dire on sait pas quel est le tache du filtre et donc les polynomiaux γ et δ sont inconnus.

Pour avoir une solution unique de, les ordres des polynomiaux de l'équation (34) doivent avoir les valeurs suivantes :

$$\deg(S_{fix}) = n_{S_{fix}} = n_{H_S} + n_B + n_{H_R} - 1$$

$$\deg(\beta) = n_\beta = n_\gamma - 1$$

$$n_\gamma = n_\delta = n_\alpha$$

Evidement, pour assurer le même nombre des pôles et des zéros dans le filtre γ/δ , le polynomial β doit avoir l'ordre inférieur de $\alpha \Rightarrow$ même conclusion comme dans la section 3.2.1. Ce qui ne correspond pas avec l'approximation de Ritz utilisée. Si on insiste sur l'approximation de Ritz actuelle, on aura la condition $n_\alpha=n_\beta$. Puis, les ordres du filtre seront décrits par la relation :

$$n_\gamma = n_\delta + 1 \quad (35)$$

Mais on ne connaît pas une algorithm pour la synthèse du filtre de ce type. Quand même, il est possible d'essayer diminuer l'ordre de β (ça n'influence pas la convexité du problème) et tester quoi donne l'optimisation convexe.

Approche par équation de Bezout

En considérant les relations (20) et (24), l'équation (22) peut avoir la forme suivante :

$$AS_{fix}\gamma + BH_R(R_0\alpha + AH_S\beta) = P_D\alpha \quad (36)$$

où, S était remplacé par $S_{fix}\gamma$ et R par $H_R(R_0\alpha + AH_S\beta)$. Supposons nous comme dans les cas précédents $\alpha=\delta$. L'équation (36) devient :

$$AS_{fix}\gamma + BH_RAH_S\beta = (P_D - BH_RR_0)\delta$$

On a une nouvelle équation de Bezout avec les inconnus β et S_{fix} où les ordres des polynômes ont ces valeurs :

$$\deg(S_{fix}) = n_{S_{fix}} = n_A + n_{H_S} + n_B + n_{H_R} - 1$$

$$\deg(\beta) = n_\beta = n_A + n_\gamma - 1$$

$$n_\gamma = n_\delta = n_\alpha$$

Cette fois les ordres de α et β sont encore plus différents. C'est à dire, la condition de l'approximation de Ritz $n_\alpha=n_\beta$ n'est pas rempli. On a donc la même conclusion comme dans l'approche par S_{yp} .

3.2.3. Conclusion

On a déterminé dans ce chapitre, que la paramétrisation convexe avec l'approximation de Ritz actuelle n'est pas capable de créer un filtre 2pôles/2zéros (ou bien n pôles/ n zéros). Le filtre attaché à la fonction de sensibilité créé par la paramétrisation a toujours un nombre de pôles différent par rapport au nombre de zéros. On ne peut pas donc déterminer une relation évidente entre le filtre 2pôles/2zéros et les paramètres α et β si on suppose l'approximation de Ritz actuelle. Cependant, on a vu qu'il est théoriquement possible de modifier l'approximation, c'est à dire fixer $n_\alpha=n_\beta+1$. Et ensuite de tester l'optimisation si elle trouve un filtre proche de celui calculé par la méthode proposée dans le chapitre 1.

On peut aussi utiliser certains résultats de cette partie pour mieux comprendre le traitement des fonctions de sensibilité par l'optimisation convexe. L'équation (26) dérivée dans le chapitre 3.2.1 peut être visualisée sur un exemple. Pour le modèle de bras souple d'ordre 12 et son régulateur calculé par optimisation convexe on a les résultats dans Figure 2, Figure 3 et Figure 4. L'ordre de paramétrisation du régulateur $N=4$ nous donne les paramètres α et β avec le même ordre. Dans la Figure 4 on voit le placement des pôles de α c'est à dire en fait du dénominateur δ du filtre (voir les relations (26), (27)). La Figure 3 visualise quels sont les zéros du polynôme $Z_c \Rightarrow$ les zéros communs de S_0 et BH_R . Finalement la Figure 2 présente la partie d'une fonction de sensibilité, qui est fixée pendant l'optimisation (la partie sans le filtre) en comparaison avec la fonction de sensibilité complète.

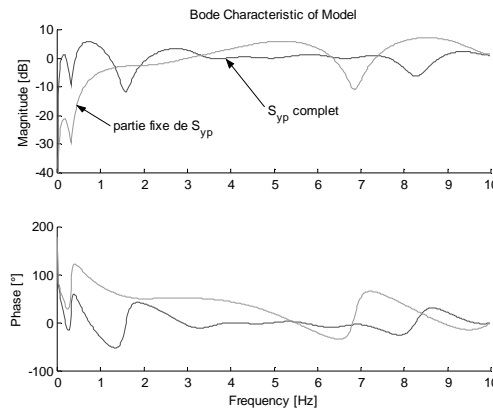


Figure 2 : La fonction S_{yp} complet et sa partie fixe $\frac{AH_s Z_c}{P_x}$ pour le modèle de bras souple

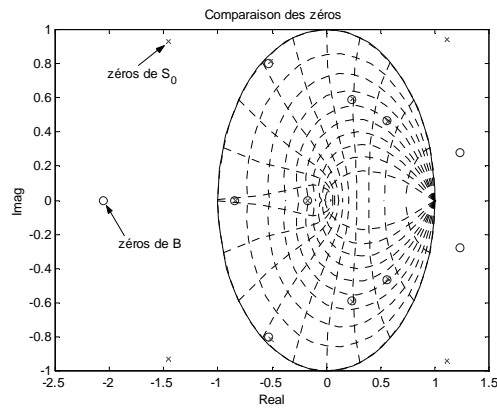


Figure 3 : Visualisation des zéros communs de S_0 et B

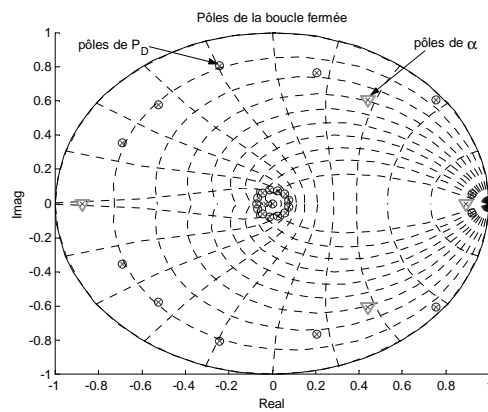


Figure 4 : Pôles de la boucle fermée et ses partition aux pôles de P_D et aux pôles de α

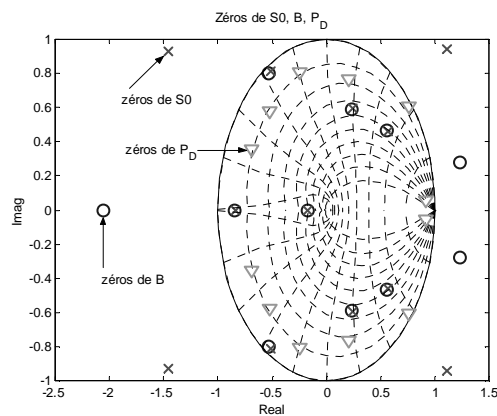


Figure 5 : Comparaison entre les zéros de S_0 , B et P_D

4. Analyse de paramétrisation convexe sur un exemple

4.1. Description de l'analyse

Pour l'instant on traite la fonction de sensibilité S_{yp} , qui a la forme (24). On a vu dans la chapitre 3.2 que pour sortir un filtre γ/δ de 2pôles/2zéros à partir de la paramétrisation convexe, l'ordre n_β de β doit être $n_\beta = n_\alpha - 1$. Cette modification de l'approximation de Ritz sera mis en place dans la logiciel OPTREG et testé sur un exemple. A rappeler que jusqu'à maintenant l'approximation de Ritz utilisée avait β et α de même ordre ($n_\alpha = n_\beta$).

Dans l'exemple qu'on va étudier on créera artificiellement le cas où l'optimisation convexe devra diminuer S_{yp} dans un seul endroit ($|S_{yp}|$ dépasse le gabarit dans un seul région assez étroit). Après, on va laisser l'optimisation de rentre S_{yp} dans le gabarit. La partie de S_{yp} qui est optimisée, c'est à dire qui est modifié dans chaque pas de l'itération a la forme suivante (à partir de (24)):

$$F_{opt} = \frac{(S_0 \alpha - B H_R \beta)}{\alpha} = \frac{S'}{\alpha} = S_f \cdot \frac{\gamma_f}{\alpha} = S_f F_{zp} \quad (37)$$

où F_{zp} est notre filtre 2pôles/2zéros optimisé. Le dénominateur de F_{opt} a l'ordre n_α par contre le numérateur a l'ordre beaucoup plus élevé. On suppose que l'optimisation convexe fait une chose pareille comme nous pour résoudre le problème posé. C'est à dire, qu'elle construit un filtre F_{zp} qui atténue dans un région fréquentiel et n'influence pas le reste. Il faut chercher quels zéros de S' nous forme avec α notre filtre 2pôles/2zéros et éventuellement déterminer à quoi serve le reste S_f de S' . Pour l'optimisation on mettra l'ordre de l'optimisation $N=2$. Donc α sera un polynôme d'ordre 2 et on va chercher 2 zéros dans S' qui nous formeront le polynôme γ_f .

4.2. Exemple analysé

L'exemple a les paramètres suivants: (workspace: wspbrasN2_1pbred3z0.mat)

Modèle

Polynomiaux	A	B	z^{-n}
	1.0	0	1
	-2.193580	0	z^{-1}
	1.858930	0.057230	z^{-2}
$T_c=0.07\text{sec}$	-1.000160	-0.1640750	z^{-3}
	1.125980	0.2560	z^{-4}
	-1.436430	-0.136230	z^{-5}
	0.7268610	0.029460	z^{-6}

Pôles du modèle

Eigenvalue	Magnitude	Damping	Freq.(rad/s)	Freq.(Hz)
0.971210-0.207125i	0.993051	0.033170	3.003326	0.477994
0.971210+0.207125i	0.993051	0.033170	3.003326	0.477994
0.645353-0.733382i	0.976898	0.027515	12.135435	1.931414
0.645353+0.733382i	0.976898	0.027515	12.135435	1.931414
-0.519773-0.708645i	0.878830	0.058513	31.534558	5.018881
-0.519773+0.708645i	0.878830	0.058513	31.534558	5.018881

Zéros du modèle

Eigenvalue	Magnitude	Damping	Freq.(rad/s)	Freq.(Hz)
0.361361-0.240022i	0.433811	0.818447	14.577186	2.320031
0.361361+0.240022i	0.433811	0.818447	14.577186	2.320031

1.072110-1.259323i	1.653878	-0.502553	14.301905	2.276219
1.072110+1.259323i	1.653878	-0.502553	14.301905	2.276219

Paramètres de placement de pôles

Pôles dominants	$f_n=0.477994\text{Hz}$ $\xi=0.800000$	Pôles dominants du modèle avec l'amortissement modifié
Pôles auxiliaires	$f_n= 1.931414\text{Hz}$ $\xi=0.25$ $f_n= 5.018881\text{Hz}$ $\xi= 0.058513$ $(1 - 0.3q^{-1})^5$	Pôles oscillants du modèle plus un pôle multiple réel
Hs	$(1-q^{-1})$	Intégrateur
Hr	$(1+q^{-1})$	Ouverture de boucle

