

Université A.MIRA – BEJAIA

Faculté de la Technologie

DEPARTEMENT DE GENIE ELECTRIQUE



Mémoire de fin d'étude

**En vue de l'obtention du diplôme de Master
En électrotechnique
Option automatisme industriel**

Thème

**RÉALISATION D'UNE INTERFACE GRAPHIQUE
D'AIDE AU DIMENSIONNEMENT D'UN
RÉGULATEUR RST AVEC
GUIDE MATLAB**

Réaliser par :

- BOUHAFS Lyes
- BOUAMAMA Ramadhane

Encadrer par :

Mr. AZZI Abdelmalek

Promotion 2013

Remerciements

Nous remercions le Dieu de nous avoir donné la force et le courage pour réaliser ce modeste travail. Nous remercions aussi Mr M.AZZI pour nous avoir dirigé et aider pendant toute la durée de réalisation de ce travail et dont les critiques et les conseils nous ont été très précieux.

Nous adressons nos plus vifs remerciements aux membres du jury pour l'honneur qu'ils nous font en acceptant la charge de juger ce travail.

Nous sommes reconnaissants envers nos enseignants auxquels nous devons notre formation en électrotechnique grâce à l'enthousiasme qu'ils ont su nous communiquer en tant qu'étudiants lors de notre cursus d'études à l'UAMB.

*Enfin, nous ne pourrions terminer ces remerciements sans remercier **notre famille** pour leurs aides, compréhensions, encouragements et soutiens, qu'elles nous ont apportés tout le long de nos études et à tous nos amies.*

Dédicaces

A qui puis je dédier cet humble travail si ce n'est à mes très chers

Parents, dont le sacrifice, la tendresse, l'amour, la patience, le soutien, l'aide et les encouragements sont l'essence de ma réussite. Sans eux, je ne serais pas ce que je suis aujourd'hui.

Je dédie ce travail également :

A mes frères et sœurs, Belkacem, Sifou, Lilya, Lina

A toute la famille BOUHAFS.

A mes amis Ferhat, nono, Bodo, Farouk, Nadir, Azzeddine, Younes, Moussa, jigou, halim, housine ...

A tous ceux dont les noms m'ont échappé.



BOUHAFS Lyes

Dédicaces

A qui puis je dédier cet humble travail si ce n'est à mes très chers

Parents, dont le sacrifice, la tendresse, l'amour, la patience, le soutien, l'aide et les encouragements sont l'essence de ma réussite. Sans eux, je ne serais pas ce que je suis aujourd'hui.

Je dédie ce travail également :

A mes frères et sœurs,

A toute la famille BOUAMAMA.

A mes amis Marsiel, Shafik, Hamza, Bilal, Lamine, ...

A tous ceux dont les noms m'ont échappé.



BOUAMAMA Ramadhane

SOMMAIRE

Introduction générale	01
Chapitre I : Description des Méthodes RST	
I.1 Introduction	03
I.2 Régulation numérique	03
I.2.1 Principe de la régulation numérique	03
I.3 Régulateur RST	04
I.3.1 Définition	04
I.4 Synthèse du Régulateur RST	05
1) détermination du polynôme $T(z)$	05
2) Détermination des polynômes $R(z)$ et $S(z)$	09
I.5 Exemples d'applications	10
1) Système 2 ^{ème} ordre	10
2) Système 1 ^{er} ordre	12
I.6 Conclusion	16
Chapitre II Réalisation de l'interface graphique d'aide au dimensionnement des polynômes R, S et T	
II.1 Introduction	18
II.2 Définition d'une interface graphique	19
II.3 L'interface de Guide Matlab	20
II.4 Les notions principales d'interface graphique	21
II.4.1 Les identifiants	21
II.4.2 Les propriétés des objets	22
II.5 Principaux objets graphiques	22
II.5.1 Bouton poussoir	22
II.5.2 Menus	23
II.5.3 Ascenseur ou slider	23
II.5.4 Texte Editable	23
II.5.5 Liste de choix	24
II.6 L'outil SIMULINK	24
II.6.1 Présentation	25

SOMMAIRE

II.6.2 Paramètres de simulation	25
II.6.3 Mode de simulation	26
II.6.4 Instant de départ et d'arrivée	27
II.6.5 Pas initial et maximum de simulation	27
II.6.6 Tolérance relative et la Tolérance Absolue	27
II.7 l'interface graphique réalisée	27
II.7.1 Cahier des charges	27
II.7.2 présentation de l'interface graphique	29
A. 1er panneau	29
A.1 premier sous panneaux	30
A.2 Deuxième sous panneau	31
A.3 Troisième sous panneau	31
A.4 Quatrième sous panneau	31
A.5 Cinquième sous panneau	32
A.6 Sixième sous panneau	32
B. 2eme panneau	33
C. 3eme panneau	34
C.1 premier sous panneau	34
C.2 Deuxième sous panneau	35
C.3 troisième sous panneau	35
D. 4eme panneau	35
E. Barre d'outils	36
II.7.3 Exemple d'application	36
1) Avec un seul pôle	36
a) Cas système avec intégrateur	37
b) Cas système sans intégrateur	44
2) Avec deux pôles	46
II.8 Conclusion	48
Chapitre III Application de la commande d'un moteur à courant continu	

SOMMAIRE

III.1 Introduction	50
III.2 Carte DSPACE RTI1104	51
III.2.1 Création d'un model en temps réel dans Simulink/DSPACE RTI1104	51
III.2.2 configuration des paramètres de simulation du modèle	52
III-3 Identification de la fonction de transfert du moteur à courant continu	52
III.4 Application	53
III.5 Essai pratique	56
III.6 Conclusion	59
Conclusion générale	61
Références bibliographiques	
ANNEXES	

NOMENCLATURE

RST	régulateur a trois polynômes $R(z)$, $S(z)$ et $T(z)$
PID	régulateur à action Proportionnel, Intégral et Dérivé
CAD	convertisseur analogique numérique
CDA	convertisseur numérique analogique
$FTBO$	fonction de transfert en boucle ouverte
$FTBF$	fonction de transfert en boucle fermé
$G(p)$	système continu
$H(z)$	système discret
$A_0(z)$	polynôme observateur
$Y_c(z)$	La consigne
$Y(z)$	mesure de la grandeur à maitriser
$U(z)$	signale de commande
ζ	coefficient d'amortissement
ρ	degré du polynôme R
σ	degré du polynôme S
τ	degré du polynôme T
ℓ	degré d'intégration.
p	variable complexe de la transformé de LAPLACE
$A(z)$	dénominateur du système
$B(z)$	numérateur du système
$B^+(z)$	polynôme intervenant dans la factorisation spectrale de $B(z)$
$B^-(z)$	polynôme intervenant dans la factorisation spectrale de $B(z)$
$H_m(z)$	fonction de transfert du modèle à poursuivre
$B_m(z)$	numérateur du modèle à poursuivre
$A_m(z)$	dénominateur du modèle à poursuivre
$R(z)$	polynôme intervenant dans le régulateur RST
$R'(z)$	polynôme en facteur dans le polynôme $R(z)$
$S(z)$	polynôme intervenant dans le régulateur RST
$T(z)$	polynôme intervenant dans le régulateur RST

NOMENCLATURE

δR	degré du polynôme $R(z)$
δS	degré du polynôme $S(z)$
δT	degré du polynôme $T(z)$
δA_m	degré du polynôme $A_m(z)$
δB_m	degré du polynôme $B_m(z)$
δA	degré du polynôme $A(z)$
δB	degré du polynôme $B(z)$
δA_0	degré du polynôme $A_0(z)$
δB^+	degré du polynôme $B^+(z)$
d	nombre de périodes d'échantillonnage dans un retard pur
$P(z)$	polynôme en facteur dans le dénominateur du modèle à poursuivre
c_1, c_2	coefficients du polynômes $P(z)$

LISTE DES FIGURES

Figure I.1	Schéma de base de la commande numérique	03
Figure I.2	Modélisation discrète du système asservis	04
Figure I.3	Schéma fonctionnel de montage en asservissement avec un régulateur	04
Figure I.4	Principe de la synthèse du régulateur RST	06
Figure I-5	Réponse indiciel d'un système discret de premier ordre en fonction de son pôle	08
Figure I-6	La réponse indicielle du système continu et celle du système discret en boucle ouverte	10
Figure I.7	La réponse indicielle du système continu et celle du système discret en boucle fermé en présence du Régulateur RST	12
Figure I-8	Boucle de Régulation d'un processus du premier ordre avec Régulateur RST en boucle fermé et sans Régulateur RST en boucle ouverte	13
Figure I-9	réponse indiciel de processus en boucle ouverte et fermé sollicité par une perturbation	14
Figure I.10	Réponse indicielle de système bouclé	14
Figure I.11	schéma de simulation système premier ordre avec le régulateur RST en boucle fermé	15
Figure I.12	réponse indiciel de système premier ordre corrigé avec le régulateur RST avec intégrateur	15
Figure I.13	réponse indiciel du système corrigé avec le régulateur RST, sans intégrateur	16
Figure II.1	Environnement de Matlab	18
Figure II.2	la hiérarchie parent-enfants des objets	20
Figure II.3	fenêtre principal du guide	21
Figure II.4	Librairie de Simulink	25
Figure II.5	présentation de l'interface	29
Figure II.6	présentation du 1 ^{er} panneau	30
Figure II.7	présentation de sous panneau 1	30
Figure II.8	présentation de sous panneau 2	31
Figure II.9	présentation de sous panneau 3	31
Figure II.10	présentation de sous panneau 4	31
Figure II.11	présentation de sous panneau 5	32

LISTE DES FIGURES

Figure II.12	présentation de sous panneau 6	32
Figure II.13	présentation du 2eme panneau	33
Figure II.14	présentation du 3eme panneau	34
Figure II.15	présentation de sous panneau 1	34
Figure II.16	présentation de sous panneau 2	35
Figure II.17	présentation de sous panneau 3	35
Figure II.18	présentation du 4eme panneau	35
Figure II.19	La barre d'outils	36
Figure II.20	L'ordre du système choisi	37
Figure II.21	Introduire les paramètres	38
Figure II.22	choix de la méthode	38
Figure II.23	les signaux de commande et de réponse du système, sans simplification des zéros du système à régler et sans intégrateur, Axes	40
Figure II.24	les signaux de commande, réponse du système et la référence, sans simplification des zéros du système à régler et sans intégrateur, figure	41
Figure II.25	les signaux de commande, réponse du système et la référence, sans simplification des zéros du système à régler et sans intégrateur, figure, pour $c_1 = -0.6$	42
Figure II.26	les signaux de commande et de réponse du système, avec simplification des zéros du système à régler et sans intégrateur, Axes	43
Figure II.27	les signaux de commande, réponse du système et la référence, avec simplification des zéros du système à régler et sans intégrateur, figure	44
Figure II.28	les signaux de commande et de réponse du système, avec simplification des zéros du système à régler et avec intégrateur	45
Figure II.29	les signaux de commande, réponse du système et la référence, avec simplification des zéros du système à régler et avec intégrateur	46
Figure II.30	la réponse du système pour un seul pôle et le pôle multiple	47
Figure III.1	Carte DS1104	51
Figure III.2	signaux de commande, de référence, réponse de système pour système 1er ordre corriger avec le régulateur RST avec intégrateur, figure	56
Figure III.3	interface utilisateur Control Desk	57
Figure III.4	signaux de sortie et de la référence essai pratique	58

LISTE DES FIGURES

Figure III.4	signal de commande essai pratique	58
--------------	-----------------------------------	----

Introduction générale

Introduction générale

Pour un système de régulation, les spécifications restent souvent vagues en raison surtout de la grande diversité de problèmes de régulation. Les critères qualitatifs à imposer dépendent d'abord de la nature du processus.

