



الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieure et de la Recherche Scientifique



جامعة وهران 2 محمد بن أحمد
Université d'Oran 2 Mohamed Ben Ahmed

معهد الصيانة والأمن الصناعي
Institut de Maintenance et de Sécurité Industrielle

Département De Maintenance En Instrumentation

MÉMOIRE

Pour l'obtention du diplôme de Master

Filière : *Génie Industriel*

Spécialité : *Ingénierie de la Maintenance en Instrumentation*

Thème

Commande numérique RST d'un moteur à courant continu via une carte d'acquisition

Présenté et soutenu publiquement par :

MAICHE Ali

KAOUBI Nabil

Devant le jury composé de :

Nom et Prénom	Grade	Etablissement	Qualité
HASSINI Abdelatif	Professeur	IMSI	Président
KACIMI Abderrhmane	MCB	IMSI	Encadreur
DJELGHOUM Farida	MAA	IMSI	Examinatrice

Juin 2016

REMERCIEMENT

*Nous tenons à remercier tout particulièrement et très
Chaleureusement notre encadreur, **Mr Kacimi.A**, avec qui
nous avons eu tant de plaisir à travailler et qui nous a fait profiter
de son expérience. Qu'il trouve ici notre sincère gratitude.
Nous remercions également le personnel du département de génie
électrique de l'ENPO pour nous avoir ouvert les portes de leurs
laboratoires*

*Nous exprimons notre profonde reconnaissance aux membres du
jury d'avoir accepté de juger notre travail.*

*Que tous nos professeurs qui ont contribué à notre formation
trouvent ici notre plus profonde gratitude.*

*Nous adressons nos plus sincères remerciements à tous ceux qui ont
contribué, de près ou de loin, à l'aboutissement de ce travail.*

KAOUBI.N

MAICHE.A

À mes chers parents,

À mes frères,

À mes sœurs, ...

À tous les miens,

Je dédie ce travail.

Àlissou

*À mes chers parents,
À mes frères,
À ma sœur, ...
À tous les miens,*

Je dédie ce travail.

Nabil

Table des matières

Introduction générale.....	1
-----------------------------------	----------

CHAPITRE I : GENERALITE SUR LA COMMANDE NUMERIQUE

I.1 Introduction	4
I.2 Procédés et signaux	4
I.3 Structure des systèmes à commande numérique.....	4
I.4 Discrétisation et choix de la période d'échantillonnage	6
I.4.1 L'échantillonnage d'un signal.....	6
I.4.2 Le choix de la période d'échantillonnage	6
I.4.3 Quantification.....	7
I.4.4 le bruit de quantification	7
I.5 Fonction de transfert de la boucle d'asservissement.....	8
I.5.1 En boucle ouverte.....	8
I.5.2 En boucle fermée.....	9
I.6 La transmittance en présence d'un bloqueur d'ordre zéro	9
I.7 Système comportant un retard pur	10
I.8 Le système dans le cas discret et continu.....	10
I.8.1 Système du 1er ordre.....	10
I.8.2 Système du 2ème ordre	11
I.9 Stabilité du système numérique	12
I.9.1 Critère Nyquist	12
I.9.2 Critère de jury	12
I.10 Précision des systèmes asservis échantillonnés	13
I.11 Conclusion	16

CHAPITRE II : SYNTHSE DE REGULATEUR NUMERIQUE RST

II. 1 Introduction	17
II. 2 Régulateur RST	17
II. 2. 1 Définition.....	17
II. 2. 2 Principe	18
II. 2. 3 Effet de l'intégrateur	19
II. 2. 4 Effet de zéro instable sur le comportement du système.....	20
II. 3 Synthèse des correcteurs RST	21

II. 3. 1 Définition de la fonction de transfert modèle	21
II. 3. 2 Choix des pôles et zéros à compenser	22
II. 3. 3 Calcul des polynômes R, S et T.....	23
II. 3. 4 Résolution de l'équation diophantienne	24
II. 3. 5 Objectifs supplémentaires pour la synthèse.....	25
II. 4 Méthode de placement de pôles.....	26
II. 4. 1 Placement des pôles avec RST	26
II. 4. 1. 1 Structure.....	26
II. 4. 1. 2 Choix des pôles en boucle fermée (Polynôme P)	28
II. 4. 1. 3 Régulation (Calcul de $R(Q-1)$ Et $S(Q-1)$).....	28
II. 4. 1. 4 Poursuite (Calcul de $T(q - 1)$).....	32
II. 4. 2 Poursuite et régulation à objectif indépendant.....	36
II. 4. 2. 1 Structure.....	38
II. 4. 2. 2 Régulation (Calcul de $R(z - 1)$ Et $S(z - 1)$)	39
II. 4. 2. 3 Poursuite (calcul de $Tq - 1$)	43
II. 4. 3 Effets des saturations – dispositif anti-saturation	45
II. 4. 4 Avantages et inconvénients du régulateur RST.....	48
II. 4. 4. 1 Avantages.....	48
II. 4. 4. 2 Inconvénient.....	48
II. 5 Conclusion	49

CHAPITRE III : MODELISATION ET IDENTIFICATION D'UN MOTEUR A COURANT CONTINU

III.1 Introduction	51
III.2 Description d'un moteur à courant continu.....	51
III.2.1 Introduction	51
III.2.2 Fonction d'usage	51
III.2.3 Principe.....	52
III.3 Modélisation d'un moteur à courant continu	53
III.3.1 Généralité sur la commande en vitesse variable des machines électriques.....	53
III.3.2 Contrôle de vitesse d'une machine à courant continu (Opération en boucle fermée).....	54
III.3.3 Modélisation de la machine à courant continu (modèle électro-magnéto-mécanique)	54
III.4 Identification des modèles dynamiques d'un moteur à courant continu.....	57
III.4.1 Principe de l'identification des modèles dynamiques de procédés	57
III.4.2 Le principe de l'algorithme des moindres carrés	60
III.4.3 Méthode d'identification par moindre carré.....	61

III.4.4 Cueillette des entrées sorties du système.....	62
III.4.5 Choix de l'ordre de système.....	62
III.4.6 Calcul des paramètres du modèle.....	63
III.5 Conclusion.....	65

CHAPITRE IV : APPLICATION DE LA COMMANDE RST SUR LE SYSTEME feedback MS 150

IV.1 Introduction	66
IV.2 Système dSPACE.....	67
IV.2.1 Configuration requise.....	67
IV.2.2 Carte dSPACE.....	67
IV.2.3 Control Desk	69
IV.3 Identification	70
IV.3.1 Système à identifier (maquette feedback MS150)	70
IV.3.2 Qu'est-ce que la SBPA ?.....	71
IV.3.3 Génération d'un signal SBPA	72
IV.4 Identification de la transmittance du système étudié par la méthode erreur de sortie utilisant une SBPA	73
IV.4.1 Essais	80
IV.4.2 Validation du modèle	82
IV.5 Application de la commande numérique sur le système identifié	84
IV.5.1 Synthèse de la loi de commande (placement de pôles).....	84
IV.5.2 Simulation de la commande sous simulink:.....	86
IV.6 Commande en temps reel sous la plateforme dSPACE / control desk	88
IV.6.1 Réalisation du schéma Simulink	88
IV.6.2 Simulation du régulateur RST avec le dispositif anti-saturation.....	89
IV.6.3 Réalisation du schéma bloc à implémenter sous le système dSPACE.....	90
IV.6.4 Lancement de la commande en pratique	92
IV.7 Conclusion	94

Conclusion générale.....	95
---------------------------------	-----------

Bibliographie.....
---------------------------	--------------

Annexe.....
--------------------	--------------

INTRODUCTION GENERALE

L'automatique est une science de l'ingénieur qui a pour but majeur de résoudre des problématiques d'analyse, de modélisation et d'identification en vue d'élaborer des lois de commande adéquates. En d'autres mots de l'automatisation de tâches par des machines fonctionnant sans intervention humaine. Une loi de commande peut être développée et appliquée en boucle ouverte à partir d'un signal d'entrée, cependant la stabilité, l'amélioration des performances et le rejet des perturbations externes qui affectent les systèmes dynamiques sont loin d'être garantis, voir difficile à obtenir parce que la loi de commande ne prend pas en compte les changements qui affectent la sortie du système à commander durant son fonctionnement, on dit que le correcteur est aveugle par rapport à l'environnement du système. Ce qui n'est pas le cas en boucle fermée, grâce au retour de sortie, la loi de commande et par l'intermédiaire de l'erreur entre la valeur de la sortie à réguler et la valeur de la consigne peut agir continuellement sur le système grâce à un signal de commande adéquat et à compenser tout effet causé par un changement éventuel sur le système durant son fonctionnement.

La loi de commande est générée par un système de commande appelé correcteur ou régulateur. Dans le domaine continu, sa mise en œuvre est une réalisation physique avec des circuits électriques ou électroniques analogiques et dans le domaine discret sa mise en œuvre est un algorithme à implanter dans un ordinateur numérique.

De nos jours, grâce aux développements de l'automatique et de l'informatique, la plupart des lois de commande sont implémentées sur des micro-ordinateurs ou processus numériques.

L'implémentations d'un algorithme de commande sur l'ordinateur en comparaison à une réalisation analogique, offre de nombreux atouts parmi lesquels, un coût faible, une précision élevée, une insensibilité au bruit, une facilité d'implémentation, une portabilité et une souplesse par rapport aux modifications.

Dans l'industrie parmi les algorithmes les plus utilisés dans la commande numérique, on trouve le PID numérique 1 et 2, le placement de pôles avec ses deux méthodes les plus répondues, la méthode polynomiale basée sur les polynômes RST avec toutes ses variantes et la méthode de commande par retour d'état ou retour de sortie.

Dans le but d'une maîtrise recherchée de cet aspect théorique et pratique de cette commande numérique des systèmes tant sollicitée en industrie, l'objectif principal de ce travail est de rassembler et présenter un ensemble de méthodes de synthèse de régulateurs polynomiaux par placement des pôles les plus perfectionnées de la littérature spécialisée et les plus utilisées dans l'industrie, en essayant d'exhiber leurs avantages et leurs défauts par des applications simples réalisées par simulations sous l'environnement Matlab. Notre travail s'articule essentiellement sur une méthode moderne de synthèse de la commande a priori des régulateurs polynomiaux, connue sous le nom de la méthode de Landau [1]. La méthode proposée unifie et complète en même temps la synthèse d'une loi de commande par placement des pôles avec des objectifs de poursuite et de régulation.

Les lois de commandes développées seront appliquées à la commande d'un moteur à courant continu, un système non linéaire pour lequel on présente une étude permettant d'aboutir à la synthèse d'une loi de commande linéaire, pour cela une modélisation classique abondamment utilisée dans la littérature aboutissant à un modèle linéaire du système sera présentée, mais les simulations seront développées sur un modèle paramétrique sous forme d'une fonction de transfert en z (discrète) obtenu par identification réalisée dans un laboratoire du moteur à courant continu en question.

Ce mémoire est divisé en quatre chapitres :

Le premier chapitre sera consacré aux notions de base de la commande numérique, entre autre le choix de la période d'échantillonnage qui est très important pour la conversion des signaux entrants et sortants du système ainsi que l'analyse du système corrigé soit en boucle ouverte ou en boucle fermée en terme de stabilité et de performances, et les différents critères utiles à cette analyse tel que le entrée de Nyquist et le critère de Jury pour la stabilité du système et l'analyse gain/phase de Bode .Quelques notions sur la conversion analogique/numérique (la quantification) et la conversion numérique /analogique (le bloqueur d'ordre zéro) sont présentées ainsi que l'environnement matériel assurant le fonctionnement d'un système continu piloté par un calculateur dans une commande numérique .

Le second chapitre de ce mémoire est voué aux régulateurs numériques structuré à trois branches plus particulièrement les régulateur RST On examine ensuite les méthodes suivantes :

-Placement de pole et des zéro avec poursuite et régulation objectifs.

-Placement de pole avec poursuite et régulation à objectif indépendant, qui permettent de traiter des systèmes de n'importe quel ordre avec ou sans retard pur. Le dernier paragraphe est consacré à l'effet de saturation de l'actionneur, qui peut être gênant en particulier si le régulateur contient un intégrateur. On a aussi imposé une dynamique de sortie de saturation différente d'un simple gain pour éliminer cet effet.

Le troisième chapitre est consacré à la modélisation du moteur à courant continu. Le système sur lequel on a choisi de développer et appliquer les différentes méthodes de la commande numérique par le régulateur RST déjà développer dans le chapitre précédant ainsi qu'un aperçu sur les principes de base de l'identification des méthodes dynamique de procédé. On présente aussi la méthode d'acquisition des entrée /sortie sous un protocole d'expérimentation, l'estimation des paramètres du modèle pour un certain ordre choisi selon des connaissances apriori sur le système, puis la validation du modèle. Dans ce même chapitre, nous donnerons un aperçu bref sur les principaux types d'algorithme d'adaptation paramétrique intervenant dans les méthodes d'identification paramétrique plus particulièrement l'algorithme des moindres carrés.

Dans le quatrième et dernier chapitre purement pratique de ce travail nous effectuerons l'implantation d'un algorithme de commande du moteur à courant continu de la maquette *FeedBack MS150* à travers un *DSP* (Digital Signal Procesor) *dSPACE 1104* basée sur un régulateur numérique RST suivant la méthode de placement de pôles afin de commander notre moteur en vitesse. Nous réaliserons aussi sous l'environnement *Matlab/Simulink,dSPACE/ControlDesk* d'une manière pratique l'identification du modèle de comportement de notre système. Apres implémentation nous comparerons les résultats des simulations développées sous *Simulink* avec les résultats de la pratique récupérés sous *ControlDesk*. Des figures et des interprétations des résultats seront fournis en fin de ce chapitre.

Nous terminons ce mémoire par une conclusion générale sur le travail présenté ainsi que les perspectives permettant sa continuation qui servira de support pour d'autres désirant le développement et l'amélioration de ce travail.



I.1 Introduction

Dans ce chapitre, on va introduire les structures de base et le fonctionnement de la commande numérique ainsi que les différentes classes de systèmes dynamiques.

I.2 Procédés et signaux

Un procédé est décrit comme une boîte noire qui introduit une relation entre deux catégories de signaux :

- Les signaux d'entrées ou signal de commande du procédé.
- Les signaux de sorties/sortie commandées.

Le cas le plus simple est celui des procédés mono-variables (SISO) qui est étudié sans un premier temps, la généralisation se fait aux procédés plus complexes. La représentation d'un procédé mono-variable est donc le suivant :



Figure I.1 : La représentation d'un procédé

Dans tout le manuscrit l'entrée ou commande de procédé est notée par $u(t)$ et la sortie par $y(t)$.

I.3 Structure des systèmes à commande numérique

Le pilotage par ordinateur des procédés physique, notamment leur asservissement, est de plus en plus utilisé dans le milieu industriel pour des raisons de coûts et de rapidité d'implémentation. La commande numérique du procédé est illustrée par la figure I.2

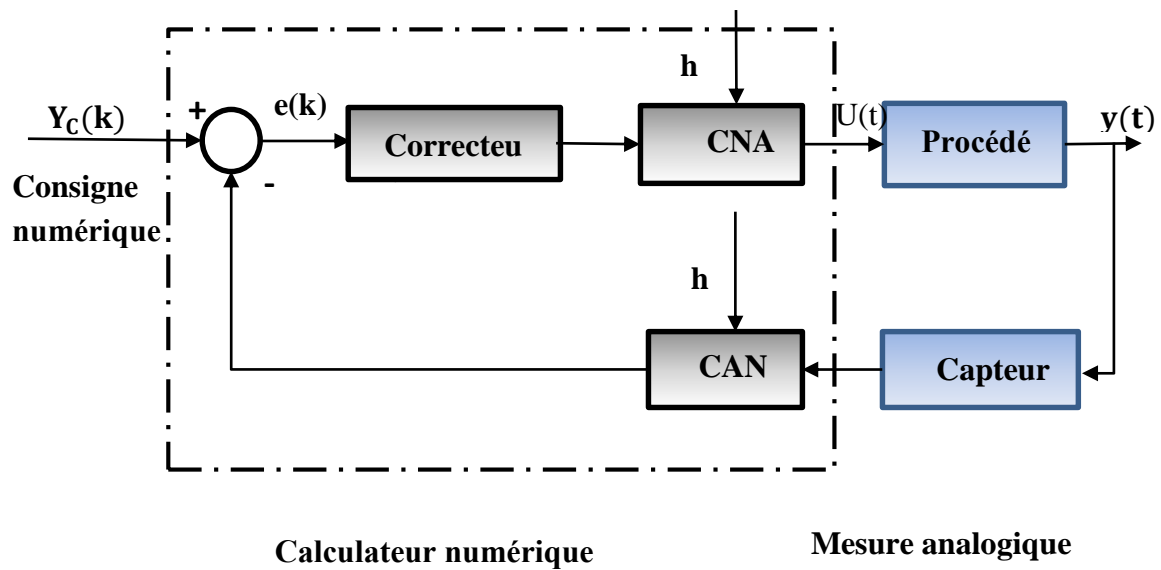


Figure I.2 : Structure de la commande numérique

Le principe d'un système numérique est de remplacer la commande analogique du système par des algorithmes mis en œuvre sur ordinateur. Le ordinateur nécessite un certain temps pour effectuer ces opérations, on introduit alors un découplage temporel des signaux au niveau du ordinateur. Les systèmes à commande numérique considérés présentent donc un caractère hybride, temps continu – temps discret .

Par conséquent, il est nécessaire de réaliser une interface entre le ordinateur et le procédé et cette interface est obtenue à l'aide :

- D'un convertisseur numérique-analogique (CNA) pour convertir les signaux numériques issus du ordinateur dans des signaux analogiques constituant l'entrée du procédé.
- D'un convertisseur analogique-numérique (CAN) pour convertir les mesures effectuées sur le procédé et les fournir au ordinateur. Il peut arriver que le capteur soit lui-même discret et qu'il n'y ait donc pas de conversion analogique-numérique à faire.

I.4 Discrétisation et choix de la période d'échantillonnage

I.4.1 L'échantillonnage d'un signal

L'échantillonnage consiste à transformer un signal analogique (continu) en un signal numérique (discret), en capturant des valeurs à intervalle de temps régulier (ici temps est à prendre au sens large et s'applique à tout signal). C'est une étape nécessaire pour pouvoir enregistrer, analyser et traiter un signal par ordinateur, car celui-ci ne peut traiter que des nombres.

Il faut distinguer l'échantillonnage de la quantification, mais ce sont toutes les deux des étapes nécessaires à la numérisation d'un signal.

La fréquence à laquelle les valeurs sont capturées est la fréquence d'échantillonnage, appelée aussi cadence d'échantillonnage, ou taux d'échantillonnage, exprimée en Hertz (Hz).

I.4.2 Le choix de la période d'échantillonnage

le choix de la période d'échantillonnage dépend de la bande passante du système à échantillonner : fréquence de coupure f_c .

La règle adoptée en régulation numérique [1] est la suivante $f_e = 6 \text{ à } 25 f_c$, dont f_e est la fréquence d'échantillonnage.

f_c est la fréquence de coupure du système en boucle fermé, donc la fréquence de coupure désirée.

On peut traduire cette formule dans le cas simple où le système en boucle fermé est assimilé à un premier ou à un second ordre : c'est ce que l'on fait en pratique lorsqu'on choisit la fonction de transfert désirée en boucle fermé.

$$\text{➤ Premier ordre : } \frac{\tau}{4} < T_e < \tau, \quad f_c = \frac{1}{2\pi\tau} \text{ tel que}$$

T_e est la période d'échantillonnage. Ceci correspond à une réponse indicielle qui dure 2 et 10 échantillons environ.

τ est la constante de temps.

- Second ordre : on peut se référer à la pulsation propre ω_0 et au facteur d'amortissement ξ

Exemple : $\xi = 0.8$ à $0.9 \Rightarrow \omega_c = 0.8 \omega_0$

Ordre de grandeur commode : $\omega_0 T_e = 1$

On peut aussi choisir notre période d'échantillonnage à partir du temps de réponse et du nombre d'échantillons désirés : c'est le cas d'une réponse en temps fini.

Exemple : t_r désiré = 6s, réponse en trois échantillons, la période d'échantillonnage sera $T_e = 2s$.

Une règle encore plus simple consiste à choisir $T_e = \frac{trd}{5 \text{ à } 10}$ ou trd est le temps de réponse désiré.

Le théorème de Nyquist-Shannon, énonce que la fréquence d'échantillonnage d'un signal doit être égale ou supérieur au double de la fréquence maximale contenue dans ce signal, afin de convertir ce signal d'une forme continu à une forme discrète (discontinue dans le temps). Ce théorème est à la base de la conversion analogique-numérique des signaux.

I.4.3 Quantification

La figure I.3 montre la différence fondamentale entre le signal analogique et le signal échantillonné et quantifié que le microcontrôleur perçoit. Ce signal est en fait une suite de nombres codés sur n bits (typiquement 8 et 16 bits)

En observant le signal échantillonné (en bas à droite de la figure I.3) la tentation est grande de vouloir s'approcher du signal analogique en essayant de faire du « quasi-continu » par un surdimensionnement des deux paramètres principaux qui sont la fréquence et le pas de quantification.

I.4.4 le bruit de quantification

Pour un signal sinusoïdal d'amplitude égale à la dynamique du convertisseur, le rapport signal à bruit SNR est de :

$$\text{SNR} = 6n / 4.76 \quad [I.1]$$

Où n est le nombre de bits du convertisseur

En pratique on choisit n tel que le rapport signal à bruit de l'ensemble ne soit pas trop diminuer par la quantification. Il existe des convertisseurs de : 8 ,10 ,12 ,16 ,24 bits. Il existe aussi des convertisseurs ayant une caractéristique logarithmique.

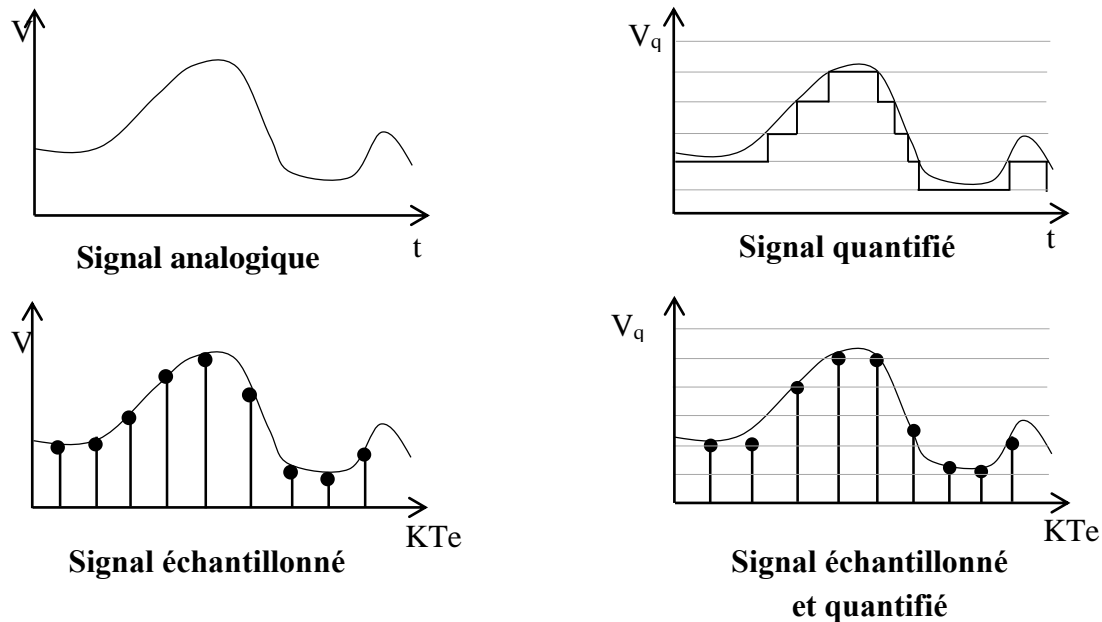


Figure I.3 : Echantillonnage et quantification d'un signal

I.5 Fonction de transfert de la boucle d'asservissement

I.5.1 En boucle ouverte

La figure I.4 montre que la boucle ouverte est formée de deux systèmes numériques en cascade : le calculateur et le processus numérisé, donc la fonction de transfert en boucle ouverte est le produit de fonction de transfert $C(z) H(z)$.

$C(z)$ est la fonction de transfert du correcteur (calculateur).

$H(z)$ est la fonction de transfert du processus échantillonné mini de son bloqueur d'ordre zéro.

I.5.2 En boucle fermée

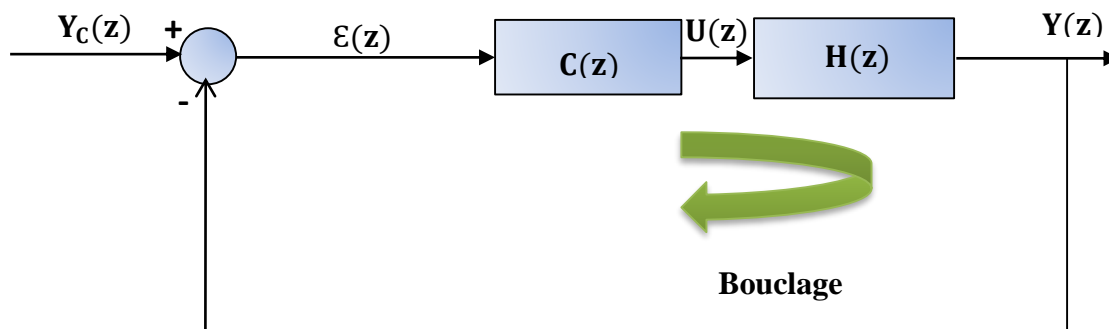


Figure I.4 : Schéma de commande en boucle fermée

Tous les signaux $y_c(k)$, $u(k)$, $e(k)$ ont une transformée en z données par les fonctions $Y_c(z)$, $U(z)$, $Y(z)$, $E(z)$ respectivement. La figure 1.4 exprime la structure de la fonction de la boucle dans le domaine z , les fonctions de transfert sont :

- En boucle ouverte

$$\frac{Y(z)}{E(z)} = H(z)C(z)$$

[I . 2]

- En boucle fermé :

$$\frac{Y(z)}{Y_c(z)} = \frac{C(z)H(z)}{1 + C(z)H(z)}$$

[I . 3]

I.6 La transmittance en présence d'un bloqueur d'ordre zéro

Si le système échantillonné en entrée est bloqué sur la période (cas le plus fréquent en commande numérique), il devient (figure I.5) :

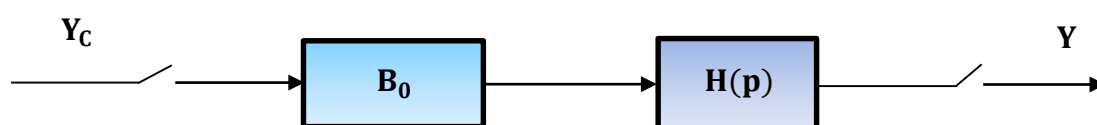


Figure I.5 : Système avec échantillonnage -bloqueur

$$TZ((B_0(p)(p))) = (1 - z^{-1})TZ\left(\frac{H(p)}{p}\right) \quad [I.4]$$

I.7 Système comportant un retard pur

Soit le processus comportant un retard pur $Tr = dh + tr$ avec d entier et $tr < h$

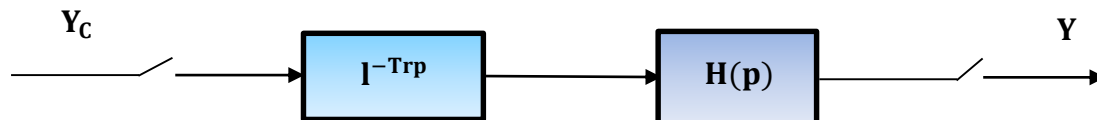


Figure I.6 : Système avec retard pure

En notant $Hr(p) = l^{-Trp} H(p)$, il vient :

$$TZ(Hr(p)) = z^{-d}TZ(l^{-Trp}H(p)) \quad [I.5]$$

Si h est choisi sous multiple exacte de Tr , on a : $tr = 0$ soit :

$$TZ(l^{-Trp}H(p)) = z^{-d}H(z) \quad [I.6]$$

I.8 Le système dans le cas discret et continu

I.8.1 Système du 1^{er} ordre

On prend un modèle qui correspond à une fonction de transfert sous forme canonique

$$H(p) = K \frac{1}{\tau p + 1} \quad [I.7]$$

$H(p)$: c'est la fonction de transfert en transformée de Laplace, tel que

K : le gain statique.