Paramètres de l'optimisation

z_0	0.3	Constant de paramétrisation
N	2	Ordre de paramétrisation

Régulateur - placement de pôles

Les zéros de S_0

Eigenvalue	Magnitude	Damping	Freq.(rad/s)	Freq.(Hz)
0.361640-0.239880i	0.433966	0.818622	14.567864	2.318548
0.361640+0.239880i	0.433966	0.818622	14.567864	2.318548
0.261318-0.782136i	0.824636	0.152645	18.045016	2.871953
0.261318+0.782136i	0.824636	0.152645	18.045016	2.871953
-0.600836-0.404625i	0.724379	0.125501	36.703301	5.841512
-0.600836+0.404625i	0.724379	0.125501	36.703301	5.841512

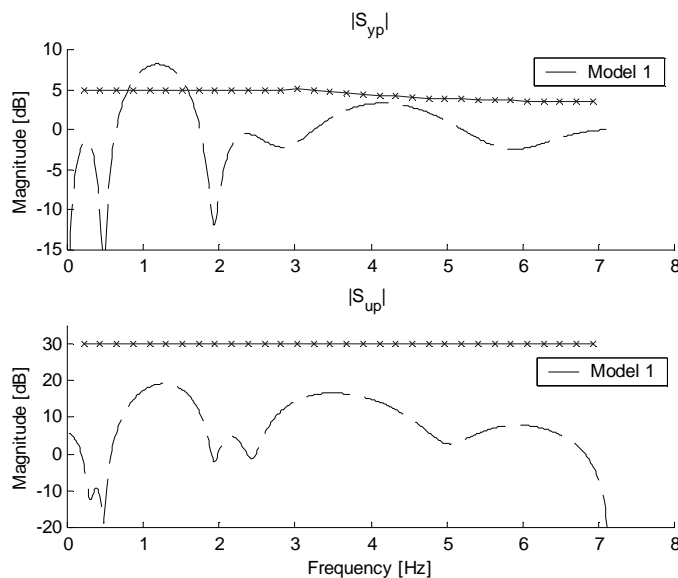


Figure 6: Fonctions de sensibilité pour le régulateur nominal (S_0 , R_0) calculé par placement de pôles

Le région de dépassement est entre 0.9Hz et 1.8Hz de $|S_{yp}|$. Le gabarit de S_{up} est assez haut et l'optimisation du logiciel OPTREG qui en général gère les deux fonctions en même temps va donc optimiser le régulateur que pour satisfaction du gabarit sur S_{yp} .

Régulateur - optimisation convexeLes zéros de α

Eigenvalue	Magnitude	Damping	Freq.(rad/s)	Freq.(Hz)
0.461160+0.000000i	0.461160	1.000000	11.057287	1.759822
0.627751+0.000000i	0.627751	1.000000	6.651595	1.058634

Les zéros de S'

Eigenvalue	Magnitude	Damping	Freq.(rad/s)	Freq.(Hz)
0.361280-0.239959i	0.433709	0.818531	14.579787	2.320445
0.361280+0.239959i	0.433709	0.818531	14.579787	2.320445
0.693297-0.429960i	0.815798	0.344323	8.446740	1.344340
0.693297+0.429960i	0.815798	0.344323	8.446740	1.344340
0.072621-0.891806i	0.894758	0.074448	21.338419	3.396115
0.072621+0.891806i	0.894758	0.074448	21.338419	3.396115
-0.695959-0.363231i	0.785045	0.090589	38.165037	6.074154
-0.695959+0.363231i	0.785045	0.090589	38.165037	6.074154

Premières 2 zéros complexe conjugués ($f_n=2.320445\text{Hz}$) sont à peu près communs avec 2 zéros du modèle. Son tache est donc simplifié ses zéros dans la fonction de la boucle ouvert. Les autres zéros peuvent être considéré comme les zéros potentiels de notre filtre 2pôles/2zéros. Les seuls deux zéros complexes conjugués sont placés dans notre région critique (0.7Hz - 1.8Hz). Ces sont les zéros à $f_n=1.344340\text{Hz}$ avec l'amortissement $\xi=0.344323$. Appelons le polynôme correspondant à cette paire des zéros γ_f et le polynôme ressemblant les autres zéros, sauf les zéros communs avec B, S'_f . Finalement supposons le polynôme des 2 zéros communs Z_c . Ensuite, le filtre 2pôles/2zéros a la forme $F_{zp} = \frac{\gamma_f}{\alpha}$ et

le filtre complet optimisé c'est $F_{opt} = Z_c S'_f \frac{\gamma_f}{\alpha}$. Le filtre F_{zp} atténue le plus à 1.344Hz de -9.4dB. Pour comparer on a calculé un filtre 2pôles/2zéros avec la méthode analytique proposé dans le chapitre 1. On a utilisé les spécifications suivantes: $f_t=1.344$; $M_t=-9.4$; $\xi_d=0.9$. Les réponse fréquentielles des deux filtres sont dans la Figure 7. Evidement les réponses sont très proches même si les dénominateurs des filtres sont différents. Les pôles et les zéros du filtre analytique ont les valeurs suivants:

Eigenvalue	Magnitude	Damping	Freq.(rad/s)	Freq.(Hz)
pôles				
0.537023-0.071727i	0.541791	0.977327	8.958461	1.425783
0.537023+0.071727i	0.541791	0.977327	8.958461	1.425783
zéros				
0.700654-0.443581i	0.829265	0.314839	8.494864	1.352000
0.700654+0.443581i	0.829265	0.314839	8.494864	1.352000

Les zéros des filtres sont assez proches (fréquences: 1.352000 et 1.344340 ;amortissements: 0.314839 et 0.344323). Par contre les pôles sont une fois (cas de synthèse analytique) complexes conjugués et une fois (le cas du filtre optimisé) réels.

La réponse fréquentielle du reste $Z_c S'_f$ du filtre optimisé et dans la Figure 8. Finalement, la partie de S_{yp} qui n'est pas touché par l'optimisation convexe qui a la forme $F_{fix} = \frac{AH_s}{P_D}$ et sa réponse fréquentielle est dans la Figure 9.

On a découpé notre fonction de sensibilité S_{yp} aux plusieurs parties la fonctions complet a la forme :

$$S_{yp} = F_{fix} \underbrace{S'_f F_{zp}}_{F_{opt}} = \frac{AH_s}{P_D} Z_c S'_f \frac{\gamma_f}{\alpha}$$

où sa réponse fréquentielle avec la réponse de S_{up} est dans la Figure 10. La Figure 11 compare les deux fonctions de sensibilité S_{yp} , une calculée pour le régulateur nominal (placement de pôles) et l'autre calculée pour le régulateur optimisé (optimisation convexe).

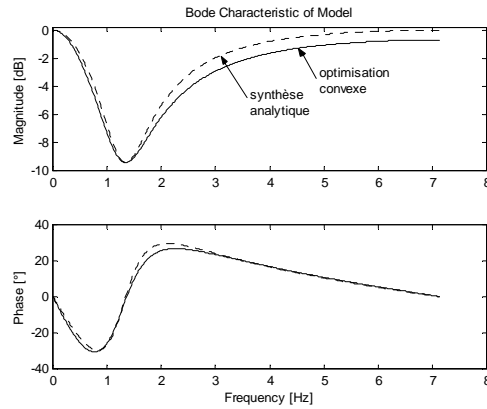


Figure 7: Réponses fréquentielles des filtres 2pôles/2zéros, un calculé par optimisation convexe, l'autre par la méthode analytique

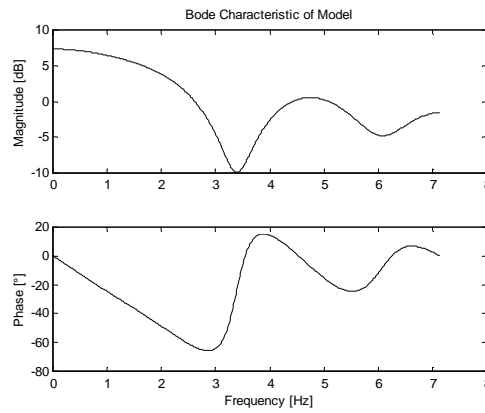


Figure 8: Réponse fréquentielle de S_f

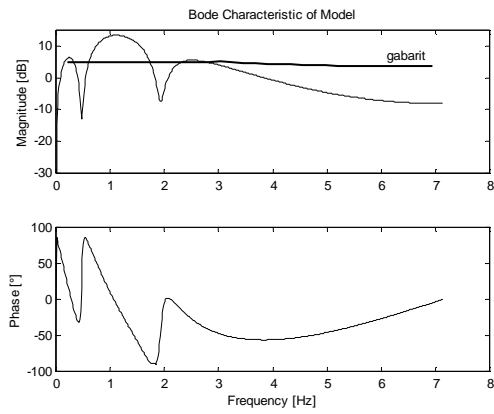


Figure 9: Réponse fréquentielle de F_{fix}

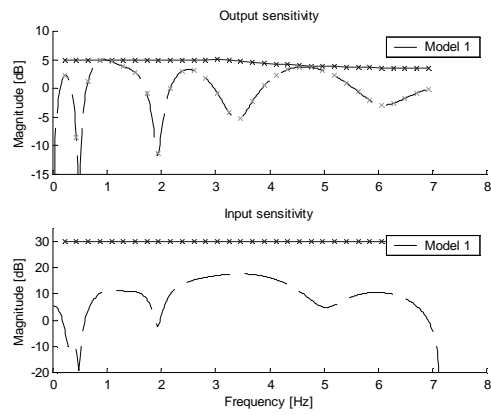


Figure 10: Réponse fréquentielle de S_{yp} et S_{up} pour le régulateur optimisé

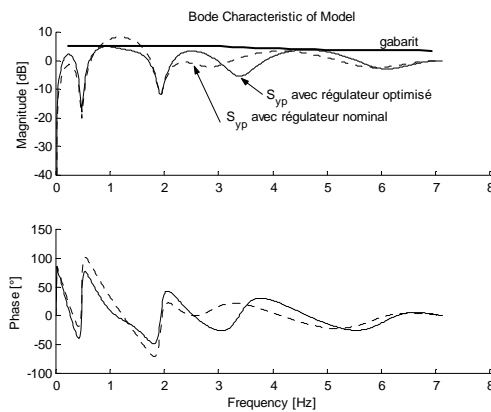


Figure 11: Comparaison de S_{yp} du régulateur nominale et celle calculée avec le régulateur optimisé

4.3. Conclusions et remarques sur l'analyse

On a remarqué que la diminution de 1 de l'ordre n_β de β diminue aussi la complexité du régulateur (les ordres n_S et n_R) de 1. Mais la diminution plus grande de n_β n'influence plus la complexité et les ordres reste de mois 1 plus petits. Ce conclusion est évident en vérifiant les ordres des polynomiaux dans l'équation (20).

Le filtre 2pôles/2zéros créé par l'optimisation dans notre exemple a dans dénominateur α 2 pôles réels. Mais la méthode analytique propose 2 pôles complexe conjugués. Ce phénomène a été observé dans trois exemples des modèle de bras souple (les modèles d'ordre 6 et 8), mais par exemple pour un modèle de poulie (workspace poulieN2_bred.mat) on a obtenu un paire des pôles complexes conjugués. Quand on a calculé le filtre analytique de tel façon qu'on obtiendrait à peu près le même amortissement dans le dénominateur, les réponses fréquentielles du filtre analytique et du filtre optimisé sont très proches (voir Figure 12).