Le choix du régulateur à insérer dans un système, au moyen des paramètres choisis, permet l'obtention de comportement désiré en boucle fermée.

Cependant Les performances d'une régulation peuvent se définir à partir de l'allure du signal de mesure suite à un échelon de consigne afin de répondre aux critères désirés

Notons toutefois que les critères de performances peuvent se résumer comme suit :

- ✓ Stabilité.
- ✓ Précision.
- ✓ Rapidité.
- ✓ Dépassement.

L'objectif est de dimensionner le régulateur à interpoler dans le procédé après ça modélisation par une fonction de transfert en formons une boucle fermer est d'aboutir à des performances désirée.

Ce document résume notre travail d'une manière synthétique avec un apport expérimental permettant de valider la partie théorique et à l'aide d'une interface graphique sous guide Matlab.

De plus encore on va réaliser dans ce travail une interface graphique avec le guide MATLAB qui va nous permettre de dimensionner le régulateur RST. Et pour cela on a partagé notre travail en trois parties suivantes :

Le premier chapitre est consacré à la description des méthodes de synthèse du régulateur RST et de déterminer les algorithmes pour le dimensionnement.

Dans le deuxième chapitre, on présente l'outil de programmation des interfaces GUIDE MATLAB ensuite l'interface graphique réalisée.

Le dernier chapitre consacré pour la partie pratique a objectif d'identifier notre système qui est un moteur à courant continue afin d'élaborer la commande RST par l'exploitation de l'interface graphique et la carte Dspace Rti1104 et son logiciel control Desk.

Chapitre I

Description des méthodes

RST

I.1 Introduction

Ce chapitre est entièrement dévolu au régulateur RST, ce nom prévenant des trois polynômes qu'il fait intervenir. Le régulateur RST est défini dans le reste de chapitre, temps en temps, il est obligatoire d'introduire un intégrateur pour assurer l'erreur statique nulle, ainsi la compensation des zéros du système à régler. La méthode de synthèse du régulateur RST avec retard est mise en œuvre.

Dimensionné ce régulateur qui va être associé en boucle fermée avec un système modélisé par une fonction de transfert.

I.2 Régulation numérique

I.2.1 Principe de la régulation numérique

Ce type de régulation réalisé principalement à l'aide de micro-ordinateurs prend également le nom de régulation échantillonnée, le processus à réguler étant généralement de nature continue même si sa modulation est strictement discrète.

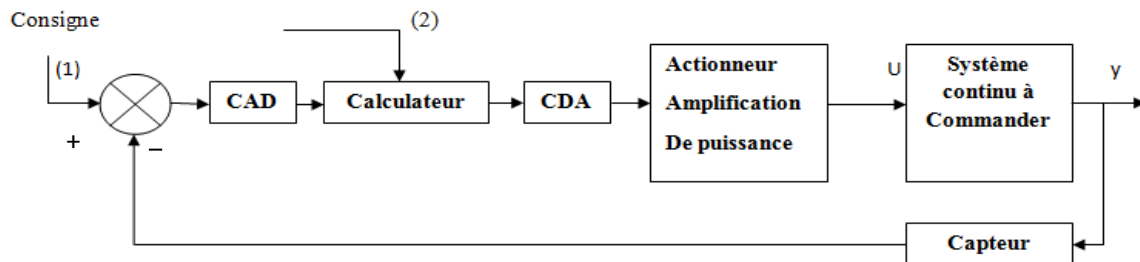


Figure I.1 Schéma de base de la commande numérique

Sur le schéma de la (figure I.1), on voit apparaître un convertisseur analogique digital (CAD) qui transforme en valeur numérique, les valeurs échantillonnées des variables à manipuler, et un convertisseur digital analogique (CDA) qui permet d'obtenir, le plus souvent sous la forme d'un signal électrique continu, l'information fournie par le calculateur.

Ce signal est alors adapté sous une forme permettant la commande du processus avec en général amplification de Puissance.

La consigne peut être de façon équivalente appliquée sous forme analogique (en 1) ou directement sous forme numérique au niveau du calculateur (en2) .modélisé sous forme

discrète un tel système échantillonné peut être décrit à l'aide de la transformé en z selon le schéma de la (figure I.2). [1]

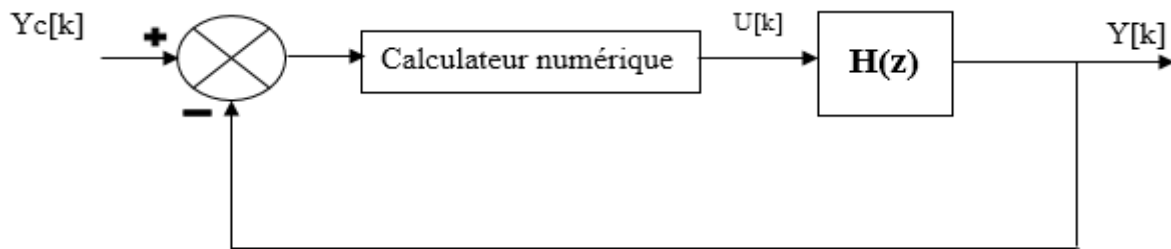


Figure I.2 Modélisation discrète du système asservi

I.3 Régulateur RST

I.3.1 Définition

Le régulateur RST peut être semblable à un régulateur PID digital mais avec quelques avantages de plus. On utilise une structure qui agit différemment sur la consigne et sur la sortie. Pour obtenir un comportement différent pour une variation de consigne ou une perturbation, on peut appliquer un filtre polynomial sur la consigne, et son contrôle permet d'effectuer une régulation en boucle fermée d'un système industriel. Autre mot dit, c'est un correcteur couramment utilisé dans les systèmes de commande numérique. Le sigle RST vient du nom des 3 polynômes 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 met en œuvre un polynôme de poursuite [2].

La structure générale du régulateur de RST est représentée à la (figure I.3) suivante :

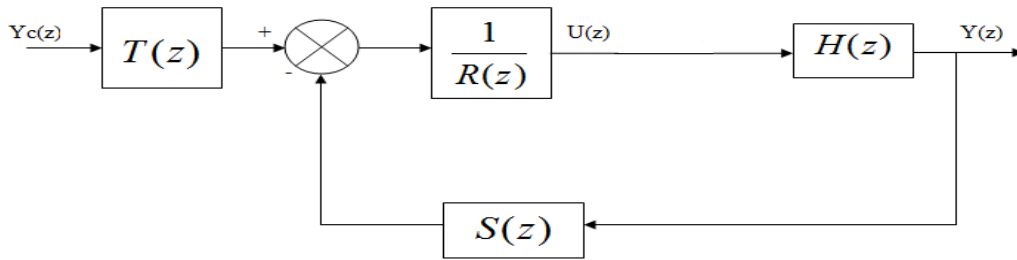


Figure I.3 Schéma fonctionnel de montage en asservissement avec un régulateur

Le régulateur RST est composé de deux parties :

- ✓ partie du régulateur interne : $S(z) 1/R(z)$
- ✓ partie externe pour garantir le comportement dynamique face à la consigne : $T(z) 1/S(z)$

Les polynômes R, S, T se présentent sous la forme générale suivante. [1]

$$\begin{cases} R(z) = z^\rho + r_{\rho-1} * z^{\rho-1} + \dots + r_0 & \text{'}\rho\text{' désigne le degré du polynôme R.} \\ S(z) = s_\sigma * z^\sigma + s_{\sigma-1} * z^{\sigma-1} + \dots + s_0 & \text{'}\sigma\text{' désigne le degré du polynôme S} \\ T(z) = t_\tau * z^\tau + t_{\tau-1} * z^{\tau-1} + \dots + t_0 & \text{'}\tau\text{' désigne le degré du polynôme T} \end{cases}$$

$B(z)$: représente le numérateur du système discret.

$A(z)$: représente le dénominateur du système discret.

La fonction de transfert en boucle fermée :

$$\frac{y(z)}{y_c(z)} = \frac{B(z) * T(z)}{A(z) * R(z) + B(z) * S(z)} \quad (I.1)$$

Ce régulateur est dit "poursuite et régulation à objectifs indépendants".

Le calcul de la loi de commande est basé sur un critère polynomial qui permet de spécifier la dynamique de réjection de l'erreur entre le signal de référence et le signal de sortie (perturbation). [4]

Une des racines de $S(z)$ sert à compenser le zéro subsistant en boucle fermée, l'autre racine forme une paire avec celle de $T(z)$ qui permet de diminuer ou d'augmenter le dépassement selon que la racine de $T(z)$ est plus grande ou plus petite que celle de $S(z)$. Dans tous les cas, le gain statique de $[T(z)/S(z)]$ doit être à 1.

La loi de commande sous forme d'équation canonique :

$$u(z) = \frac{T(z)}{R(z)} * y_c(z) - \frac{S(z)}{R(z)} * y(z) \quad (I.2)$$

I.4 Synthèse du Régulateur RST

Dans le but de dimensionner le régulateur RST, plusieurs étapes à suivre

1) détermination du polynôme $T(z)$

Les polynômes $R(z)$, $S(z)$ et $T(z)$ du régulateur RST vont être dimensionnés afin que cette fonction de transfert en boucle fermée soit identique à la fonction de transfert $H_m(z)$ d'un modèle à poursuivre, ou modèle de référence, donnée par l'utilisateur :

$$H_m(z) = \frac{B_m(z)}{A_m(z)} \quad (I.3)$$

$H_m(z)$ est une fonction rationnelle propre ; de plus le polynôme $A_m(z)$ est Monique et ses zéros sont tous à l'intérieur du cercle unité, la (figure I.4) montre le principe de la synthèse.

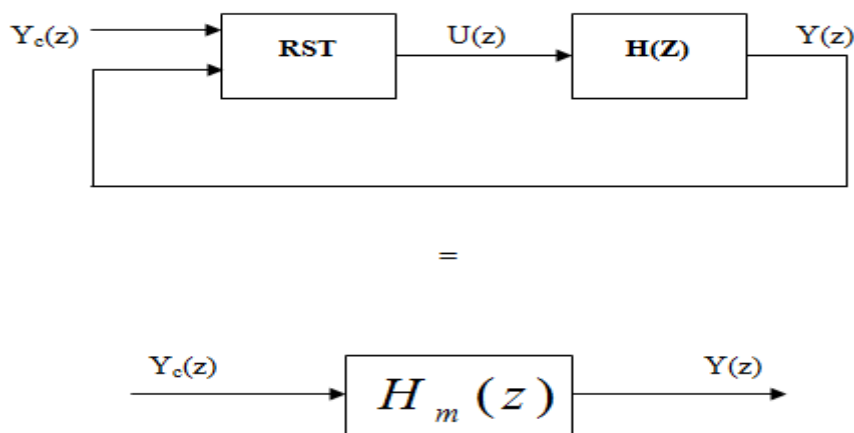


Figure I.4 Principe de la synthèse du régulateur RST

Le dimensionnement du régulateur RST généralise considérablement la synthèse dans le lieu des pôles. On désire toujours placer les pôles du système en boucle fermée dans le but de maîtriser le régime transitoire.