τ : la constante de temps.

I.8.2 Système du 2^{ème} ordre

Système de seconde ordre est représenté par la fonction de transfert $H(p)$ donnée par:

$$H(p) = \frac{\omega_n^2}{p^2 + 2\xi\omega_n p} \quad [I.8]$$

ω_n^2 : la pulsation propre de système.

ξ : le coefficient d'amortissement.

La fonction du transfert équivalente en discret est :

$$H(z) = \frac{b_1 z + b_2}{z^2 + a_1 z + a_2} \quad [I.9]$$

Tel que :

$$a_1 = -2\alpha\beta, a_2 = \alpha^2, b_1 = 1 - \alpha\left(\beta + \frac{\xi\omega_n}{\omega_d}\gamma\right) \text{ Et } b_2 = \alpha^2 + \alpha\left(\frac{\xi\omega_n}{\omega_d}\gamma - \beta\right)$$

Avec

$$\omega_d = \omega_n \sqrt{1 - \xi^2}, \beta = \cos(\omega_d h), \gamma = \sin(\omega_n h), \alpha = e^{-\xi\omega_n h}$$

I.9 Stabilité du système numérique

I.9.1 Critère Nyquist

La fonction de transfert du système en boucle fermée est :

$$F_{BF}(z) = \frac{C(z)H(z)}{1+C(z)H(z)}$$

[I . 10]

L'étude de la stabilité du système asservi revient à la position par rapport à un cercle centré au point (0,0) des racines du polynôme caractéristique $1 + C(z)H(z) = 0$.

Le contrôle de Nyquist est défini comme étant le cercle unité parcouru dans le sens trigonométrique, donc le contour fermé C s'écrit comme : $C = \{ z=1^{j\omega}, \omega_n \in [-\pi, \pi] \}$

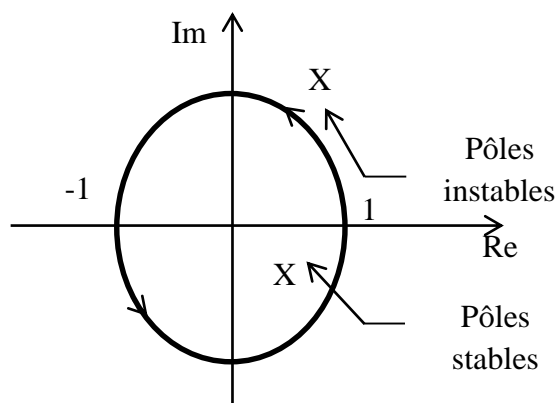


Figure 1.7 : Contour de Nyquist

I.9.2 Critère de jury

Un système à temps discret est dit stable si chaque séquence d'entrée bornée produit une séquence de sortie bornée.

Pour qu'un système à temps discret de la fonction de transfert $H(z)$ soit stable il faut que tous les pôles de $H(z)$ soient situés à l'intérieur du cercle unité d'un plan z .

Cette condition peut être déduite du système continu et sachant que $z = l^{hp}$ et cette application transforme le demi plan ouvert gauche au plan p à l'intérieur du cercle unité de plan z .

Dans le cas de système continu, on utilise le critère de Routh pour tester la stabilité.

Pour le cas de systèmes discrets on utilise le critère de Jury. Pour une équation caractéristique de la forme :

$$D(z) = a_0 z^N + a_1 z^{N-1} + \dots + a_n$$

[I . 11]

Jury a démontré que toutes les racines de l'équation sont dans le cercle unité si et seulement si tous les premiers élément ($b_0 ; c_0 ; d_0 ; \dots$) sont tous positifs. Il faut au départ arranger $D(z)$ pour que $a_0 = 1$ en suite procéder. Dans ces conditions $\{ (b_0 ; c_0 ; d_0 ; \dots)$ positive}, la condition $e_0 > 0$ et équivalente aux conditions $\{ (D(1) > 0 \text{ et } ((- 1)^n D(-1) > 0) \}$.

Ces deux conditions nécessaires pour la stabilité doivent être tester avant même de former la table. Il faut aussi noter qu'en cas d'instabilité le nombre de pôles en dehors du cercle unité est égale au nombre des a_0, b_0, c_0, d_0 et $e_0 \dots$

I.10 Précision des systèmes asservis échantillonnés

L'erreur d'asservissement est définie comme étant l'écart entre la consigne et la grandeur à régler. L'analyse de l'erreur en régime permanent est très importante parce qu'elle nous donne une mesure de la qualité de l'asservissement en terme de précision statique. Dans cette section nous étudions l'erreur permanente d'asservissement en supposant que le système bouclé est stable. Le signal d'erreur est :

$$E(z) = \frac{Yc(z)}{1+C(z)H(z)}$$

[I. 12]

D'après le théorème de la valeur finale pour le système à temps discret, on peut calculer l'expression de l'erreur en régime établi par :

$$\varepsilon_{\infty} = \lim_{z \rightarrow 1} (z - 1)E(z)$$

[I. 13]

Dans le cas d'un système à temps discret, une intégration est caractérisée par la présence d'un pôle en $z = 1^0 = 1$ ainsi si la somme des coefficients du dénominateur de la fonction de transfert du système considéré est nulle alors le système comporte au moins un intégrateur.

Mettons maintenant en évidence le pôle $z = 1$ de multiplicité m de la fonction de transfert en boucle ouverte :

$$F_{BO}(z) = \frac{1}{(z-1)^m} \frac{N(z)}{D(z)}$$

[I. 14]

Avec $N(z)$, $D(z)$ des polynômes de degrés appropriés. Soit la constante K définie par $N(1) / D(1)$.

L'entier m est appelée la classe du système en boucle ouverte.

- **Entrée échelon :** Dans ce cas l'erreur statique est également appelée écart permanent d'ordre 0. Alors l'entrée s'écrit comme :

$$Yc(z) = \frac{E_0}{z-1}$$

[I. 15]

Et donc :

$$\varepsilon^\infty = \lim_{z \rightarrow 1} \frac{E_0}{1 + F_{BO}(z)} = \begin{cases} \frac{E_0}{1+K} & \text{si } m = 0 \\ 0 & \text{si } m > 0 \end{cases}$$

[I. 16]

➤ **Entrée rampe** : l'entrée s'écrit comme :

$$Y_C = \frac{V_0 z}{(z - 1)^2}$$

[I. 17]

Et donc :

$$\varepsilon^\infty = \lim_{z \rightarrow 1} \frac{E_0}{(z - 1) + F_{BO}(z)} = \begin{cases} \infty & \text{si } m = 0 \\ V_0 & \text{si } m = 1 \\ \frac{K}{K} & \text{si } m > 1 \\ 0 & \text{si } m > 1 \end{cases}$$

[I. 18]

I.11 Conclusion

Dans ce chapitre, on a présenté quelques aspects de la commande numérique en outre le rôle et la place qu'occupe le calculateur dans le processus en général, qui comporte un régulateur implanté comme algorithme dans le calculateur, un procédé physique continu et une carte d'acquisition qui servira de lien de compatibilité entre les deux systèmes différents de leur nature structurelle ainsi que du type des signaux que développe chacun d'eux .

La commande numérique permet de commander plusieurs systèmes continus ou discrets définis par leur fonction de transfert. Le test de stabilité peut être effectué par plusieurs méthodes, on peut citer les plus utilisées à savoir la méthode graphique appelée le contour de Nyquist et la méthode algébrique nommée critère de Jury.

II.1 Introduction

La sensibilité des algorithmes de commande classique aux variations des paramètres du procédé à contrôler nécessite l'utilisation d'une nouvelle loi de commande, qui va prendre en charge les caractéristiques du processus et les performances que l'on souhaite imposer au système. Ce chapitre est entièrement dédié au régulateur RST, ce nom prévenant des trois polynômes qu'il fait intervenir. Le régulateur RST est défini dans le reste du chapitre, temps en temps. Nous aborderons plus particulièrement les méthodes de synthèse par placement de pôles dans le cas d'un régulateur de forme RST. Ensuite identifierons plus spécifiquement un problème de saturation du régulateur RST.

II.2 Régulateur RST

II.2.1 Définition

Un régulateur RST est un organe de contrôle permettant une régulation en boucle fermée d'un système industriel. Autrement dit, c'est un correcteur couramment utilisé dans les systèmes de commande numérique. Le sigle RST vient des trois polynômes qui doivent être déterminés afin d'obtenir une commande efficace. La synthèse de ce type de correcteur s'effectue par placement de pôles. La résolution du système mis en œuvre un polynôme de poursuite. La structure générale du régulateur RST est représentée ci-dessous.

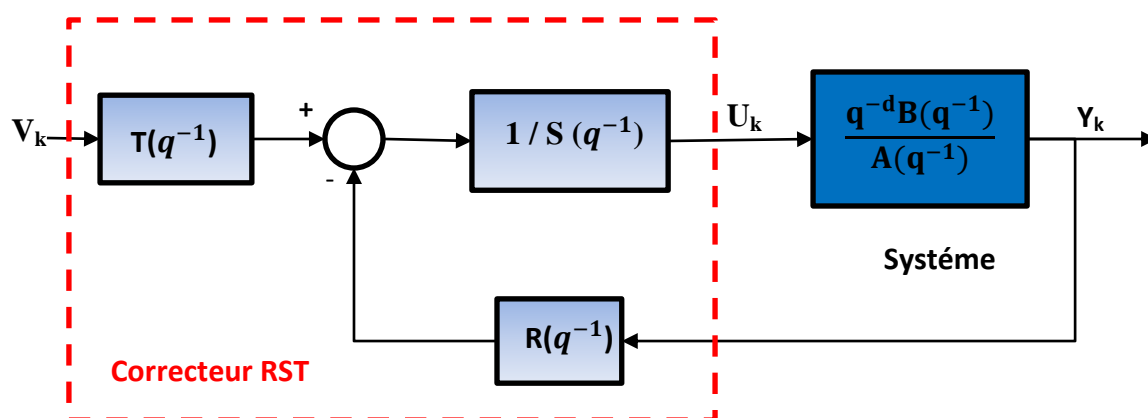


Figure II.1 : Représentation général du régulateur RST

Le régulateur RST possède deux degrés de liberté : le premier classiquement défini sur le signal d'erreur consigne/mesure, le second autorisant la poursuite via une trajectoire de référence. La structure du contrôleur permet aussi d'imposer des pôles et certains zéros en boucle fermée.

II.2.2 Principe

Le procédé est donné par :

$$F(q^{-1}) = \frac{q^{-d}B(q^{-1})}{A(q^{-1})} \quad \left| \begin{array}{l} \text{Deg}(B)=m \\ \text{Deg}(A)=m \end{array} \right.$$

[II . 01]

Et les polynômes du régulateur RST par :

$$T(q^{-1}) = t_0 + t_1q^{-1} + t_2q^{-1} + \dots$$

$$R(q^{-1}) = r_0 + r_1q^{-1} + r_2q^{-1} + \dots$$

$$S(q^{-1}) = s_0 + s_1q^{-1} + s_2q^{-1} + \dots$$

[II . 02]

Le système en boucle ouverte s'écrit :

$$G(q^{-1}) = \frac{q^{-d} \cdot T(q^{-1}) \cdot B(q^{-1}) \cdot R(q^{-1})}{S(q^{-1}) \cdot A(q^{-1})}$$

[II . 03]

Et le système en boucle fermée devient :

$$H(q^{-1}) = \frac{T(q^{-1}) \frac{F(q^{-1})}{S(q^{-1})}}{1 + R(q^{-1}) \cdot \frac{F(q^{-1})}{S(q^{-1})}}$$

$$H(q^{-1}) = \frac{q^{-d} \cdot T(q^{-1}) \cdot B(q^{-1})}{S(q^{-1}) \cdot A(q^{-1}) + q^{-d} \cdot R(q^{-1}) \cdot B(q^{-1})}$$

[II . 04]

Soit en notant P le polynôme du déterminant :

$$P(q^{-1}) = S(q^{-1}) \cdot A(q^{-1}) + q^{-1} \cdot R(q^{-1}) \cdot B(q^{-1}) \quad [II . 05]$$

D'où :

$$H(q^{-1}) = q^{-d} \cdot T(q^{-1}) \cdot B(q^{-1}) / P(q^{-1}) \quad [II . 06]$$

II.2.3 Effet de l'intégrateur

Considérons maintenant le montage régulateur, dans lequel la consigne $y_c(k) = 0$ tandis qu'une addition analogique $w(t)$ agit de manière additive en amont du processus à régler.

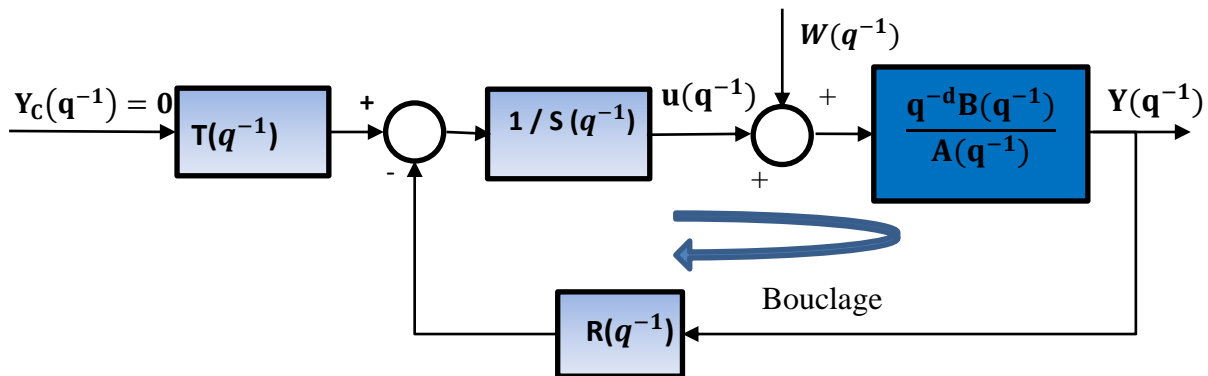


Figure II.2 : Schéma fonctionnel du montage en intégrateur avec un régulateur RST

On déduit l'influence de perturbation sur la grandeur à régler par :

$$Y(q^{-1}) = W(q^{-1})H(q^{-1}) + H(q^{-1}) \left(-\frac{S(q^{-1})}{R(q^{-1})} Y(q^{-1}) \right) \quad [II . 07]$$

D'où, en posant

$$K(q^{-1}) = \frac{S(q^{-1})}{R(q^{-1})} \quad [II . 08]$$

Alors

$$Y(q^{-1}) = \frac{W(q^{-1})H(q^{-1})}{1 + K(q^{-1})H(q^{-1})} \quad [\text{II.09}]$$

La nécessité pour rejeter des perturbations, de la présence de α effets intégrateurs, obtenus en remplaçant dans tous les développements précédents et avenir $R(z^{-1})$ par $(1 - z^{-1})^\alpha R(z^{-1})$, la fonction de transfert $(q^{-1}) = \frac{S(q^{-1})}{R(q^{-1})}$, doit alors être échangée avec :

$$W(q^{-1}) = \frac{W(q^{-1})}{(1 - q^{-1})^\alpha W(q^{-1})} \quad [\text{II.10}]$$

$$\text{Et } \frac{S(q^{-1})}{R(q^{-1})} \quad \text{avec} \quad \frac{T(q^{-1})}{(1 - q^{-1})^\alpha R(q^{-1})}$$

Pour annuler l'erreur permanente d'ordre n vis-à-vis d'une entrée de perturbation, il faut que la fonction de transfert de la partie chaîne directe située entre la sortie de lu comparateur et le point d'application de la perturbation comporte au moins $n+1$ pôles en $q=1$ soit ($n+1$ intégrateurs) [5].

II.2.4 Effet de zéro instable sur le comportement du système

Les zéros instables par abus de langage, sont les zéros situés dans la partie réelle positive du plan complexe sur lequel sont représentés les pôles et les zéros de la fonction de transfert désignant le modèle du système.

Ces zéros sont responsables de réponses imprévisibles et non désirables du système à des signaux de consignes tel que l'échelon ou autre signal pratique utilisé.

La compensation de ces zéros affecte la stabilité interne du système corrigé en boucle fermée. Leur apparition comme pôles instable dans le transfert consigne/commande est responsable de signaux de commandes de grandes amplitudes endommagent ainsi les actionneurs.

On trouve dans la littérature tous les développements et analyses concernant l'effet de ces zéros instables, les systèmes possédant ces zéros sont appelés systèmes à non minimum de phase en référence au tracé de leur graphe de phase dans le diagramme de Bode (gain/phase), ce sont des systèmes plus difficiles à commander et présentent certaines particularités surtout en ce qui concerne leur stabilité en boucle fermée, dans certains cas elle est difficile à prouver.

II.3 Synthèse des correcteurs RST

Un correcteur RST est synthétisé par une méthode de placement de pôles algébrique. L'objectif est de donner à la boucle fermée le comportement d'un système décrit par une fonction de transfert modèle $F_m = N_m(q^{-1}) / A_m(q^{-1})$ exprimée comme une fraction rationnelle en q^{-1} . Usuellement cette fonction est du deuxième ordre, et on prendra pour A_m un polynôme monique, c'est-à-dire tel que $A_m(q^{-1} = 0) = 1$ (cela est toujours possible).

La structure d'un correcteur RST est donnée tel que R, S et T sont des polynômes en q^{-1} .

Dans la suite nous considérons que le procédé à asservir est donné par la fonction de transfert $G(q^{-1})$ qui s'écrit en puissance de q^{-1} .

$$G(q^{-1}) = q^{-d} B(q^{-1}) / A(q^{-1})$$

[II . 11]

Où q^{-d} est le retard pur du procédé.

II.3.1 Définition de la fonction de transfert modèle

La fonction de transfert modèle est définie à partir de critères du type :

- Amortissement.
- Gain statique.
- Pulsation propre.
- Dépassement.
- Temps de premier maximum.
- Etc.

Ces critères permettent généralement de définir le dénominateur A_m de la fonction de transfert modèle. Le choix de N_m en revanche ne peut pas être complètement imposé. Comme l'influence des zéros de la fonction de transfert est moindre que celle des pôles, ces contraintes n'auront pas d'effets trop importants sur le comportement du système.

La fonction de transfert du système corrigé par un correcteur RST dont la structure est donnée par le schéma de la figure vaut :

$$FTBF = \frac{BT q^{-d}}{AS + BR q^{-d}} \quad [II . 12]$$

Une analyse simple que le numérateur de la fonction de transfert doit nécessairement contenir

- D'une part le retard pur q^{-d} du procédé. En effet il est possible de compenser un retard pur, car il faudrait concevoir un système non causal. Donc N_m contient q^{-d} ou éventuellement $q^{-(d+t)}$ avec $t > 0$.
- D'autre part tous les zéros du procédé qui ne seront pas compensés par S . ces zéros sont notés B^- .

Finalement, on a donc $N_m = q^{-d}B^-B_m^+$ ou seul B_m^+ peut être choisi librement. En l'absence de contraintes particulières, on choisira souvent B_m^+ de degré minimal (zéro).

II.3.2 Choix des pôles et zéros à compenser

Il est possible de choisir de compenser ou de ne pas compenser certains des pôles et zéros du procédé. On décomposera le procédé de la façon suivante :

$$G(q^{-1}) = \frac{q^{-d}B(q^{-1})}{A(q^{-1})} = \frac{q^{-d}B^+(q^{-1})B^-(q^{-1})}{A^+(q^{-1})} \quad [II . 13]$$

Où le polynôme B^+ contient les zéros que l'on souhaite compenser, et A^+ les pôles que l'on souhaite compenser.

Pour compenser un pôle il faut le mettre dans le polynôme R. pour compenser un zéro, il faut le mettre dans le polynôme S. cela se comprend aisément en observant la structure du correcteur.

On aura donc :

$$\begin{aligned} R &= A^+R' \\ S &= B^+S' \end{aligned}$$

[II . 14]

Comment choisir la décomposition de B et A ?

En vertu du principe de la stabilité interne du système, il est déjà interdit de compenser des pôles ou des zéros à l'extérieur du cercle unité. Par conséquence :

- B - contient tous les zéros instables du procédé.
- A - contient tous les pôles instables du procédé.

Les autres pôles et zéros peuvent être compensés si le concepteur du correcteur en fait le choix. Toutefois, on note que les zéros compensés apparaissent comme des pôles dans la fonction de transfert entre consigne Y_c et la commande U . Les zéros stables réels négatifs conduisent donc à des pôles stables réels négatifs pour le transfert consigne commande. Il est connu que de tels pôles créent des comportements oscillants alternés : la commande alternera donc entre des valeurs positives et négatives. Ce phénomène est à éviter car il crée des usures prématurées des systèmes mécaniques et des systèmes électriques de puissance (commutation des hacheurs). Par conséquent, on évitera de compenser les zéros stables réels négatifs.

II.3.3 Calcul des polynômes R, S et T

Le calcul des polynômes R, S et T se fait en identifiant la fonction de transfert modèle avec la fonction de transfert du système bouclé corrigé :

$$\frac{q^{-d}B^+(q^{-1})B^-(q^{-1})}{A^+(q^{-1})A^-(q^{-1})} = \frac{q^{-d}B^-B_m^+}{A_m}$$

[II . 15]

Après simplification et en égalisant (à un polynôme près A_0) numérateur et d'énomérateur, on obtient :

$$\begin{aligned} A^{-1}S' + q^{-d}B^{-1}R' &= A_0A_m \\ T &= A_0A_mB_m^+ \end{aligned}$$

[II . 16]

La première équation est une équation diophantienne où S_0 et R_0 sont les inconnues.

Le polynôme A_0 est généralement choisi égal à 1, sauf si le cahier des charges impose de filtrer (filtrer pas rejeter) des perturbations au-delà d'une certaine fréquence.

II.3.4 Résolution de l'équation diophantienne

Les équations diophantiennes sont de la forme :

$$AX + BY = C$$

[II . 17]

Où A , B et C sont des polynômes connus et X et Y des polynômes recherchés.

Ces équations ont une infinité de solutions, mais on s'intéressera aux solutions de degré minimal pour X et pour Y .

Deux cas se présentent :

- $\deg(A) + \deg(B) > \deg(C)$ l'équation est régulière et il existe une solution à la fois minimal pour X et pour Y . on a alors $\deg(X) = \deg(B) - 1$ et $\deg(Y) = \deg(A) - 1$
- $\deg(A) + \deg(B) \leq \deg(C)$ il existe une solution minimale pour X pour laquelle $\deg(X) = \deg(B) - 1$ et $\deg(Y) = \deg(C) - \deg(B)$ et une solution minimale pour Y pour laquelle $\deg(Y) = \deg(A) - 1$ et $\deg(X) = \deg(C) - \deg(A)$.

Une fois les degrés de X et Y connus, l'équation diophantienne peut être mise sous la forme d'une équation matricielle qu'on peut résoudre simplement avec Matlab par exemple.

Lorsque S_0 et R_0 ont été déterminés, on obtient R, S et T par :

$$\begin{aligned} R &= R'A^+ \\ S &= S'B^+ \\ T &= A^+A_0B_m^+ \end{aligned}$$

[II . 18]

II.3.5 Objectifs supplémentaires pour la synthèse

Un des objectifs de la synthèse sera souvent d'avoir une erreur statique de position nulle, autrement dit un gain statique unitaire. Pour un correcteur série, ce type de contraintes est respecté si le procédé contient un intégrateur, ou en ajoutant un dans le correcteur.

Ici, en raison du préfiltre T, la présence d'un intégrateur ne permet pas de garantir l'annulation de l'erreur statique.

Pour cela, il suffit en fait d'imposer que la fonction de transfert modèle ait un gain statique unitaire, autrement dit, on veut que :

$$A_m(1) = B_m^+(1)B_m^-(1)$$

[II . 19]

Un autre objectif peut être de rejeter des perturbations. Pour cela, on procède comme pour un correcteur série classique : il faut ajouter le bon nombre d'intégrateur dans le système via le correcteur. R, S et T étant des polynômes, seul le bloc 1/S peut contenir des intégrateurs.

Pour imposer un intégrateur dans S, on réécrit simplement l'équation diophantienne de la façon suivante :

$$A^-(1 - q^{-1})S_1 + q^{-d}B^-R' = A_0A_m$$

$$\text{Avec } S' = (1 - q^{-d})S_1$$

[II . 20]

II.4 Méthode de placement de pôles

II.4.1 Placement de pôles avec RST

La stratégie de placement de pôles permet la conception d'un contrôleur numérique RST pour les cas d'un système stable ou instable :

- Sans restriction sur les degrés des polynômes $A(q^{-1})$ et $B(q^{-1})$ du modèle du procédé à temps discret (à condition qu'ils n'ont pas des facteurs communs).
- Sans restriction sur le retard du procédé.
- Sans restriction sur les zéros du procédé (stables ou instables).

Cette méthode ne simplifie pas les zéros du système (c'est pourquoi ils peuvent être instables).

La seule restriction concerne les facteurs communs de $A(q^{-1})$ et de $B(q^{-1})$, ce qui doit être simplifié avant que les calculs soient effectués [1].

II.4.1.1 Structure

La structure du système en boucle fermée est donnée sur le schéma II.3 le procédé à commander est caractérisé par la fonction de transfert échantillonnée (irréductible).

$$H(q^{-1}) = \frac{q^{-d}B(q^{-1})}{A(q^{-1})} \quad [\text{II.21}]$$

Où d est le nombre entier de la période d'échantillonnage contenues dans le retard pur.

Avec

$$A(q^{-1}) = 1 + \alpha_1 q^{-1} + \dots + \alpha_{n_A} q^{-n_A} \quad [\text{II.22}]$$

$$B(q^{-1}) = b_1 q^{-1} + b_2 q^{-2} + \dots + B_{n_B} q^{-n_B} = q^{-1} B^*(q^{-1}) \quad [\text{II.23}]$$

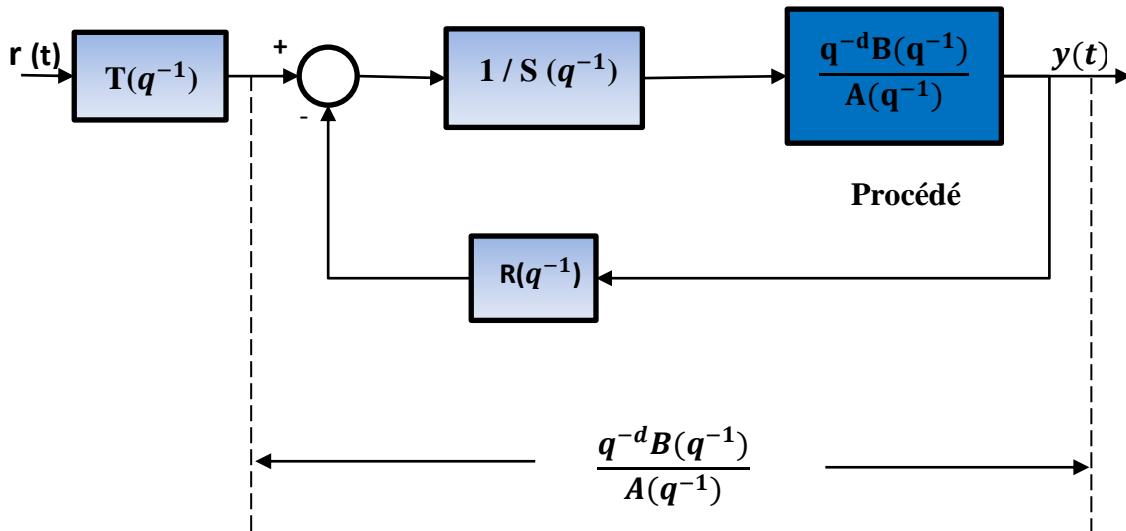


Figure II.3 : Placement de pôles avec le correcteur RST

F.T de la boucle fermée ($r \rightarrow y$) (poursuite de consigne) est donnée par :

$$H_{BF}(q^{-1}) = \frac{q^{-d}T(q^{-1})B(q^{-1})}{A(q^{-1})S(q^{-1}) + q^{-d}B(q^{-1})R(q^{-1})} = \frac{q^{-d}T(q^{-1})B(q^{-1})}{P(q^{-1})} \quad [II . 24]$$

Où

$$P(q^{-1}) = S(q^{-1}).A(q^{-1}) + q^{-d}.R(q^{-1}).T(q^{-1}).B(q^{-1}) = 1 + p_1q^{-1} + p_2q^{-2} + \dots \quad [II . 25]$$

$P(q^{-1})$ définit les pôles de la boucle fermée et respectivement le comportement en régulation et le comportement vis-à-vis d'une perturbation est donné par la fonction de sensibilité perturbation sortie ($p \rightarrow y$) (rejet de perturbation) :

$$H_{YP}(q^{-1}) = \frac{A(q^{-1})S(q^{-1})}{A(q^{-1})S(q^{-1}) + q^{-d}B(q^{-1})R(q^{-1})} = \frac{A(q^{-1})S(q^{-1})}{P(q^{-1})} \quad [II . 26]$$

II.4.1.2 Choix des pôles en boucle fermée (polynôme P)

Les pôles en boucle fermée qui sont les racines du polynôme $P(q^{-1})$ définissent en grande partie les performances en régulation.