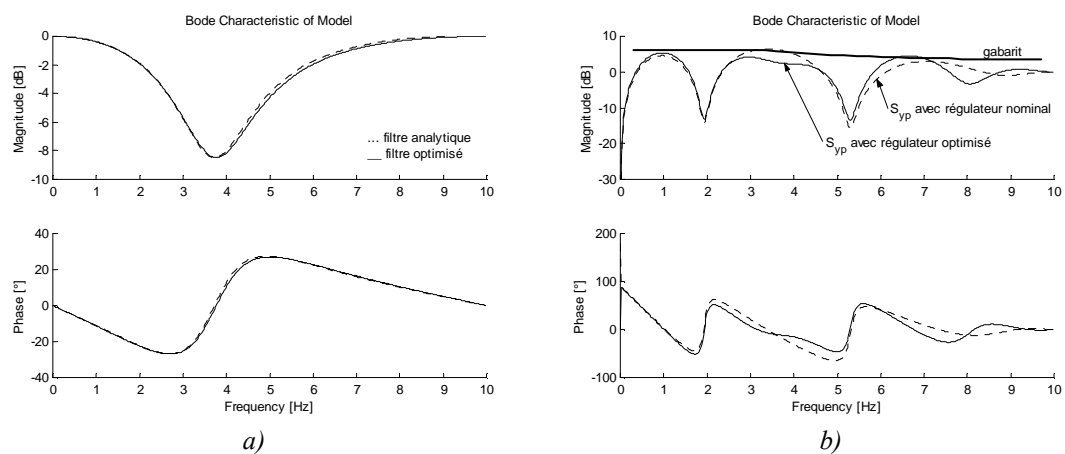


Figure 12: Exemple du poulie a) Les réponses fréquentielles d'un filtre calculé analytiquement et d'un filtre calculé par optimisation convexe. b) S_{yp} pour le régulateur nominal et pour le régulateur optimisé

La constante de paramétrisation $z_0 \in (-1,1)$ joue aussi un rôle assez important dans la construction du filtre par l'optimisation convexe. Sa valeur influence la vitesse des pôles optimisé α (voir [4]). Plus elle est grande plus les pôles sont lents. En connaissant ce règle on peut estimer la valeur de z_0 . Surtout si le logiciel OPTREG propose une valeur correspondant au moyen des pôles du modèle et des valeurs raisonnables des limites. Dans la Figure 13 on a pour notre exemple de bras souple des réponses fréquentielles des filtres 2pôles/2zéros optimisé. Ces filtres étaient calculés en utilisant 3 différents

valeurs de z_0 (10^{-8} ; 0.3; 0.5). Les pôles des filtres sont vraiment déplacés vers haut fréquences avec z_0 plus petit.

Pôles de α pour des différents valeurs de z_0

Eigenvalue	Magnitude	Damping	Freq.(rad/s)	Freq.(Hz)
$z_0=10^{-8}$				
-0.329327+0.000000i	0.329327	0.333329	47.602241	7.576132
0.824920+0.000000i	0.824920	1.000000	2.749548	0.437604
$z_0=0.3$				
0.461160+0.000000i	0.461160	1.000000	11.057287	1.759822
0.627751+0.000000i	0.627751	1.000000	6.651595	1.058634
$z_0=0.5$				
0.538995+0.000000i	0.538995	1.000000	8.829268	1.405222
0.811736+0.000000i	0.811736	1.000000	2.979708	0.474235

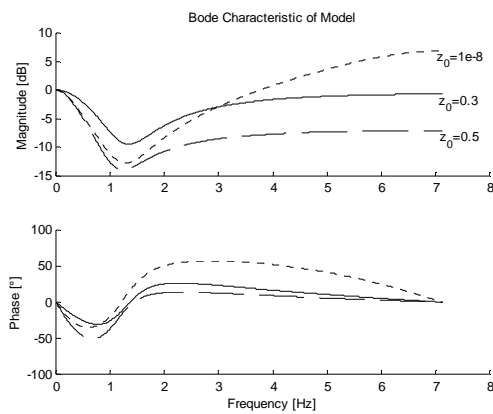


Figure 13: Réponses fréquentielles des filtres 2pôles/2zéros optimisés, calculés avec les différents valeurs des z_0

5. Bibliography

[1] A.V. Oppenheim, R. W. Schaffer, « **Digital Signal Processing** », Prentice-Hall, New Jersey, 1975

[2] F. Taylor, The Athena Group, INC., J. Mellott, « **Hands-on Digital Signal Processing** », McGraw-Hill, 1998

[3] S.A. Tretter, « **Introduction to Discrete-time Signal Processing** », John Wiley & Sons, New York, 1976

[4] Langer, J., « **Synthèse de régulateurs numériques robustes, application aux structures souples** », thèse LAG, 1998

[5] ADAPTECH, «User's Guide: OPTREGTM, Automated Design of Robust Digital Controllers», 1999

User's Guide for RMC tool MATLAB[®] graphical toolbox
for robust MIMO digital controller design

Hynek Procházka

September 23, 2003

Contents

1	About toolbox	3
2	Basic Theory	4
2.1	Closed loop	4
2.1.1	Multi-input multi-output linear robust control problem	4
2.1.2	Robustness and performance by sensitivity shaping . .	5
2.2	Convex design specifications	6
2.2.1	Sensitivity shaping criteria	7
2.2.2	Closed-loop δ -stability criterion	8
2.2.3	Multi-criterion function and optimisation algorithm . .	9
2.3	Controller structure	9
2.4	Design specifications as convex criteria	11
2.5	Design procedure	12
3	Description of application	14
3.1	Software requirements	14
3.2	Getting started	14
3.3	Principal window	15
3.3.1	Menu bar	16
3.4	Model in detail window	17
3.5	Parameters in detail window	19
3.5.1	General description	19
3.5.2	Pole placement window	20
3.6	Optimisation in detail window	21
3.6.1	General description	21
3.6.2	Template window	24
3.6.3	δ -region window	25
3.7	Closed loop in detail window	25
3.7.1	Sensitivity response analysis window	27
3.7.2	Transfer matrix analysis window	28
3.7.3	Time simulation window	28

Annexe B9. H.Prochazka, User's Guide for RMC tool. 2

Bibliography 32

Chapter 1

About toolbox

The RMC (Robust Multivariable Control) controller design tool is a toolbox programmed in MATLAB[®] 5.3 environment¹ (it works also under MATLAB 6.0 and higher). It is user friendly application with graphical user interface (GUI) for design of digital MIMO (multi-input, multi-output) robust controllers. The implemented controller design procedure is multivariable pole placement with sensitivity shaping and convex optimisation presented in [PL03a, PL03b].

The toolbox was developed at Laboratoire d'Automatique de Grenoble as a part of doctoral thesis "H. Procházka, Synthèse de régulateurs numériques robustes multivariables par optimisation convexe, Institut National Polytechnique de Grenoble" under direction of I.D.Landau.

¹MATLAB is a registered trademark of The MathWorks, Inc.

Chapter 2

Basic Theory

2.1 Closed loop

2.1.1 Multi-input multi-output linear robust control problem

It is supposed that we dispose with a linear discrete-time multi-input multi-output time-invariant (MIMO) model $G(z)$ of a plant. The model is supposed to have the following state space representation form:

$$\begin{aligned}x_p(t+1) &= Ax_p(t) + Bu_p(t) \\y_p(t) &= Cx_p(t) + Du_p(t)\end{aligned}\tag{2.1}$$

with $A \in R^{n_x \times n_x}$, $B \in R^{n_x \times n_u}$, $C \in R^{n_y \times n_x}$, $D \in R^{n_y \times n_u}$. The matrices A , B , C are some constant state space matrices and $u_p(t)$, $y_p(t)$ and $x_p(t)$ are any vectors of plant inputs, outputs, and states respectively. The variables n_u , n_y , and n_x represent the number of plant inputs, outputs and states respectively. In the case of digitalized continuous time plants (considered in here), the matrix D is zero. The time variable t is discrete time: $t = kT_e$, $k = 0, 1, 2, \dots$ with a sampling time T_e . A corresponding model transfer function matrix is then:

$$G(z) = C(zI - A)^{-1}B\tag{2.2}$$

The considered closed loop system configuration to achieve any desired closed-loop performance is shown in Fig. 2.1. The block K is an appropriate MIMO controller described either in state space representation with A_K , B_K , C_K , D_K , where the matrix D_K is non-zero, or by a transfer matrix $K(z)$. The transfer matrix T is an eventual tracking filter to modify tracking performances.

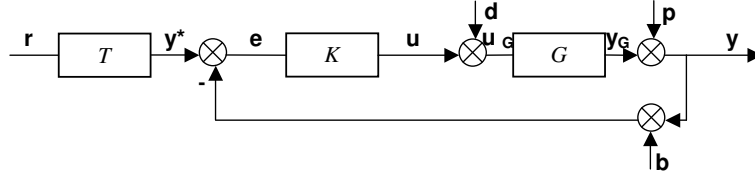


Figure 2.1: Considered closed loop configuration

2.1.2 Robustness and performance by sensitivity shaping

Generally, output disturbance $p(t)$, measuring noise $b(t)$ and command disturbance $d(t)$ are expected to act on the system (see Fig. 2.1). Influence of these disturbances onto CL output $y(t)$ and command $u(t)$ is expressed in the form of sensitivity functions. Typically, the following sensitivity matrices are usually treated:

- $S_{yp}(z)$: Sensitivity matrix output disturbance - output

$$S_{yp}(z) = \frac{y(z)}{p(z)} = (I + G(z)K(z))^{-1} \quad (2.3)$$

- $S_{up}(z)$: Sensitivity matrix output disturbance - command

$$S_{up}(z) = \frac{u(z)}{p(z)} = -K(z)(I + K(z)G(z))^{-1} \quad (2.4)$$

- S_{yb} : Sensitivity matrix measuring noise - output

$$S_{yb}(z) = \frac{y(z)}{b(z)} = -G(z)K(z)(I + G(z)K(z))^{-1} \quad (2.5)$$

- S_{yd} : Sensitivity matrix command disturbance - output

$$S_{yd}(z^{-1}) = \frac{y(z^{-1})}{d(z^{-1})} = G(z^{-1})(I + K(z^{-1})G(z^{-1}))^{-1} \quad (2.6)$$

- S_{ud} : Sensitivity matrix command disturbance - command

$$S_{ud}(z^{-1}) = \frac{u(z^{-1})}{d(z^{-1})} = -K(z^{-1})G(z^{-1})(I + K(z^{-1})G(z^{-1}))^{-1} \quad (2.7)$$

Imposing bounds on these sensitivity function matrices one can match a majority of control specifications, especially the ones related to robustness. Restrictions on $S_{yp}(z)$, $S_{yd}(z)$, $S_{ud}(z)$ and $S_{yb}(z)$ can guarantee closed-loop robust stability with respect to different uncertainties and disturbances [Mac89, GL95, PL03b], and certain bounds on S_{yb} adjust closed loop tracking properties. The $S_{up}(z)$ matrix allows to assess the stress on the actuator in the frequency domain, in other words adjusting this sensitivity mainly the controller gain is shaped.