Le degré de $A_m(z)$ n'est pas forcément égal à celui de $A(z)*R(z) + B(z)*S(z)$; en fait, un modèle à poursuivre très simple, avec un polynôme $A_m(z)$ de degré nettement inférieur à celui de $A(z)*R(z) + B(z)*S(z)$, est généralement sélectionnée. [4]

Il est possible de compenser des zéros du procédé à l'aide du correcteur, à conditions que ces zéros Soient à l'intérieur du cercle unité afin de garantir la stabilité en boucle fermée.

On décompose le numérateur de la fonction de transfert du procédé de la façon suivante :

$$B(z) = B^+(z) * B^-(z) \quad (I.4)$$

- Le polynôme $B^-(z)$ Contient les zéros du procédé qui ne sont pas compensés.
- Le polynôme $B^+(z)$ contient les zéros du procédé qu'on choisit de compenser.

Si on veut pas simplifiée les zéros du système à régler, on obtient $B^+(z)=1$ et l'équation (I.4) devient :

$$B(z) = B^-(z) \quad (I.5)$$

Le modèle à poursuivre $H_m(z)$ est donné sous forme :

$$H_m(z) = \frac{B^-(z) * P(1)}{\frac{B^-(1)}{Z^d * P(z)}} \quad (I.6)$$

Dans ce cas : $B_m(z) = \frac{B^-(z) * P(1)}{B^-(1)} , \quad A_m(z) = Z^d * P(z)$

$$B'_m(z) = \frac{P(1)}{B^-(1)} \quad (I.7)$$

Le facteur Z^d dans $A_m(z)$, provoquant un retard de d périodes d'échantillonnage.

Le nombre $\frac{P(1)}{B^-(1)}$ assure $H_m(1)=1$, le polynôme $P(z)$ est Monique et de degré 1 ou 2 selon

que des oscillations de la grandeur à régler sont bannies ou tolérées, respectivement :

$$\left\{ \begin{array}{l} P(z) = z + c_1 \\ \text{Ou} \\ P(z) = (z + c_1) * (z + c_2) \end{array} \right. \quad (I.8)$$

Les coefficients c_1 et c_2 sont sélectionnés de manière à ce que des conditions absolue et relative d'amortissement soient remplies. [5]

Dans la (figure I.5), on trouvera quelques exemples de comportements de systèmes du premier ordre en fonction du lieu du pole. [6]

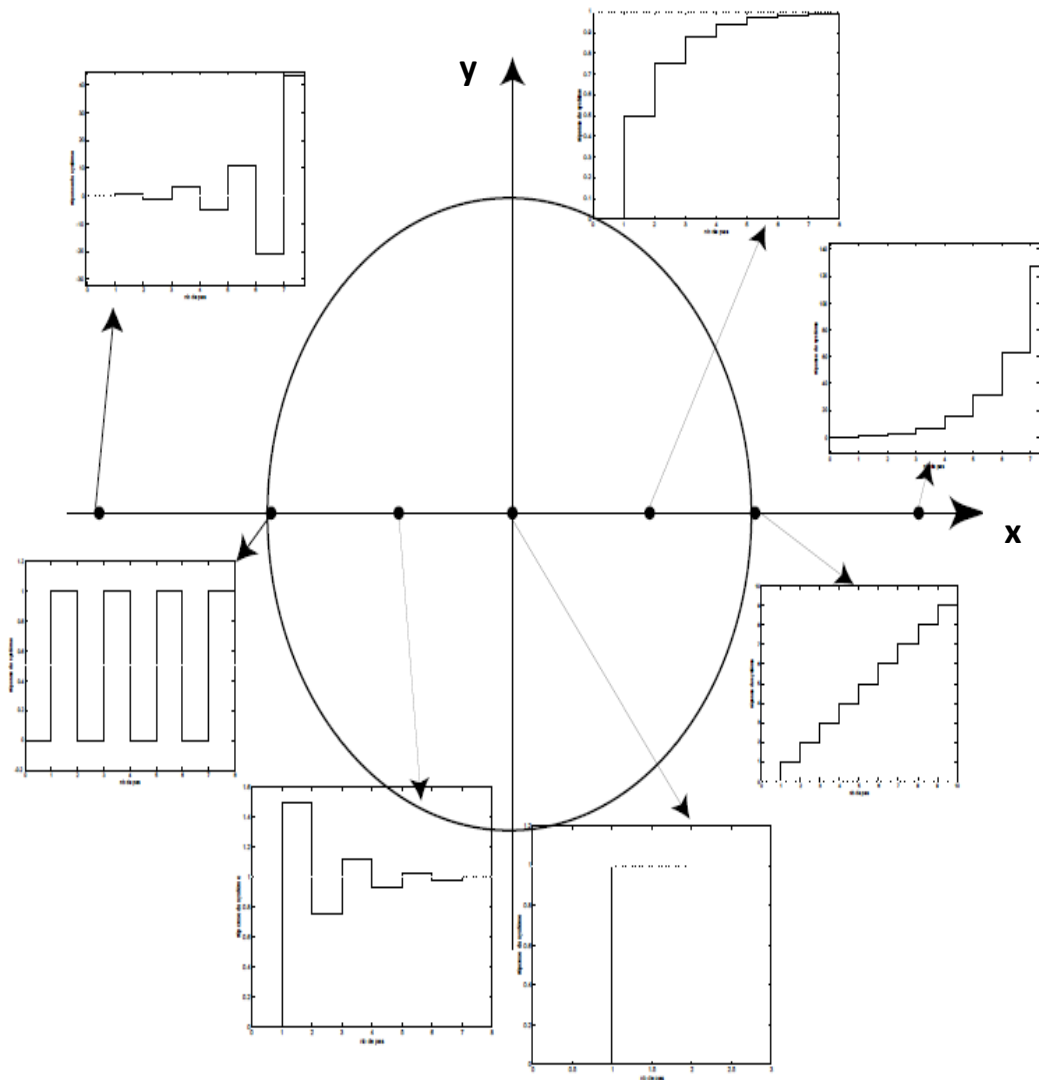


Figure I.5 Réponse indiciel d'un système discret de premier ordre en fonction de son pôle

Un autre polynôme $A_0(z)$, appelé polynôme observateur, ses zéros doivent nécessairement être à l'intérieur du cercle, il permet d'ajuster le gain statique à 1. [1]

$$A_0(z) = z^{\delta A_0} \quad (I.9)$$

Tel que : δA_0 représente le degré du polynôme $A_0(z)$

La détermination de δA_0 dépend d'un autre paramètre, c'est l'effet intégrateur ℓ

Son insertion est selon les cas suivants :

- Système possède un intégrateur, dans ce cas $\ell=0$
- Système ne possède pas un intégrateur, dans ce cas $\ell \neq 0$

Ou

$$\delta A_0 \geq 2 * \delta A - \delta A_m - \delta B^+ + \ell - 1$$

Détermination de polynôme T(z) du régulateur RST :

$$T(z) = B_m'(z) * A_0(z) \quad (I.10)$$

2) Détermination des polynômes R(z) et S(z)

Cette fois on passe à déterminer les polynômes R(z) et S(z) on commençant par leurs degrés :

a) Cas avec intégrateur :

$$\begin{aligned} \delta R' &= \delta A_m + \delta A_0 - \delta A \\ \delta S &= \delta A + \ell - 1 \end{aligned}$$

sachant que : $R(z) = (z-1)^\ell * R'(z)$

b) cas sans intégrateur :

$$\begin{aligned} \delta R' &= \delta A_m + \delta A_0 - \delta A \\ \delta S &= \delta A - 1 \\ R(z) &= R'(z) \end{aligned}$$

L'équation de Diophante :

$$A(z) * (z-1)^\ell * R'(z) + B^-(z) * S(z) = A_m(z) * A_0(z)$$

La résolution de l'équation de Diophante permet de déterminer les polynômes R(z) et S(z), consiste à évaluer les coefficients des termes de même degré des polynômes $A(z) * R(z) + B(z) * S(z)$ et $C(z)$. Il en découle un système d'équations algébrique linéaire donnant les coefficients inconnus des polynômes R(z) et S(z).

Dans le contexte de la synthèse du régulateur RST, le système linéaire associé à l'équation Diophante présente le plus souvent l'allure suivante :

$$\begin{bmatrix} 1 & 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ a_1 & 1 & \dots & 0 & b_0 & 0 & \dots & 0 \\ a_2 & a_1 & \dots & 1 & b_1 & b_0 & \dots & 0 \\ \vdots & a_2 & \dots & a_1 & b_2 & b_1 & \dots & 0 \\ \vdots & \vdots & \dots & a_2 & \vdots & b_2 & \dots & b_0 \\ a_{\delta A} & \vdots & \dots & \vdots & b_{\delta B} & \vdots & \dots & b_1 \\ 0 & a_{\delta A} & \dots & \vdots & 0 & b_{\delta B} & \dots & b_2 \\ \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & a_{\delta A} & 0 & 0 & \dots & b_{\delta B} \end{bmatrix} \begin{bmatrix} r_0 \\ r_1 \\ \vdots \\ r_{\delta R} \\ s_0 \\ s_1 \\ \vdots \\ s_{\delta S} \end{bmatrix} = \begin{bmatrix} c_1 - a_1 \\ c_2 - a_2 \\ \vdots \\ c_{\delta A} - a_{\delta A} \\ c_{\delta A+1} \\ c_{\delta A+2} \\ \vdots \\ s_{\delta S + \delta R + 1} \end{bmatrix}$$

La matrice d'ordre $\delta R + \delta S + 1$ de ce système est dite matrice de Sylvestre. [5]

Sa résolution peut s'effectuer par :

- élimination gaussienne.
- ou via une factorisation standard de la matrice de Sylvestre.

I.5 Exemples d'applications

1) Système 2^{ème} ordre

On considère un système du second ordre très faiblement amorti ($\zeta = 0,1$) et de pulsation propre non-amortie $\omega_n = 0,7$ rad/s. Son gain statique est égal à 1.

$$G(p) = \frac{0.49}{p^2 + 0.14p + 0.49}$$

On désire un système mieux amorti ($\zeta' = 0,8$) pour lequel $\omega'_n = 1,4$ rad/s, le gain statique étant égal à l'unité. Pour obtenir cela on insère le second ordre dans une boucle de commande numérique (calculateur numérique et son BOZ associé) à retour unitaire. On adopte :

$T_e = (1/1,4)$ s. On calcule $H(z) = [B_0(p) * G(p)]$ avec Matlab .

Le système discret est : $H(z) = \frac{0.1185z + 0.1145}{z^2 - 1.672z + 0.9048}$

La réponse indicielle du système continu $G(p)$ et celle du système discret $H(z)$ sans Régulation :

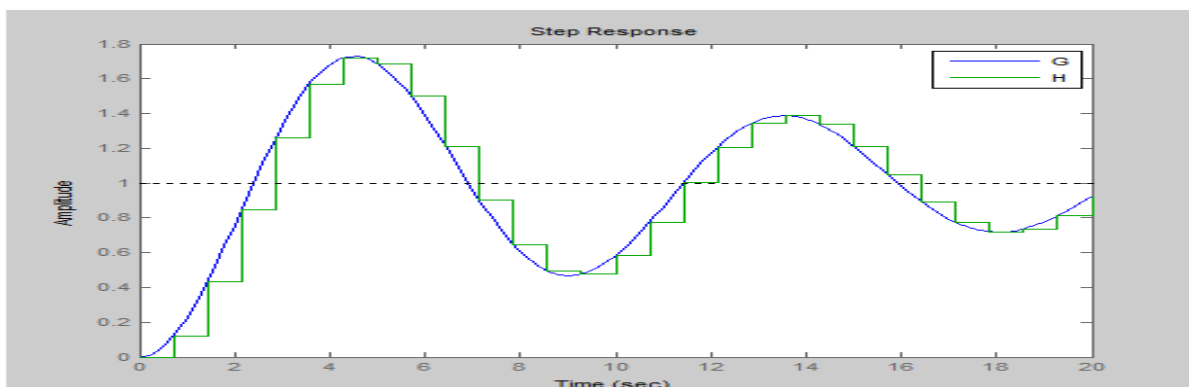


Figure I.6 La réponse indicielle du système continu et celle du système discret en boucle ouverte

On notera :

- l'effet du faible amortissement sur la réponse indicielle ;
- la correspondance des 2 modèles.