Nous avons vu dans le cas du régulateur P.I.D qu'on peut spécifier un polynôme $P(q^{-1})$ définissent les pôles en boucle fermée à partir d'un modèle du deuxième ordre continu ayant la fréquence propre et l'amortissement désiré. On peut aussi spécifier directement le polynôme $P(q^{-1})$ à partir des performances désirées. A titre d'illustration.

II.4.1.3 Régulation (calcul de $R(q^{-1})$ et $S(q^{-1})$)

$P(q^{-1})$ étant spécifié, pour calculer $R(q^{-1})$ et $S(q^{-1})$, il faut résoudre l'équation :

$$A(q^{-1})S(q^{-1}) + q^{-1}B(q^{-1})R(q^{-1}) = P(q^{-1}) \quad [\text{II} . 27]$$

Connue sous le nom de « identité de BEZOUT »

En définissant :

$$n_A = \deg A(q^{-1}) \quad ; \quad n_B = \deg B(q^{-1}) \quad [\text{II} . 28]$$

Cette équation polynomiale a une solution unique ($A(q^{-1})$ et $B(q^{-1})$ étant premier entre eux) pour :

$$n_P = \deg P(q^{-1}) \leq n_A + n_B + d - 1$$

$$n_S = \deg S(q^{-1}) = n_A + d - 1$$

$$n_R = \deg R(q^{-1}) = n_B - 1$$

[II . 29]

Où

$$S(q^{-1}) = 1 + s_1 q^{-1} + \dots + s_{n_s} q^{-n_s}$$

$$= 1 + q^{-1} s^*(q^{-1})$$

$$R(q^{-1}) = r_0 + r_1 q^{-1} + \dots + r_{n_R} q^{-n_R}$$

[II . 30]

Pour effectivement résoudre l'équation [II.27] celle-ci se met sous forme matricielle :

$$MX=P$$

[II . 31]

où

$$X^T = [1, s_1 \dots s_{n_s}, r_0 \dots r_{n_R}]$$

$$P^T = [1, p_1 \dots s_i \dots p_{n_p}, 0 \dots 0]$$

[II . 32]

Et la matrice M a la forme :

$$\begin{array}{c}
 \overbrace{\hspace{10em}}^{n_B + d} \quad \overbrace{\hspace{10em}}^{n_A} \\
 \left[\begin{array}{cccccccc}
 1 & 0 & \dots & 0 & 0 & \dots & \dots & 0 \\
 a_1 & 1 & & \cdot & b'_1 & & & \\
 a_2 & & 0 & b'_2 & & & & b'_1 \\
 & & & 1 & \cdot & & & b'_2 \\
 & & & a_1 & \cdot & & & \\
 a_{n_A} & & a_2 & b'_{n_B} & & & & \\
 0 & & \cdot & \cdot & \cdot & \cdot & & \\
 0 & \dots & 0 & a_{n_A} & 0 & 0 & 0 & b'_{n_B}
 \end{array} \right]
 \end{array}
 \left. \vphantom{\begin{array}{c} \overbrace{\hspace{10em}}^{n_B + d} \quad \overbrace{\hspace{10em}}^{n_A} \\ \left[\begin{array}{cccccccc} \dots \end{array} \right] } \right\} n_A + n_B + d$$

Où : $b_i = 0$ pour $i = 0, 1 \dots d$; $b_i = b'_{i-b}$ pour $i \geq d + 1$

Le vecteur X qui contient les coefficients des polynômes $R(z^{-1})$ et $S(z^{-1})$ s'obtient après inversion de la matrice M par la formule :

$$X = M^{-1}P \quad [\text{II} . 33]$$

Différentes méthodes sont utilisées pour résoudre l'équation [II 27]

Pour des raisons variées les polynômes $R(q^{-1})$ et $S(q^{-1})$ contiennent en général des parties fixes spécifiées avant la résolution de l'équation par exemple la nécessité d'avoir une erreur statique nulle pour une consigne ou une perturbation en échelon implique la présence d'un intégrateur numérique dans la voie directe respectivement la présence d'un terme $(1 - q^{-1})$ dans le polynôme $S(q^{-1})$.

C'est pour prendre en compte ces parties fixes pré-spécifiées que les polynômes $R(q^{-1})$ et $S(q^{-1})$ sont factorisés sous la forme :

$$R(q^{-1}) = R'(q^{-1})H_R(q^{-1}) \quad [\text{II} . 34]$$

$$S(q^{-1}) = S'(q^{-1})H_S(q^{-1}) \quad [\text{II} . 35]$$

Où $H_R(q^{-1})$ et $H_S(q^{-1})$ sont des polynômes pré-spécifiés

Avec cette paramétrisation des polynômes $S(q^{-1})$ et $R(q^{-1})$, la fonction de transfert en boucle fermée sera alors :

$$\begin{aligned} H_{BF} &= \frac{q^{-d}T(q^{-1})B(q^{-1})}{A(q^{-1})S'(q^{-1})H_S(q^{-1}) + q^{-d}B(q^{-1})R'(q^{-1})H_R(q^{-1})} \\ &= \frac{q^{-d}T(q^{-1})B(q^{-1})}{P(q^{-1})} \end{aligned} \quad [\text{II} . 36]$$

Au lieu de l'équation [II.27] il faudra maintenant résoudre l'équation [II.37]

$$A(q^{-1})S'(q^{-1})H_S(q^{-1}) + q^{-d}B(q^{-1})R'(q^{-1})H_R(q^{-1}) = P(q^{-1}) \quad [\text{II.37}]$$

Pour résoudre l'équation [II.37] il suffit de résoudre l'équation [II.27] en remplaçant $A(q^{-1})$ par $A(q^{-1})H_S(q^{-1})$ et $B(q^{-1})$ par $B(q^{-1})H_R(q^{-1})$ avec la restriction que les polynômes $A(q^{-1})H_S(q^{-1})$ et $B(q^{-1})H_R(q^{-1})$ soient premiers entre eux, par contre pour la mise en œuvre $S(q^{-1})$ sera remplacé par $S(q^{-1})H_S(q^{-1})$ et $R(q^{-1})$ par $R(q^{-1})H_R(q^{-1})$.

- **Erreur statique**

Comme on l'a vu précédemment $S(q^{-1})$ doit contenir un terme $(1 - q^{-1})$ pour avoir une erreur nulle à une entrée ou perturbation en échelon, $(S_{yp}(q^{-1}))$ doit être nulle en régime stationnaire c'est-à-dire pour $q = 1$. Dans ce cas :

$$H_S(q^{-1}) = (1 - q^{-1}) \quad [\text{II.38}]$$

- **Rejet de perturbation harmonique**

Si on impose une réjection parfaite d'une perturbation harmonique à une certaine fréquence, ce qui équivaut à imposer que $H_S(q^{-1})$ introduise une paire de zéros complexes non amortis à cette fréquence.

Dans ce cas :

$$H_S(q^{-1}) = 1 + \alpha q^{-1} + q^{-2} \quad [\text{II.39}]$$

Où

$$\alpha = -2\cos\omega T_e = -2\cos 2\pi \frac{f}{f_e}$$

S'il s'agit uniquement d'une atténuation donnée, $H_S(q^{-1})$ introduira une paire de zéros complexes amortis avec un facteur d'amortissement dépendant de l'atténuation souhaitée.

- **Blocage d'un signal**

Dans certaines applications le signal mesuré contient des composantes à certaines fréquences sur lesquelles la régulation ne doit pas agir pour ne pas atténuer leur effet (il s'agit en général de signaux inhérents au processus de fabrication). Dans ce cas, le système doit être en boucle ouverte à ces fréquences, ce qui est équivalent à imposer des zéros correspondants dans $R(q^{-1})$ conduisant à un gain nul pour $R(q^{-1})$ et respectivement pour la fonction de sensibilité perturbation-entrée $S_{up}(q^{-1})$.

La partie fixe de $R(q^{-1})$ aura la forme :

$$H_R(q^{-1}) = 1 + \beta q^{-1} + q^{-2} \quad [\text{II} . 40]$$

Où

$$\beta = -2\cos\omega_0 T_e = -2\cos 2\pi \frac{f}{f_e}$$

Ou en général la forme d'un polynôme de deuxième ordre ayant des zéros complexes amortis si une certaine atténuation est acceptée.

II.4.1.4 Poursuite (calcul de $T(q^{-1})$)

Dans le cas idéal, lors d'un changement de consigne, on souhaite faire suivre la sortie du procédé $y(t)$ une trajectoire $y^*(t)$. Cette trajectoire peut être mémorisée ou engendrée à chaque changement de consigne à l'aide d'un modèle de référence (Figure II.4)

La fonction de transfert du modèle de référence est :

$$H_m(q^{-1}) = \frac{q^{-d} B_m(q^{-1})}{A_m(q^{-1})} \quad [\text{II} . 41]$$

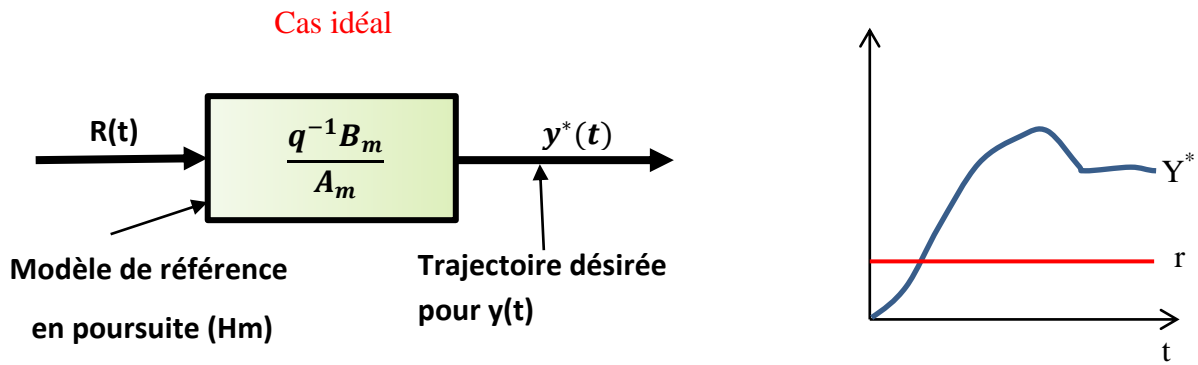


Figure II.4 : Génération de la trajectoire désirée y^*

En règle générale, elle se détermine à partir de performances souhaitées (temps de réponse, dépassement, temps d'établissement) en passant par exemple par un modèle continu normalisé du 2eme ordre (paramètres ω_0, ξ) défini à l'aide des courbes données dans la figure II.4.

En connaissant la fonction de de transfert continu et la période d'échantillonnage T_e . On obtient par discrétisation la fonction de transfert du modèle de référence (à l'aide de PC-REG ou des tables de Transformation), la fonction de transfert échantillonnée du modèle de référence sera de la forme :

$$\frac{q^{-1}B_m(q^{-1})}{A_m(q^{-1})} = \frac{q^{-1}(b_{m_0} + b_{m_0}q^{-1})}{1 + a_{m_1}q^{-1} + a_{m_2}q^{-1}}$$

[II . 42]

C'est cette équation de transfert que le régulateur doit réaliser entre consigne « r » et la sortie « y » éventuellement multipliée par q^{-1} dans le cas de la présence d'un retard de « d » pas dans le du « placement de pôles » ceci n'est pas possible car on conserve les zéros du procédé (polynôme $B(q^{-1})$).

L'objectif sera alors d'approcher la trajectoire de référence retardée :

$$y^*(t) = r(t) \frac{q^{-(d+1)}B_m(q^{-1})}{A_m(q^{-1})}$$

[II . 43]

Pour ceci, on engendre d'abord
 $y^*(t + d + 1)$ à partir de $r(t)$:

$$y^*(t + d + 1) = r(t) \frac{B_m(q^{-1})}{A_m(q^{-1})}$$

[II . 44]

Et on choisit $t(q^{-1})$ pour assurer

- Un gain statique unitaire entre y^* et y
- La compensation de la dynamique de régulation car la dynamique du régulation est en général différente de la dynamique de poursuite $A_m(q^{-1})$ ceci conduit à choisir :

$$T(q^{-1}) = GP(q^{-1})$$

[II . 45]

Où

$$G = \begin{cases} 1 \\ B(1) \\ 1 \end{cases}$$

Si $B(1) \neq 0$ et si $B(1) = 0$ respectivement.

[II . 46]

L'équation de la loi de commande devient :

$$S(q^{-1})u(t) + R(q^{-1})y(t) = T(q^{-1})y^*(t + d + 1)$$

[II . 47]

Le schéma complet pour le placement des pôles est donné dans la figure

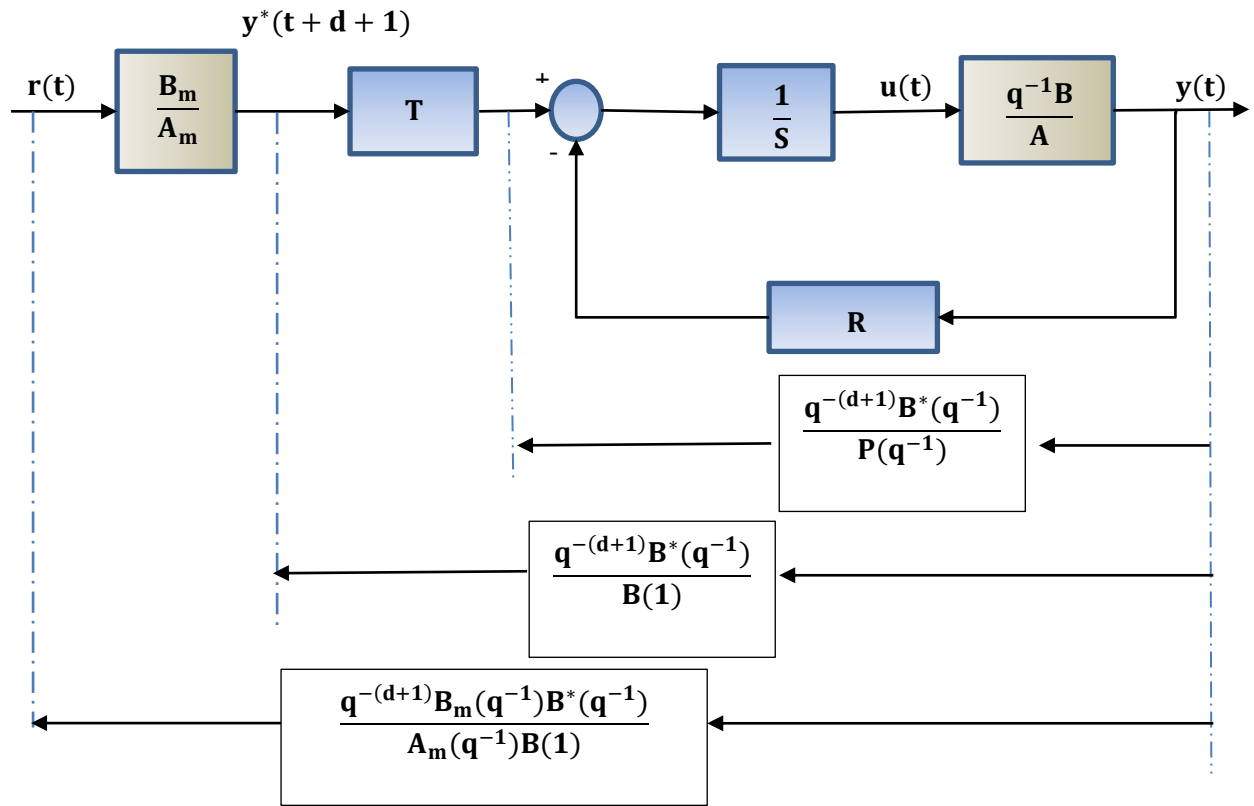


Figure II.5 : Placement des pôles – poursuite et régulation

La fonction de transfert réalisée entre la consigne et la sortie est :

$$H_{BF}(q^{-1}) = \frac{q^{-(d+1)}B_m(q^{-1})B^*(q^{-1})}{A_m(q^{-1})B(1)}$$

[II . 48]

Les équations de la régulation pour placement de pôles sous différentes formes sont résumées au - dessous (les équations de récurrents servants à la programmation sont encadrées).

$$u(t) = \frac{T(q^{-1})y^*(t+d+1) - R(q^{-1})y(t)}{S(q^{-1})}$$

$$S(q^{-1})u(t) + R(q^{-1})y(t) = GP(q^{-1})y^*(t + d + 1) = T(q^{-1})y^*(t + d + 1)$$

$$S(q^{-1}) = 1 + q^{-1}S^*(q^{-1})$$

$$u(t) = P(q^{-1})Gy^*(t + d + 1) - S^*(q^{-1})u(t - 1) - R(q^{-1})y(t)$$

$$y^*(t + d + 1) = \frac{B_m(q^{-1})}{A_m(q^{-1})} r(t)$$

$$A_m(q^{-1}) = 1 + q^{-1}A_m^*(q^{-1})$$

$$y^*(t + d + 1) = A_m^*(q^{-1})y(t + d) + B_m(q^{-1})r(t)$$

$$B_m(q^{-1}) = b_{m0} + b_{m1} q^{-1} + \dots$$

$$A_m(q^{-1}) = a_{m1} q^{-1} + a_{m2} q^{-2} + \dots$$

II.4.2 Poursuite et régulation à objectif indépendant

Cette méthode de calcul des régulateurs numériques c'est un cas particulier du placement de pôles (les pôles de la boucle fermée contiennent les zéros du procédé). Permet d'obtenir le comportement désiré en poursuite (changement de consigne) indépendamment du comportement désiré en régulation (rejet d'une perturbation). La spécification des performances est illustrée dans la figure II.6 ou le temps de réponse en régulation est sensiblement plus petit que le temps de réponse à une généralisation (mais la situation inverse peut être considérée). Cette méthode est une généralisation de la prétendue « commande modèle de référence » [1].

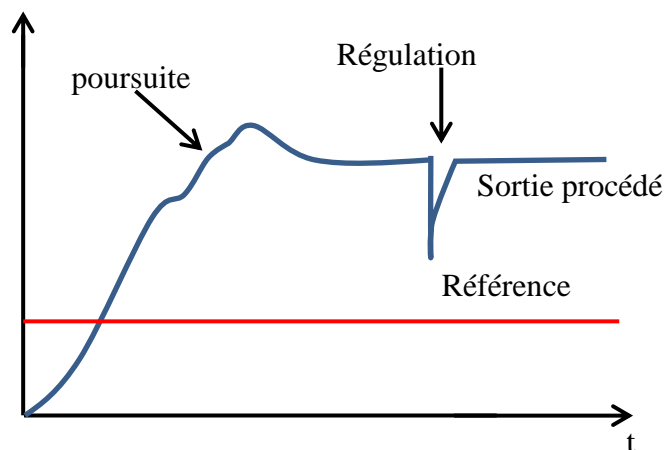


Figure II.6 : Performance en poursuite et en régulation

A la différence de la méthode de placement de pôles cette méthode conduit à la simplification des zéros du modèle échantillonné du procédé, ce qui permet de réaliser sans approximation des performances imposées en poursuite et en régulation.

Cette stratégie de commande permet de calculer un régulateur numérique RST pour :

- Système stable ou instable
- Sans restriction sur les degrés des polynômes $A(q^{-1})$ et $B(q^{-1})$ de la fonction de transfert échantillonné du procédé
- Sans restriction sur le retard entier d du procédé discrétisé
- Modèle échantillonnés ayant des zéros stables

Par contre à cause de la simplification des zéros, elle ne s'applique qu'aux modèles échantillonnés de procédé ayant des zéros stables. Cette méthode ne tolère donc pas des retards fractionnaires supérieurs à $0.5 T_e$.

Notons que des zéros instables permettent également apparaître suite à échantillonnage trop rapide des systèmes continus ayant une déférence de degré supérieur à 2, entre le numérateur et le dénominateur de la fonction de transfert [1].

II.4.2.1 Structure

La structure du système en boucle représentée dans la figure II.7

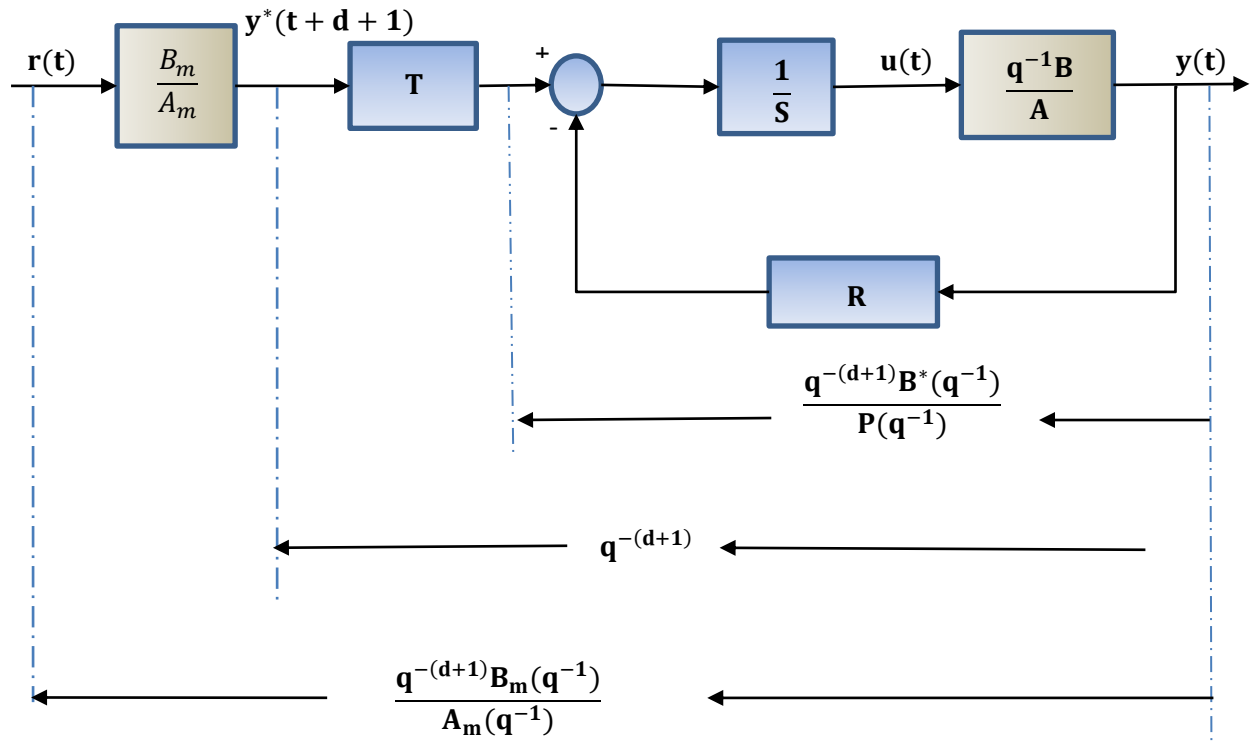


Figure II.7 : Poursuite et régulation à objectif indépendant

Les pôles en boucle fermée sont définis par un polynôme $P(q^{-1})$ qui spécifie le comportement désiré en régulation.

En règle générale, $P(q^{-1})$ est le produit de deux polynômes :

$$P(q^{-1}) = P_D(q^{-1}) \cdot P_F(q^{-1})$$

Où $P_D(q^{-1})$ est déterminé en fonction de performances souhaitées et $P_F(q^{-1})$ représente les pôles additionnels.

La fonction de transfert entre la consigne $r(t)$ et la sortie du procédé $y(t)$ qui définit le comportement en poursuite est :

$$\frac{q^{-(d+1)}B_m(q^{-1})}{A_m(q^{-1})}$$

La sortie du modèle de poursuite $B_m(q^{-1})/A_m(q^{-1})$ spécifie la trajectoire désirée y^* avec $d + 1$ pas d'avance.

Le procédé à réguler est caractérisé par la fonction de transfert échantillonnée dans l'équation [II.21], les polynômes $A(q^{-1})$ et $B(q^{-1})$ étant spécifiés par les équations et [II.22] notons que dans ce cas les polynômes $A(q^{-1})$ et $B(q^{-1})$ peuvent avoir des facteurs communs.

Le calcul de $R(q^{-1})$, $S(q^{-1})$ et $T(q^{-1})$ se fera en deux temps. Dans un premier temps à l'aide de $R(q^{-1})$ et $S(q^{-1})$, on placera les poles en boucle fermée aux valeurs désirées, spécifier par un polynôme $P(q^{-1})$ (objectif de régulation) et on simplifiera les zéros du modèle échantillonné du procédé. Dans un deuxième temps, on déterminera le pré-compensateur $T(q^{-1})$ pour obtenir en poursuite que la sortie du procédé suive, au retard pres, la trajectoire de référence y^* .

II.4.2.2 Régulation (calcul de $R(z^{-1})$ et $S(z^{-1})$)

Sans pré-compensateur $T(z^{-1})$, la fonction de transfert en boucle fermée est :

$$\begin{aligned} H_{BF}(q^{-1}) &= \frac{q^{-(d+1)}B^*(q^{-1})}{A(q^{-1})S(q^{-1}) + q^{-(d+1)}B^*(q^{-1})R(q^{-1})} \\ &= \frac{q^{-(d+1)}}{P(q^{-1})} = \frac{q^{-(d+1)}B^*(q^{-1})}{B^*(q^{-1})P(q^{-1})} \end{aligned}$$

[II . 49]

On souhaite que les pôles soient ceux définis par $P(z^{-1})$ et que les zéros du procédé soient simplifier (pour pouvoir ultérieurement obtenir une poursuite parfaite).

De l'équation [II.1], il résulte que les pôles en boucle fermée doivent en fait contenir en plus les zéros du modèle du procédé.

$P(z^{-1})$ étant spécifié, il résulte de l'équation [II.49] que pour calculer $R(z^{-1})$ et $S(z^{-1})$, il faut résoudre l'équation :

$$A(q^{-1})S(q^{-1}) + q^{-(d+1)}B^*(q^{-1})R(q^{-1}) = B^*(q^{-1})P(q^{-1}) \quad [\text{II. 50}]$$

D'autre part, pour compenser $B^*(q^{-1})$, $S(q^{-1})$ devra avoir $B^*(q^{-1})$ comme facteur commun :

$$S(q^{-1}) = s_0 + s_1q^{-1} + \dots + s_{n_s}q^{-n_s} = B^*(q^{-1})S'(q^{-1}) \quad [\text{II. 51}]$$

En introduisant l'expression de $S(q^{-1})$ donnée par l'équation [II.51] dans l'équation [II.50] et après simplification par $B^*(q^{-1})$, on obtient :

$$A(q^{-1})S'(q^{-1}) + q^{-(d+1)}R(q^{-1}) = P(q^{-1}) \quad [\text{II. 52}]$$

Cette équation a une solution unique pour :

$$n_p = \text{deg}A(q^{-1}) \leq n_A + d \quad , \quad \text{deg}S'(q^{-1}) = d \quad , \quad \text{deg}R(q^{-1}) = n_A - 1 \quad [\text{II. 53}]$$

Ou

$$R(q^{-1}) = r_0 + r_1q^{-1} + \dots + r_{n_A-1}q^{n_A-1} \quad [\text{II. 54}]$$

Et

$$S'(q^{-1}) = 1 + s_1' q^{-1} + s_0' q^{-1} + \dots + s_d' q^{-d}$$

[II . 55]

L'équation [II.52] peut se mettre sous une forme matricielle :

$$MX = P$$

[II . 56]

Où M est une matrice triangulaire inferieure de dimension $(n_A + d + l) * (n_A + d + l)$

la matrice M a la forme :

$$\begin{array}{c}
 \underbrace{\hspace{10em}}_{n_A + n_B + 1} \\
 \left[\begin{array}{cccccccc}
 \mathbf{1} & \mathbf{0} & & & & & & \mathbf{0} \\
 \mathbf{a}_1 & \mathbf{1} & & & & & & \cdot \\
 \mathbf{a}_2 & & \mathbf{0} & & & & & \cdot \\
 & & & \mathbf{1} & \mathbf{0} & & & \cdot \\
 & & & \mathbf{a}_1 & \mathbf{1} & & & \cdot \\
 \mathbf{a}_d & \mathbf{a}_{d-1} \dots & \mathbf{a}_1 & \mathbf{1} & & & & \cdot \\
 \mathbf{a}_{d+1} & \mathbf{a}_d & & \mathbf{a}_1 & \mathbf{1} & & & \cdot \\
 \mathbf{a}_{d+2} & \mathbf{a}_{d+1} & & \mathbf{a}_2 & \mathbf{0} & & & \cdot \\
 & & & \cdot & \cdot & \cdot & & \cdot \\
 & & & \cdot & \cdot & \mathbf{0} & & \cdot \\
 \mathbf{0} & & \mathbf{0} & \mathbf{a}_{n_A} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \cdot
 \end{array} \right]
 \end{array}$$

$\underbrace{\hspace{8em}}_{d+1}$

$\underbrace{\hspace{2em}}_{n_A}$

$\left. \begin{array}{c} \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \end{array} \right\} n_A + n_B + d$

$$X^T = [1, s_1', \dots, s_d', r_0, r_1, \dots, r_{n_A-1}] \quad [\text{II.57}]$$

$$P^T = [1, P_0, P_1, \dots, P_{n+1}] \quad [\text{II.58}]$$

Certain coefficient P_i peuvent être nuls. Du fait que M est une matrice triangulaire inférieure, l'équation [II.8] et respectivement l'équation [II.4] a toujours une solution.