A typical robustness condition in H_∞ -optimisation approach for one sensitivity matrix $S_{ij}(z)$ has the following general form:

$$\|W_j(z)S_{ij}(z)W_i(z)\|_\infty \leq \gamma \quad (2.8)$$

where γ is an appropriate constant assuring satisfactory performances and $\|\cdot\|_\infty$ is the norm of H_∞ . The indices i, j denote any considered output ($y(t)$, $u(t)$) - input ($p(t)$, $b(t)$, $d(t)$) signals. The $W_i(z)$ is a weighting transfer matrix and its inverse represents a template imposed on the sensitivity matrix. The weighting transfer matrix $W_j(z)$ can be interpreted as a disturbance model for the corresponding input signal j .

Replacing the norm $\|\cdot\|_\infty$ by its expression with the largest singular value $\bar{\sigma}(\cdot)$ and substituting z with $z = e^{j\omega_d}$, where $\omega_d \in \langle 0, \pi \rangle$ is normalized discrete frequency:

$$\omega_d \doteq \omega T_e \mid \omega \in \langle 0, \frac{\pi}{T_e} \rangle \quad (2.9)$$

the condition (2.8) becomes:

$$\bar{\sigma}(W_j(e^{j\omega_d})S_{ij}(e^{j\omega_d})W_i(e^{j\omega_d})) \leq \gamma \mid \forall \omega_d \quad (2.10)$$

To implement the sensitivity function shaping [LK98, PL02, PL03a, Lan98], a template $F_{ij}(\omega_d)$ is imposed on the largest singular value evolution $\bar{\sigma}(S_{ij}(e^{j\omega_d}))$. This template replaces the weighting matrix $W_i(e^{j\omega_d})$ (the template corresponds to the singular value evolution of the weighting matrix inverse). Next, to simplify expressions suppose $W_j(e^{j\omega_d}) = I$ and we obtain:

$$\bar{\sigma}(S_{ij}(e^{j\omega_d})) \leq F_{ij}(\omega_d) \mid \forall \omega_d \quad (2.11)$$

where the developed sensitivity shaping condition (2.11) is non-convex with respect to the controller matrix $K(z)$.

2.2 Convex design specifications

Convex optimisation can be applied only if all required specifications on robustness and on closed-loop performances are expressed via a convex cri-

terion function. In this section, convex criterion functions will be developed for these two design specifications:

- *Sensitivity shaping* - Based on (2.11), templates are imposed on the largest singular value of sensitivity matrix. These templates imposed on different closed-loop sensitivities can guarantee a major of closed-loop specifications concerning robustness and performance.
- *Closed-loop δ -stability* - fixed region imposed on closed loop pole locations. It allows to assure eventual specifications concerning closed-loop dynamic - pole natural frequency and damping.

2.2.1 Sensitivity shaping criterions

Applying "Youla-Kucera" parametrization of all stabilizing controllers [YJB76, Kuc91, Mac89, BB91] on the controller and using a specific structure of the free parameter Q [PL03a]:

$$Q(z) = d^{-1}(z)N(z) \quad (2.12)$$

the sensitivity shaping condition (2.11) becomes (after some algebraic operations):

$$|d(e^{j\omega_d})|\bar{\sigma}(S_{ij}(e^{j\omega_d})) - \text{Re}\{d(e^{j\omega_d})\}F_{ij}(\omega_d) \leq 0 \quad \forall \omega_d \quad (2.13)$$

All the terms in the inequality (2.13) are depending on the frequency ω_d . We will simplify the inequality expression by omitting the notation of frequency dependence $e^{j\omega_d}$.

$$|d|\bar{\sigma}(S_{ij}) - \text{Re}\{d\}F_{ij} \leq 0 \quad \forall \omega_d \quad (2.14)$$

Unfortunately, the new sensitivity shaping condition 2.14 is more conservative than 2.11. On the other hand, stability of the polynomial $d(z)$, and consequently the closed loop stability is assured with the new condition, since verifying the condition (2.14) the polynomial $d(z)$ fulfills the condition: $\text{Re}\{d(e^{j\omega_d})\} > 0, \forall \omega_d$, which means that $d(z)$ is strictly positive real and this is sufficient condition of polynomial stability.

Similar condition can be developed for every element (transfer function) of any sensitivity matrix S_{ij} . In the condition (2.14), the largest singular value $\bar{\sigma}(\cdot)$ is replaced with modulus $|\cdot|$ and hence we obtain a matrix of frequency response modulus. Then, multiplying both sides of sensitivity matrix with two vectors-selectors v_1 and v_2 , we can choose one element of that matrix. The two vectors v_1, v_2 have all elements except one null. The one non-null

element is equal to one and its position determines which transfer function is chosen. The developed one-element sensitivity shaping condition has the following form:

$$|dv_1^T S_{ij} v_2| - \operatorname{Re}\{d\} F_{ij} \leq 0 \quad | \quad \forall \omega_d \quad (2.15)$$

Define now criterion functions ϕ_{ij} and $\phi_{ijv_1v_2}$ suitable for optimisation of the conditions (2.14) and (2.15):

$$\phi_{ij}(X) = \sup_{\omega_d} [|d| \bar{\sigma}(S_{ij}) - \operatorname{Re}\{d\} F_{ij}] \quad (2.16)$$

$$\phi_{ijv_1v_2}(X) = \sup_{\omega_d} [|dv_1^T S_{ij} v_2| - \operatorname{Re}\{d\} F_{ij}] \quad (2.17)$$

where the criterion functions are convex with respect to the optimized vector X .

To evaluate the criterion, a frequency point grid is defined and the function $\phi_{ij}(X)$ is computed at these points. The maximum value from the obtained results will approximate the real criterion value.

2.2.2 Closed-loop δ -stability criterion

The poles of the observer and of the state feedback are fixed before the optimisation during central controller calculation, generally according to some desired performances. It remains to restrict the optimized poles of $d(z)$. First, recall that verification of the inequality (2.14) guarantees stability of $d(z)$, because the condition (with $z = e^{j\omega_d}$):

$$-\operatorname{Re}\{d(z)\} < 0 \quad | \quad \forall \omega_d \quad (2.18)$$

is fulfilled and this is a sufficient condition of stability.

Based on the condition (2.18) [Lan98, LL99] have introduced (for SISO case) a condition of δ -stability by defining bounds in the form of a circle with its center on real axis. The modified condition of δ -stability is defined as follows:

$$-\operatorname{Re}\{\bar{d}(z)\} < 0 \quad | \quad \forall \omega_d \quad (2.19)$$

where

$$\bar{d}(z) = d(\bar{z})|_{\bar{z}=rz+c} \quad (2.20)$$

The variables r and c are respectively a radius of circle and a circle center position on real axis.

The criterion function ϕ_δ for the inequality (2.19) has the following form:

$$\phi_\delta(X) = \sup_{\omega_d} [-\operatorname{Re}\{\bar{d}(z)\}] \quad (2.21)$$

The criterion function ϕ_δ is convex and the proposed evaluation of this function is equivalent to that one provided in the case of sensitivity function shaping criterion.

2.2.3 Multi-criterion function and optimisation algorithm

The function $\max(\cdot)$ is used to assembly several criterion functions. Consequently, the multi-criterion convex function ϕ_{mc} is defined as follows:

$$\phi_{mc}(X) = \max_k [\alpha_k \phi_k(X)] \quad (2.22)$$

where the index k denote appropriate criterion function ($k = \{\delta, yp, ypv_1v_2, up, \dots\}$) and α_k is weighting coefficient ($\alpha_k \in R$) giving desired weight to this function.

The optimisation algorithm is a standard ellipsoid algorithm described in [BB91] (the same was used by [Lan98, LL99]). The algorithm is gradient based and it allows also optimisation with constrains. Using optimisation with constrains, we can distinguish an objective function $\phi_{OBJ}(X)$ and a constraint function $\phi_{CST}(X)$. Both of this criterion functions have to be convex and, in our case, they have the general form (2.22). The optimisation algorithm proceeds as follows:

$$\begin{array}{ll} \text{if } (\phi_{CST}(X) < 0) & \rightarrow \text{ minimizing } \phi_{OBJ}(X) \\ \text{else} & \rightarrow \text{ minimizing } \phi_{CST}(X) \end{array}$$

2.3 Controller structure

Suppose the controller $K_c(z)$ parameterized by "Youla-Kucera" parametrization with Q a free parameter of a certain structure [PL03a]. Moreover, suppose that the final controller contains a pre-fixed block named $H_K(z)$, which is any stable transfer matrix (for example MIMO integrator). Then, the final controller $K(z)$ is defined as follows:

$$K(z) = H_K(z)K_c(z) = H_K(z)(U_0(z) + M(z)Q(z))(V_0(z) - N(z)Q(z))^{-1} \quad (2.23)$$

The optimized controller $K_c(z)$ is constructed from the model $G(z)$, the central controller $K_0(z)$ and the optimized block $Q(z)$. It was shown (see for example [Zho98, BB91]), that the optimized controller $K_c(z)$ can be represented as an observer with state feedback matrix, where the block $Q(z)$ is

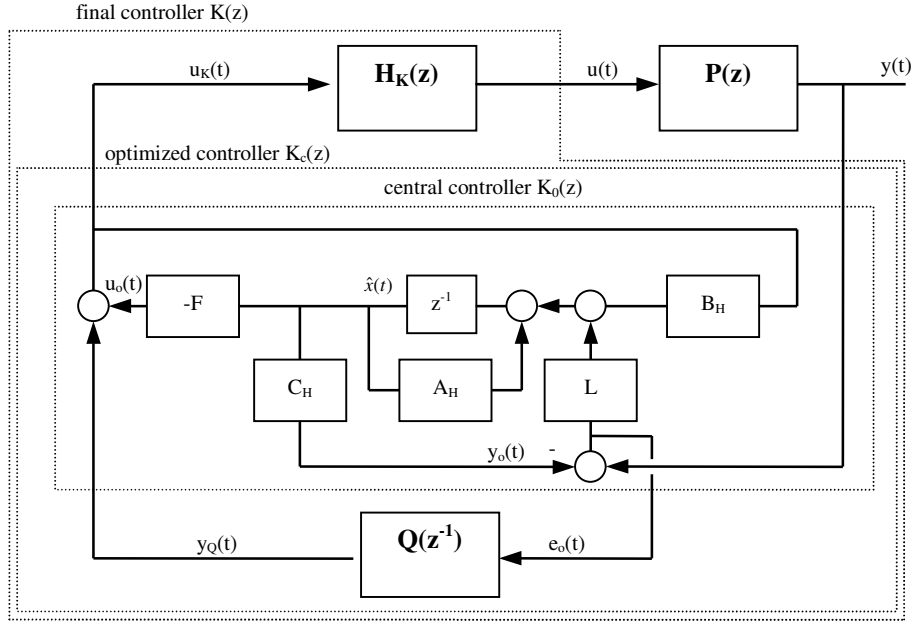


Figure 2.2: Controller structure

connected between the output prediction error $e_o(t)$ and the state feedback command $u_K(t)$. The corresponding diagram is shown in Fig. 2.2.