On ne désire pas compenser le zéro égal à $-0,9669$. Ainsi

$$\begin{cases} B^-(z) = 0.118 * (z + 0.967) \\ B^+(z) = 1 \end{cases}$$

Déterminons le modèle à poursuivre :

$$\delta A_m - \delta B_m \geq \delta A - \delta B \quad \Longrightarrow \quad \delta A_m - \delta B_m \geq 1$$

Admettons que

$$\begin{cases} B_m(z) = B^-(z) * B_m^+(z) = 0.118 * (z + 0.967) \\ \delta A_m = 2 \end{cases}$$

On désire que le système en boucle fermée admette $\zeta' = 0,8$ et $\omega'_n = 1,4$ rad/s.

Etablissons le modèle à poursuivre. Nous trouvons : $H_m(z) = \frac{0.2336(z+0.967)}{z^2 - 0.7417z + 0.2019}$

$A_0(z)$ permet d'ajuster le gain statique à 1.

$$t_1 = B_m^-(z) = \frac{P(1)}{B^-(1)} = 1.98$$

Sans intégrateur : $\ell = 0$

$\delta A_0 \geq 2 * \delta A - \delta A_m - \delta B^+ + \ell - 1$	$\delta A_0 = 4 - 2 - 0 + 0 - 1 = 1$, le choix de $A_0(z) = z$
$\delta R' = \delta A_m + \delta A_0 - \delta A - \ell$	$\delta R' = 2 + 1 - 2 - 0 = 1$, le choix de $R(z) = z + r_0$
$\delta S = \delta A + \ell - 1$	$\delta S = 2 + 0 - 1 = 1$, le choix de $S(z) = s_1 * z + s_0$

Il faut résoudre l'équation de Diophante : $A(z) * (z-1)^\ell * R'(z) + B^-(z) * S(z) = A_m(z) * A_0(z)$

$$z * (z^2 - 0.74 * z + 0.2) = (z^2 - 1.672 * z + 0.905) * (z + r_0) + 0.118(z + 0.967) * (s_1 * z + s_0)$$

$$\begin{cases} r_0 = 0.449 \\ s_1 = 4.086 \\ s_0 = -3.563 \end{cases} \quad \Longrightarrow \quad \begin{cases} R(z) = z + 0.449 \\ S(z) = 4.086 * z - 0.563 \\ T(z) = 1.98 * z \end{cases}$$

La loi de commande :

$$u(z) = \left(\frac{1.98 * z}{z + 449} \right) * Y_c(z) - \left(\frac{4.086 * z - 3.563}{z + 449} \right) * Y(z)$$

$$\Longrightarrow \boxed{u(k) = -0.449 * u[k - 1] + 1.98 * Y_c[k] - 4.086 * Y[k] + 3.563 * Y[k - 1]}$$

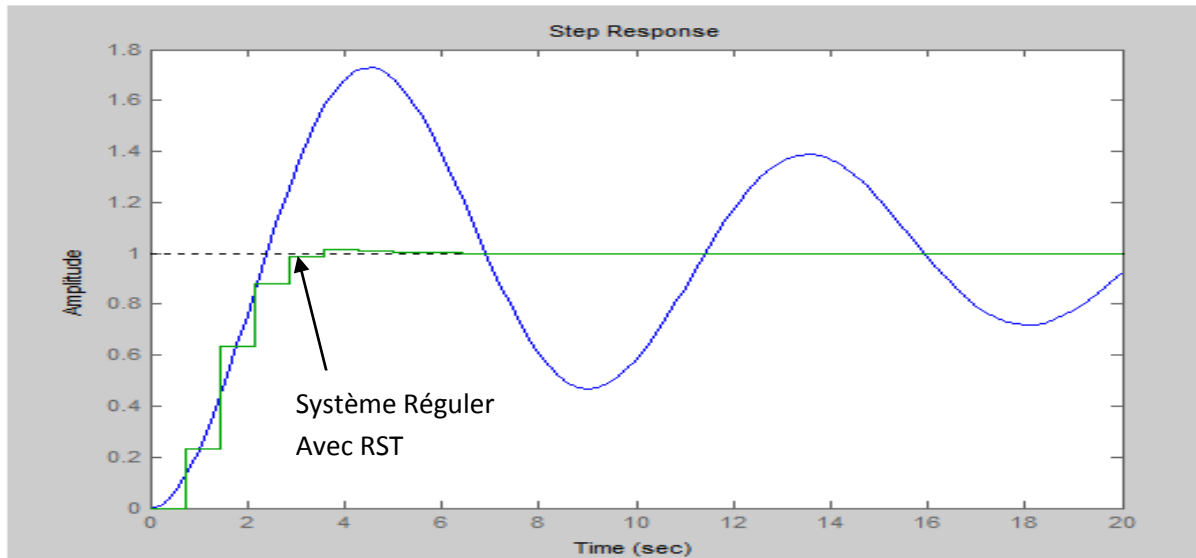


Figure I.7 La réponse indicielle du système continu et celle du système discret en boucle fermée en présence du Régulateur RST

On remarque l'effet de la boucle de commande sur la réponse indicielle du système ainsi corrigé

2) Système 1^{er} ordre

Soit un processus à réguler de transmittance : $G(p) = \frac{4}{p+2}$

La période d'échantillonnage retenue étant $T = 0,025s$, déterminons $H(z)$

$$H(z) = (1 - z^{-1}) * Z \left[\frac{4}{p * (p + 2)} \right]$$

$$\text{D'où : } H(z) = \frac{0.09754}{z-0.9512}$$

On désire améliorer la dynamique du processus à régler. On spécifiera de la manière suivante l'amélioration des performances dynamiques :

On désire que le régime transitoire dû au pôle réel s'atténue d'un facteur en β périodes d'échantillonnage. [7]

Ainsi le pôle réel doit être égal à : $c1 = e^{-\frac{\ln \alpha}{\beta}}$

Ici on a choisit $\alpha = 20$ et $\beta = 5 \implies c1 = -0.55$

$$\text{Le modèle à poursuivre : } H_m(z) = \frac{B^-(z) \frac{p(1)}{B^-(1)}}{z^d * p(z)} = \frac{0.45}{z-0.55}$$

Application de la méthode :

Etapes	Actions	Résultats
Etape1	Vérifie que : $\delta A_m - \delta B_m \geq \delta A - \delta B$	Vérifié puisque : $1-0=1-0$
Etape2	Factoriser $B(z) = B^-(z) * B^+(z)$; $B^+(z)$ monique.	$B^+(z) = 1$ et $B^-(z) = 0.09754$
Etape3	Déterminer $B_m'(z)$ à partir $B_m(z) = B^-(z) * B_m'(z)$	$B_m'(z) = \frac{p(1)}{B^-(1)} = \frac{0.45}{0.0975} = 4.615$
Etape4	Déterminer le nombre ℓ d'intégrations à rajouter	$\ell = 0$
Etape5	$\delta A_0 \geq 2 * \delta A - \delta A_m - \delta B^+ + \ell - 1$ $\delta R' = \delta A_m + \delta A_0 - \delta A$ $\delta S = \delta A + \ell - 1$	$\delta A_0 \geq 2 - 1 - 0 + 0 - 1 = 0$ $\delta R' = 1 + 0 - 1 = 0$ $\delta S = 1 + 0 - 1 = 0$ $A_0(z) = 1 ; R(z) = 1 ; S(z) = s_0$
Etape6	Résoudre : $A(z) * (z-1)^\ell * R'(z) + B^-(z) * S(z) = A_m(z) * A_0(z)$	$(z - 0.55) * 1 = (z - 0.95) * 1 + 0.975 * s_0$ $s_0 = 4.1$
Etape7	Calculer $R(z) = B^+(z) * R'(z)$ et $T(z) = B_m'(z) * A_0(z)$	$R(z) = 1$ $T(z) = 4.615$

Simulons ce régulateur avec SIMULINK.

Sur ce schéma nous avons représenté le processus en boucle avec le régulateur RST et le processus en boucle ouverte. Dans les deux cas nous excitions le dispositif par une perturbation injectée en entrée du processus tel que $y_c = \text{perturbation} = 3$.

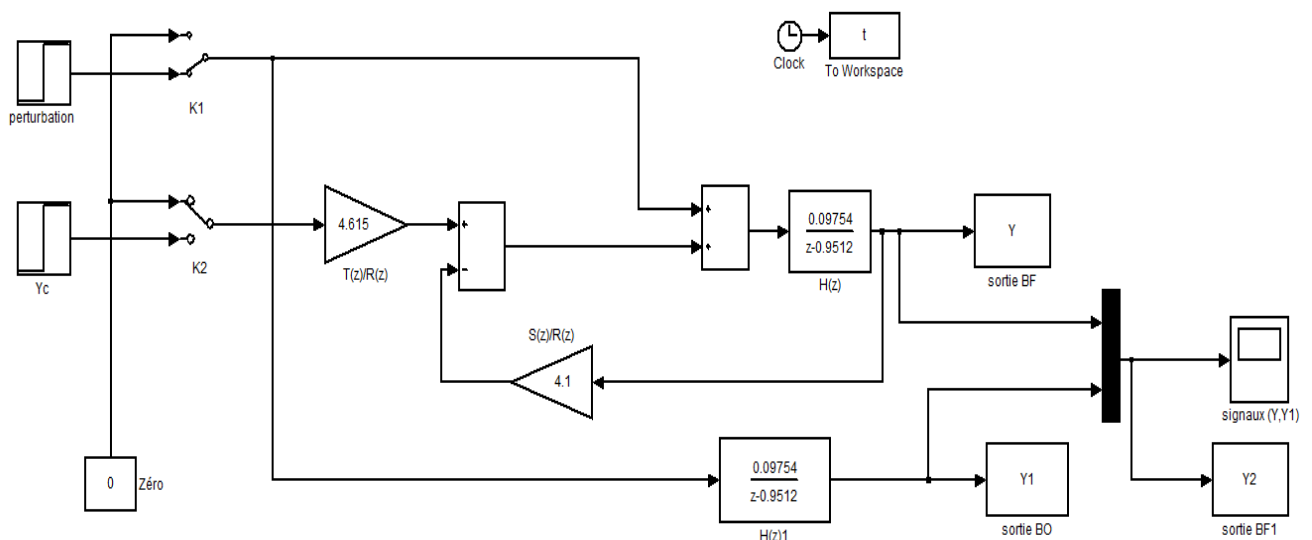


Figure I.8 Boucle de Régulation d'un processus du premier ordre avec Régulateur RST en boucle fermé et sans Régulateur RST en boucle ouverte