On observe que déjà, par la nature de la stratégie de commande, $R(q^{-1})$ contient une partie fixe (équation [II.3]) spécifiée avant la résolution de l'équation [II.4].

Il convient donc de considérer, comme dans le cas du placement, une paramétrisation de $R(q^{-1})$ et $S(q^{-1})$ de la forme :

$$R(q^{-1}) = H_R(q^{-1})R'(q^{-1}) \quad [\text{II.59}]$$

$$S(q^{-1}) = H_S(q^{-1})S'(q^{-1}) \quad [\text{II.60}]$$

Où $H_R(q^{-1})$ et $H_S(q^{-1})$ représentent les parties pré spécifiées de $R(q^{-1})$ et $S(q^{-1})$.

Dans le cas de la poursuite et régulation à objectifs indépendants, $H_S(q^{-1})$ aura en général la forme :

$$S(q^{-1}) = B^*(q^{-1})(q^{-1}) \quad [\text{II.61}]$$

Et l'équation devient dans le cas général :

$$A(q^{-1})S'(q^{-1})H_S'(q^{-1}) - q^{-(d+1)}H_R(q^{-1})R(q^{-1}) = P(q^{-1}) \quad [\text{II.62}]$$

- **Erreur statique**

Pour avoir une erreur statique nulle à une entrée ou perturbation échelon, le voie directe doit contenir un intégrateur numérique. C'est-à-dire que le polynôme $S(q^{-1})$ doit contenir un terme $(1 - q^{-1})$:

$$\begin{aligned} S(q^{-1}) &= B^*(q^{-1})(1 - q^{-1})S'(q^{-1}) \\ &= B^*(q^{-1})H_S'(q^{-1})S'(q^{-1}) \end{aligned}$$

[II . 63]

En introduisant cette expression dans l'équation [II.50] et après simplification on obtient :

$$A(q^{-1})(1 - q^{-1})S'(q^{-1}) + q^{-(d+1)}H_R(q^{-1})R(q^{-1}) = P(q^{-1})$$

[II . 64]

Qui devra être résolue pour obtenir les coefficients correspondant de $S'(q^{-1})$ et $R(q^{-1})$ lorsqu'on utilise un integrateur

II.4.2.3 Poursuite (calcul de $T(q^{-1})$)

Le pré-compensateur $T(q^{-1})$ est calculer pour assurer (confortement à la figure [II.7]) entre la consigne $r(t)$ et $y(t)$ une fonction de transfert :

$$F_{BF}(q^{-1}) = \frac{q^{-(d+1)}B_m(q^{-1})}{A_m(q^{-1})} = \frac{q^{-(d+1)}B_m(q^{-1})T(q^{-1})}{B_m(q^{-1})P(q^{-1})}$$

[II . 65]

De l'équation [II.65] il résulte :

$$T(q^{-1}) = P(q^{-1})$$

[II . 66]

l'entrée de $T(q^{-1})$ étant constituée par la prédiction de la trajectoire $y^*(t + d + 1)$ obtenue par le passage de $r(t)$ à travers le modèle de poursuite $B_m(q^{-1})/A_m(q^{-1})$.

$$y^*(t + d + 1) = B_m(q^{-1})/A_m(q^{-1}) \quad [\text{II} . 67]$$

Et l'équation de la régulation sera donnée par :

$$S(q^{-1}).u(t) + R(q^{-1}).y(t) = P(q^{-1})y^*(t + d + 1) \quad [\text{II} . 68]$$

L'équation [II.68] peut se mettre aussi sous la forme :

$$u(t) = \frac{P(q^{-1})y^*(t + d + 1) - R(q^{-1}).y(t)}{S(q^{-1})} \quad [\text{II} . 69]$$

En tenant compte que $S(q^{-1})$ est de la forme :

$$S(q^{-1}) = s_0 + s_1q^{-1} + \dots + s_{n_s}q^{-n_s} = s_0 + q^{-1}S^*(q^{-1}) = B^*(q^{-1})S'(q^{-1}) \quad [\text{II} . 70]$$

En considérant les expressions de $B^*(q^{-1})S(q^{-1})$, il résulte :

$$s_0 = b_0 \quad [\text{II} . 71]$$

L'équation [II.68] peut encore se mettre sous la forme :

$$u(t) = \frac{1}{b_1} [P(q^{-1})y^*(t + d + 1) - S^*(q^{-1})u(t) - R(q^{-1})y(t)] \quad [\text{II} . 72]$$

Les équations du régulateur pour la poursuite et la régulation à objectif indépendants sous différentes formes sont résumées dans le tableau au- de sous :

$$u(t) = \frac{T(q^{-1})y^*(t+d+1) - R(q^{-1})y(t)}{S(q^{-1})}$$

$$S(q^{-1})u(t) + R(q^{-1})u(t) = T(q^{-1})y^*(t+d+1) = P(q^{-1})y^*(t+d+1)$$

$$S(q^{-1}) = s_0 + q^{-1}S^*(q^{-1})$$

$$u(t) = \frac{1}{b_1} [P(q^{-1})y^*(t+d+1) - S^*(q^{-1})u(t-1) - R(q^{-1})y(t)]$$

$$y^*(t+d+1) = \left(\frac{B_m(q^{-1})}{A_m(q^{-1})} \right) r(t)$$

$$A_m(q^{-1}) = 1 + q^{-1}A_m^*(q^{-1})$$

$$y^*(t+d+1) = -A_m^*(q^{-1})y^*(t+d) + B_m(q^{-1})r(t)$$

$$B_m(q^{-1}) = b_{m1} + b_{m1} q^{-1} \dots$$

$$A_m(q^{-1}) = 1 + a_{m1} q^{-1} + a_{m2} q^{-2} \dots$$

II.4.3 Effets des saturations – Dispositif anti-saturation

L'effet de saturation de l'actionneur peut être gênant, en particulier si le régulateur contient un intégrateur. Comme dans le calcul de la commande $u(t)$ interviennent les valeurs précédentes de cette commande, on remplace ces valeurs calculées précédemment par les valeurs réellement appliquées ce qui est équivalent à l'incorporation d'une « copie » de la caractéristique non linéaire de l'actionneur dans le régulateur. Dans ce cas, la loi de commande s'écrira :

$$u(t) = \frac{T(q^{-1})y^*(t+d+1) - R(q^{-1})y(t) - S^*(q^{-1})\bar{u}(q^{-1})}{s_0}$$

[II . 73]

Dans l'équation [II.73], $\bar{u}(q^{-1})$ est le vecteur constitué par les valeurs précédentes de $u(t)$ passées par la caractéristique non linéaire [1].

Ceci est illustré dans la figure II.8 pour la loi de commande :

$$u(t) = \frac{T(q^{-1})y^*(t+1) - r_0y(t) - s_1\bar{u}(t-1)}{s_0} \tag{II.74a}$$

$$\bar{u} = \begin{cases} u(t) & \text{si } |u(t)| < u_{sat} \\ u_{sat} & \text{si } u(t) \geq u_{sat} \\ -u_{sat} & \text{si } u(t) \leq -u_{sat} \end{cases} \tag{II.7b}$$

Correspondant à la poursuite et régulation à objectifs indépendants pour un modèle échantillonné de procédé décrit par :

$$y(t+1) = -a_1y(t) + b_1u(t) + b_2u(t-1) \tag{II.75}$$

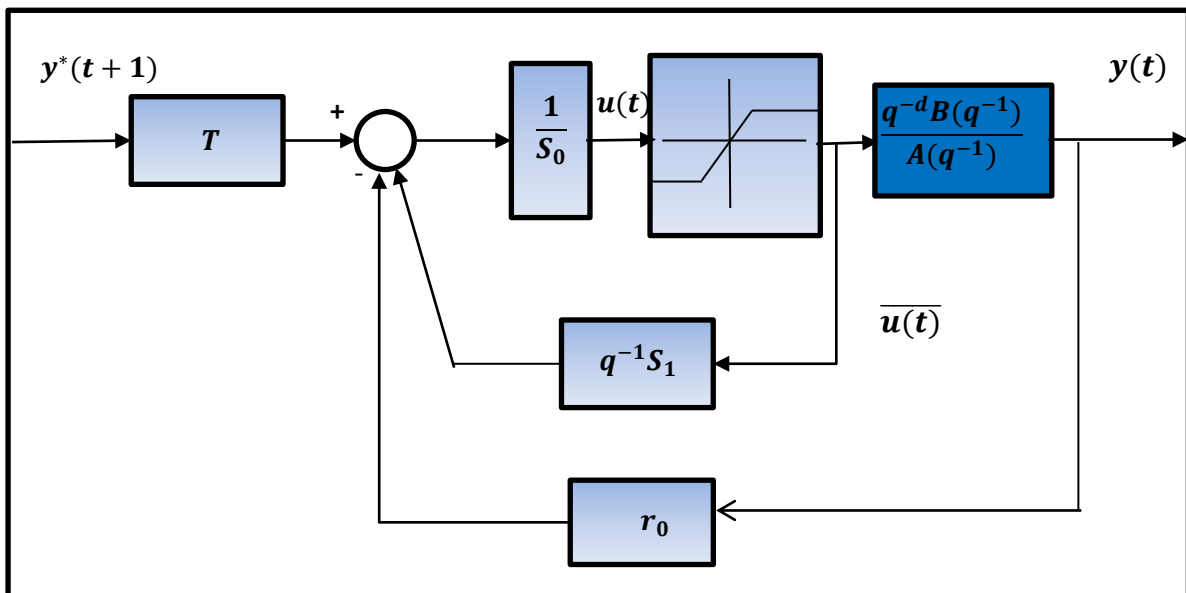


Figure II.8 : Régulation numérique en présence de saturation(dispositif anti-saturation)

On peut aussi imposer une dynamique de sortie de saturation différente d'un simple gain. Un tel schéma est représenté Figure, ou la dynamique de sortie de saturation est définie par la polynôme :

$$P_S(q^{-1}) = s_0 + q^{-1}P_S^*(q^{-1})$$

[II . 76]

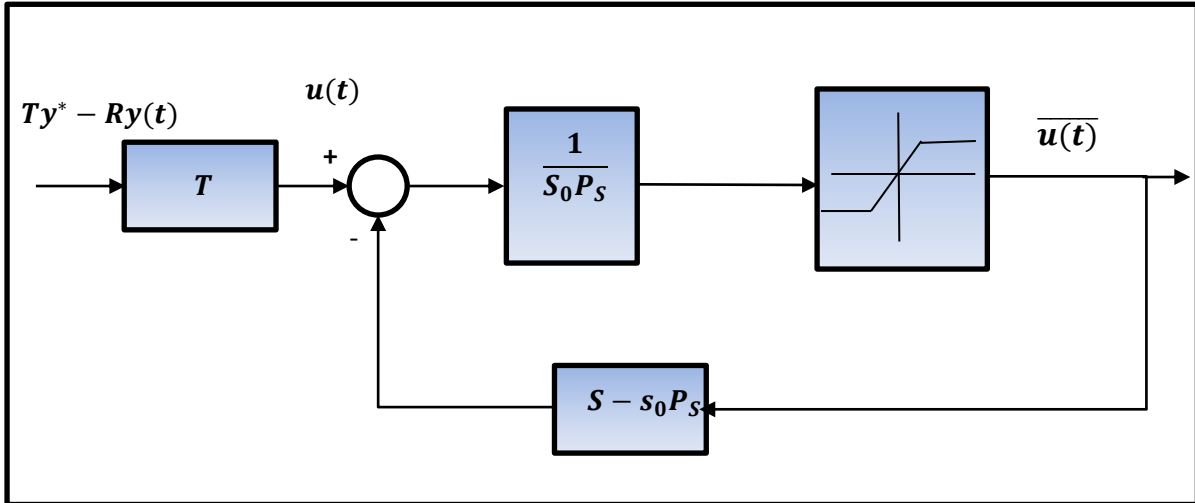


Figure II.9 : Dispositif anti-saturation avec dynamique de sortie de saturation

Dans le domaine linéaire $|u(t)| < |u_{sat}| \rightarrow \bar{u}(t) = u(t)$, ce système a la fonction de transfert $1/S(q^{-1})$. En effet, de la figure II.9 il vient :

$$\frac{\frac{1}{P_S(q^{-1})}}{1 - \frac{S(q^{-1}) - P_S(q^{-1})}{P_S(q^{-1})}} = \frac{1}{P(q^{-1}) + S(q^{-1}) - P(q^{-1})} = \frac{1}{S(q^{-1})}$$

[II . 77]

L'équation du régulateur s'écrit dans le cas général :

$$P_S(q^{-1})u(t) = T(q^{-1})y^*(t + d + 1) - R(q^{-1})y(t) - [S(q^{-1}) - P_S(q^{-1})]\bar{u}(q^{-1})$$

[II . 78]

Et respectivement :

$$u(t) = \frac{T(q^{-1})y^*(t+d+1) - R(q^{-1})y(t) - [S(q^{-1}) - P_S(q^{-1})]\bar{u}(q^{-1}) - P_S^*(q^{-1})u(t-1)}{s_0} \quad [\text{II.79}]$$

Mais

$$S(q^{-1}) - P(q^{-1}) = s_0 + q^{-1}S^*(q^{-1}) - [s_0 + q^{-1}P_S^*(q^{-1})] = q^{-1}[S^*(q^{-1}) - P_S^*(q^{-1})] \quad [\text{II.80}]$$

On obtient alors, pour une régulation numérique les équations suivantes :

$$u(t) = \frac{1}{s_0} (T(q^{-1})y^*(t+d+1) - R(q^{-1})y(t) - [S(q^{-1}) - P_S(q^{-1})]\bar{u}(q^{-1}) - P_S^*(q^{-1})u(t-1)) \quad [\text{II.81a}]$$

$$\bar{u} = \begin{cases} u(t) & \text{si } |u(t)| < u_{sat} \\ u_{sat} & \text{si } u(t) \geq u_{sat} \\ -u_{sat} & \text{si } u(t) \leq -u_{sat} \end{cases}$$

[II.81b]

II.4.4 Avantages et inconvénients du régulateur RST

II.4.4.1 Avantages

L'intérêt de l'utilisation du régulateur RST contrairement à beaucoup d'autres correcteurs numériques tel le PID classique est de spécifier la loi de rejet des perturbations indépendantes de celles de la poursuite du signal de consigne. Pour cette raison le signal est dit de « poursuite de régulation à objectif indépendant ». Un autre avantage du RST est que la grandeur de commande peut s'écrire sous forme d'une équation récurrente qui peut être facilement implantée par des algorithmes récursifs [4].

II.4.4.2 Inconvénient

Si le système à corriger comporte des racines stables, ces racines peuvent être simplifiées si on les place dans $R(z)$, ce qui rappelle la méthode des régulateurs classiques.

II.5 Conclusion

Dans ce chapitre, plusieurs méthodes de calcul de régulateurs numériques opérant dans un environnement déterministe ont été présentées.

Tous les régulateurs numériques ont une structure canonique à trois branches (R-S-T), correspondant à une lois de commande de la forme :

$$S(q^{-1})u(t) + R(q^{-1})y(t) = T(q^{-1}) y^*(t + d + 1)$$

Qui peut être exprimée aussi sous la forme :

$$u(t) = \frac{T(q^{-1})y^*(t + d + 1) - R(q^{-1})y(t)}{S(q^{-1})}$$

Où u est la commande, y la sortie du processus, y^* la trajectoire désirée en poursuite et d le retard pur discret .

Le calcul du régulateur s'effectue en 2 étapes :

1. Calcul des Polynômes $S(q^{-1})$ et $R(q^{-1})$ pour obtenir les performances désirées en régulation .
2. Calcul du Polynômes $T(q^{-1})$ pour approcher (ou atteindre parfaitement) les performances désirées en poursuite.

Les régulateurs numériques sont parfaitement adaptés pour la régulation des procédés ayant des retards purs. La complexité du régulateur (c'est -à-dire les degrés des polynômes $S(q^{-1}), R(q^{-1}), T(q^{-1})$) dépend uniquement des degrés des polynômes de la fonction de transfert échantillonnée du modèle du procédé.

Ce modèle peut être obtenu soit directement par une technique d'identification, soit par discrétisation d'un modèle continu de procédé.

La stratégie de commande poursuite et régulation à objectif indépendant permet d'atteindre les performances désirées en poursuite et en régulation, mais pour qu'elle soit applicable, la fonction de transfert échantillonnée du modèle discret du procédé doit avoir des zéros stables.

La stratégie de commande placement de pôles s'applique aux procédés ayant des modèles discrets avec zéro aussi bien stable. Elle permet d'atteindre les performances désirées en régulation et les performances désirées en poursuite filtrées par les zéros du procédé.

III.1 Introduction :

L'évolution des technologie conduit à utiliser des machines nécessitant des vitesses de rotation précises et variable pour l'entraînement d'engins industriels, système robotique...etc .Dans notre étude on s'intéresse aux moteur a courant continu, car ils offrent une introduction facile aux fonctionnement de ses homologues, en donnant des repères clairs. Les moteurs à courant continu ont pendant longtemps été les seuls à la vitesse variable à large bonde passante, comme dans le domaine robotique. Ils ont donc fait l'objet de nombreuses améliorations, et beaucoup de produits commercialisés aujourd'hui.

Dans ce chapitre, nous donnerons une description succincte d'un moteur à courant continu avec son fonctionnement non exposons ensuite la méthode de modélisation d'un moteur à courant continu basé sur le principe fondamental de la dynamique .finalement les principes de base de l'identification paramétrique plus particulièrement l'algorithme des moindres carrés.

III.2 Description d'un moteur à courant continu

III.2.1 Introduction

Beaucoup d'application nécessite un couple de démarrage élevé. Or, le moteur à courant continu par nature, possède une caractéristique couple /vitesse de pente importante, ce qui permet de vaincre un couple résistant élevé, et d'absorber facilement les à couple de charge, la vitesse de moteur s'adapte à sa charge. d'autre part la miniaturisation recherchée par le concepteurs trouve dans le moteur continu une solution idéale, puisque présentant un rendement élevé, en comparaison aux autre technologies.

III.2.2 Fonction d'usage

Le moteur à courant continu est un convertisseur électrique en énergie mécanique avec quelque perte. Un moteur à courant continu est une machine électrique. Il s'agit d'un convertisseur électromécanique permettant la conversion bidirectionnelle d'énergie entre une installation électrique parcourue par un courant continu et un dispositif mécanique en rotation d'où l'énergie électrique est transformée en énergie mécanique.

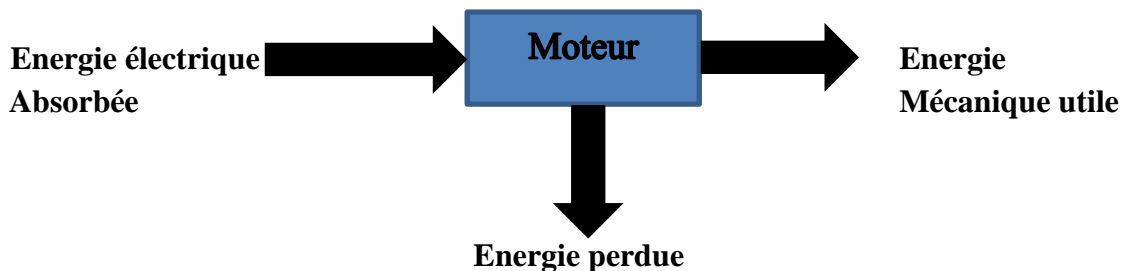


Figure III.1 : Fonctionnement d'un moteur à courant continu

III.2.3 Principe

le moteur à courant continu est essentiellement composé de trois parties :

- **le stator ou inducteur** : il est formé d'aimants pour les « petits » et de bobine parcourues par un courant pour les moteurs plus puissants. grâce aux progrès sur les aimants ses derniers sont plus en plus employés. Les aimants ou bobines permettent la création d'un flux magnétique.
- **le rotor ou l'induit** : il est alimenté par une tension u et parcouru par un courant I . le circuit électrique appelé induit est obtenu en associant en série des conducteurs logés dans des encoches du rotor.
- **Le collecteur** : il est formé d'un ensemble de lames de cuivre isolées latéralement les unes des autres et disposées suivant un cylindre en bout rotor. Deux balais portés par le stator frottent sur les lames du collecteur.

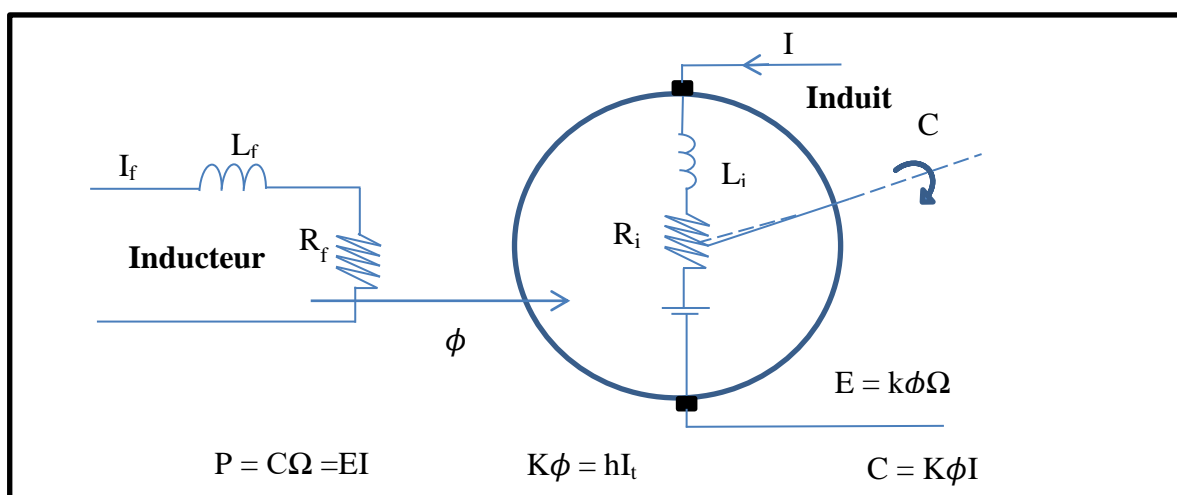


Figure III.2 : Schéma équivalent d'un moteur à courant continu

III.3 Modélisation d'un moteur à courant continu

III.3.1 Généralité sur la commande en vitesse variable des machines électriques

Comme tous les moteur électrique, la machine à courant continu doit être considérée comme un générateur de couple .ce couple est alors transmis à la charge qui peut entrer en rotation. La vitesse de l'ensemble moteur + charge n'est donc pas imposée par le moteur, mais par la caractéristique couple /vitesse de l'ensemble moteur + charge. le point de fonctionnement de l'ensemble de la motorisation est situé à la l'intersection de la caractéristique mécanique de l'ensemble moteur + charge et de la caractéristique $C_{moteur} = c^{te}$.pour faire vérifier la vitesse de la rotation de l'ensemble, il faut mouler le couple fournis par le moteur à la charge. Si l'on souhaite augmenter la vitesse de rotation de l'ensemble moteur + charge, il faut augmenter le couple fournis par le moteur, qui devient alors supérieur au couple de charge. Il apparait alors un couple d'accélération $(j \frac{d\omega}{dt})$, ce qui entraine une augmentation de la vitesse de la charge [3].

L'ensemble d'une commande à vitesse variable, quel que soit le type de motorisation utilisée, peut donc se décomposer en deux boucles d'asservissement : une boucle de coupe et une boucle de vitesse.

III.3.2 Contrôle de vitesse d'une machine à courant continu (opération en boucle fermée)

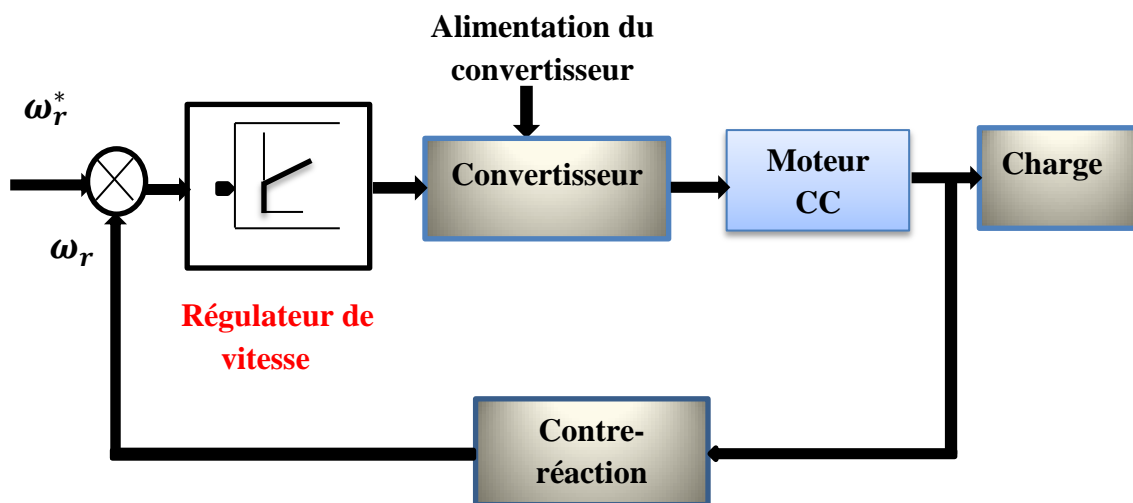


Figure III.3 : Schéma synoptique d'un contrôleur de vitesse machine à courant continu

Pour pouvoir calculer les paramètres du régulateur de vitesse, il faut d'abord avoir un modèle complet du moteur à courant continu et sa charge, ainsi celui de convertisseur.

III.3.3 Modélisation de la machine à courant continu (modèle électro-magnéto-mécanique)

La modélisation de l'ensemble moteur + charge est réalisable à partir des équations de base de la machine à courant continu et de principe fondamental de la dynamique (PFD).

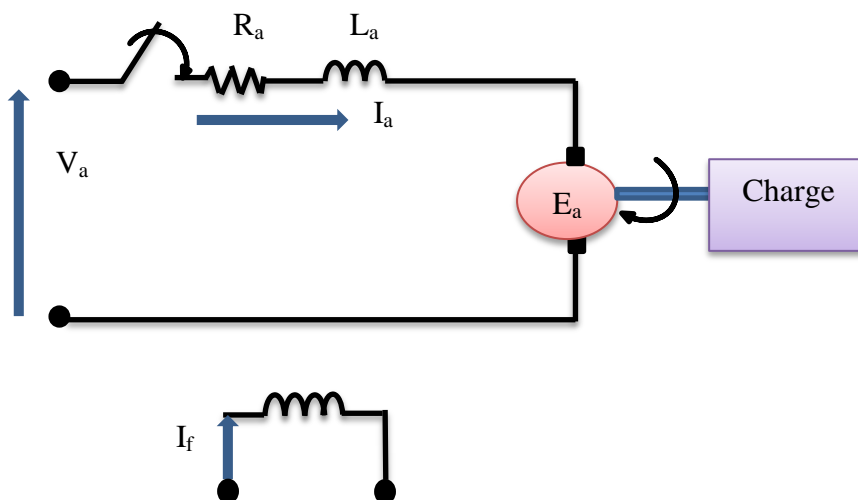


Figure III.4 : Modèle électro-magnéto-mécanique d'un moteur à courant continu

➤ **Modèle électrique :** $V_a = R_a I_a + L_a \frac{dI_a}{dt} + E_a$

[III . 1]

Où :

- E_a : force contre électromotrice générée par le moteur.- V_a : la tension d'induit.- L_a, R_a : l'inductance d'induit et sa résistance.