To construct final (complete) controller $K(z)$, we first define a system $H(z)$ of the cascade connection $H(z) = G(z)H_K(z)$ with the state vector $x_H = [x_{H_K}, x_p]^T$. The state space representation (SSR) of $H(z)$, $H_K(z)$ and $Q(z)$ are as follows:

$$\begin{aligned}
 x_H(t+1) &= A_H x_H(t) + B_H u(t) \\
 y(t) &= C_H x_H(t) \\
 x_{H_K}(t+1) &= A_{H_K} x_{H_K}(t) + B_{H_K} u_o(t) \\
 u(t) &= C_{H_K} x_{H_K}(t) + D_{H_K} u_o(t) \\
 x_Q(t+1) &= A_Q x_Q(t) + B_Q e_o(t) \\
 y_Q(t) &= C_Q x_Q(t) + D_Q e_o(t)
 \end{aligned} \tag{2.24}$$

where A_i, B_i, C_i, D_i ($i \equiv (H, H_K, Q)$) are some constant state space matrices and $x_i(t)$, are corresponding states. The signal $y_Q(t)$ is an output vector of the matrix $Q(z)$, and it is added to the estimated state feedback vector $u_o(t)$ - see Fig. 2.2.

The state observer has the following form:

$$\begin{aligned}
 \hat{x}(t+1) &= A_H \hat{x}(t) + L e_o(t) + B_H u_K(t) \\
 y_o(t) &= C_H \hat{x}(t)
 \end{aligned} \tag{2.25}$$

where $\hat{x}(t)$ is an estimation of the state vector x_H . The estimated state feedback vector $u_K(t)$ and the output estimation error vector $e_o(t)$ are defined as follows:

$$\begin{aligned} u_K(t) &= -F\hat{x}(t) + y_Q(t) \\ e_o(t) &= y(t) - y_o(t) \end{aligned} \quad (2.26)$$

The central controller $K_0(z)$ includes the state space matrices of the cascade connection $H(z)$ together with the observer matrix L - forming observer (2.25), and the estimated state feedback matrix F - estimated state feedback part. The matrices L and F determine assignment of one part of closed loop poles (the rest of poles are in $Q(z)$), where eigenvalues of $(A_H - LC_H)$ and $(A_H - B_H F)$ matrices define respectively observer and estimated state feedback closed loop poles. Two major methods exist for the computation of the central controller: eigenstructure assignment and linear quadratic gaussian -LQG method (kalman filter with optimal state feedback). In this paper, we consider only assignment method, and particularly the algorithms of CL eigenvector optimisation [KND85, TY96, Var00], but LQG technique can be applied as well. The algorithms of multivariable pole assignment define some criterion (guaranteeing robust placement - a small shift of model poles causes only small changes in CL pole locations, low controller gain, etc.) and then search by an optimisation technique for a criterion minimum (not necessarily global) by optimizing the closed-loop eigenvectors.

The relations (2.26) and the state space representation of $Q(z)$ together with the equation (2.25) give the state space representation of the optimized controller $K_c(z)$ with the state vector $x_c(t) = [\hat{x}(t), x_Q(t)]^T$:

$$\begin{aligned} x_c(t+1) &= A_c x_c(t) + B_c y(t) \\ u_K(t) &= C_c x_c(t) + D_c u_K(t) \end{aligned} \quad (2.27)$$

with A_c , B_c , C_c and D_c controller state space matrices. The final controller $K(z)$ is then a cascade connection of the optimized controller $K_c(z)$ and the pre-fixed transfer matrix block $H_K(z)$ as it shows (2.23) and Fig. 2.2.

2.4 Design specifications as convex criterions

Convex optimisation can be applied only if all required specifications on robustness and on closed-loop performances are expressed via a convex criterion function. In this section, convex criterion functions will be developed for these two design specifications:

- *Sensitivity shaping* - Based on (2.11), templates are imposed on the largest singular value of sensitivity matrix. These templates imposed on different closed-loop sensitivities can guarantee a major of closed-loop specifications concerning robustness and performance.
- *Closed-loop δ -stability* - fixed region imposed on closed loop pole locations. It allows to assure eventual specifications concerning closed-loop dynamic - pole natural frequency and damping.

2.5 Design procedure

The complete MIMO controller design procedure consists of following steps:

- Transform desired closed-loop specifications to design specifications compatible with the proposed procedure. As it was shown in previous sections, basically these two design specifications can be used:
 - Templates on sensitivity matrices. The templates should express desired robustness and performances (robust stability, disturbance rejection in specific frequencies, etc.)
 - δ -stability region for optimized poles. Using the criterion function (2.16) as the optimisation objective, the stability region is automatically the entire unit circle (because of the element $Re\{d_{N_q}(X)\}$). To restrain more the region for optimized CL poles, one has to set the function (2.21) as the constraint function of optimisation. The radius r and the center c from (2.20) determine the desired region-circle in complex plain for optimized CL pole locations.

Finally, it has to be decided which design specifications will be considered as constraints and which will form optimisation objective.

- Select the pre-fixed block H_K from (2.23). Some of required performances can be directly assured by incorporating a fixed filter into the controller. This block often contains the internal model of a disturbance or a filter opening the loop at certain frequencies. The most common example is an integrator to eliminate static error.
- Central controller $K_0(z)$ design. Closed-loop poles of observer and estimated state feedback have to be fixed and accordingly the matrices L, F are computed. The central controller is then constructed from (2.25) and (2.26) as observer with estimated state feedback. This step is important for final controller complexity and performance. To obtain

a simple high-performance final controller, the central controller should reflect desired closed-loop performance and robustness specifications. These several principal rules are proposed to deal with the choice of desired closed-loop poles:

- use the plant model dynamics (eventually with appropriately attenuated vibration modes) to define the observer poles together with some real multiple auxiliary pole to provide corresponding roll-off.
- to specify the estimated state feedback poles use a complex pair of poles with a damping > 0.75 , at the frequency corresponding to a desired dominant closed loop dynamics and with the multiplicity n_y -number of outputs (maximal multiplicity allowed by assignment algorithms).

Remark: The matrices L and F were computed by the algorithm [KND85, TY96] or [Var00]. In the case of [KND85], sometimes only a small pole modification produces an important changes in solutions (the problem is ill-conditioned), and thus some iteration effort is needed. As we already mentioned, the controller can also be computed using LQG method.

- Set optimisation parameters. The criterion function weights α_i from (2.22) providing an appropriate scaling has to be chosen and the parameterization order N_q of the matrix Q is set. In general, one starts with the order $N_q = 1$ and rises the order if the results are not satisfactory.
- Run optimisation and verify obtained solution. If some obtained results are not satisfactory, one has to modify some of the optimisation parameters, to re-adjust the central controller or to change some design specifications.

Chapter 3

Description of application

3.1 Software requirements

The RMC controller design tool is MATLAB toolbox, and so it can work only in the MATLAB environment. Specifically, the toolbox was programmed for MATLAB version 5.3, but it works also under version 6.0 or version 6.5.

Moreover, the RMC tool requires the standard MATLAB toolboxes and Control System Toolbox.

3.2 Getting started

1. To run RMC toolbox:
 - Run MATLAB version 5.3 or higher.
 - Change the MATLAB current directory on the directory with m-files / p-files of RMC tool (using command `cd` or Path Browser) or add this directory to the PATH variable (for example using `path-browser`).
 - Type "mv" to MATLAB command line
2. To load a model and start design you go to menu bar:
Model→Load model→Matlab format / MidSys format
3. To load a whole design workspace (an example is in the directory "Examples") go to menu bar:
File→Load Environment.
4. To save a whole design workspace go to menu bar:
File→Save Environment.

For any problems contact: hynek_prochazka@yahoo.com.

3.3 Principal window

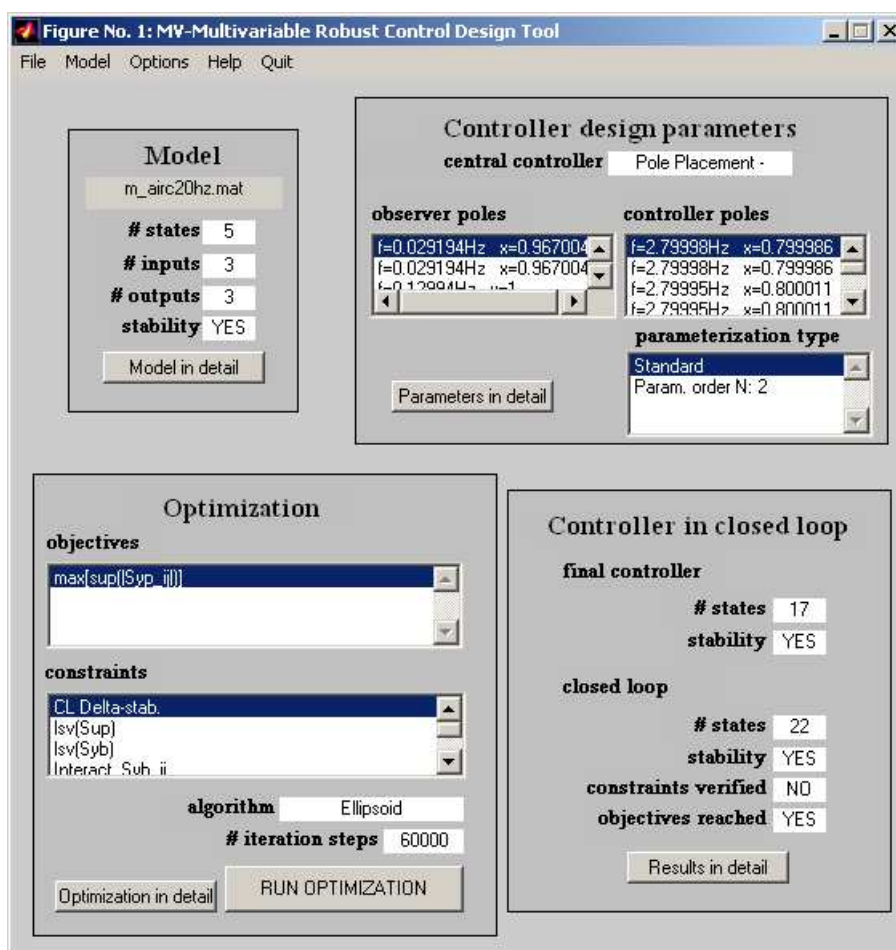


Figure 3.1: Application principal window

The principal window (see Fig.3.1) consists of a bar menu and of four sections:

- *Model section* - It displays the basic properties of the actual model: the name of the model, the number of states, the number of inputs, the number of outputs, and the model stability (yes/no). The button "Model in detail" opens a model analysis window (see description section "Model in detail window" below).

- *Controller parameter section* - This section shows principal properties of central controller and of convex optimisation parametrization. The displayed properties are: evaluation of central controller (pole placement / optimal control), observer poles and state feedback poles, actual parametrization type (standard / Ritz approximation) and the order of parametrization N_q (for Ritz approximation, the parameter z_0 is shown yet). The button "Parameters in detail" opens a corresponding window allowing to modify the central controller and the structure of the matrix Q (see description in the corresponding section below).
- *Optimization section* - In this section, the properties concerning the convex optimisation criterions and the optimisation algorithm are displayed. The section shows: the selected objective criterion functions, the selected constraint criterion functions, the used optimisation algorithm, and the maximal allowed number of optimisation steps. The button "Optimisation in detail" opens a window for treating the criterion functions and for setting the parameters of the optimisation algorithm. The button "RUN OPTIMIZATION" runs controller optimisation.
- *Closed loop and result section* - Some basic information about the final controller, the closed loop and the optimisation results are displayed in this section: number of controller states, controller stability (yes/no), number of closed loop states, stability of the closed loop, and if the constraints and objectives are satisfied (yes/no). The button "Results in detail" opens "Results window" allowing to analyze the obtained controller and the closed loop properties. For details on this window see the corresponding section below.