Dans le cas ; $K_1=1$ et $K_2=0$

Après simulation de notre model en visualise la réponse indicielle du système discret en boucle ouverte calculer dans cette exemple

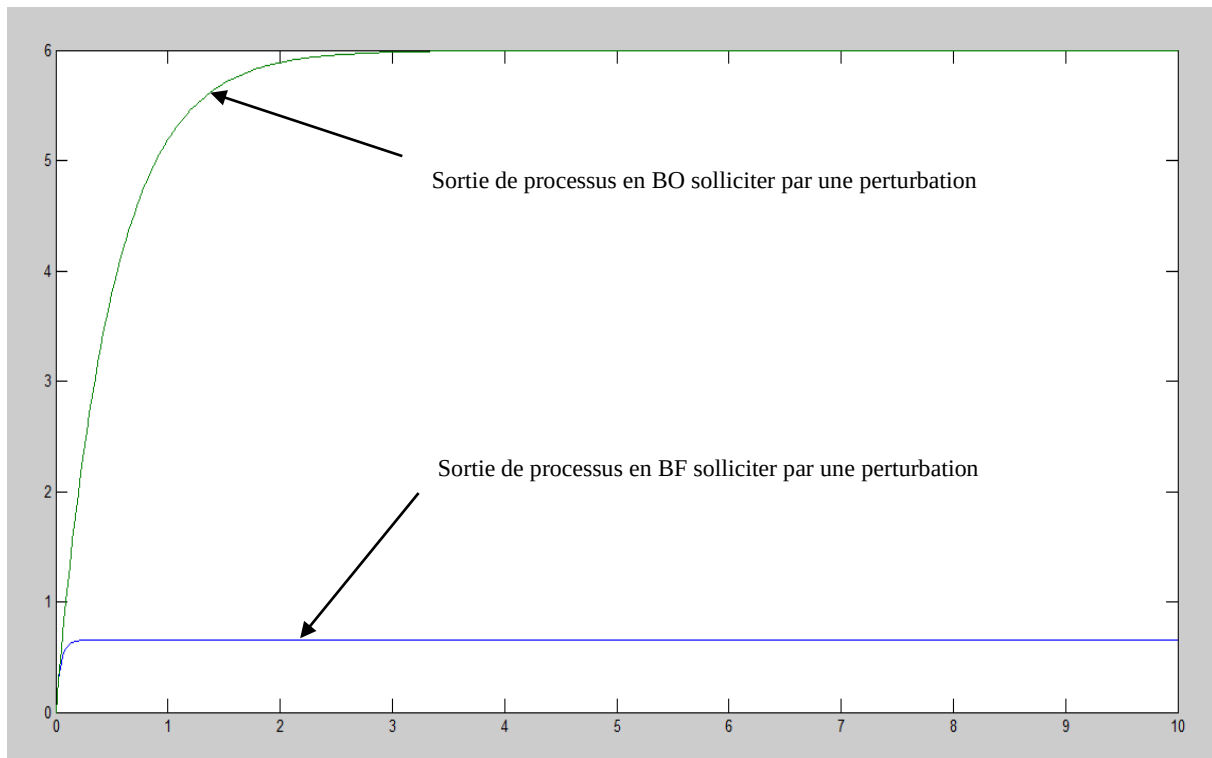


Figure I.9 réponse indiciel de processus en boucle ouverte et fermé sollicité par une perturbation

Dans le cas, $K_1=0$ et $K_2=1$.

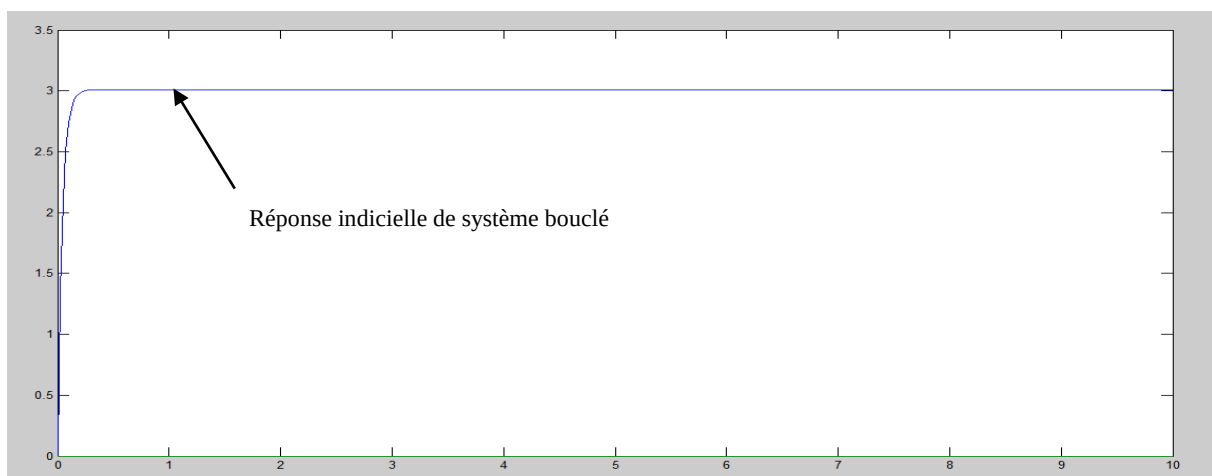


Figure I.10 Réponse indicielle de système bouclé

Pour voir l'effet intégrateur, en redimensionne le régulateur RST on insère l'intégrateur $\ell = 1$, on trouve :

Le polynôme $T(z) = t_1 * z$ avec $t_1 = 4.574$

Le polynôme $s(z) = s_1 * z + s_0$ avec $s_1 = 14.326$ et $s_0 = -9.7519$

La loi de commande est : $u(k) = u[k-1] + t_1 * y_c[k] - s_1 * y[k] - s_0 * y[k-1]$

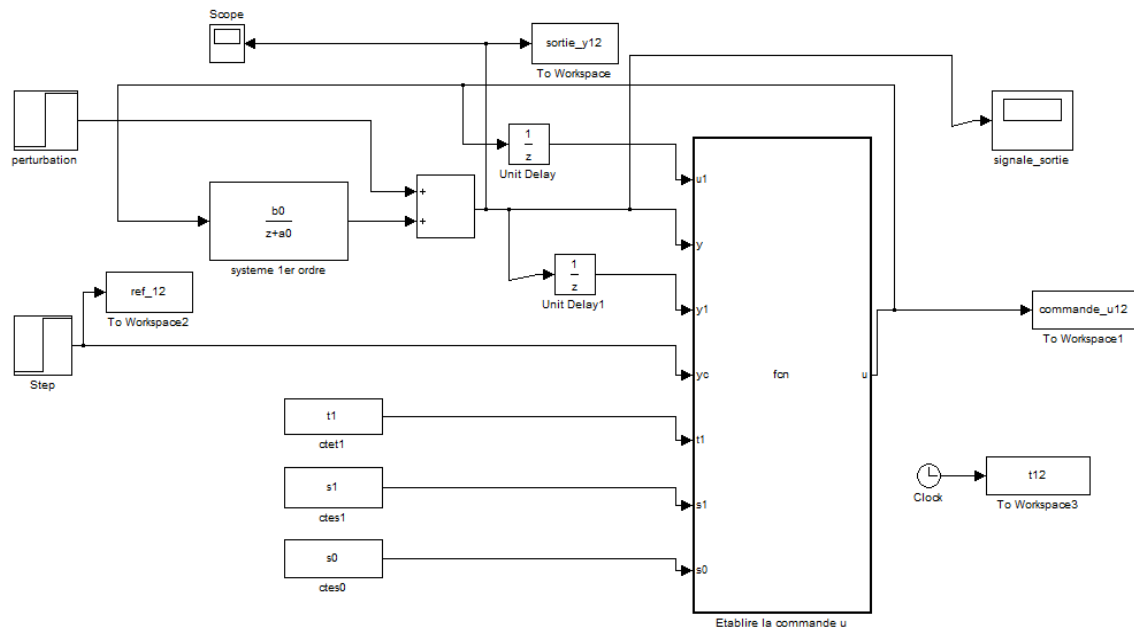


Figure I.11 schéma de simulation système premier ordre avec le régulateur RST en boucle fermé

La (figure I.12) montre la réponse indiciel du système corrigé avec le régulateur RST en présence de la perturbation

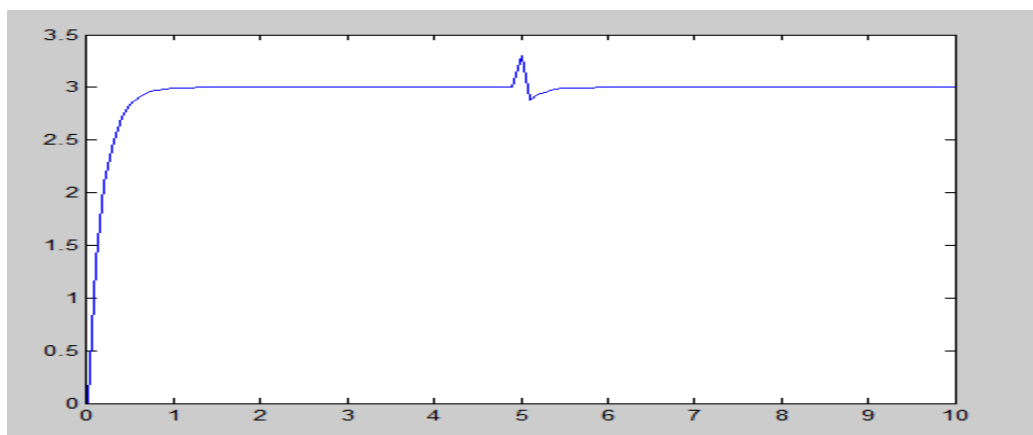


Figure I.12 réponse indiciel de système premier ordre corrigé avec le régulateur RST avec intégrateur

On prend le schéma de simulation de la (figure I.8) avec $k_1=1$ et $k_2=1$ pour visualiser la réponse indicielle du système corrigé avec le régulateur RST mais sans intégrateur

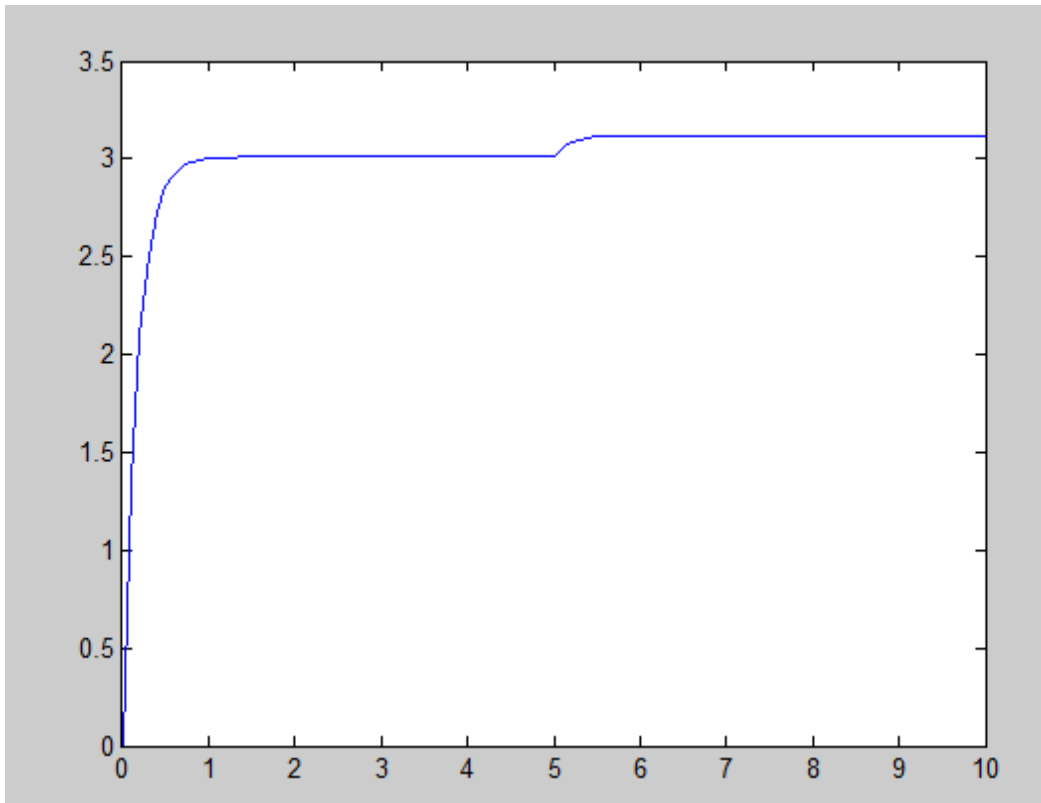


Figure I.13 réponse indicielle du système corrigé avec le régulateur RST, sans intégrateur

D'après la (figure I.13), on remarque que la perturbation n'est pas rejetée

Donc, on compare entre, (figure I.12) et (figure I.13), la présence de l'intégrateur pour le dimensionnement du régulateur RST assure le rejet de la perturbation.