➤ **Modèle magnétique :**

$$\begin{cases} E_a = K_a J \omega_m \\ C = K_a J I_a \end{cases}$$

[III . 2]

Où :

- C : le couple électromagnétique fourni par le moteur.- K_a : le coefficient de tension induit.- ϕ : le flux d'excitation.

➤ **Modèle mécanique :** $C - C_r = J \frac{d\omega}{dt} + F\omega$

[III . 3]

Où :

- F : le coefficient de frottement visqueux de l'ensemble moteur + charge.- J : le moment d'inertie de l'ensemble moteur + charge.- C : le couple moteur.- C_r : le couple résistant.- ω : la vitesse angulaire de moteur.- $J \frac{d\omega}{dt}$: le couple d'accélération.

➤ **Modèle électro- magnéto-mécanique :**

$$(V_a - E_a) = \frac{1}{R_a + sL_a} I_a$$

$$(C - C_r) \frac{1}{F + sJ} = \omega$$

$$E_a = K_a J \omega_m$$

$$C = k_a J I_a$$

[III . 4]

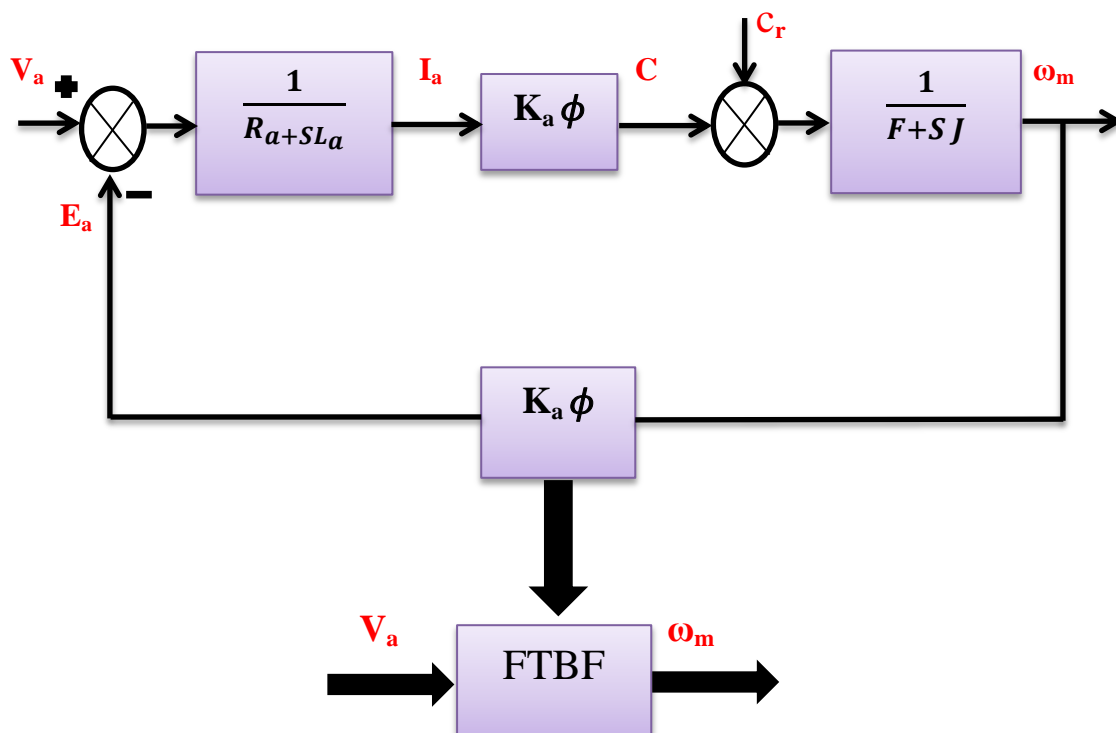


Figure III.5 : Modèle électro-magnéto- mécanique d'moteur à courant continu en boucle fermée.

$$FTBF = \frac{\omega_m}{V_0} = \frac{G}{1+GH} \tag{ III . 5 }$$

$$K = \frac{(K_a \phi)}{(K \phi)^2 + R_a F} \quad \text{Gain statique} \tag{ III . 6 }$$

$$\omega_0^2 = \frac{1}{K} \frac{(K_a \phi)}{L_a J} \quad \text{fréquence propre} \tag{ III . 7 }$$

$$\frac{\omega_m}{Va} = K \frac{1}{S^2 + 2\xi\omega_0 S + \omega_0^2} \quad \text{tel que} \quad \xi = K \frac{\omega_0 R_a J + L_a F}{(K_a \phi)} \quad \text{Amortissement}$$

[III . 8]

III.4 Identification des modèles dynamiques d'un moteur à courant continu

III.4.1 Principe de l'identification des modèles dynamiques de procédés

L'identification, est l'opération de détermination des caractéristiques dynamiques d'un procédé (système) en les traduisant sous forme d'un modèle structural (fonction de transfert ou espace d'état) dont la connaissance est nécessaire pour la conception et la mise en œuvre d'un système performant de commande (régulation et asservissement).

La figure III.6 résume les principes généraux de conception et calcul d'un régulateur. Pour concevoir et ajuster correctement un régulateur, il :

1. Spécifier les performances désirées pour la boucle de commande régulation.
2. Connaître le modèle dynamique d'un procédé (appelé aussi « modèle de commande ») qui décrit la relation entre les variations de la commande et les variations de la sortie.
3. Disposer d'une méthode appropriée de calcul régulateur compatible avec les performances spécifiées et les caractéristiques du modèle du procédé.

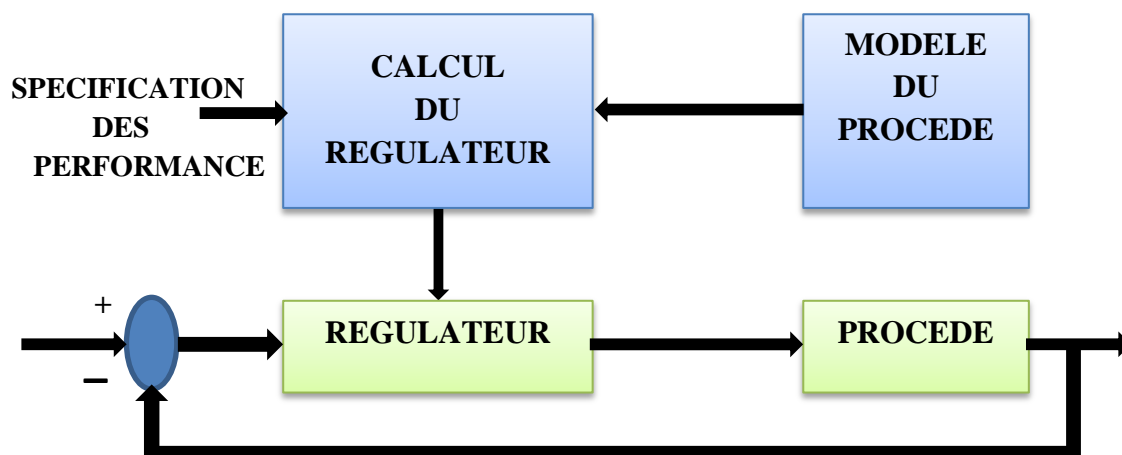


Figure III.6 : Principe de la conception et du calcul d'un régulateur.

La notion de modèle mathématique d'un système ou d'un phénomène est un concept fondamental. En général il existe une multitude de type de modèle, chaque type de modèle étant destiné à une application particulière.

Par exemple, les modèle de connaissance (basé sur les lois de la physique, chimie, etc...) permettant une description assez complète des système et son utilisé pour la simulation et la conception des procédés.

Ces modèles sont en générale extrêmement complexes et rarement directement utilisable pour l'automatique. Les modèle dynamique de la commande, qui donnent le relation entre les variations des entrées d'un système et les variations de la sortie sont comme on l'a indiqué plus haut de type de modèle dont on doit disposer pour la conception et l'ajustement des système de commande régulation.

Bien que des indications sur la structure de ces modèles de commande puissent s'obtenir à partir de la structure du modèle de connaissance, il est en générale très difficile de déterminer les valeurs des paramètres significatifs à partir de ces modèles [1].

C'est la raison pour laquelle, dans la majorité des situations pratiques, on est amené à mettre en œuvre une méthodologie d'identification directe de ces modèles dynamiques.

Notons que les modèles dynamiques sont de deux types :

1. Modèles non paramétriques

Exemple : (réponse fréquentielle, réponse à un échelon).

2. Modèles paramétriques

(Exemple : fonction de transfert, équation différentielle ou aux différences).

Nous nous intéresserons par la suite à l'identification des modèles dynamiques paramétrique échantillonnés qui sont les plus appropriés pour la conception et l'ajustement des systèmes à commandés et régulation numériques.

L'identification est une approche expérimentale pour la détermination du modèle dynamique d'un système. elle comporte quatre étapes :

1. Acquisition des entrées /sortie sous un protocole d'expérimentation.
2. Choix de la structure du « modèle » (complexité).
3. Estimation des paramètres du modèle.
4. Validation de modèle identifié (structure et valeur des paramètres).

La première étape fournit les données entrées /sorties pour un type de modèle permet de déterminer l'extraction d'un modèle de procédé significatif, le protocole d'acquisition est un problème clé car il conditionne en grande partie le succès d'une opération d'identification .il doit tenir en compte d'une part des contraintes pratiques, ce qui requiert en général l'utilisation des signaux d'excitation de faible amplitude et d'autre part de la nécessité

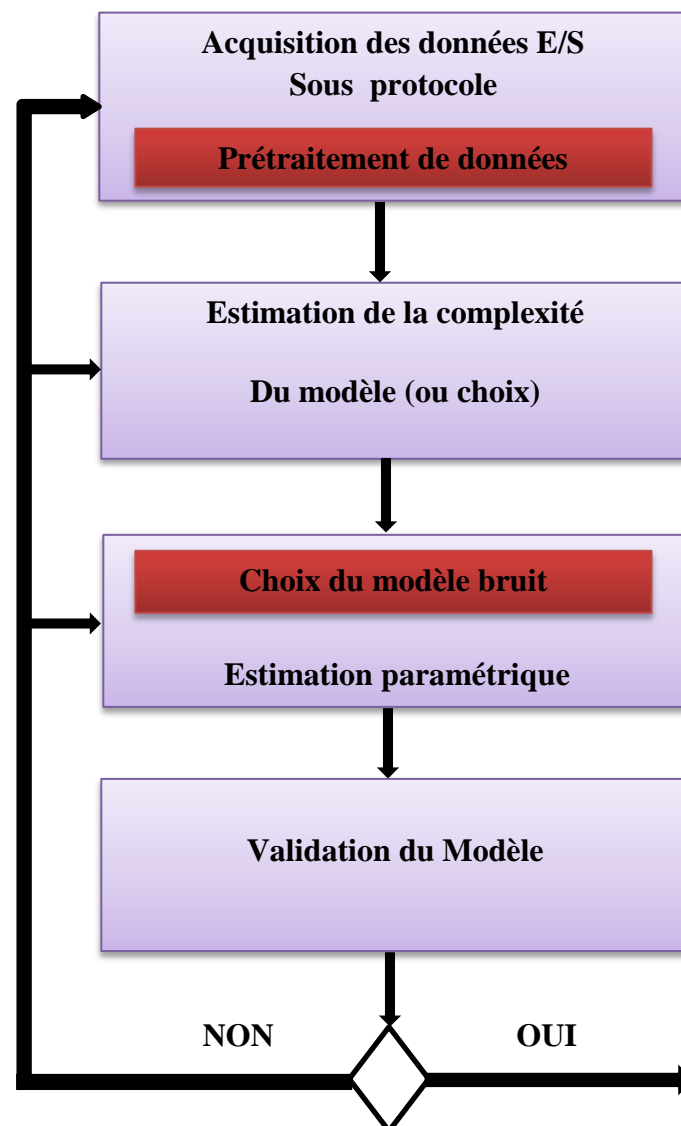


Figure III.7 : Méthodologie de l'identification

d'exciter le système identifié dans une boucle de fréquence suffisamment large afin d'obtenir un modèle significatif pour le calcul de la commande.

L'estimation ou le choix de la complexité pour un type de modèle permet de déterminer quels sont les paramètres à estimer et quel est leur nombre.

L'estimation paramétrique utilise des algorithmes qui, à partir des mesures entrées/sorties disponible fournissent les paramètres du modèle, en fin, la dernière étape doit permettre de mettre en évidence si le modèle identifier est représentatif des comportements entrées/sorties du procédé.

Une opération d'identification complète doit nécessairement comporter les quatre phases indiquées plus haut. les méthodes spécifique utilisées dans chaque phase dépendent du type de modèle recherche (paramétrique ou non paramétrique, continu ou échantillonné).

Dans notre étude on s'intéresse par la suite à l'identification des modèles paramétriques échantillonné [1].

III.4.2 Le principe de l'algorithme des moindres carrés

Un algorithme d'identification de moindre carrés peuvent être utilisé pour l'estimation des paramètres d'un modèle à partir des entrées/sortes .la référence a été donnée au algorithme pour différentes raison parmi lesquelles nous mentionnerons :

- Estimation des paramètres du modèle au fur et à mesure de l'évolution du procédé.
- Economie de mémoire et de puissance de calcul.
- Capacité d'identification en temps réel.

Il n'y a pas de structure unique procédé + perturbation pour décrire toutes les situations rencontrées en pratiques, et il n'y a pas non plus une méthode unique d'identification donnant dans toutes les situations des estimations non biaisées.

Pour identifier effectivement un modèle de procédé, un système interactif d'identification est nécessaire .il doit comporter :

- Différentes structures « modèle +perturbation ».
- Différentes méthodes d'identification.
- Des méthodes de validation de modèle identifié.
- Un système d'acquisition et traitement des données entrées /sortie.
- Des outils d'analyse des modèles.
- Un système de visualisation graphique.

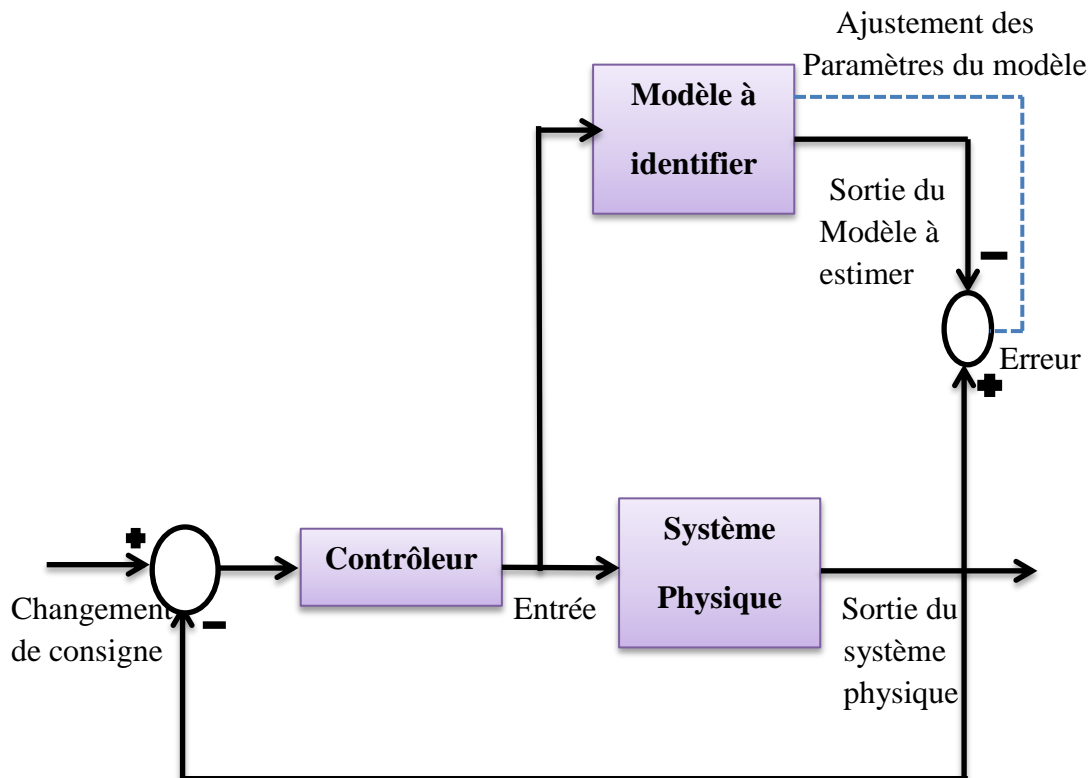
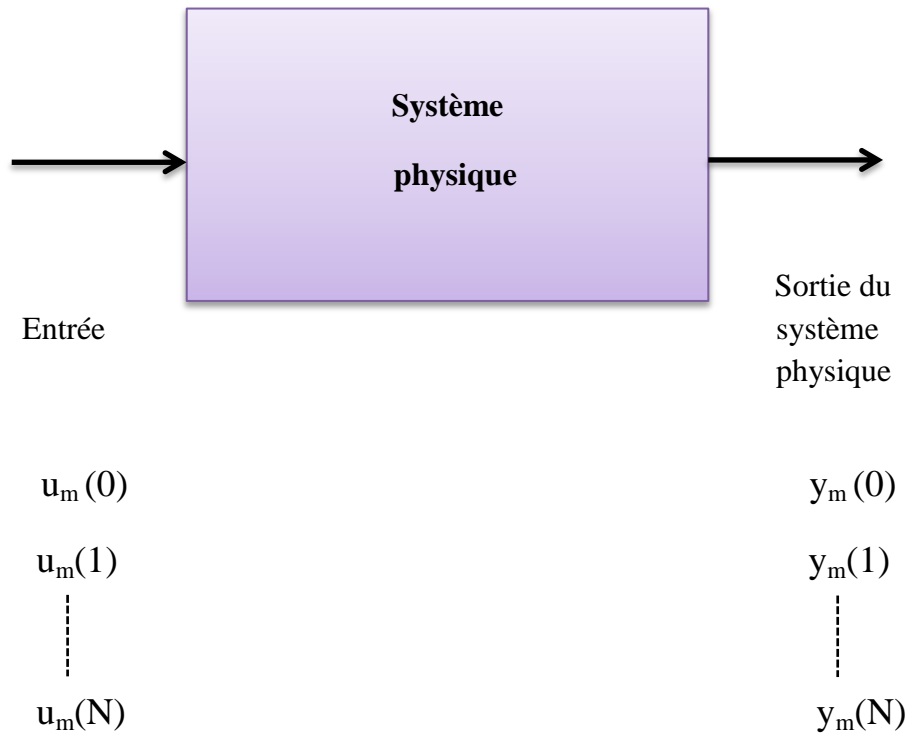


Figure III.9 : Identification d'un système en fonction

III.4.3 Méthode d'identification par moindre carré

- Cueillette des entrées sorties du système.
- Choix de l'ordre du système a identifié.
- Calcul des paramètres du modèle de façons à minimiser l'erreur quadratique.

III.4.4 Cueillette des entrées sorties du système



III.4.5 Choix de l'ordre de système

Si l'ordre du système est inconnu, en se basant sur la connaissance a priori sur le système ou suite à une analyse de la boucle de sa sortie mesurée on peut faire un choix approximatif, sinon on choisit par défaut l'ordre égale à 1, puis à 2, ainsi de suite jusqu'à ce que le modèle soit satisfaisant.

La fonction de transfert en Z d'ordre n du modèle prend alors la forme suivante :

$$G(z) = \frac{Y(z)}{U(z)} = \frac{b_{n-1} z^{n+1} + b_{n-2} z^{n+2} + \dots + b_0}{z^n + a_{n-1} z^{n-1} + a_0}$$

[III . 9]

La transformation en Z inverse nous permet alors d'obtenir :

$$Y(k) = - a_{n-1}y(k-1) - a_{n-2}y(k-2) - a_0y(k-n)$$

$$+ b_{n-1}u(k-1) + b_{n-2}u(k-2) + \dots + b_0u(k-n)$$

[III . 10]

III.4.6 Calcul des paramètres du modèle

L'équation récurrente du modèle

$$Y(k) = -a_{n-1}y(k-1) - a_{n-2}y(k-2) - \dots - a_0y(k-n)$$

$$b_{n-1}u(k-1) + b_{n-2}u(k-2) + \dots + b_0u(k-n)$$

Peut-être réécrite sous la forme :

$$Y(k) = \phi(k)p$$

[III . 11]

$$P = [a_{n-1} \ a_{n-2} \ \dots \ a_0 \quad b_{n-1} \ b_{n-2} \ \dots \ b_0]$$

$$\phi(k) = [-y(k-1) \ -y(k-2) \ \dots \ -y(k-n) \quad u(k-1) \ u(k-2) \ \dots \ u(k-n)]$$

Ainsi :

$$y = \begin{bmatrix} y(n) \\ y(n+1) \\ \vdots \\ y(N) \end{bmatrix} = \begin{bmatrix} \phi(n) \\ \phi(n+1) \\ \vdots \\ \phi(N) \end{bmatrix} P = \phi p$$

[III . 12]

Où

$$\phi = \begin{bmatrix} -y(n-1) & -y(n-2) & \dots & -y(0) & u(n-1) & u(n-2) & \dots & u(0) \\ -y(n) & -y(n-1) & \dots & -y(1) & u(n) & u(n-1) & \dots & u(1) \\ \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \dots & \vdots \\ -y(N) & -y(N-2) & \dots & -y(N-n) & u(N-1) & u(N-2) & \dots & u(N-n) \end{bmatrix}$$

[III . 13]

L'erreur quadratique entre la sortie de modèle et celle du système physique est alors :

$$\begin{aligned}\varepsilon &= \sum_{k=n}^N (y_m(k) - \phi_m(k)p)^2 = (y_m - \phi_m p)^T (y_m - \phi_m p) \\ &= p^T \phi_{Tm}^T \phi_m p - 2 p^T \phi_m^T Y_m^T Y_m\end{aligned}$$

[III . 14]

Où

$$Y_m = \begin{bmatrix} y_m(n) \\ y_m(n+1) \\ \vdots \\ y_m(N) \end{bmatrix}$$

$$\phi_m = \begin{bmatrix} -y_m(n-1) & -y_m(n-2) & \dots & -y_m(0) & u_m(n-1) & u_m(n-2) & \dots & u_m(0) \\ -y_m(n) & -y_m(n-1) & \dots & -y_m(1) & u_m(n) & u_m(n-1) & \dots & u_m(1) \\ \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \dots & \vdots \\ -y_m(N) & -y_m(N-2) & \dots & -y_m(N-n) & u_m(N-1) & u_m(N-2) & \dots & u_m(N-n) \end{bmatrix}$$

Minimisation de l'erreur quadratique

$$\frac{\partial \phi}{\partial p} = 2\phi_{Tm}^T \phi_m p - 2\phi_m^T Y_m^T Y_m = 0$$

D'où

$$P = [\phi_{Tm}^T \phi_m]^{-1} \phi_m^T Y_m^T Y_m \quad -$$

III.5 Conclusion

Le processus d'identification /modélisation est une étape essentielle pour mener à bien la synthèse des boucles de régulation des systèmes, notamment en commande numérique par ordinateur.

Dans ce chapitre, nous avons présenté le modèle de connaissance du moteur à courant continu candidat à une application de la commande numérique RST dans le chapitre prochain.

Nous avons aussi présenté une autre méthode de modélisation de moteur à courant continu, jugée de notre part plus importante et plus sollicitée dans le domaine industriel, c'est l'identification paramétrique d'un processus dynamique. c'est cette méthode qui va nous fournir dans le chapitre prochain, le modèle utilisé dans la synthèse des lois de commande pour notre moteur à courant continu.

|

IV.1 Introduction

Malgré les progrès scientifiques et technologiques qu'ont connus les machines asynchrones, la machine à courant continu reste encore la machine par excellence pour les applications demandant un contrôle de vitesse.

Dans ce chapitre purement pratique L'objectif est de travailler sur la réalisation d'une commande numérique RST sur un bon d'essai *FeedBack MS150*.

En première partie on présentera l'environnement de cette étude qui nous permettra d'élaborer les étapes de l'identification du process et de la commande en temps réel dans le but d'introduire le lecteur à la compréhension de notre travail.

Ensuite on procédera à l'identification de la fonction de transfert de notre système à commander qui est principalement un moteur à courant continu.

En dernier, en utilisant le logiciel de calcul *Matlab* on va générer la loi de commande et vérifier son exactitude sous *Simulink* afin de pouvoir implémenter l'algorithme sur une carte *DSP* (digital signal processing) *dSPACE* 1104 qui nous offre la possibilité de commander le process en temps réel et le superviser sous l'interface graphique *ControlDesk*.

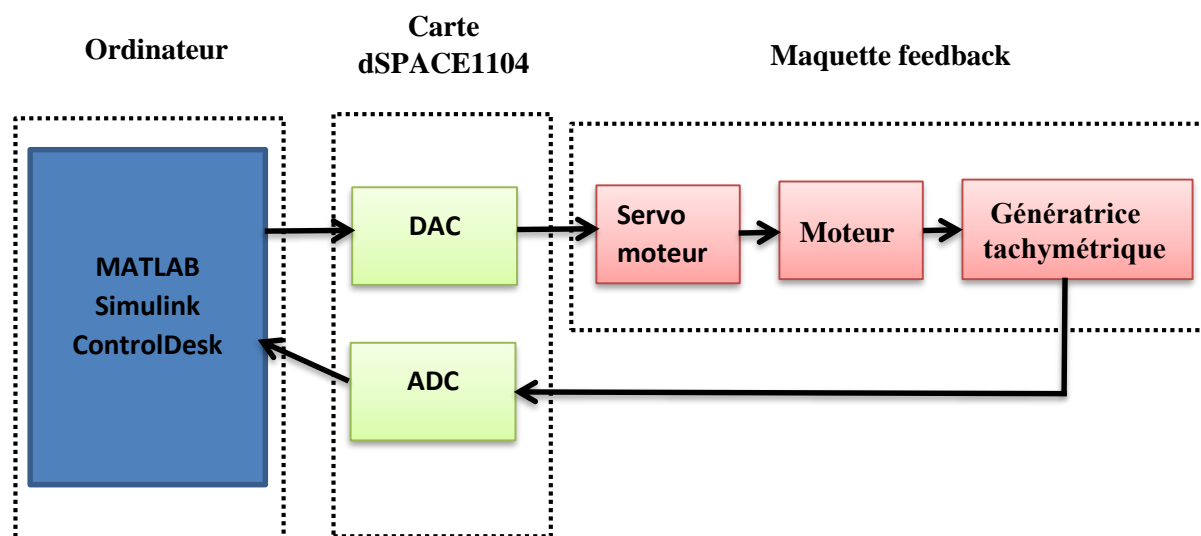


Figure IV.1 : Environnement de l'étude

IV.2 Système *dSPACE* :

IV.2.1 Configuration requise

La configuration minimale requise pour utiliser le système *dSPACE* est :

Processeur : Pentium 4 at 2 GHz (ou équivalent)

Mémoire vive : 2 GB RAM

Espace disque : 5.5 GB

Pour la licence (Dongle License) : un port USB pour installer le Dongle

Un slot : pour installer la DS1104, il faut un slot de libre dans la carte mère du PC (33 MHz/32-bit 5 V PCI slot).

IV.2.2 Carte *dSPACE* :

Le pilotage en temps réel des systèmes continus se fait à l'aide d'un PC connecté à la carte *dSPACE* DS1104. La programmation se fait à l'aide de l'outil de modélisation *Simulink*, qui aide à poser le problème d'une manière graphique en utilisant les blocs interconnectés. En fait, beaucoup de systèmes de développement en temps réel à base de *DSP* viennent maintenant avec une interface à *Simulink* par laquelle ils peuvent convertir les blocs de *Simulink* en un code machine qui peut être exécuté sur un système à base de *DSP*. Ceci réduit considérablement le temps de développement et de prototypage pour le contrôle des systèmes. Le prototypage passe alors par trois étapes :

- 1. Construction du système de commande en utilisant les blocs de *Simulink*.
- 2. Simulation du système pour voir les résultats dans différents scénarios.
- 3. Exécution du modèle en temps réel à travers la carte DS1104.

Le processeur principal est un MPC8240, avec une horloge de 250mHz. Les caractéristiques de la carte sont données dans le tableau de la figure 1 (document constructeur).