3.3.1 Menu bar

The following options are available in the bar menu:

- *File* - This option makes access to the following three functions:
 - "Load Environment" allows to load any saved controller design parameter set and make it actual.
 - "Save Environment" saves all current design parameters to one file and so any design stage can be recorded.
 - "Get actual opt.vec." loads an optimization vector from "OptVec.x" file. This file is a result of optimization algorithm "ocpp.exe" (in toolbox named "C_Ellipsoid").

- *Model* - In this option one can use the following functions :
 - "New model" allows to generate a new model with desired number of inputs, outputs and states. It opens a dialog window asking the desired values of these numbers.
 - "Load Model" gives a possibility to choose a format of multivariable model ("Matlab format" or "Midsys format") and then it loads a model in the chosen format. The Matlab format model file is a standard Matlab workspace containing the state space matrices of the model: A , B , C , and D and sampling time T_s . The Midsys format model file is a file saved in Midsys multivariable identification software package.
- *Options* - The only item here is "Scales". It allows to change the scales of frequency response graphs. It opens a small dialog window, where desired scale is chosen for X axe (in hertz-"Hz", in rad/s-"rad/s", and normalized from 0 to 0.5 - "ratio f/fs") and for Y axe ("deciBell" or "linear").
- *Help* - opens user's guide in format pdf.
- *About Toolbox* - opens a small window with some basic facts about the toolbox.

3.4 Model in detail window

This dialog window shows principal properties of a multivariable model actually used for controller design (see Fig. 3.2). It also allows to modify the actual model. The purpose of each graphical object in the window is (the objects are described from up to down and then from left to right):

- popup selector in the upper-left corner shows actual model used for controller design. If several model is loaded, this popup allows to choose, which model is actual.
- "observability", "controllability", "stability" - they denote if these properties are true or not for actual model, i.e. if the model is observable, controllable and stable.
- "poles" list, "transmission zeros" list - they displays all actual model poles and transmission zeros either in real + imaginary part form or in natural frequency and damping continues equivalent form.

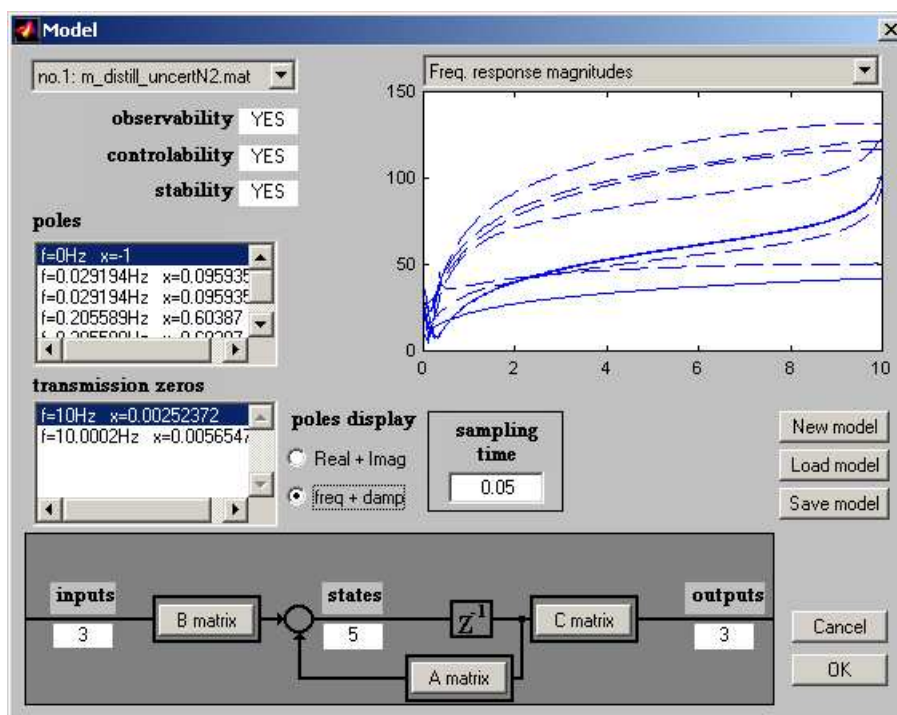


Figure 3.2: Model analysis window

- "poles display" radio buttons define the form of pole/zero displaying: real + imaginary part, or natural frequency + damping.
- "inputs", "outputs", "states" shows the number of inputs, outputs and states for actual model.
- "A matrix", "B matrix", "C matrix" buttons open a dialog window allowing to see coefficients of model state space matrix and to modify these coefficients.
- the popup selector above the graph allows to choose what kind of model characteristic (pole-zero map, frequency response maximal singular value, frequency response module or phase) is displayed in the graph.
- "sampling time" edit line display actual sampling time and allows to change it.
- "New model", "Load model" buttons have the same effect as the items of menu-bar option "Model". The button "Save model" saves actual

model in Matlab format.

- "OK" button confirm all changes made on actual model.

3.5 Parameters in detail window

The upper of the window concern central controller design, the lower part is related to the controller parametrisation.

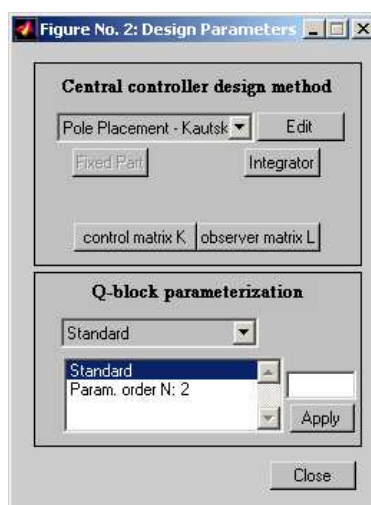


Figure 3.3: Design parameters window

3.5.1 General description

In the upper part the popup selector determine what kind of central controller design method will be applied ("Pole placement - Kautsky" [KND85], "Pole placement - Tits" [TY96], "Pole placement - Varga" [Var00]) and by pressing the "Edit" button a Pole placement window is opened (see below). The last popup choice: "any observer+state feed." consider any observer and state feedback matrix loaded by pressing the button "Edit".

The button "Integrator" opens a small dialog window allowing to set an integrator to any of the channel and to choose the position of the integrator from 0.95 (almost integrator) to 1.0 (pure integrator).

The buttons "st.feed. matrix F" and "observer matrix L" open respectively a dialog displaying current state feedback or observer matrix.

The lower part (parametrization) popup selector determine what structure of Q matrix will be used ("Standard" structure, "Ritz approximation" based structure).

The list display parameters of actual Q structure. In the case of standard structure it is parametrization order N_q , for Ritz approximation it is parametrization order N_q and the parameter z_0 (see theory). The list together with the button and the edit line on the side allow to modify current parametrization parameters. It suffices to select a parameter (its value will appear in the edit line), then change the current value and press the "Apply" button.

3.5.2 Pole placement window

The pole placement window serves to multivariable pole placement design of central controller. The method consist of selecting closed loop poles and of optimizing closed loop eigenvectors. The graphical objects have following significance (see Fig. 3.4):

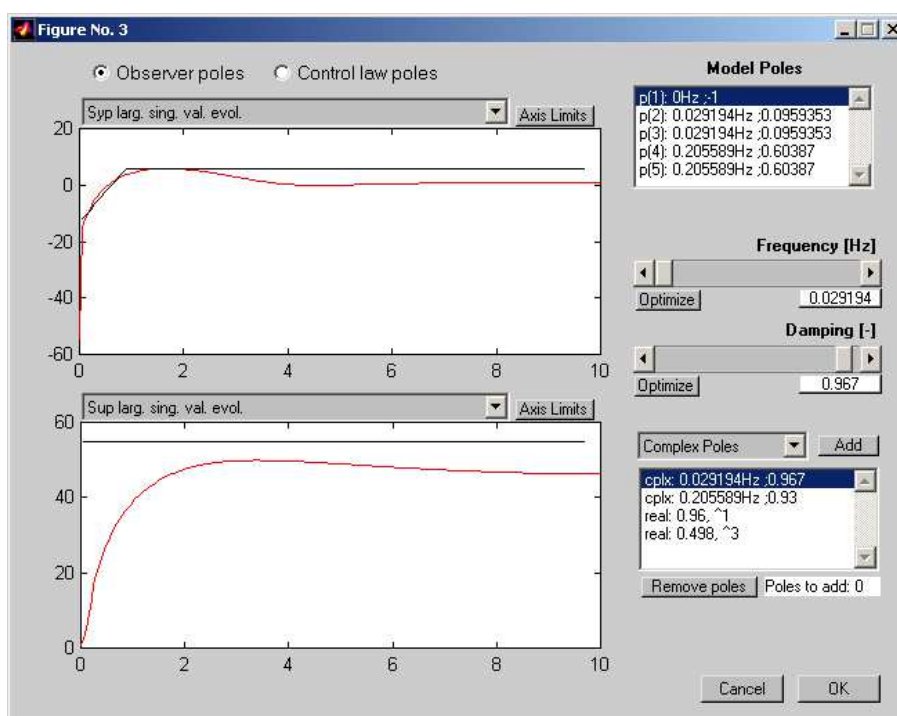


Figure 3.4: Multivariable pole placement window

- The radio buttons "Observer poles" and "Control law poles" select what closed loop poles are to be treated: observer closed loop poles or state feedback closed loop poles.
- The two graphs allows to display graphically two closed or controller properties simultaneously. The popup selector above every graph determines what property is displayed, "Axis Limits" button set the axes ranges. The displayed properties are sensitivity frequency responses: $S_{yp}(z^{-1})$, $S_{yb}(z^{-1})$, $S_{up}(z^{-1})$, $S_{yd}(z^{-1})$, $S_{ud}(z^{-1})$, $S_{uGP}(z^{-1})$, its largest singular values, closed loop poles, controller poles.
- "Model poles" list displays all model poles. If a model pole is selected in this list, then if the same type (real pole, or complex pair) of desired closed loop pole is added, the selected model pole will appear as a new desired CL pole.
- "Frequency [Hz]" and "Damping [-]" sliders and buttons with the current value allows to modify selected desired closed loop poles, i.e. its natural frequency, damping or in the case of real pole its position and multiplicity. The sliders modify the values continuously, by pressing the button with actual value the value can be changed through an edit line.
- At the right-bottom corner there is a desired closed loop pole list. The selected pole/s can be modified by sliders/buttons (see above). If a new desired pole has to be added, one choose the type of the pole in upper popup selector ("Complex Poles" adds a pair of complex poles, "Multiple Pole" adds one real pole), then press "Add" button. To Remove a pole, select the pole in the list and press "Remove poles" button.
- The value "Poles to add:" indicate the number of non-assigned poles which can be added yet. If some poles are not assigned, they are automatically placed around the origin.
- "OK" button apply the designed central controller.

3.6 Optimisation in detail window

3.6.1 General description

In this window sensitivity templates, weighting coefficients and optimization algorithm parameters are set and can be adjusted, see Fig. 3.5.