I.6 Conclusion

Dans ce chapitre, on a présenté un régulateur numérique de type RST, et la synthèse de ce dernier nous a permis de décrire les différents cas pour son dimensionnement.

La stratégie pour le dimensionnement du régulateur RST par la méthode de placement de pôles

L'élaboration de la commande RST avec l'intégrateur assure une réponse indicielle sans erreur en régime permanent.

La compensation des zéros du système à régler est parfois nécessaire afin d'obtenir un arrangement des pôles et zéros sur l'axe réel évitant tout dépassement.

Chapitre II

*Réalisation de l'interface
graphique d'aide au
dimensionnement des
polynômes $\mathcal{R}$, S et T*

II.1 Introduction

MATLAB est un progiciel qui permet de façon interactive et graphique de modéliser et de simuler des systèmes. MATLAB s'impose dans les mondes universitaire et industriel comme un outil puissant de simulation et de visualisation de problèmes numériques. Dans le domaine industriel, il est utilisé pour la résolution et la simulation de problèmes pratiques d'ingénierie et de prototypage.

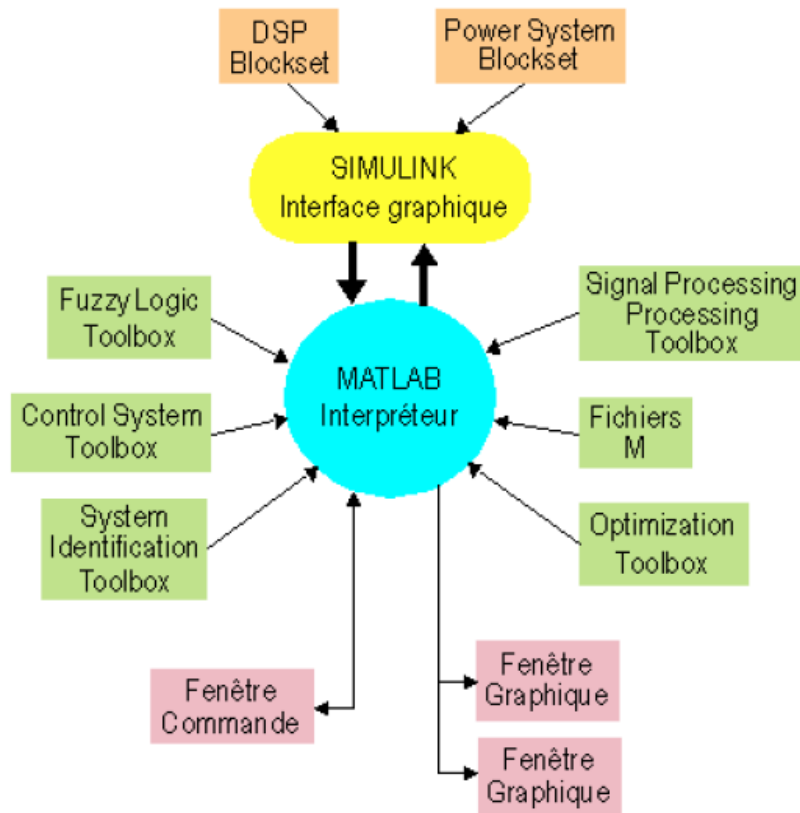


Figure II.1 Environnement de Matlab. [3]

Fenêtre Commande: Dans cette fenêtre, l'utilisateur donne les instructions et MATLAB retourne les résultats.

Fenêtres Graphique: MATLAB trace les graphiques dans ces fenêtres.

Fichiers M: Ce sont des programmes en langage MATLAB (écrits par l'utilisateur).

Toolboxes: Ce sont des collections de fichiers M développés pour des domaines d'application spécifiques (Signal Processing Toolbox, System Identification Toolbox, Control System Toolbox, u-Synthesis and Analysis Toolbox, Robust Control Toolbox,

Optimization Toolbox, Neural Network Toolbox, Spline Toolbox, Chemometrics Toolbox, Fuzzy Logic Toolbox, etc.)

Simulink: C'est l'extension graphique de MATLAB permettant de travailler avec des diagrammes en blocs.

Blocksets: Ce sont des collections de blocs Simulink développés pour des domaines d'application spécifiques (DSP Blockset, Power System Blockset, etc.).

En effet Matlab possède Le GUIDE (Graphical User Interface Development Environnement), c'est un constructeur d'interface graphique qui regroupe tous les outils dont le programmeur a besoin pour créer une interface graphique . Il s'ouvre, soit en cliquant sur l'icône appropriée, soit en tapant guide dans le Commande Window de MATLAB. [8]

II.2 Définition d'une interface graphique

Une interface graphique permet de contrôler une application interactivement avec la souris et le clavier. Elle comprend des menus, des boutons, des "ascenseurs (slider)", des cases à cocher, des listes de choix, des zones de texte...etc.

Les interfaces graphiques sous MATLAB sont constituées d'objets graphiques répartis selon une hiérarchie parent-enfant. Chacun de ces objets possède de nombreuses propriétés [8], [9].

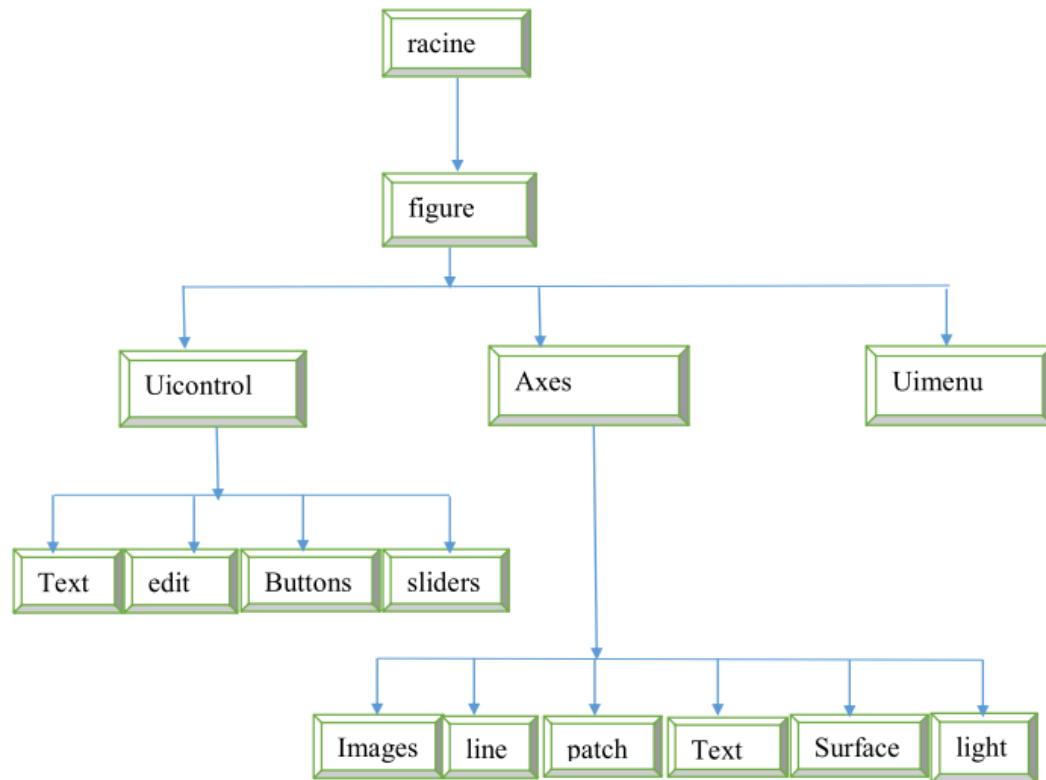


Figure II.2 la hiérarchie parent-enfants des objets

II.3 L'interface de Guide Matlab :

Guide Matlab est doté d'une interface facile à utiliser composé d'un menu, d'une barre d'outils, d'une palette d'objets et d'un espace dédié au placement des objets dont on est besoin.

La (figure II.3) présente l'interface de GUIDE Matlab et ses différents éléments de bases :