Parameter		Specification
Processor		<ul style="list-style-type: none"> ■ MPC8240 processor with PPC 603e core and on-chip peripherals ■ 64-bit floating-point processor ■ CPU clock: 250 MHz ■ 2 x 16 KB cache, on-chip ■ On-chip PCI bridge (33 MHz)
Memory	Global memory	■ 32 MB SDRAM
	Flash memory	■ 8 MB
Timer	4 general-purpose timers	<ul style="list-style-type: none"> ■ 32-bit down counter ■ Reload by hardware ■ 80-ns resolution
	1 sampling rate timer (decrementer)	<ul style="list-style-type: none"> ■ 32-bit down counter ■ Reload by software ■ 40-ns resolution
	1 time base counter	<ul style="list-style-type: none"> ■ 64-bit up counter ■ 40 ns resolution

Figure IV.2 : Les caractéristiques de la carte

Pour implémenter la commande temps réel en utilisant la carte « dSPACE DS1104 R&D Controller Board » et le logiciel Matlab, on a besoin de :

- 1) La carte de contrôle *dSPACE DS1104*
- 2) La licence sous forme de clé USB (Dongle)
- 3) La licence software (fichier Licence.dsp)
- 4) Le fichier Keys.dsp
- 5) Et enfin le panneau de connexion *CP1104*



**le panneau de connexion
CP1104**

**La licence sous forme de clé
USB (Dongle)**



**La carte de contrôle dSPACE
DS1104**

Figure IV.3 : Les composants du système dSPACE

IV.2.3 Control Desk

ControlDesk est une interface qui permet de visualiser en temps réel différentes variables du Fichier développé sous *Simulink* et de modifier également des paramètres définissant le mode de fonctionnement des blocs constituant le schéma *Simulink*. La visualisation de variables ou de signaux et la modification de paramètres sont possibles par l'intermédiaire d'instruments graphiques que l'on sélectionne.

IV.3 Identification

L'identification constitue une phase importante dans la connaissance du modèle de comportement, c'est par elle que le choix de la classe de modèle à adopter puis les valeurs des paramètres qui le caractérisent vont se préciser. Le plus souvent l'identification s'effectue en optimisant un critère de qualité qui caractérise l'écart entre le comportement du processus (repéré par un ensemble de mesures) et celui de son modèle (étudié par simulation) pour un ensemble de sollicitations données (signal SBPA dans notre travail).

Parmi les méthodes d'identification connues, la méthode d'erreur de sortie (OE).

L'opération d'identification présente une grande sensibilité pour mener à bien la synthèse et la génération de la commande donc elle doit se faire convenablement d'une manière répétitive.

IV.3.1 Système à identifier (maquette *feedback* MS150)

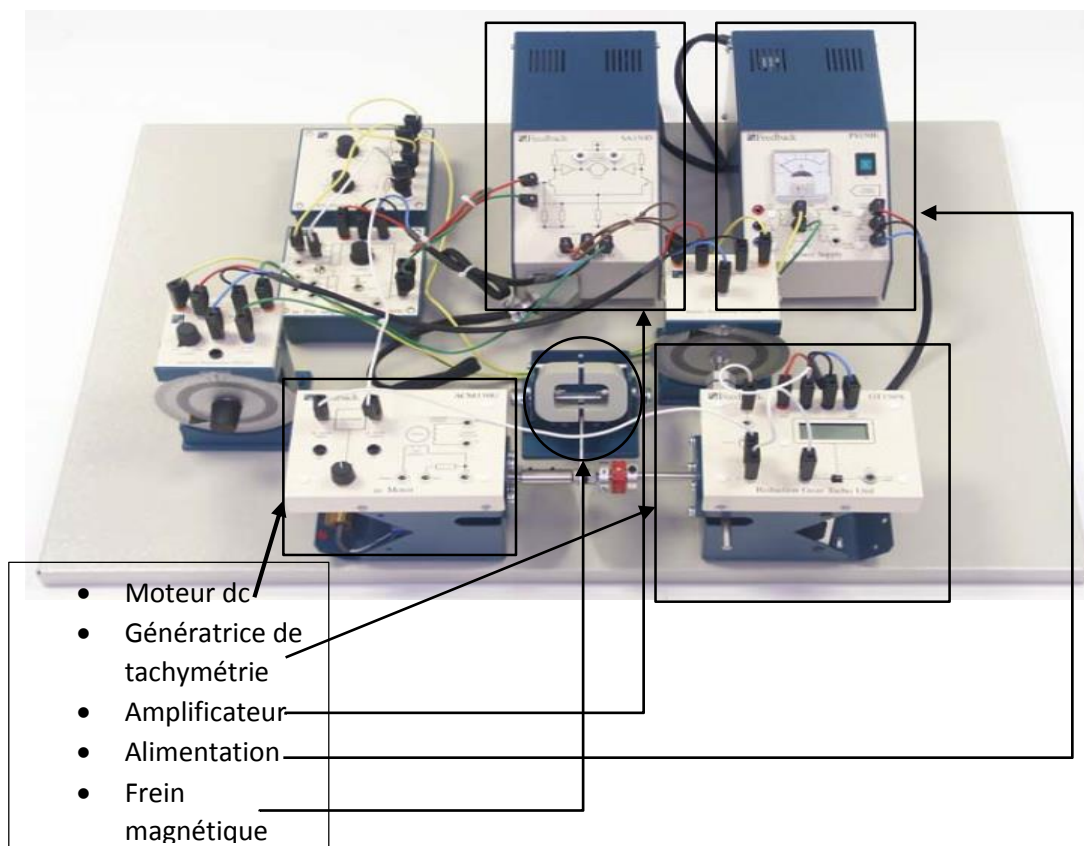


Figure IV.4 : Système à identifier maquette feedback MS150

Pour les besoins de notre travail (commande numérique) on exploitera exclusivement les composants de la maquette encadrés

- Moteur à courant continu *150F* :

C'est un moteur cc à excitation indépendante (universelle pour des raisons de rapidité de réponse aux variations de signaux) Sur sa plaque signalétique on peut lire :

- vitesse nominale 1400 tr/ min
- tension d'alimentation $V=40v$

- Servo-Amplificateur *150V* :

Contient les circuits électroniques qui lui permettent de recevoir les signaux de commandes extérieurs au système (dans notre cas de la carte de contrôle *DS1104*) et il est connecter à son tour au moteur *150F*.

- Alimentation *150E* :

Alimente le moteur, il délivre une tension de *40V*.

- Frein magnétique :

C'est le composant de la maquette *MS150* qui nous permettra de créer des perturbations une fois la commande en vitesse réalisée.

IV.3.2 Qu'est-ce que la SBPA ?

La SBPA (Séquence Binaire Pseudo-Aléatoire) est le signal le plus utilisé comme signal d'excitation en identification c'est un signal formé d'impulsions rectangulaires modulées aléatoirement en longueur, qui approxime un bruit blanc discret, donc riche en fréquence et de valeur moyenne nulle, ne modifiant pas le point de fonctionnement du procédé.il se caractérise

par :

- Centrée : $\pm 10\%$ de la consigne
- Spectre riche : recueillir le maximum d'information sur le système (exciter sur toute la bande de fréquence intéressante)
- Déterministe : Physiquement réalisable
- Amplitude limitée : ne pas trop perturber le process, rester en linéaire

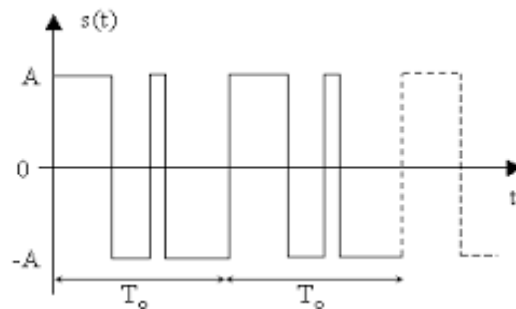


Figure IV.5 : Signal SBPA

IV.3.3 Génération d'un signal SBPA

Pour générer un SBPA on utilise un registre à décalage bouclé par un ou-exclusif. Ce qui suit est le schéma bloc SIMULINK qui nous a servi dans la génération du SBPA

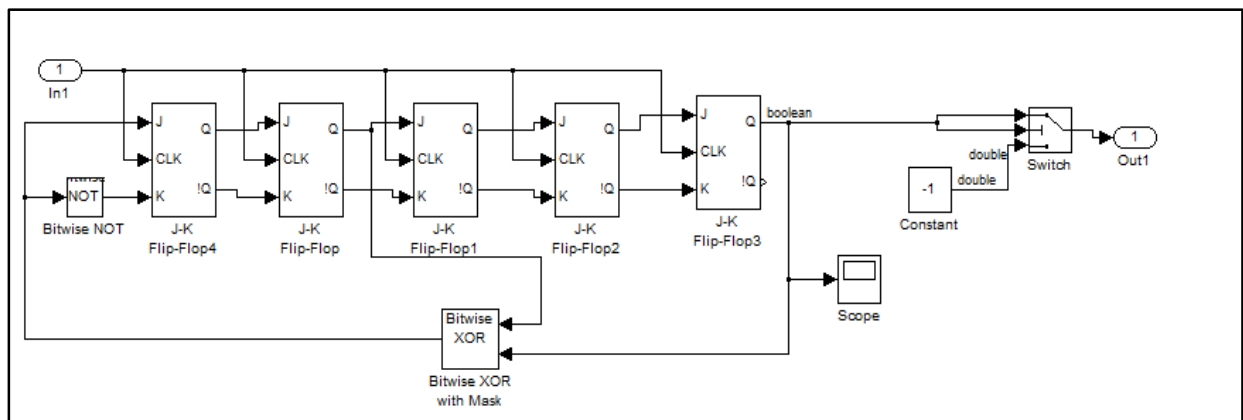


Figure IV.6 : Génération d'un signal SBPA

➤ Le signal SBPA vu dans le scope :

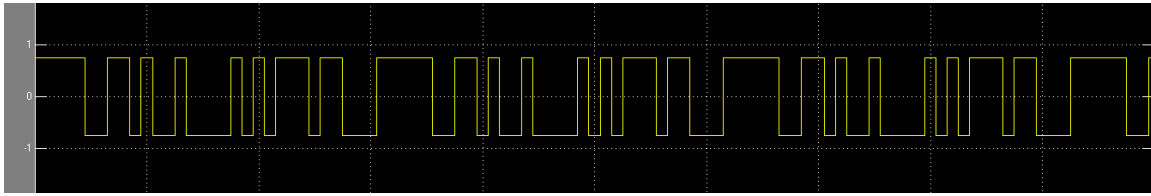


Figure IV.7 : Le signal SBPA vu dans le scope

IV.4 Identification de la transmittance du système étudié par la méthode erreur de sortie utilisant une SBPA

Etape 01: Réalisation et configuration du schéma blocs SIMULINK

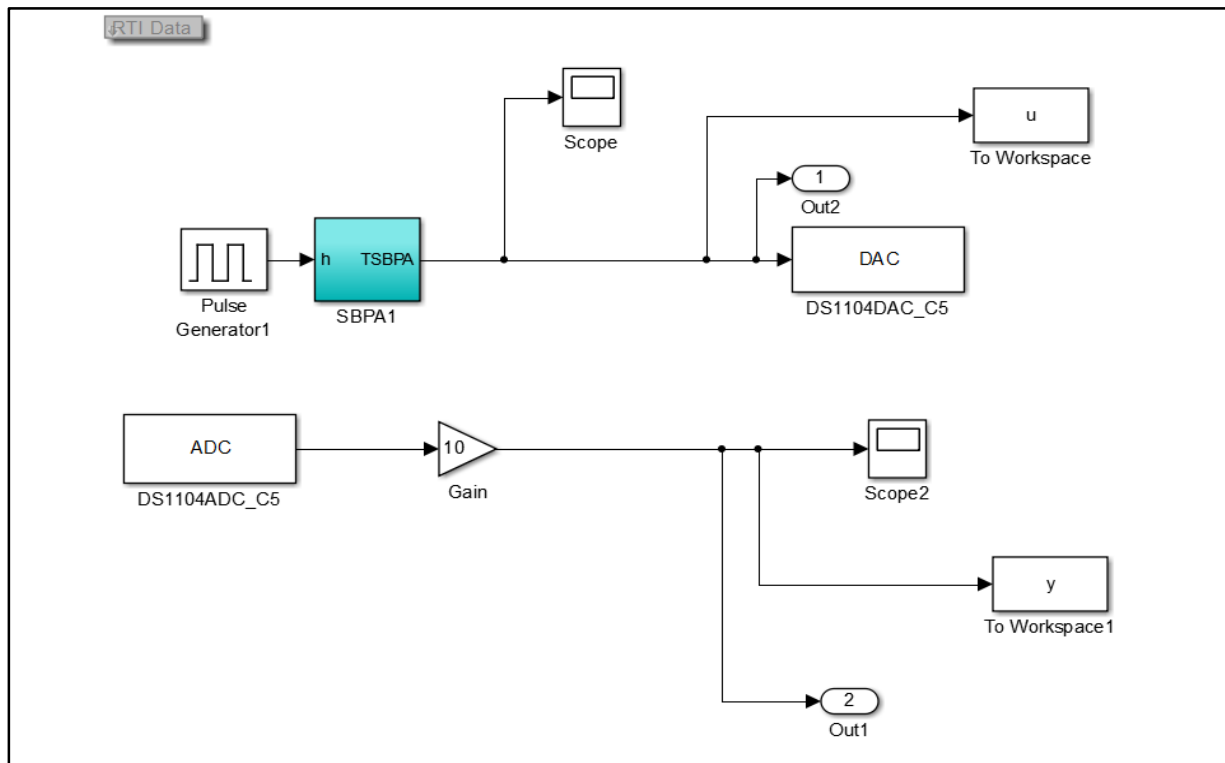


Figure IV.8 : Schéma bloc SIMULINK de l'identification par le SBPA

Avec un simple clique sur ADC et DAC, un tableau qui nous permettra de spécifier le numéro de port de la carte dSPACE CPL1104 qui communique avec notre système à identifier (dans notre cas ce sont respectivement les ports ADCH5 et DACH5).

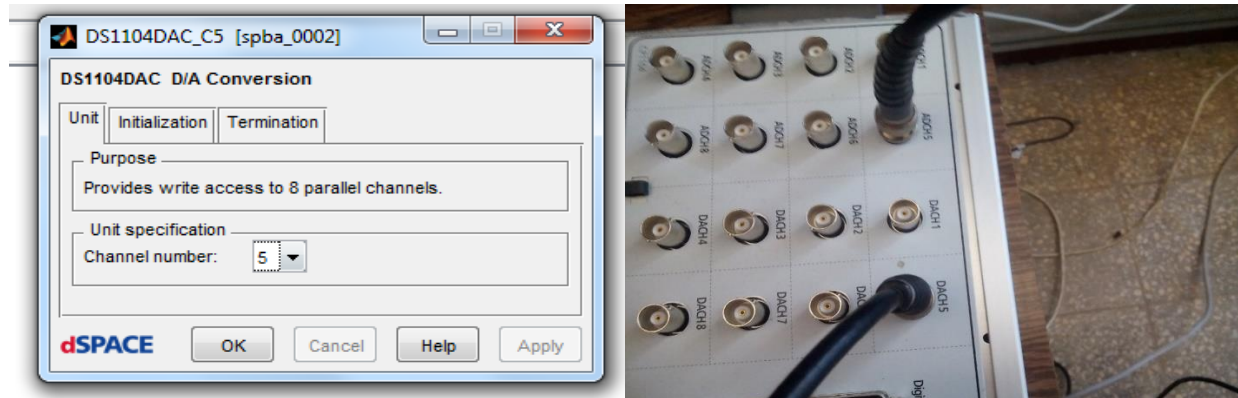


Figure IV.9 : Configuration de la carte dSPACE avec le système à identifier

Après avoir fixé la période d'échantillonnage dans la fenêtre << configuration parameter sbpa >> à une valeur adéquate (après plusieurs essais on a fixé la $T_e = 0.002s$), ensuite on compile notre SIMULINK ce qui va générer un fichier sbpa avec l'extension « .sdf » (sbpa.sdf) dont on aura besoin pour la récupération des signaux.

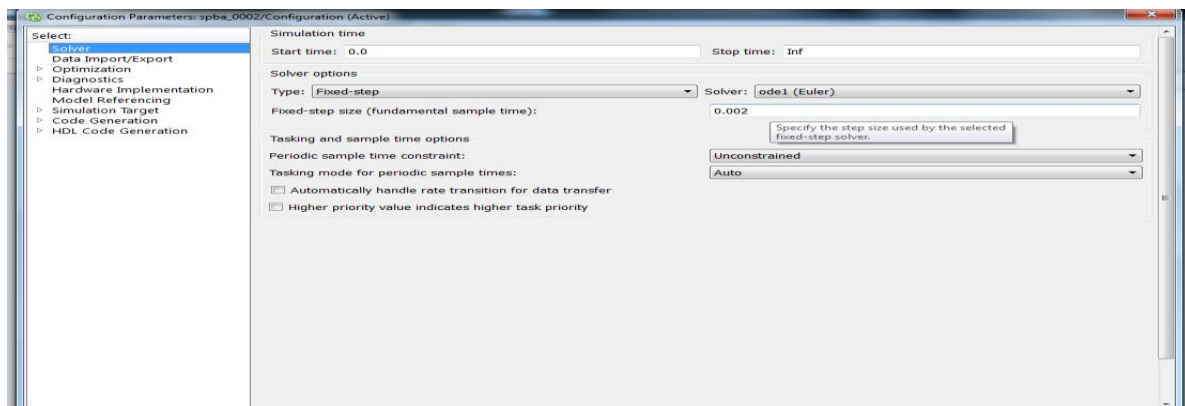


Figure IV.10 : Choisir la période d'échantillonnage

Etape 02 : récupération des signaux E/S via l'interface Control desk

Cette étape consiste à récupérer les signaux entrée /sortie,mais avant cela on doit :

- Lancer l'application control desk dans laquelle on va créer un nouveau projet

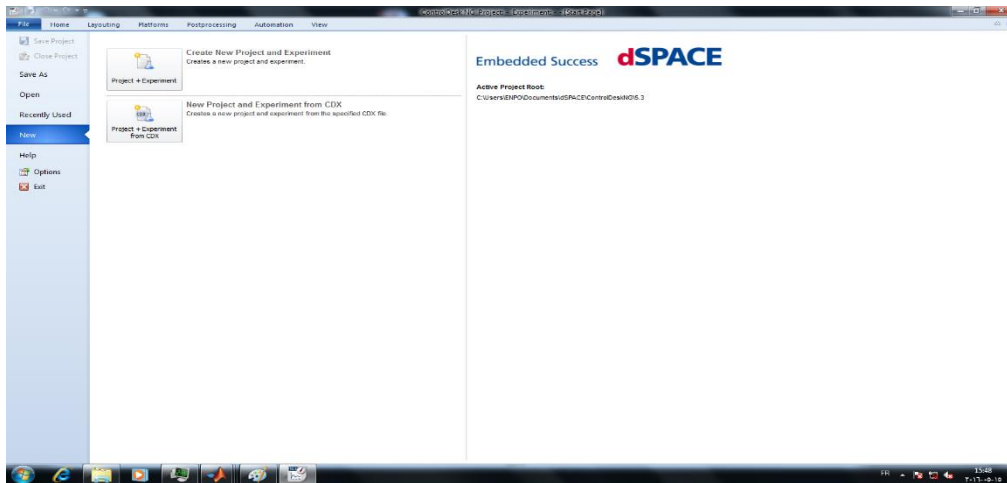


Figure IV.11 : Interface control desk

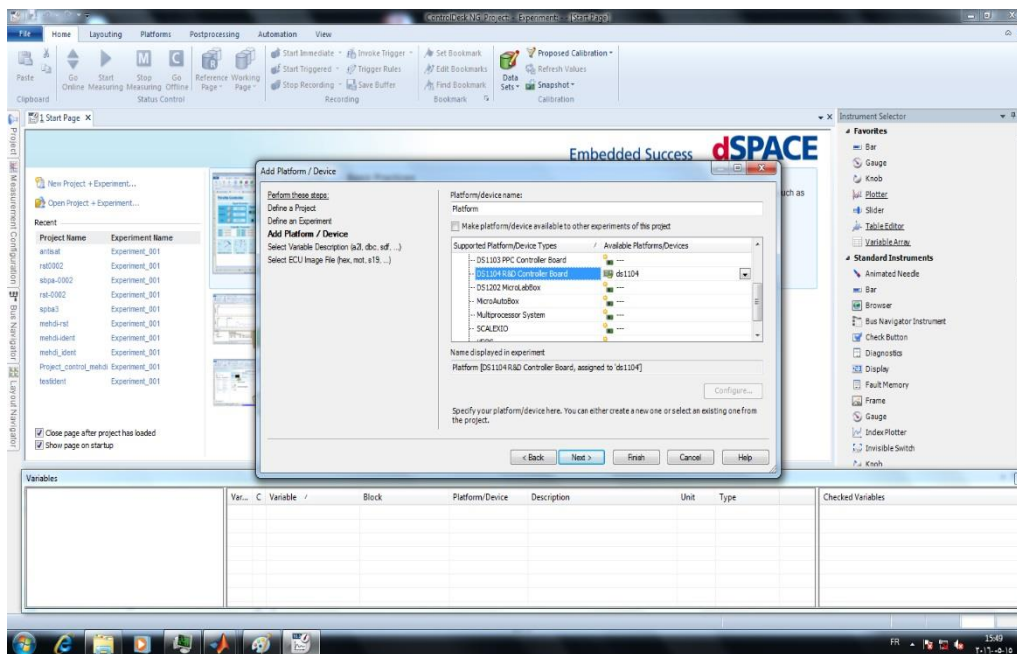


Figure IV.12 : Choisir la plateforme du travail (DS1104 R&D Controller Board)

Dans la fenêtre (figure ci-dessus) « Add Platform/Device » on sélectionne « DS1104 R&D Controler Board » qui est notre carte dSPACE.

Après avoir choisi la plateforme on passe à l'étape suivante dans laquelle on fait appel au fichier enregistré sous l'extension « .sdf » (sbpa.sdf) via la fenêtre « Select Variable Description » en cliquant sur « Import From File » comme le montre la figure suivante

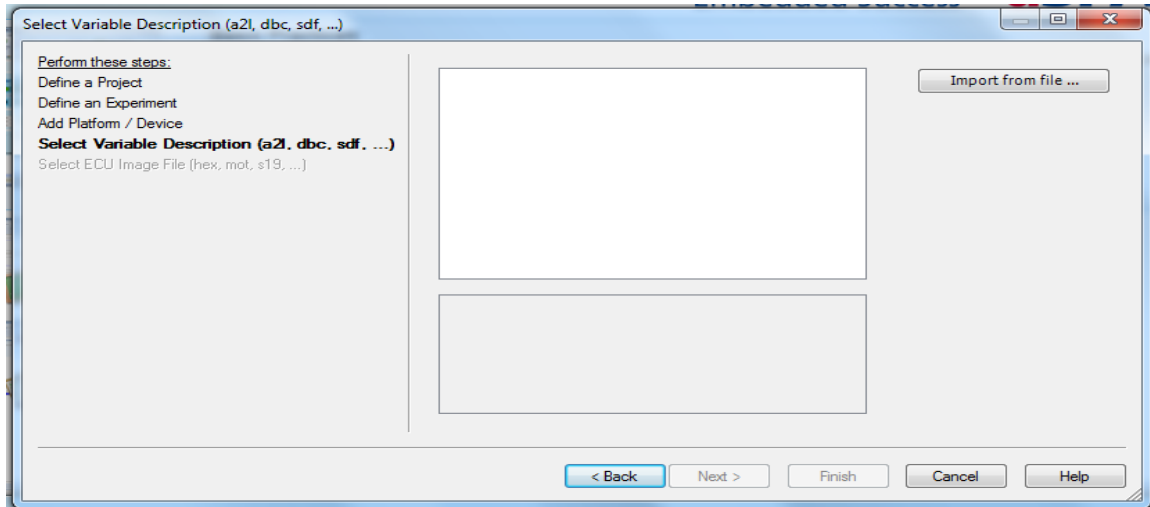


Figure IV.13 : Faire fichier enregistré sous l'extension « .sdf »

Une fois le fichier insérer le nouveau projet est créé, et on peut visualiser tous les blocs du schéma SIMULINK dans notre interface graphique.

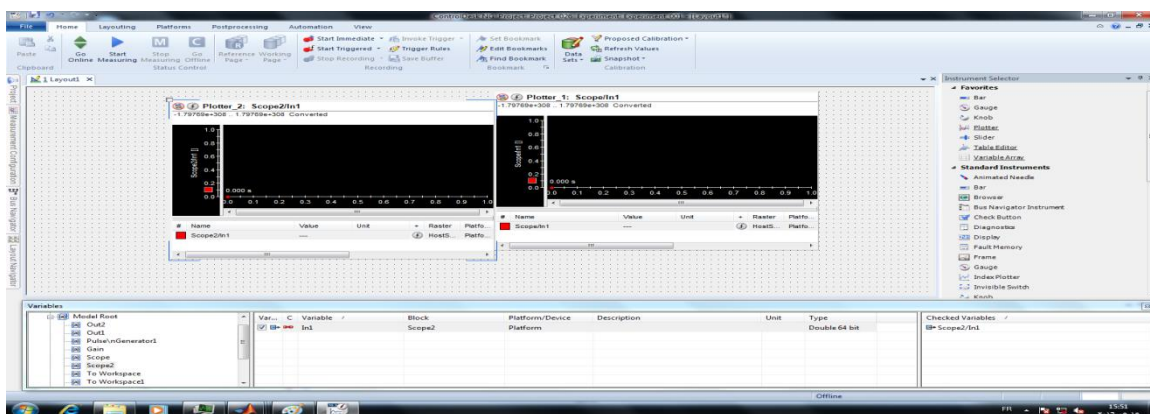


Figure IV.14 : Interface control desk avec un nouveau projet

➤ Visualisation, enregistrement et stockage des signaux entrée/sortie

Afin de réaliser cette tâche on doit commencer par importer un enregistreur « recorder01 » dans la rubrique « mesurement configuration », ensuite dans la fenêtre « propriété » on lui spécifie l'extension de fichier à stocker, pour exporter les données vers un fichier Matlab, allez dans Project, Experiment1, Measurement Data et cliquer droit sur le fichier .idf créé puis export en choisissant bien le type de fichier .mat dans l'onglet Save as Type (dans notre cas l'extension est « .mat ») pour le traiter ensuite à l'aide du logiciel MATLAB.

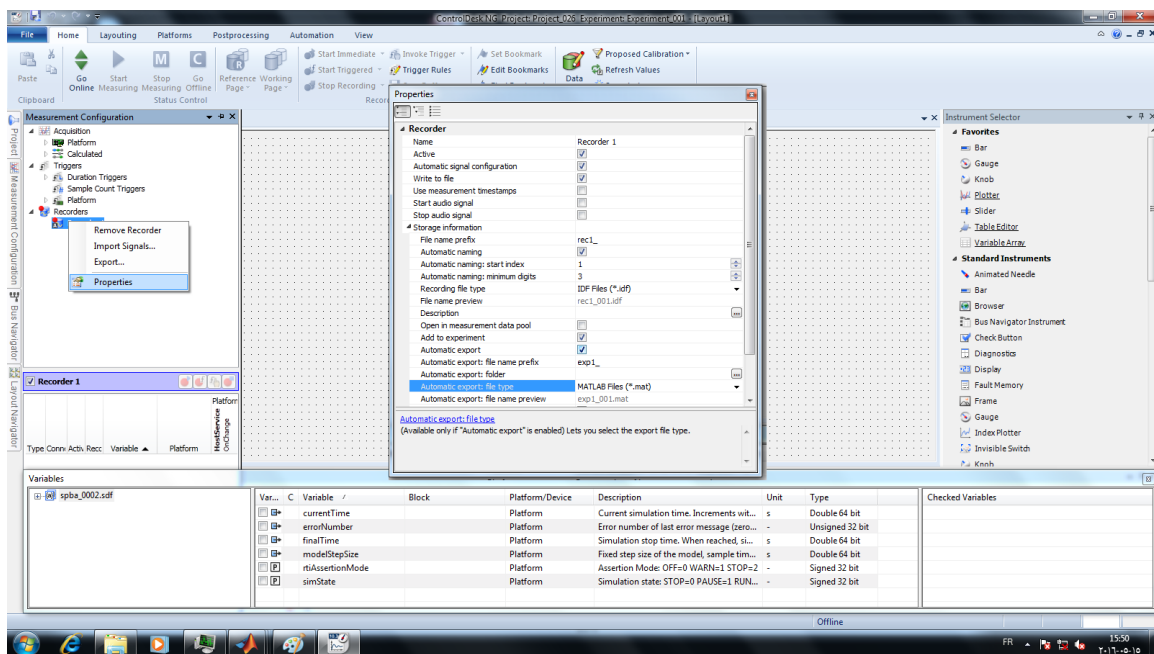


Figure IV.15 : Réglage des paramètres Visualisation, enregistrement et stockage des Signaux entrée/sortie

cliquer sur le bouton « GO ONLINE » ce qui nous permettra de lancer visualiser les deux signaux de sortie et d'entrée dans les plotter1 et 2 respectivement.