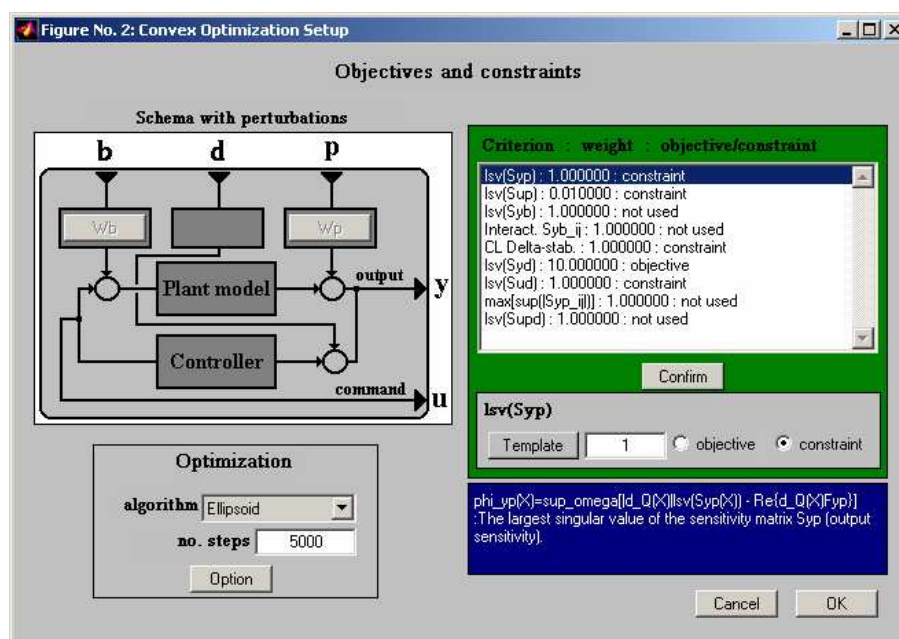


Figure 3.5: Optimisation parameters window

The left-bottom section "Optimisation" concerns optimisation algorithm. It contains the following objects:

- The popup selector "algorithm" determines what ellipsoid optimisation algorithm will be employed: "Ellipsoid" is programmed in Matlab. "C_Ellipsoid" is a .exe routine and it has to be run from system but first "RUN OPTIMISATION" button has to be pressed to generate initializing file. Only S_{yp} , S_{yb} , S_{up} sensitivity can be calibrated with this algorithm. "Ellips.Denom.Optim" optimize only denominator d_Q of the matrix Q .
- The edit line "no.steps" determine the maximum number of optimisation steps.
- The button "Option" opens a dialog for optimisation algorithm parameters, see Fig. 3.6. In this window the gradient size, the stopping criterion value, the initial ellipsoid size and step for re-displaying the criterion during optimisation are displayed and can be modified. Also one can choose the type of multi-criterion function ("max" is maximum function, "sum" is sum of criterion functions) and if the initial vector is the current vector or it is set randomly. Finally, current optimized

vector is displayed in a list.

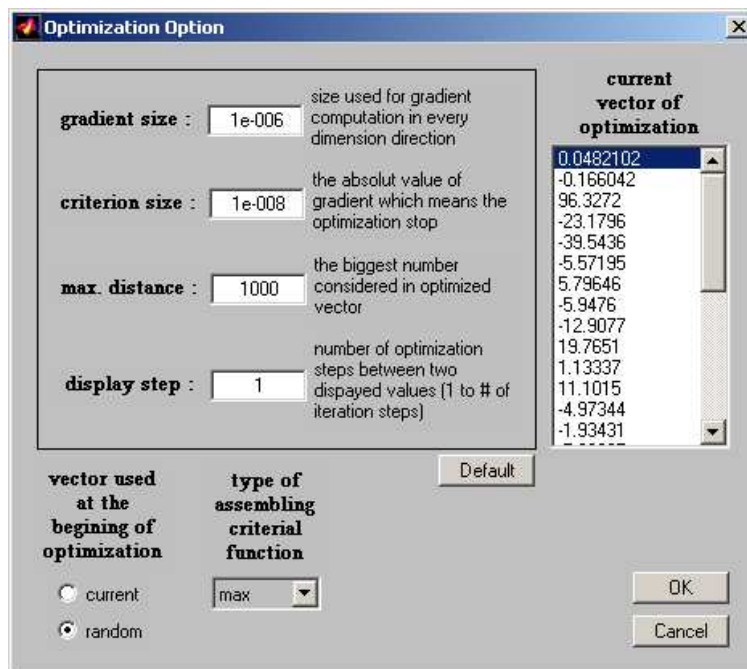


Figure 3.6: Optimisation options window

The right side of the window concerns template and weighting coefficient setting. The objects signifies the following:

- The list "Criterion : weight : objective/constraint" displays all accessible criterion functions.
- The button "Template" or "Region" under the list opens the window to modify appropriate template or δ -region (in the case of "CL Delta-stab." criterion function). To see descriptions on these windows see next subsections.
- The edit line on the side of "Template" button allow to modify weighting coefficient.
- "objective" and "constraint" radio buttons determine if the criterion function is objective, constrain or unused.
- "Confirm" button apply chosen weighting coefficient and objective/constrain selection.

- the lowest text area contains some description of actually selected criterion function.

3.6.2 Template window

In this window a shape of corresponding sensitivity templates on sensitivity frequency responses can be adjusted, see Fig. 3.7. The template consists of N points which are limits of response magnitude or of response largest singular value. The sections in the window has the following functions:

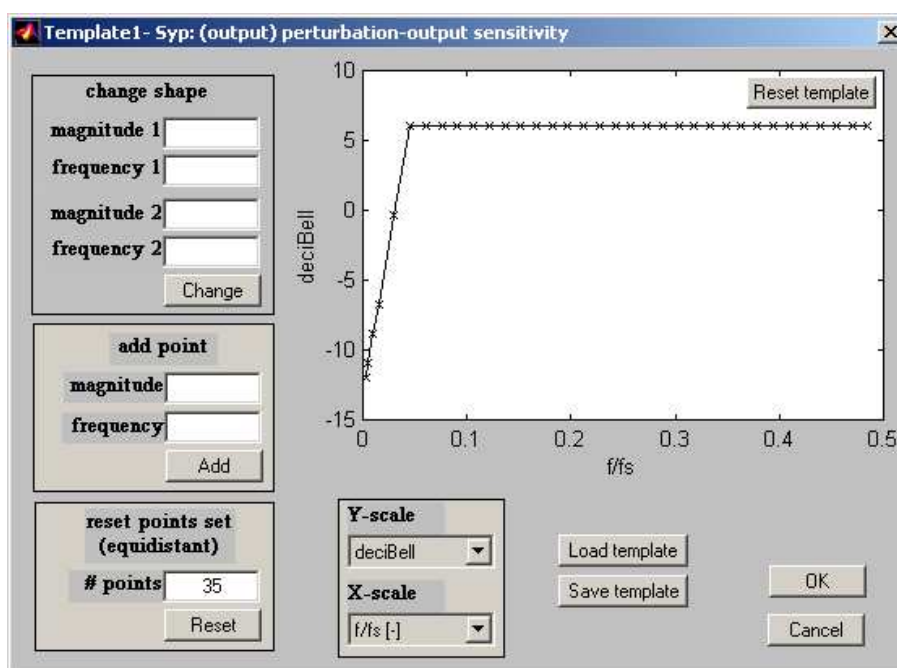


Figure 3.7: Template window

- "change shape" section allows to make a linear interpolation from frequency point 1 (defined by magnitude 1 and frequency 1) to frequency point 2 (defined by magnitude 2 and frequency 2).
- "add point" section allows to add one new point at a specific frequency and with a specific magnitude.
- "reset points set (equidistant)" section create a new template with a desired number of equidistantly distributed poles.

- the graph displays actual template shape.
- "Reset template" button set the template to a default shape.
- "Y-scale" and "X-scale" popup allows to choose desired X and Y graph scale.
- "Load template" and "Save template" buttons loads and saves the template in Matlab format.

3.6.3 δ -region window

The δ -region window allows to modify δ -region (a circle with rayon r and center c) for δ -stability criterion function. The center of the δ -region circle can be set by "coordinate X" edit line, the radius by "radius" edit line. To apply the changes, the button "Change" has to be pressed. The graph display actual δ -region and unit circle with natural frequency - damping grid. The button "Reset region" set δ -region to a default form.

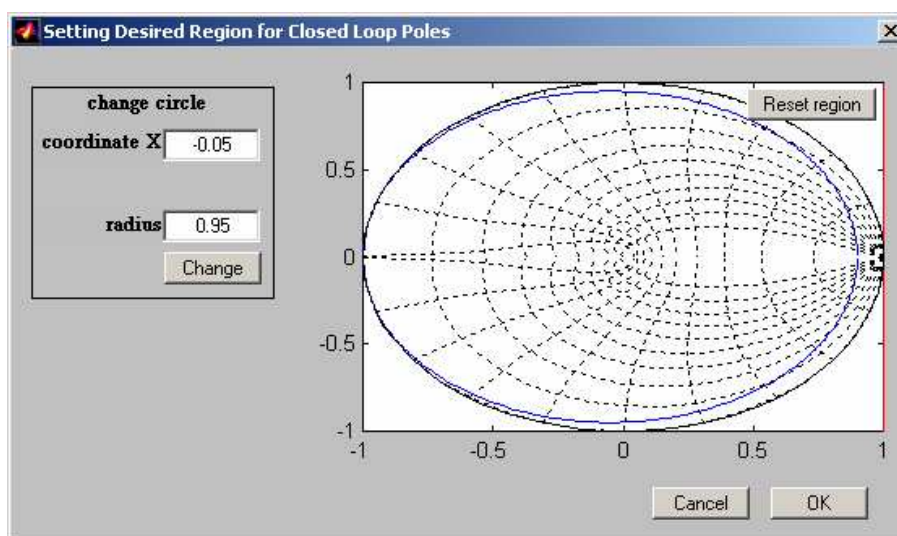


Figure 3.8: δ -region window

3.7 Closed loop in detail window

From this window all essential closed loop and controller characteristics can be displayed. The window consists of following sections:

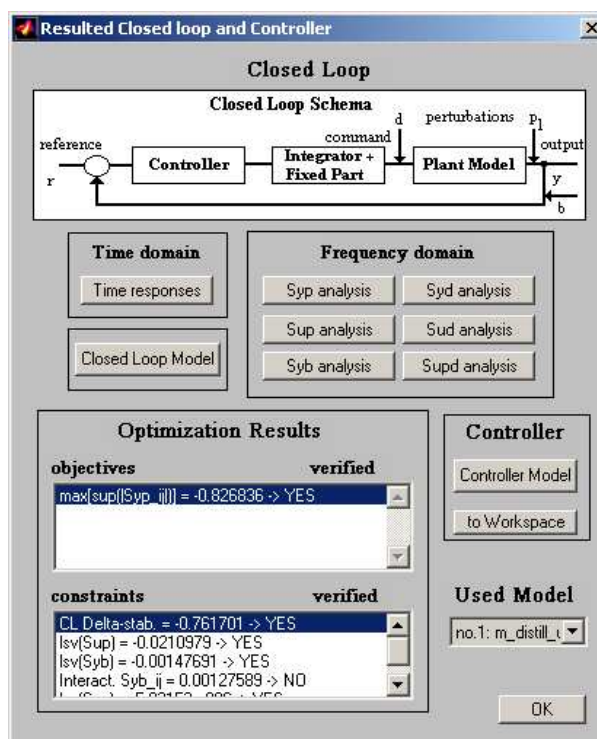


Figure 3.9: Controller and closed loop results window

- "Closed loop" section displays considered closed loop configuration.
- "Time domain" section contains "Time responses" button which opens time simulation window. This window displays some closed loop time responses, for details see subsection Time simulation window below.
- "Closed Loop Model" button opens a transfer function analysis window allowing to see closed loop system properties (for the analysis window description see subsection Transfer matrix analysis window).
- "Frequency domain" section contains 6 buttons (every button corresponds to one sensitivity). The buttons open a sensitivity response analysis window allowing to analyze corresponding sensitivity (see detailed description of the window in next subsection).
- "Controller" section contains "Controller Model" button which opens transfer matrix analysis window for actual controller analysis (see subsection describing the analysis window). The button "to Workspace"

transfer actual controller to Matlab workspace in state space representation form, hence the state space matrices A_k , B_k , C_k , D_k appear in workspace.