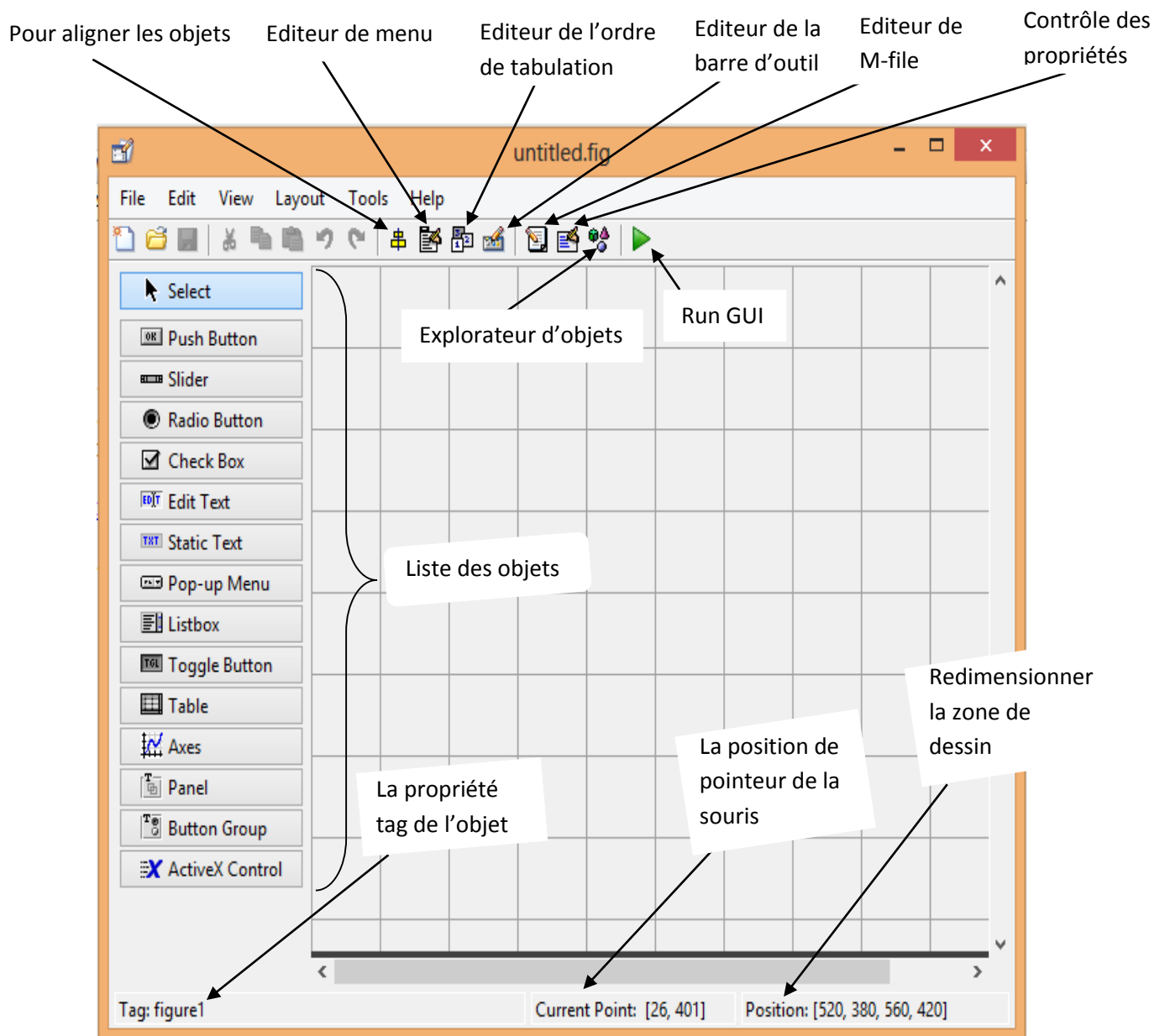


Figure II.3 fenêtre principal du guide

II.4 Les notions principales d'interface graphique

II.4.1 Les identifiants

A la création d'un objet, MA T L A B lui attribue automatiquement un identifiant (ou handle), sous la forme d'une valeur numérique unique qui peut être stockée dans une variable. Ceci permet de retrouver à tout moment un objet graphique au cours du

fonctionnement d'une interface. Cet identifiant existe tant que l'objet existe. Dès que l'objet est détruit, cet identifiant disparaît. [8]

Quelque identifiants particuliers :

- **gca** : récupère l'identifiant de l'objet Axes courant.
- **gcbf** : récupère l'identifiant de l'objet Figure où se trouve l'objet graphique dont l'action est en cours.
- **gcbo** : récupère l'identifiant de l'objet graphique dont l'action est en cours.
- **gcf** : récupère l'identifiant de l'objet Figure courant.
- **gco** : récupère l'identifiant de l'objet graphique courant.

II.4.2 Les propriétés des objets graphiques

Tous les objets graphiques possèdent des propriétés. Il est possible de configurer ces Propriétés au moment de la création de l'objet en spécifiant le nom et la valeur de la propriété. Les principales propriétés sont : le titre, la couleur, la dimension dans l'écran, la position..., on peut gérer ces propriétés par les fonctions GET et SET, GET Pour obtenir une propriété d'un objet et SET pour modifier les propriétés (modifiables) d'un objet On utilise les syntaxes suivantes [11] :

- Get (handles.nom de l'objet. 'Le paramètre').
- Set (handles.nom de l'objet. Le paramètre', 'valeur').

Tout objet graphique créé pourra être supprimé par :

- delete (handle_objet)

II.5 Principaux objets graphiques

II.5.1 Bouton poussoir

Un bouton poussoir se crée par :

```
bp1= uicontrol( objet , 'style' , 'push' , 'position' , [10 100 60 30 ] , 'String' , 'Début', ...  
'callback', 'plot (T, X)' )
```

Lorsqu'on clique sur le bouton poussoir, il provoque l'exécution de la fonction indiquée dans le 'callback'.

Un bouton-poussoir s'inactive par la commande :

```
Set (bp1, 'enable' , 'off ')
```

Par cette commande, on peut rendre inactif certaines commandes.

II.5.2 Menus

Généralement, les menus de la fenêtre d'application ne sont pas des menus standards mais des menus spécifiques. Un menu est un titre complété par une liste de sous-menu. Les actions (callbacks) sont généralement lancées à partir des sous-menus. L'ajout de menus spécifique se fait par :

```
menu1 = uimenu (fig1, 'label' , ' Statist');
```

Un sous-menu est un élément du menu principal, donc de l'entité père. Il est donc déclaré carmenu du menu principal.

II.5.3 Ascenseur ou slider

L'ascenseur a pour objectif de fixer la valeur d'un paramètre entre deux bornes fixées. La valeur prise par la variable est représentée par la position du curseur.

```
slid1=uicontrol(objet,'style','slider','position',[100,50,150,20] , 'Min' , -1 , 'Max' , 1,...  
'callback' , 'val_variable = get(slid1 , "value" )');
```

Les textes (variable affectée, valeurs..) ne sont pas définis par le slider. Il faut le compléter par des éléments textes convenablement placés et paramétrés, leur valeur est à modifier par le callback du slider.

II.5.4 Texte Editable

Permet à l'utilisateur de saisir une valeur. C'est une fonction importante.

```
Text1 = uicontrol( objet , 'style' , ' edit' , 'position', [100,50,150,20] , 'Max' , 1,...  
'string' , '0');
```

Généralement, Il faut associer un texte fixe pour préciser le rôle de la fenêtre de saisie à l'utilisateur.

II.5.5 Liste de choix

La liste de choix ou pop-up menu permet de sélectionner une valeur parmi une liste. Généralement, cette valeur est un texte. La valeur retournée lors du choix (paramètre 'Value') est le numéro de ligne du choix.

```
choix1 = uicontrol((gcf, 'Style', 'popup', 'String', 'Pierre|Paul|Jacques', 'Position', ...  
[10 10 100 80]));
```

La liste des textes est modifiable après la création de la fenêtre de choix, en modifiant la propriété 'String'.

II.6 L'outil SIMULINK

Ce logiciel est un programme de simulation destiné à traiter les systèmes dynamiques.

Comme extension de MATLAB, SIMULINK comprend plusieurs caractéristiques spécifiques aux systèmes dynamiques. Sur SIMULINK toutes les fonctions de MATLAB restent valables. Nous pouvons considérer SIMULINK comme une version graphique de ce logiciel. Ainsi, avec SIMULINK, pour intégrer un modèle, il suffit de dessiner le schéma bloc correspondant au système et suivre l'évolution de la simulation directement sur des blocs appelés "Scope", "Graph",... etc

Tous les composants nécessaires à la réalisation du schéma bloc d'un système se trouvent dans la bibliothèque standard de SIMULINK. En fait cette bibliothèque est composée de sous-systèmes dans lesquels sont groupés les différents éléments (sommateur, fonction de transfert, intégrateur, limiteur,...). Des blocs réalisant de nouvelles fonctions peuvent être créés et rajoutés soit à la bibliothèque standard ou bien à une bibliothèque. [10]

II.6.1 Présentation

Pour démarrer SIMULINK, il suffit de taper la commande "simulink" sous environnement MATLAB ou cliquer sur l'icône suivante  dans la fenêtre de MATLAB.

Et cette fenêtre apparaît :

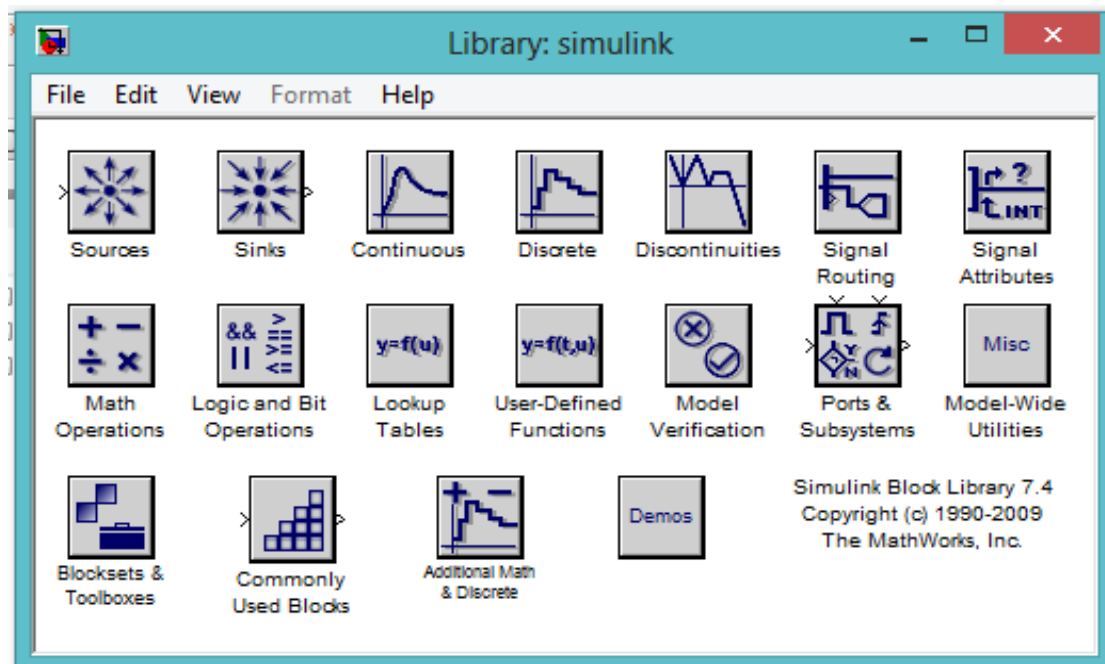


Figure II.4 Librairie de Simulink

Cette fenêtre contient des blocs comme (Sources, Sinks, Discrete, Linear, Nonlinear, Connections, Toolboxes...etc) utilisés lors des simulations. Pour avoir accès à ces blocs, il suffit de cliquer sur l'un d'entre eux.

II.6.2 Paramètres de simulation

Avant de mettre en marche une simulation avec SIMULINK, il faut établir l'environnement de simulation. Celui-ci se décompose en sept parties :

- Mode de simulation (Intégration Algorithmique),
- Instant de départ (Start Time),
- Instant final (Stop Time),
- Pas minimum de simulation (Min Step Size),

- Pas maximum de simulation (Max Step Size),
- La tolérance.
- Les variables retournées

II.6.3 Mode de simulation

Simuler un système sous SIMULINK, revient à résoudre une équation différentielle théorique. Or un ordinateur ne fonctionne pas en continu mais de façon échantillonné. En effet, toutes les opérations effectuées par ce dernier sont cadencées par une horloge interne à une certaine fréquence.

Pour effectuer la résolution d'une équation différentielle en mode échantillonné, deux modes sont proposés soit en :

- pas d'échantillonnage variable ("Variable-Step")
 - pas d'échantillonnage fixe ("Fixed-Step").
- ✓ Dans le premier mode huit méthodes de résolutions sont proposées :

La résolution "ode45 (Dormand-Prince)" est utilisée pour les systèmes linéaires composés de blocs tels que fonction de transfert, espace d'état, sommateur, gain, etc.... Cette résolution reste valable si le système contient un peu de blocs non linéaires et elle est adaptée aussi aux systèmes ayant des parties dynamiques rapides et lentes.