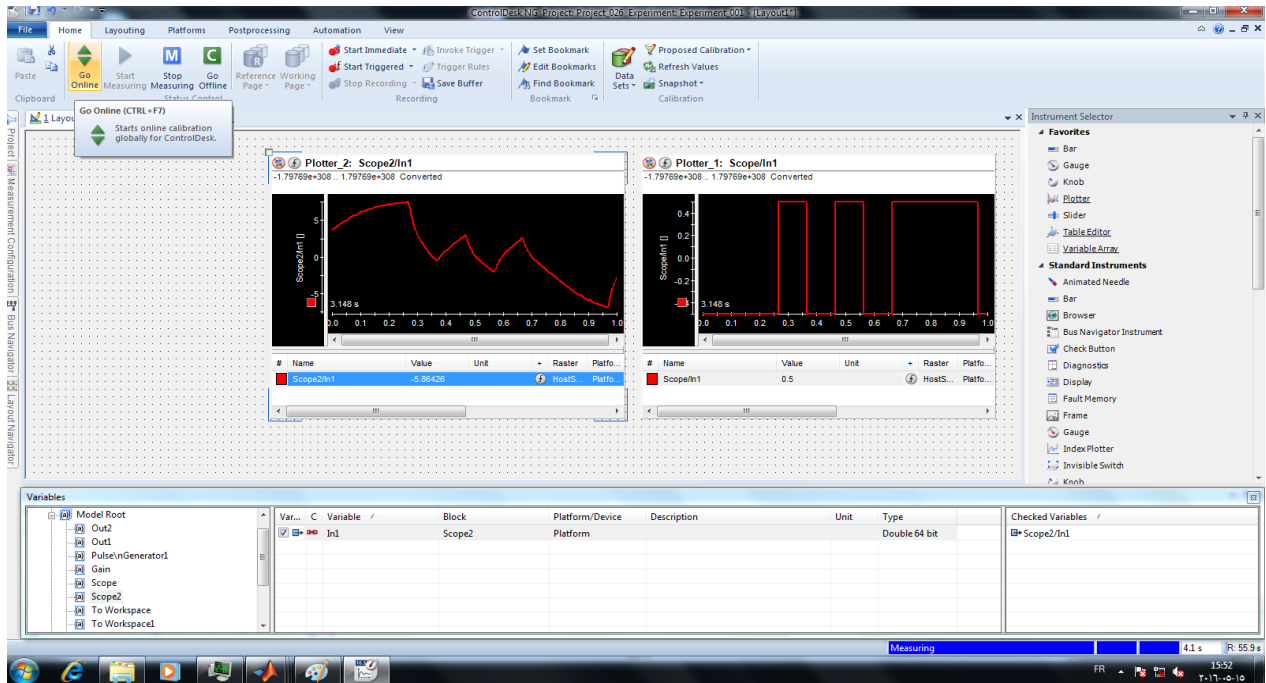


Figure IV.16 : Visualisation, des signaux entrée/sortie

Une fois les signaux visualisés sont conformes aux formes désirées on peut commencer l’enregistrement en cliquant sur le bouton « start immediate recording »

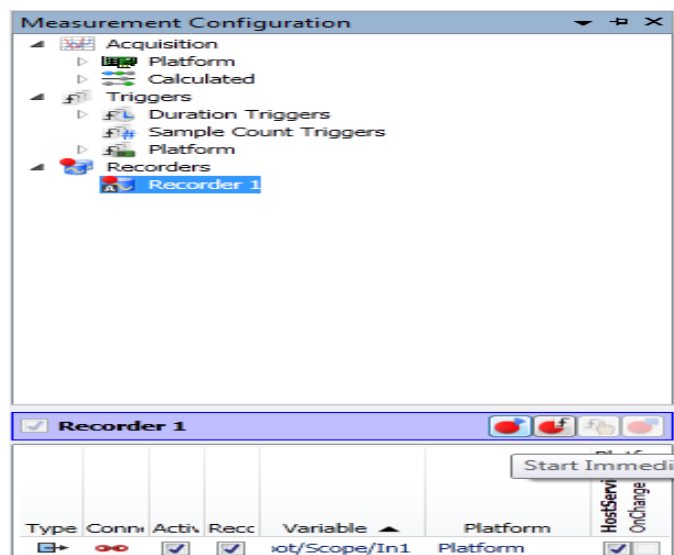


Figure IV.17: Enregistrement des signaux entrée/sortie

Remarque : un fichier exploitable en MATLAB se créera automatiquement et il contiendra les mesures de nos signaux d'entrée et de sortie.

Etape 03 : traitement et interprétation des signaux récupérés avec le logiciel MATLAB

Premièrement on importe le fichier créé suite à l'étape précédente dans le « work space » de MATLAB ensuite on va extraire les informations contenues dans ce fichier (différentes mesures du signal de sortie dans notre cas) dans la fenêtre « command window ».

En utilisant la commande MATLAB `>> ident` la fenêtre « System Identification Tool » s'ouvre ensuite en cliquant sur « import data » dans laquelle on importe le signal à identifier et où on initialise la période d'échantillonnage (dans notre cas $T_e = 0.002s$).

Pour lancer l'estimation des paramètres du signal de sortie on choisit la méthode d'estimation dans le volet « estimat » (dans notre cas la méthode OE (Output Error) erreur de sortie), le logiciel calcule un modèle selon les paramètres choisis préalablement qui s'affiche dans la fenêtre « model output » qui nous informe sur le taux de compatibilité entre le signal à identifier et le modèle calculé.

Remarque : pour une bonne estimation le modèle doit être compatible à plus de 80%

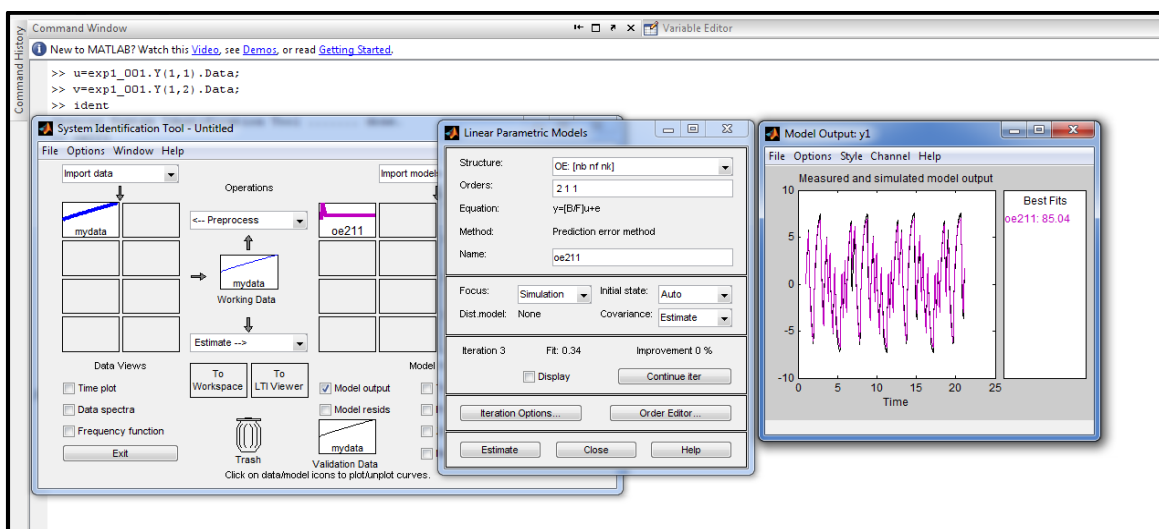


Figure IV.18: Interprétation des signaux récupérés avec le logiciel MATLAB

Après la visualisation, un fichier qui contient les coordonnées du modèle vont être enregistré dans la fenêtre « workspace » .

On glisse se fichier dans l'interface « MATLAB » en cliquant sur « entrer » le logiciel va calculer la fonction de transfert du model qui caractérise le signal.

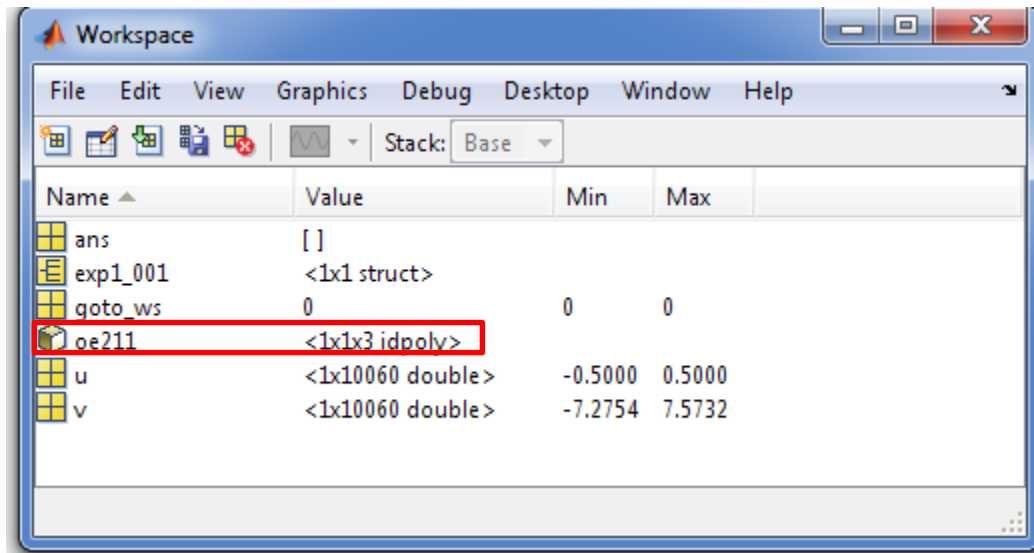


Figure IV.19: Le fichier qui contient les cordonné du modèle dans la fenêtre « workspace »

IV.4.1 Essais

Dans cette étape pratique on a effectué quatre (4) essais afin de raffiner le signal pour une identification optimale.

Par différents paramétrages de la boite de dialogue de de la fonction OE (Output Error), qui est une variante de la méthode des moindres carrés exposée précédemment (chapitre III) .

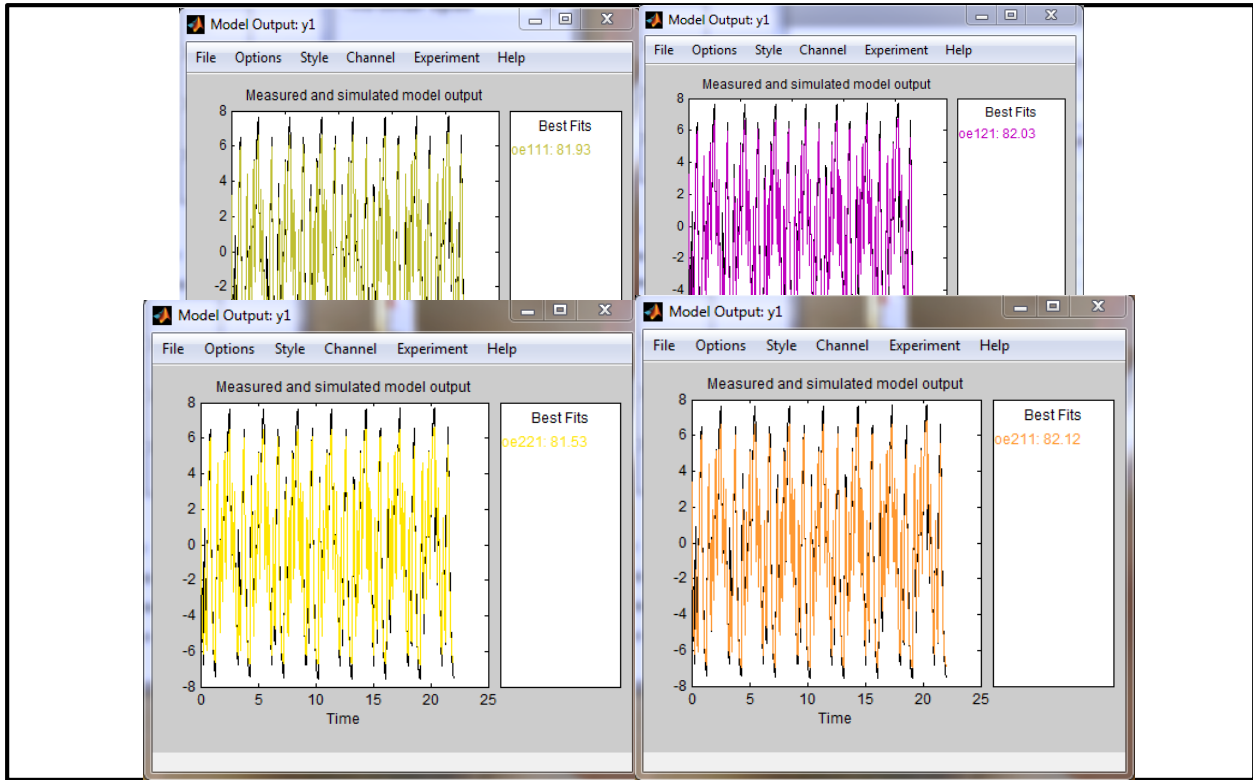


Figure IV.20: Essais de l'interprétation du fichier qui contient les cordonné du modèle

Le Tool Box Ident nous indique un critère de qualité qui caractérise l'écart entre le comportement du système et celui de son modèle.

Résultats obtenus après nos essais :

Essais	Critère de qualité
Essai 01 (oe 221)	81.53
Essai 02 (oe 111)	81.93
Essai 03 (oe 211)	82.12
Essai 04 (oe 112)	82.03

IV.4.2 Validation du modèle

La sensibilité de cette étape nous mène à une interprétation très précise car elle est la base de notre chapitre , dans cette étape On peut déduire la fonction de transfert du modèle à partir de l'un de ces essais .et pour cela on a effectué quatre essais afin de valider le signal qui génère la fonction la plus adéquate.

Le choix du modèle se base sur le critère de qualité, dès que le critère est grand le modèle sera plus représentatif de la boîte noire.

A partir du tableau l'essai 03 porte le plus grand critère de qualité avec 82.12, et pour cela on a mis sur lui le choix de validation.

Avec le logiciel MATLAB on a pu déduire la fonction de transfert qui correspond au modèle de l'Essai 03 :

$$H(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})} = \frac{-0.4848 z^{-1} + 0.2574 z^{-2}}{1 - 0.9841 z^{-1}}$$

qui peut s'écrire aussi comme suit :

$$H(z) = \frac{B(z)}{A(z)} = \frac{-0.4848 z + 0.2574}{z^2 - 0.9841 z}$$

Avec les pôles :

$$p_1 = 0 \quad , \text{et} \quad p_2 = 0.9841$$

Et un seul zéro :

$$z_1 = 0.5309$$

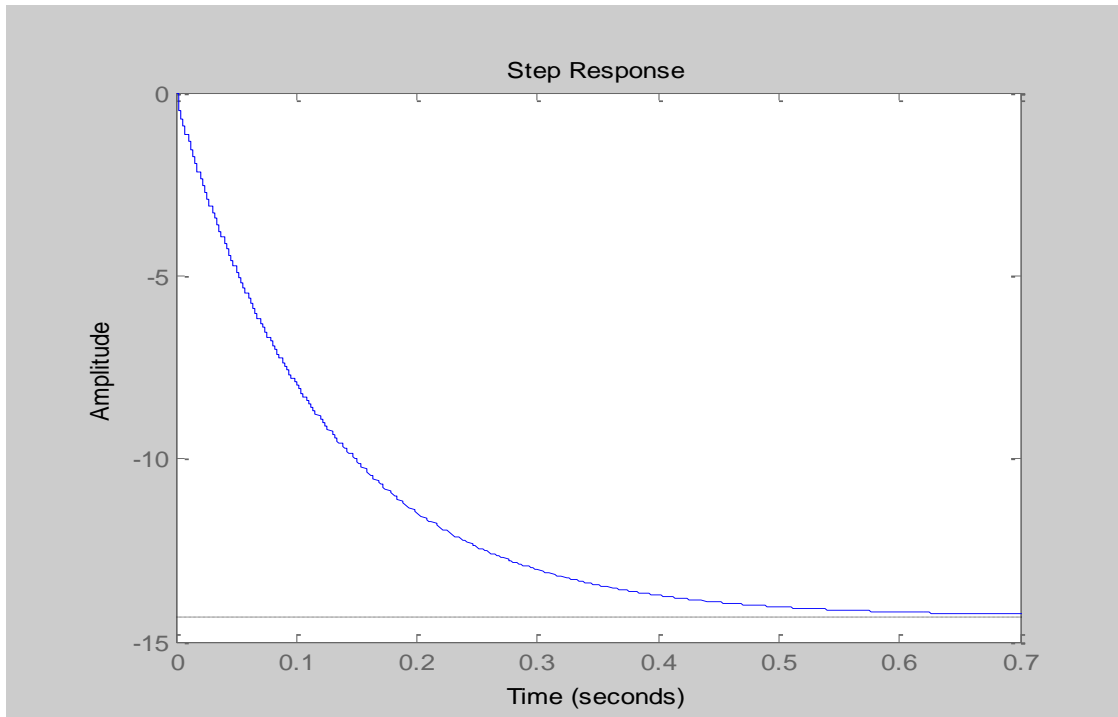


Figure IV.21: La réponse indicielle du modèle validé

Discussion :

La réponse indicielle à la fréquence d'échantillonnage $T_e = 0.002s$ du modèle validé (figure.IV.21) correspond au comportement d'un système du 2eme ordre (avec ses deux pôles p_1 et p_2) sur-amorti (absence de dépassement causé par un amortissement) que nous pouvons assimiler à un moteur à courant continu qui se caractérise par un temps de montée $T_r = 0,274s$ (relativement lent pour un moteur cc) et un erreur statique importante, d'où la nécessité d'une correction pour garantir de meilleures performances.

IV.5 Application de la commande numérique sur le système identifié

On désire commander notre système en vitesse constitué d'amplificateur E de moteur cc, et de génératrice de tachymétrie (« maquette Feedback ») qui est représenté par la fonction de transfert suivante :

$$H(z) = \frac{B(z)}{A(z)} = \frac{-0.4848 z + 0.2574}{z^2 - 0.9841 z}$$

Selon les performances suivantes :

- Amortissement $\xi = 0.707$ (cette valeur découle de l'imposition du dépassement $D\% < 20\%$)
- Temps de réponse $t_R = 0.175 \text{ s}$
- Période d'échantillonnage $T_e = 0.002 \text{ s}$
- Une pulsation propre $\omega_0 = \frac{2.16}{t_R} = 12.342 \text{ rad/s}$

IV.5.1 Synthèse de la loi de commande (placement de pôles)

Le calcul des polynômes R, S et T est donné par un programme écrit sous MATLAB (voir annexe A), en remplaçant les valeurs données on aura :

$$R(z^{-1}) = -0.8203 + 0.0794 z^{-1}$$

$$S(z^{-1}) = 1 - 1.0208 z^{-1} + 0.0208 z^{-2}$$

$$T(z^{-1}) = -0.02$$

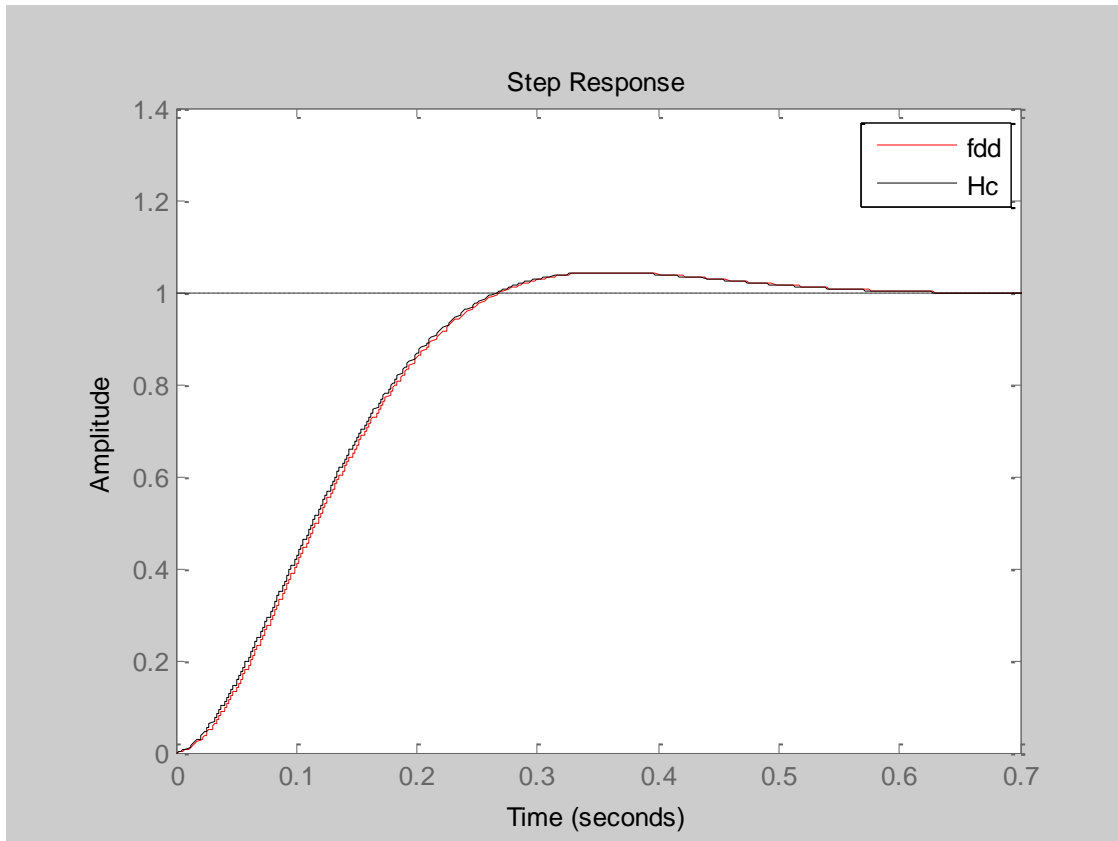


Figure IV.22: Réponse indicielle du système après correction et trajectoire désirée

Interprétation :

La réponse indicielle du système à commander H_c bouclé avec un correcteur RST pré-calculé se confond avec la trajectoire désirée f_{dd} élaborée à partir du cahier de charges. Ce qui montre la fiabilité et l'exactitude des paramètres des trois polynômes du régulateur numérique R, S et T.

IV.5.2 Simulation de la commande sous SIMULINK :

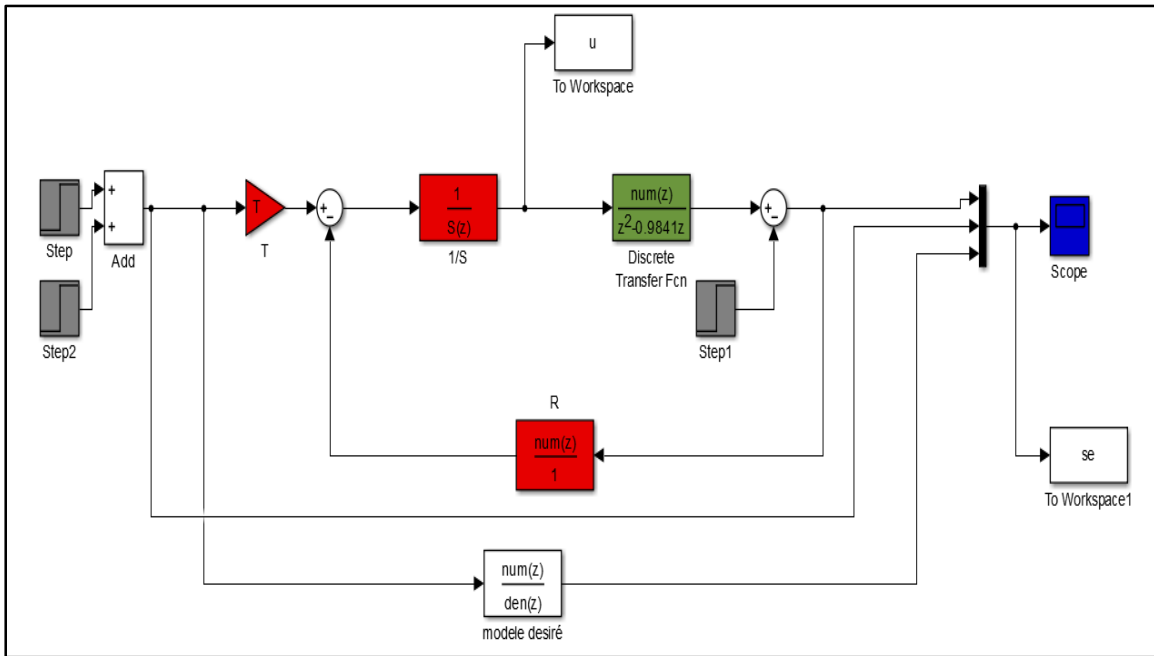


Figure IV.23: Schéma bloc SIMULINK de Simulation de la commande

Résultats de la simulation :

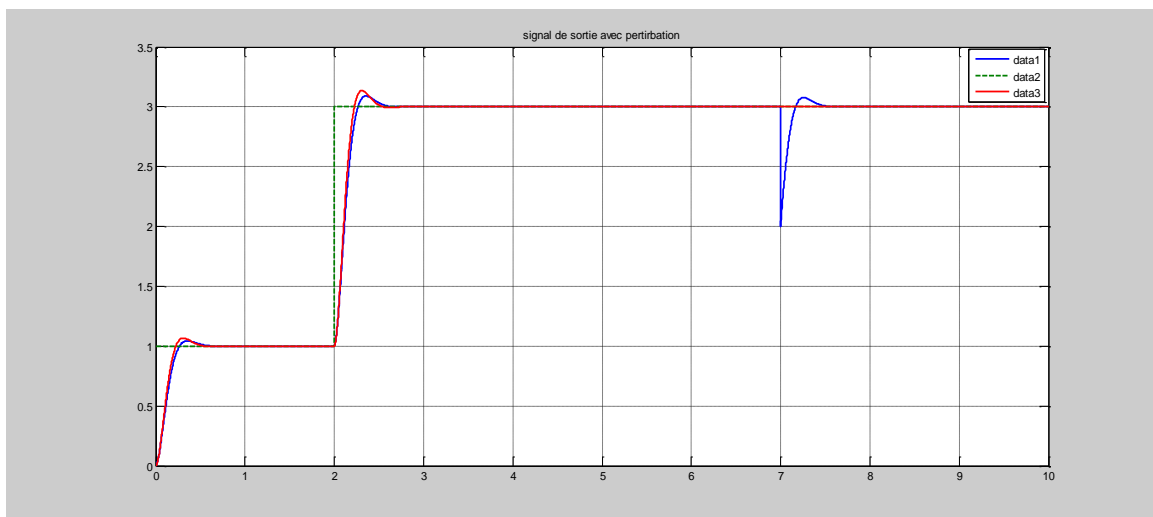


Figure IV.24: La réponse indicielle du système bouclé avec le régulateur RST et la réponse désirée

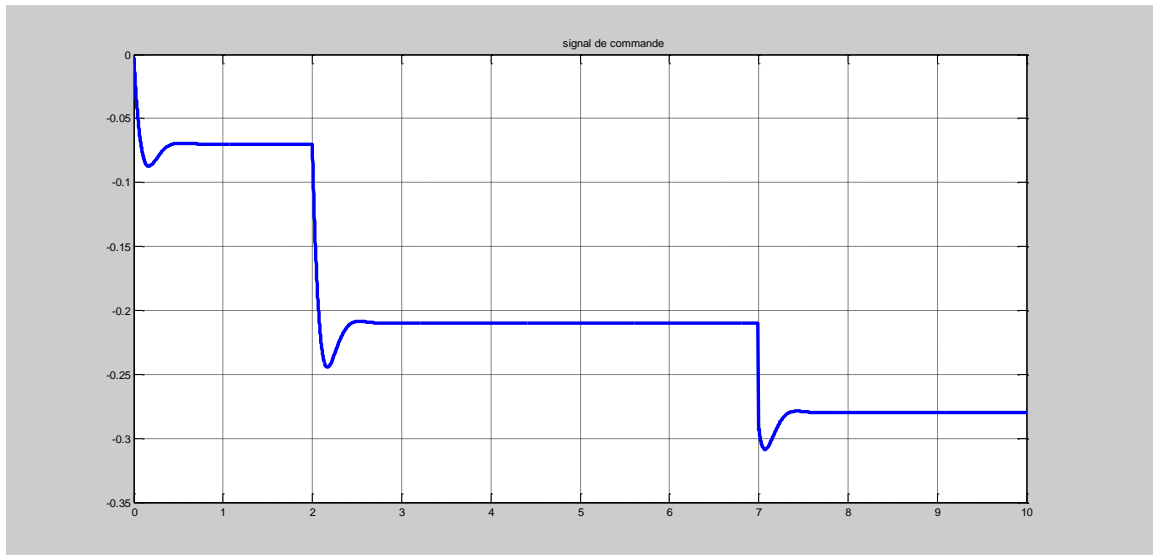


Figure IV.25: Signal de commande

Discussion des résultats :

La première remarque qu'on peut faire à propos de la figure IV.25 est que la réponse indicielle du système bouclé avec le régulateur RST (vert) est confondue avec celle du modèle de référence (rouge).