- "User model" popup selector allows to choose which model (from loaded models) is used for sensitivity characteristic evaluation and time response evaluation. This allows to verify how the computed controller works on other than nominal model.

3.7.1 Sensitivity response analysis window

This window (see Fig. 3.10) display frequency responses for chosen sensitivity S_{ij} together with fixed templates. The following graphical objects appears in the window:

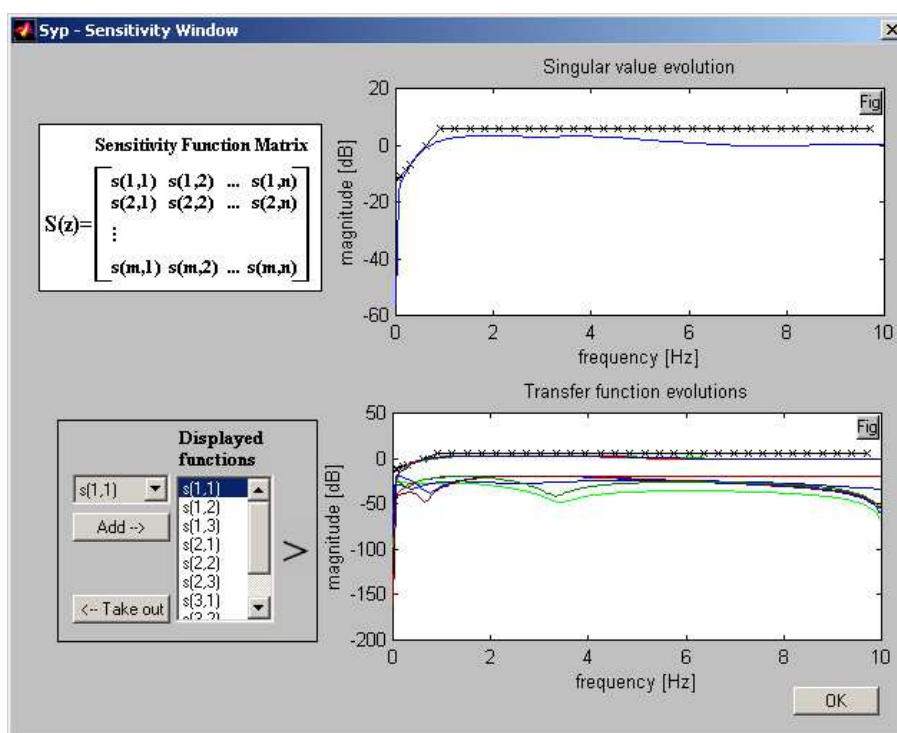


Figure 3.10: Sensitivity matrix window

- The upper "Singular value evolution" graph displays the largest singular value of sensitivity frequency response: $\bar{\sigma}(S_{ij}(e^{-j\omega_d})) \mid \forall \omega_d$ and

templates $F_{ij}(\omega_d)$. The small button "Fig." in the upper-right corner opens a new Matlab figure containing only the graph.

- The lower "Transfer function evolutions" graph displays template $F_{ij}(\omega_d)$ and sensitivity modules of multivariable frequency response:

$$|S_{ij}(e^{-j\omega_d})| = \begin{bmatrix} |s_{(1,1)}(e^{-j\omega_d})| & |s_{(1,2)}(e^{-j\omega_d})| & \dots \\ |s_{(2,1)}(e^{-j\omega_d})| & |s_{(2,2)}(e^{-j\omega_d})| & \dots \\ \vdots & & \\ |s_{(m,1)}(e^{-j\omega_d})| & \dots & |s_{(m,n)}(e^{-j\omega_d})| \end{bmatrix} \forall \omega_d \quad (3.1)$$

where $|s_{(k,l)}(e^{-j\omega_d})|$, $k = 1 \dots m$, and $l = 1 \dots m$ is frequency response module of one transfer function. The small button "Fig." in the upper-right corner opens a new Matlab figure containing only the graph.

- The list "Displayed functions" on the left shows which frequency responses of multivariable frequency response are displayed in "Transfer function evolutions" graph. To clear any response, it has to be chosen in the list and "i-Take out" button has to be pressed. To add a cleared response to the graph, the response has to be selected in the popup selector on the left of the list and "Add-i" button has to be pressed.

3.7.2 Transfer matrix analysis window

This window (see Fig. 3.11) is very similar to the model analysis window described in section 3.4. It allows to analyze some essential properties of a transfer matrix (controller or closed loop). For detailed description see the mentioned section.

3.7.3 Time simulation window

In this window some simple time simulations of multivariable closed loop system can be realized, see Fig. 3.12. the graphical objects have the following purpose:

- The list in "Simulation Settings" section on the left displays all n_y (number of outputs/references) closed loop reference signals. A type of the signal selected in the list is determined by the popup selector above the list. The signal types are: "constant", "step", "ramp", "pulse", or "trapezoid". Every signal type has its properties which can be modified by pressing "Properties" button.

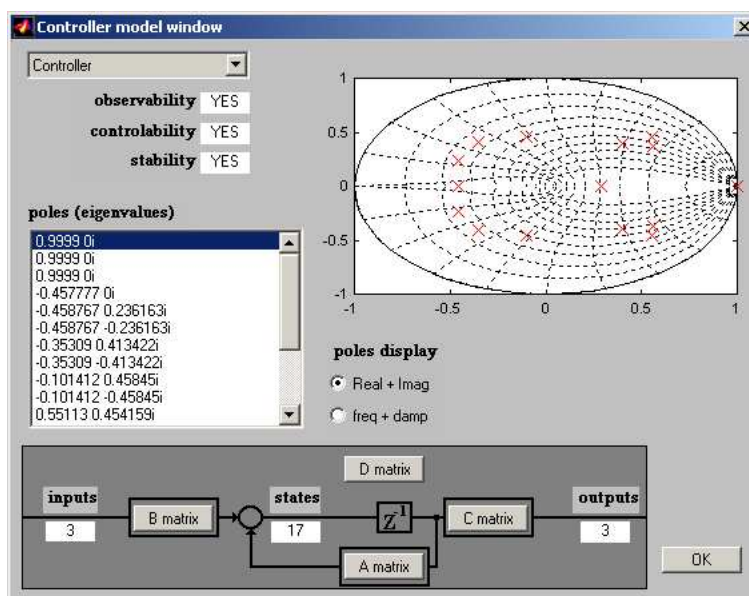


Figure 3.11: Transfer matrix analysis window

- "simulation times [s]" edit line displays actual simulation time and it can be change in this line.
- The list "current element of TF matrix" lists all reference-output channels of the closed loop. An output time response of selected channel is plotted in the graph below.
- "Display All" button plots all output time responses simultaneously.
- "Run Simulation" button activate time simulation re-computing and re-displaying. After opening the window, this button has to be pressed to see simulation results.
- The graph at the bottom shows desired output time responses and reference signal.

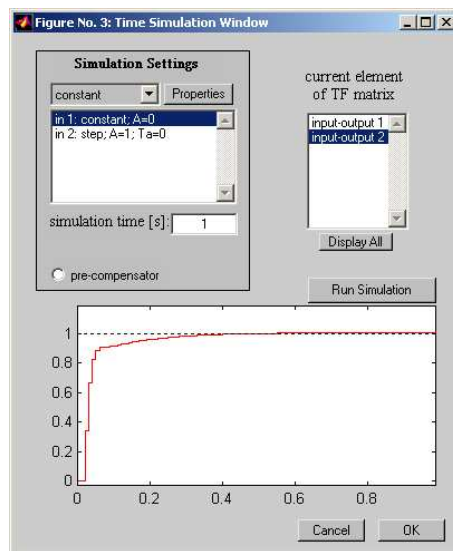


Figure 3.12: Closed loop time simulation window

Bibliography

- [BB91] S.P. Boyd and C.H. Barratt. *Linear controller design - Limits of performance*. Prentice Hall, New Jersey, 1991.
- [GL95] M. Green and D.J.M. Limbeer. *Linear robust control*. Prentice-Hall, New Jersey, 1995.
- [KND85] J. Kautsky, N.K. Nichols, and P. Van Doren. Robust pole assignment in linear state feedback. *Int. J. Control*, 41(5):1129–1155, 1985.
- [Kuc91] V. Kucera. *Analysis and Design of Discrete Linear Control Systems*. Academia, Prague and Prentice Hall International, London, 1991.
- [Lan98] J. Langer. *Synthèse de régulateurs numériques robustes - Application aux structures souples*. PhD thesis, Institut National Polytechnique de Grenoble, 1998.
- [LK98] I.D. Landau and A. Karimi. Robust digital control using pole placement with sensitivity function shaping method. *Int. J. Robust and Nonlin. Cont.*, 8:191–210, 1998.
- [LL99] J. Langer and I.D. Landau. Combined pole placement/sensitivity function shaping method using convex optimization criteria. *Automatica*, pages 1111–1120, 1999.
- [Mac89] J.M. Maciejowski. *Multivariable Feedback Design*. Addison-Wesley, 1989.
- [PL02] H. Prochazka and I.D. Landau. Logiciel pour l’enseignement et le calcul du placement de pôles robuste. *Conf. Int. Francophone d’Automatique Nante*, pages 694–698, 7 2002.
- [PL03a] H. Prochazka and I.D. Landau. Convex optimisation for robust multivariable controller design as alternative of h_∞ optimisation. *submitted to European Journal of Control*, 2003.

- [PL03b] H. Prochazka and I.D. Landau. Multivariable robust controller design by pole placement and sensitivity shaping. *Internal Rapport, Laboratoire d'Automatique de Grenoble*, 2003.
- [TY96] A.L. Tits and Y. Yang. Globally convergent algorithms for robust pole assignment by state feedback. *IEEE Trans. Autom. Control*, 41(10):1432–1452, 1996.
- [Var00] A. Varga. Robust pole assignment via sylvester equation based state feedback parametrization. *Proc. IEEE International Symposium on Computer Aided Control System Design, CACSD'2000, Anchorage, Alaska*, 2000.
- [YJB76] D.C. Youla, H.A. Jabri, and J. Bongiorno. Modern wiener-hopf design of optimal controllers. *IEEE Trans. on Automatic Control*, AC-21:319–338, 1976.
- [Zho98] K. Zhou. *Essentials of Robust Control*. Prentice Hall, New Jersey, 1998.