Donc on peut confirmer que la conception et le calcul de notre régulateur élaboré précédemment pour la méthode de placement de pôles répond aux spécifications désirées.

On remarque aussi que la dynamique de poursuite est pareil à la dynamique de rejet de perturbations ce qui confirme la robustesse de ce type de commande.

IV.6 Commande en temps réel sous la plateforme dSPACE / ControlDesk :

Afin d'implémenter la commande dans le système *dSPACE*, on doit d'abord réaliser le schéma bloc de notre boucle de régulation en prenant en considération les spécificités de notre carte de contrôle *DS 1104* qui sature à une tension de 10V ainsi on doit prévoir un bloc anti-saturation et un filtre anti-wind up élaboré d'une manière pratique tout en gardant la fonction principale de notre boucle qui est de commander le moteur en vitesse.

IV.6.1 Réalisation du schéma Simulink :

Dans l'interface Simulink nous avons élaboré un schéma blocs de notre boucle de commande en vitesse doté d'un « dispositif anti-saturation » dans le but de rendre la tâche de la commande faisable en temps réel composé :

- D'un bloc saturation initialisé à ± 0.9 , afin de ne pas saturer la carte de contrôle *DS1104* et ainsi lui permettre de délivrer le signal de commande adéquat d'une manière permanente quelle que soit la consigne donnée.
- Un filtre anti wind-up (anti emballement), dont la fonction est de vider de remettre à zéro l'effet de l'intégrateur dans le but de maintenir la stabilité du système.
- Ainsi qu'un filtre de débruitage du signal de sortie avant son admission dans le bloc de $R(z^{-1})$ tout en gardant l'information.

Tous ces apports sont les contraintes de la pratique, ils sont mis en place après plusieurs essais.

IV.6.2 Simulation du régulateur RST avec le dispositif anti-saturation

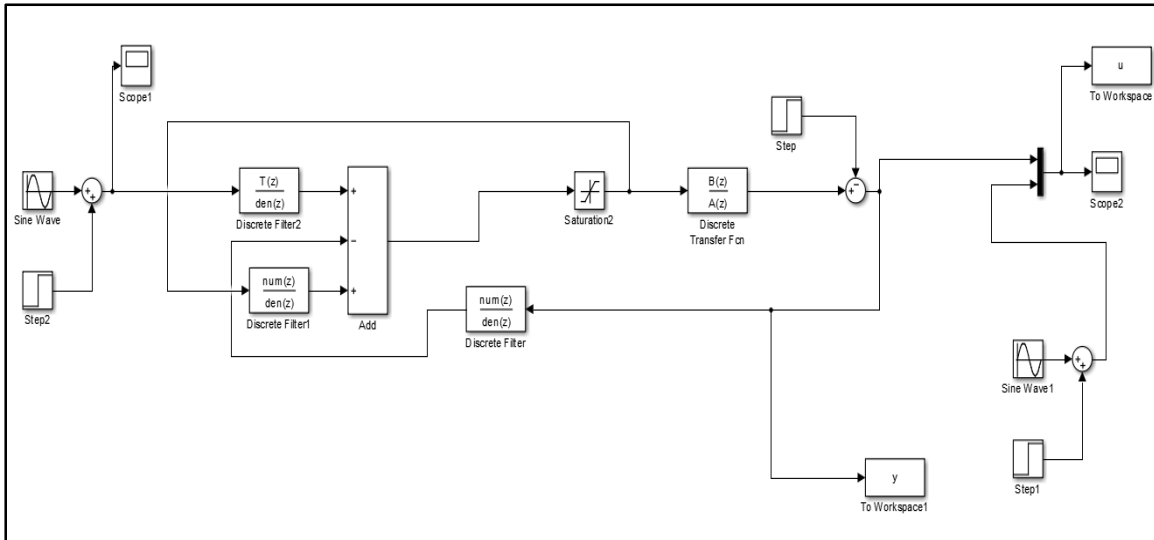


Figure IV.26: Schéma bloc Simulink temps réel de la Simulation du régulateur RST avec le dispositif anti-saturation

Résultat de la simulation :

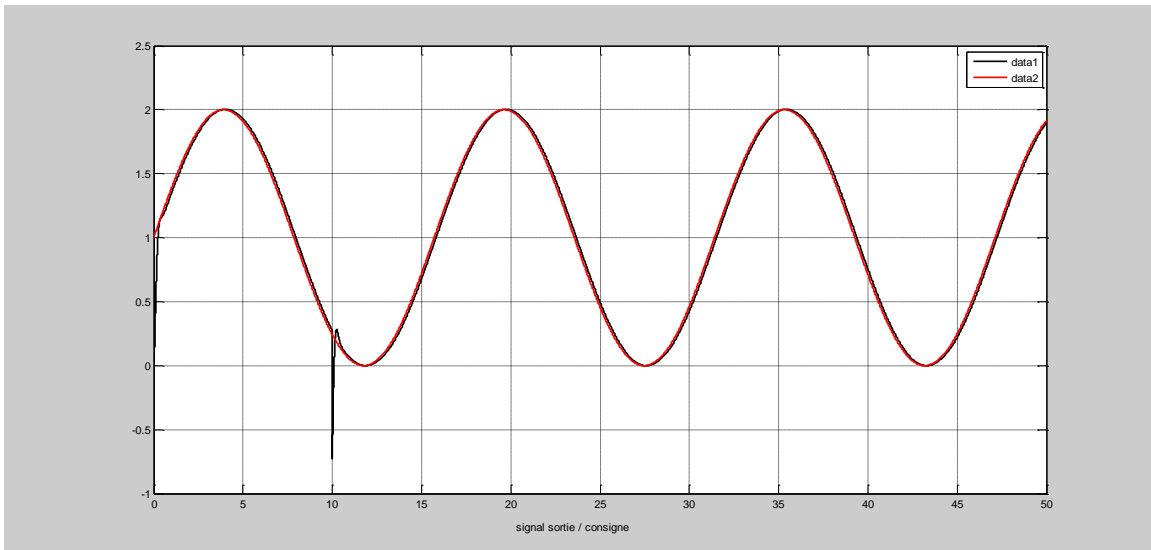


Figure IV.27: Signal de sortie de la boucle d'asservissement et celui de la consigne

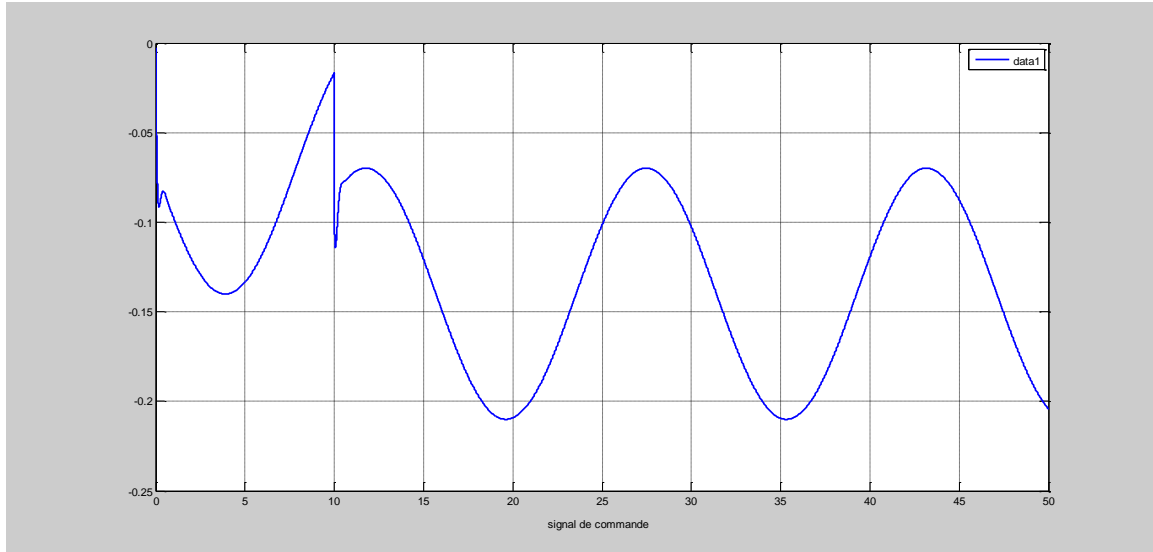


Figure IV.28: Signal de commande

Interprétations des résultats :

On remarque que le signal de sortie de cette boucle d'asservissement se confond avec la consigne d'entrée, ainsi qu'un rejet de perturbations efficace ce qui montre que le dispositif anti-saturation est fonctionnel.

IV.6.3 Réalisation du schéma bloc à implémenter sous le système *dSPACE* :

Dans l'interface temps réel de Simulink(*RTI*) on réalise le schéma blocs de notre boucle d'asservissement simulé précédemment, en faisant appel à la bibliothèque *rtlib1104* nous importons les blocs *DAC* (Digital Analog converter) et *ADC* (Analog Digital converter) qui communiquent avec notre système.

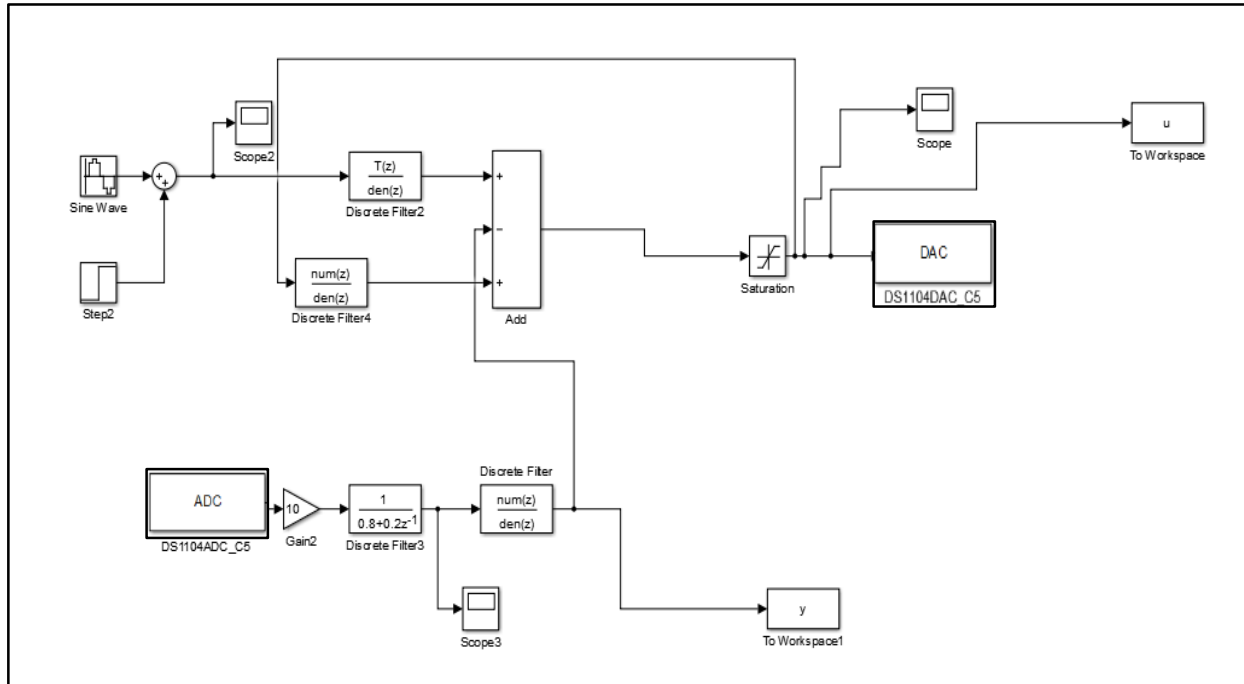



Figure IV.29: Schéma bloc Simulink à implémenter sous le système dSPACE



Figure IV.30 : Communication de système dSPACE avec le système à commander

IV.6.4 Lancement de la commande en pratique

Après compilation sous Simulink en cliquant sur cette icône  , le chargement du programme dans la carte de contrôle *DS1104* se fait instantanément.

un fichier avec l'extension « *rst.sdf* » exploitable sous *ControlDesk* se génère automatiquement, en l'exploitant nous créons un nouveau projet dans *ControlDesk* (comme expliquer dans la partie dédiée à l'identification). Nous lançons le programme après avoir fait tous les branchements correctement et avoir mis le système sous tension.

La visualisation des signaux d'entrée et de sortie comme le montre la figure IV.31 est possible après avoir importé de l'anglet des variables les *scoops 2* et *3*.

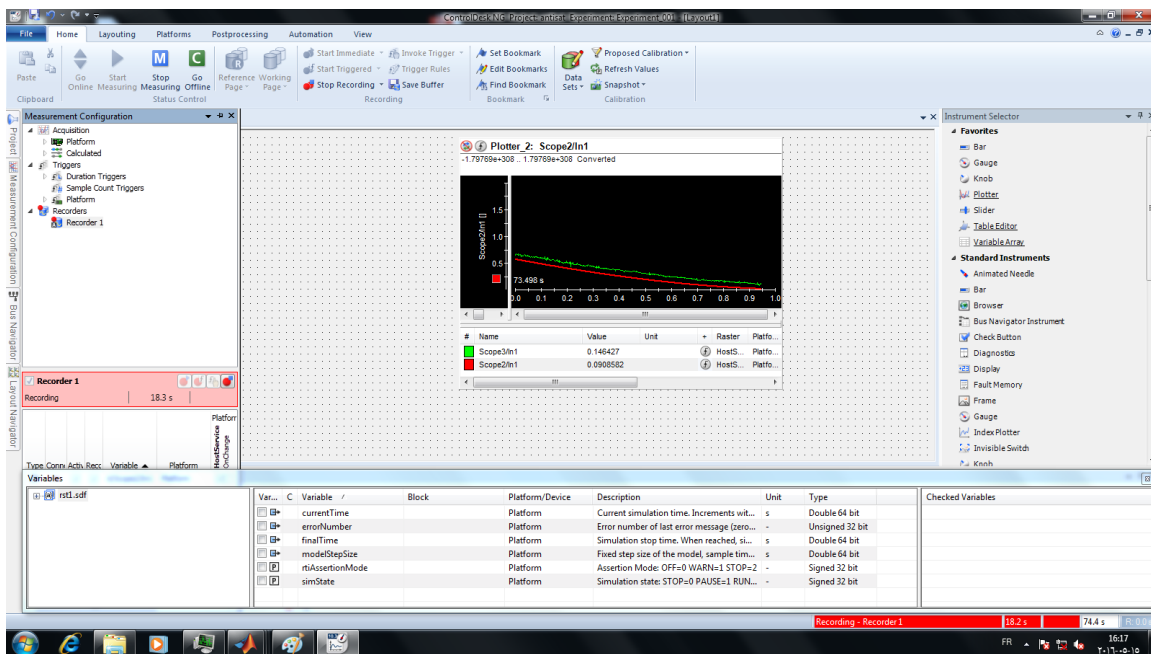


Figure IV.31: La visualisation des signaux d'entrée et de sortie sur l'interface control desk

L'enregistrement des deux signaux d'entrée et de sortie sous *ControlDesk* et leurs restitutions sous *MATLAB* (comme expliquer précédemment) à donner les résultats suivants :

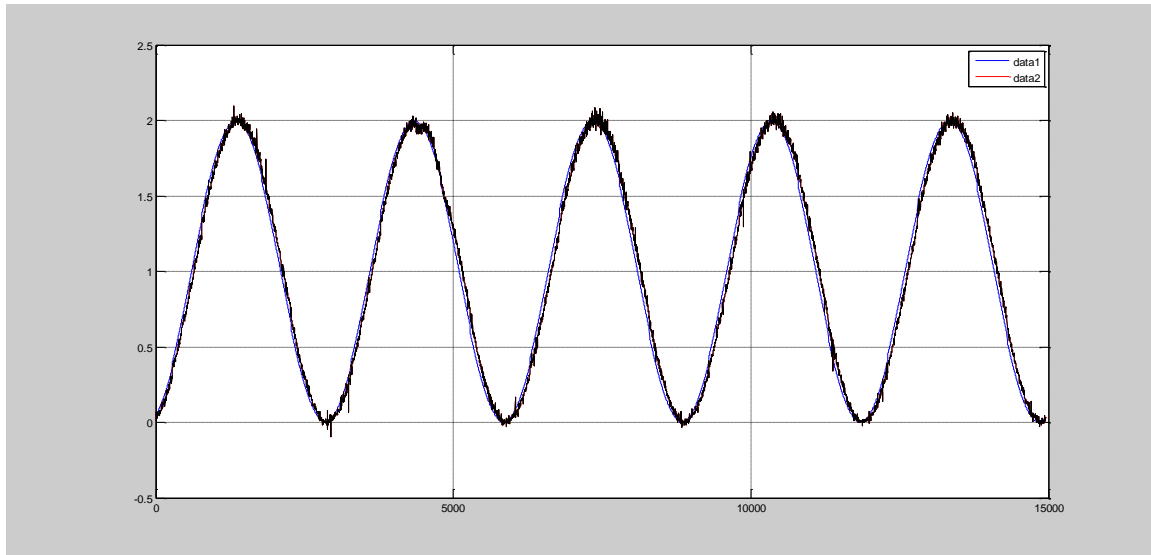


Figure IV.32:Signal de sortie réel de la boucle d'asservissement et celui de la consigne

Interprétation des résultats :

D'après les résultats obtenus en pratique, on peut conclure sur l'efficacité de notre correcteur numérique RST couplé avec le dispositif anti-saturation, de par la nature confondue des deux signaux d'entrée et de sortie ainsi que la robustesse de notre commande vu son rejet de perturbations (frein magnétique).

IV.7 Conclusion

Dans ce dernier chapitre, qui représente la consécration de notre travail, nous avons mis en pratique la théorie traitée dans les chapitres précédents.

En premier nous avons fait l'identification de notre système à commander (bon d'essai *FeedBack MS150*) d'une manière pratique sous l'environnement *dSPACE / ControlDesk* et *MATLAB* où nous avons pu manipuler les techniques appropriées à l'identification des systèmes dynamiques, pour des questions de faisabilité nous avons opté pour l'identification par les SBPA.

Ensuite nous avons pu générer un correcteur numérique de structure RST avec la méthode de placement de pôles avec même objectif pour la poursuite et la régulation en utilisant que nous jugeons satisfaisant de par la robustesse qu'il offre au système (marge de phase $P_m=71.1$ et la marge de gain $G_m=33.8$) ainsi que la poursuite et le rejet de perturbations.

Enfin nous avons pu implémenter la loi de commande en vitesse dans la *DSP DS1104* après résolutions des problèmes liés à la pratique (saturation, emballement, ...). Ainsi on a pu superviser notre système en temps réel sous *ControlDesk*.

Conclusion générale

L'objectif de ce travail était l'étude théorique et l'application pratique d'une commande numérique de structure RST sur un moteur à courant continu afin de le commander en vitesse, cet objectif a été incité par l'importance de la variation de la vitesse dans le domaine du génie électrique ainsi que l'étendu de sa sollicitation dans les domaines de l'industrie, la robotique et beaucoup d'autres branches.

Dans le volet théorique de ce travail, nous avons en premier, traité des méthodes de synthèse des régulateurs RST, car c'est le socle sur lequel notre projet s'articule, où globalement nous avons constaté plusieurs caractéristiques intéressantes et attractives du régulateur à structure RST que ce soit du point de vue théorique ou pratique. En théorie, la souplesse ainsi que la puissance de ce type de correcteurs face à de nombreux cas de configuration de commande le met en tête des régulateurs dits à commande avancée. En pratique il est conçu directement dans le domaine numérique avec une équation récurrente facile à implémenter sur une architecture programmable. Nous avons ensuite proposé une modélisation du moteur à courant continu, cette modélisation représente les principaux phénomènes qui interviennent lors du fonctionnement du moteur.

Dans le volet pratique réalisé sous l'environnement *MATLAB/Simulink* *dSPACE/ControlDesk* qui est la partie la plus importante et de notre projet, nous avons travaillé à déterminer le modèle de comportement (fonction de transfert) d'un système dynamique qui est la maquette *FeedBack MS150*, l'identification du modèle du système a été réalisée par injection et récupération de signaux via la carte de contrôle *dSPACE 1104*, ce qui nous a permis de côtoyer et de manipuler l'identification. Les signaux récupérés sont traités sous *MATLAB* par différents algorithmes d'estimation. Ensuite on a pu élaborer la loi de commande RST dédiée à la commande en vitesse de notre *FeedBack MS150* en respectant un cahier de charges prédéfini.

Nous avons ensuite réalisé des simulations sous *Simulink* du régulateur RST généré bouclé avec le système à commander afin de vérifier sa stabilité et sa robustesse ainsi que ses dynamiques de poursuite de rejection de perturbations que nous jugeons satisfaisante.

Dans la phase de l'implémentation de l'algorithme nous avons rencontré des difficultés liées aux limitations physiques de la carte de contrôle (DSP) *dSPACE 1104* notamment le problème de la saturation, de bruitage des signaux de retour que nous avons réussi à surmonter avec un dispositif anti-saturation et un filtrage mis en place d'une manière pratique. Les résultats de la commande en temps réel enregistrés sont amplement satisfaisants tant du point de vue robustesse que stabilité ainsi que la rejection de perturbations ce qui confirme la théorie développée à propos des régulateurs RST.

En fin, ce travail peut être un support didactique pour initiation à l'environnement Matlab/Simulink/dSPACE et peut aussi être suivi par d'autres travaux sur la commande adaptative robuste basés sur les techniques de placement de pôles pour le réglage automatique de contrôleurs.

Références Bibliographiques

- [1] **Ioan Doré Landau** « Identification et commande des systèmes » Editions Hermès, Paris (1993).
- [2] **P. Borne et G. Dauphin Tanguy** « Analyse et Régulation des Processus Industriels » Edition Technip, Paris (1993).
- [3] **Philippe de Larminat** « Automatique Appliquée », Edition Lavoisier, Paris(2009).
- [4] **SADOUD Hassane .DELICI Abderrazak** « Etude comparative entre la régulation à modèle interne et la régulation à approche polynômiale » Mémoire d'ingénieur U.M.M TIZI-OUZOU, Promotion 2007.
- [5] **ZADRI Samir** « Commande numérique basée sur le placement de pôles des procédés linéaires à retard » Mini Projet Master 1 U.A.M BEJAIA, Promotion 2010.
- [6] **Mohamad Alaa Eddin Alali** « Contribution à l'Etude des Compensateurs Actifs des Réseaux Electriques Basse Tension» Thèse Doctorat de l'Université Louis Pasteur -Strasbourg I Discipline : Génie électrique, Promotion 2002.
- [7] **Moussa BENDJEDIA** « Synthèse d'algorithmes de commande sans capteur de moteur pas à pas et implantation sur architecture programmable » Thèse Doctorat de l'Université FRANCH-COMTE, Promotion 2007.
- [8] **Sébastien Cabaret** « Algorithmes de Contrôles Avancés pour les Installations à Gaz du LHC au CERN suivant le Framework et l'approche dirigée par les modèles du projet GCS » Thèse Doctorat de l'Université Picardie Jules Vernes, Promotion 2008.
- [9] **FAS Mohamed Lamine** «La Commande Prédictive Généralisée robustifiée par la paramétrisation de Youla» Mémoire de magister de l'Université Mentouri Constantine, Promotion 2005.

Internet :

[http:// www.ira.eu](http://www.ira.eu) (Guide d'intégration du régulateur R-S-T sur cibles programmables, Adaptech)

[http:// www. fredgouaisbaut.free.fr](http://www.fredgouaisbaut.free.fr)

[http:// www.infranor.ch](http://www.infranor.ch)

ANNEXE

Programme pour le placement de pôles :

```
%maiche chapot
clear all
close all
clc
Gd=tf([0 -0.4848 0.2574],[1 -0.9841],0.002,'variable','z^-1');
% Gd=c2d(G,0.025);
[B,A]=tfdata(Gd,'v');
step(Gd)
Te=0.002;
tr=0.175;
ksi=0.707;
wn=2.16/tr;
fd=tf([wn^2],[1 2*ksi*wn wn^2]);
fdd=c2d(fd,Te);
pf=pole(fdd)
[num,P1]=tfdata(fdd,'v')
p1=-ksi*wn+i*sqrt(1-ksi^2);
p2=-ksi*wn-i*sqrt(1-ksi^2);
z1=exp(p1*0.025);
z2=exp(p2*0.025);
% P=conv([1 -z1],[1 -z2])
% Hm=tf(sum(P),P,0.025);
I=[1 -1];
Ai=conv(A,I);
m1=[Ai 0]';m2=[0 Ai]';m3=[B 0 0]';m4=[0 B 0]';m5=[0 0 B]';
M=[m1 m2 m3 m4 m5]
% P=[P 0 0]';
P1=[P1 0 0]';
X=inv(M)*P1;
```

ANNEXE

```
S1=X(1:2)';
R=X(3:5)'
S=conv(I,S1)
C=tf(R,S,0.002);
H=feedback(C*Gd,1);
z=zero(H)
p=pole(H)
step(H)
hold on
% step(Hm,'r')
% hold on
step(fdd,'r')
T=sum(R)
C=tf(1,S,Te,'variable','z^-1');%%%%%%%%%%C1=filt(1,S,Te)
R=filt(R,1,Te);
Hc=T*feedback(C*Gd,R);
zc=zero(Hc)
pc=pole(Hc)
step(Hc,'black',1)
```

Programme de test de robustesse :

```
%teste de Robustesse
clear all
close all
clc
Gd=tf([0 -0.4848 0.2574],[1 -0.9841],0.002,'variable','z^-1');
% Gd=c2d(G,0.025);
[B,A]=tfdata(Gd,'v');
% step(G)
Te=0.002;
```

ANNEXE

```
tr=0.175;
ksi=0.707;
wn=2.16/tr;
fd=tf([wn^2],[1 2*ksi*wn wn^2]);
fdd=c2d(fd,Te);
pf=pole(fdd)
[num,P1]=tfdata(fdd,'v')
p1=-ksi*wn+i*sqrt(1-ksi^2);
p2=-ksi*wn-i*sqrt(1-ksi^2);
z1=exp(p1*0.002);
z2=exp(p2*0.002);
I=[1 -1];
Ai=conv(A,I);
m1=[Ai 0]';m2=[0 Ai]';m3=[B 0 0]';m4=[0 B 0]';m5=[0 0 B]';
M=[m1 m2 m3 m4 m5]
% P=[P 0 0]';
P1=[P1 0 0]';
X=inv(M)*P1;
S1=X(1:2)';
R=X(3:5)'
S=conv(I,S1)
C=tf(R,S,0.002);
H=feedback(C*Gd,1);
[Nbo,Dbo]=series(R,S,B,A); %Fonction de transfert en boucle ouverte.
[Nbo,Dbo]=minreal(Nbo,Dbo); %Simplifications.
w=tf(Nbo,Dbo,Te)
w=logspace(-1,log10(pi/Te),400); %Réponse fréquentielle en boucle ouverte.
dbode(Nbo,Dbo,Te,w) %Tracé des diagrammes de Bode en boucle ouverte.
[mo,ph]=dbode(Nbo,Dbo,Te,w) %
[Gm,Pm,Wcg,Wcp]=margin(mo,ph,w) %Calcul des marges de phase et de gain.
Gmdb=20*log10(Gm)
```