La résolution "adams" est utilisée pour des systèmes lisses non linéaires et quasi-stationnaires. (Paramètres ne variant pas beaucoup dans le temps).

- ✓ Dans le deuxième mode cinq méthodes de résolutions sont proposées :

La résolution "ode4(Runge-Kutta)" est basée sur Runge-Kutta du 4^{ème} ordre. Cette résolution est valable pour des systèmes non linéaires et/ou discontinus. Cette résolution n'est pas valable pour des systèmes ayant une dynamique rapide mais reste valable pour des systèmes ayant des parties continues et discrètes.

La résolution "euler" est utilisée pour vérifier les résultats seulement.

II.6.4 Instant de départ et d'arrivée

La simulation débute à partir de la valeur donnée dans "Start Time" et s'achève à celle du "Stop Time".

II.6.5 Pas initial et maximum de simulation

Les limites de la valeur du pas d'échantillonnage de la simulation sont données par le "Initial Step Size" et le "Maximum Step Size".

Le "Initial Step Size" est le pas d'intégration utilisé en début de simulation. Généralement, sa valeur est mise en mode automatique.

Le "Maximum Step Size" doit être choisi prudemment il faut généralement le laisser en mode automatique.

II.6.6 Tolérance relative et la Tolérance Absolue

La tolérance relative est l'erreur relative de l'intégration à chaque pas de calcul. Cette erreur est fixée à $1e-3$. La tolérance absolue est l'erreur absolue de l'intégration à chaque pas de calcul. Cette erreur est fixée à $1e-6$. [10]

II.8 l'interface graphique réalisée

Après définition de l'interface graphique et l'outil Simulink dans le début de ce chapitre, on passe à présenter l'interface réalisée qui gère des modèles Simulink créés dans le but de dimensionner le régulateur numérique RST pour un système choisis et à l'aide de quelques méthodes de la synthèse

II.8.1 Cahier des charges

-  L'interface doit contenir un moyen de tracer une réponse ou une commande à une fonction de transfert de notre choix.
-  Avoir la possibilité d'identifier la fonction de transfert du système.
-  Avoir un choix sur la méthode de dimensionnement du régulateur RST à utilisées :

- ✓ Le choix d'ordre du système.
 - ✓ Avec présence ou absence d'un intégrateur.
 - ✓ Avec ou sans simplification des zéros du système à régler.
-
- + Avoir la possibilité de choisir le nombre de pôles (1 ou 2)
 - + Après avoir choisis la méthode, et à l'aide d'un slider en affecte une valeur pour le pôle puis en calcule les paramètres du régulateur RST.
 - + Introduire les paramètres de simulation :
 - ✓ La période d'échantillonnage.
 - ✓ Valeur de la perturbation.
 - ✓ Le temps d'injection de la perturbation.
 - ✓ Le temps de départ de la simulation.
 - ✓ Le temps d'arrêt de la simulation.
 - ✓ Introduire la valeur de la consigne.
 - + Puis en charge tous les paramètres via le model Simulink choisit
 - + Et on trace le signal désiré, soit la réponse ou bien la commande du système bouclé sur les axes de l'interface.
 - + Avoir la possibilité de changer les paramètres de simulation et de faire varier les pôles à l'aide d'un slider.
 - + Avoir la possibilité de visualiser les signaux (référence, commande, réponse du système) sur la même figure
 - + Avoir la possibilité d'enregistrer toute les valeurs que se trouve sur l'interface.
 - + Un bouton pour quitter et un autre pour effacer le tracer en cours.
 - + Essayer de faire le maximum possible de méthodes pour avoir une large gamme de choix.
 - + Avoir la possibilité de visualiser le model Simulink sous forme d'image sur l'interface, correspondant à la méthode choisi.
 - + L'affichage de message d'erreur.
 - + Avoir une interface facile à utiliser et moins encombrante

II.7.2 présentation de l'interface graphique

La figure suivante présente l'interface réalisée :

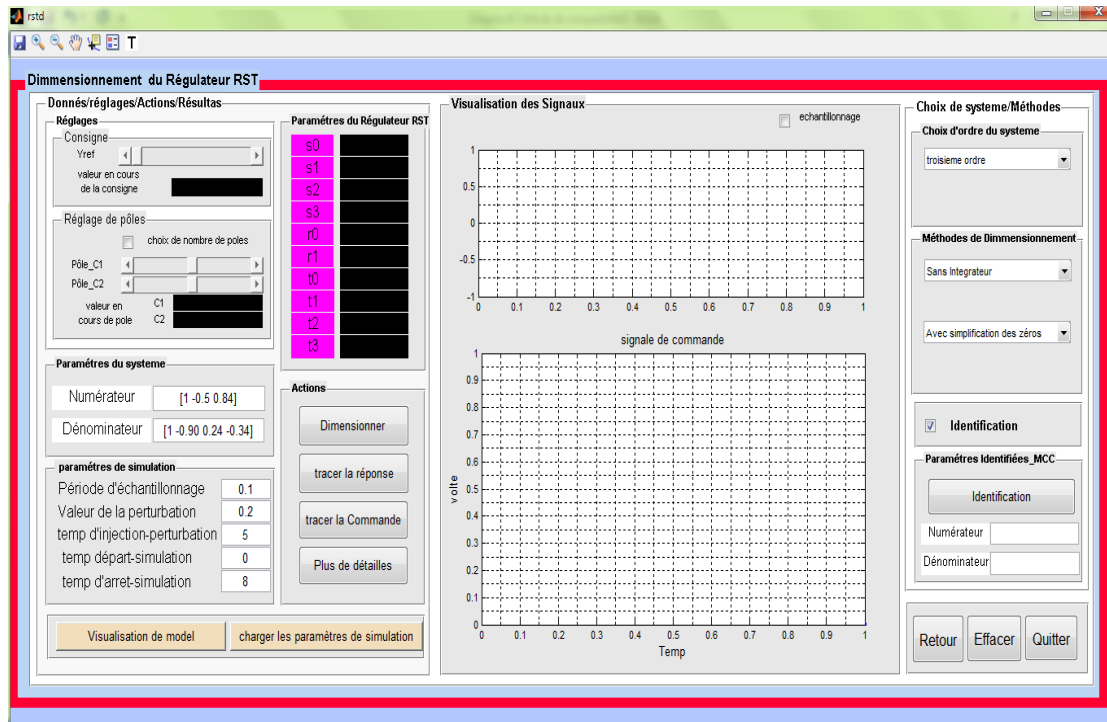


Figure II.5 présentation de l'interface

L'interface est munie de quatre panneaux principaux qui contiennent des objets, barre d'outils :

- Panneau principale 1 : Données/Réglages/Actions/Résultats.
- Panneau principale 2 : Visualisation des signaux.
- Panneau principale 3 : Choix de système/Méthodes.
- Panneau principale 4 : Retour/Effacer/Quitter.
- Barre d'outils.

A. 1^{er} panneau

Le premier panneau est situé à gauche de l'interface qui contient lui aussi six sous panneaux :

Donnés/réglages/Actions/Résultats

Réglages

Consigne
Yref
valeur en cours de la consigne

Réglage de pôles
☐ choix de nombre de poles
Pôle_C1
Pôle_C2
valeur en cours de pole C1
C2

Paramètres du Régulateur RST

s0	
s1	
s2	
s3	
r0	
r1	
t0	
t1	
t2	
t3	

Paramètres de systeme

Numérateur [1 -0.5 0.84]
Dénominateur [1 -0.90 0.24 -0.34]

paramètres de simulation

Période d'échantillonnage 0.1
Valeur de la perturbation 0.2
temp d'injection-perturbation 5
temp départ-simulation 0
temp d'arrêt-simulation 8

Actions

Dimensionner
tracer la réponse
tracer la Commande
Plus de détails

Visualisation de model charger les paramètres de simulation

Figure II.6 présentation du 1^{er} panneau

A.1 premier sous panneaux

Réglages

Consigne
Yref
valeur en cours de la consigne

Réglage de pôles
☐ choix de nombre de poles
Pôle_C1
Pôle_C2
valeur en cours de pole C1
C2

Ce panneau contient un slider et un static-text pour le réglage de la consigne et l'affichage de la valeur en cours

Ce panneau contient deux slider au choix pour le réglage de pole et l'affichage de la valeur en cours

Figure II.7 présentation de sous panneau 1

A.2 Deuxième sous panneau

Paramètres du système

Numérateur	[1 -0.5 0.84]
Dénominateur	[1 -0.90 0.24 -0.34]

Ce panneau contient deux éditeurs texte, pour introduire la fonction de transfert du système souhaiter

Figure II.8 présentation de sous panneau 2

A.3 Troisième sous panneau

paramètres de simulation

Période d'échantillonnage	0.1
Valeur de la perturbation	0.2
temp d'injection-perturbation	5
temp départ-simulation	0
temp d'arret-simulation	8

Il contient cinq éditeurs, qui nous permettent de choisir les paramètres de simulation

Figure II.9 présentation de sous panneau 3

A.4 Quatrième sous panneau

Paramètres du Régulateur RST

s0	
s1	
s2	
s3	
r0	
r1	
t0	
t1	
t2	
t3	

Ce panneau contient dix static-text, pour afficher les paramètres du régulateur RST calculé

Figure II.10 présentation de sous panneau 4

A.5 Cinquième sous panneau

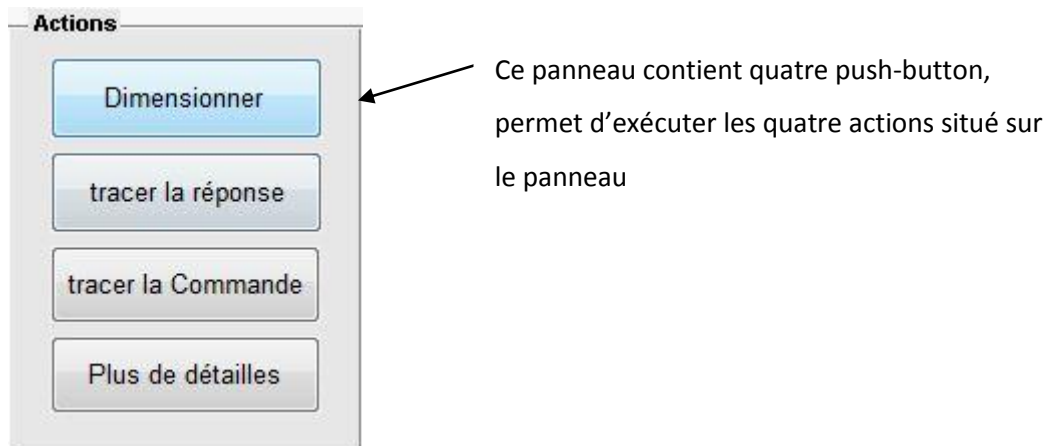


Figure II.11 présentation de sous panneau 5

- Un bouton poussoir (Dimensionner) : pour calculer les paramètres du régulateur RST pour les différents cas de la synthèse
- Un bouton poussoir (tracer la Réponse) : pour simuler puis tracer le signal de sortie du système bouclé pour les différents cas de la synthèse
- Un bouton poussoir (tracer la Commande) : pour simuler puis tracer le signal de commande du système bouclé pour les différents cas de la synthèse
- Un bouton poussoir (Plus de détails) : il fait appel à la figure pour tracer le signal de référence, de sortie et de commande du système bouclé sur la même figure

A.6 Sixième sous panneau

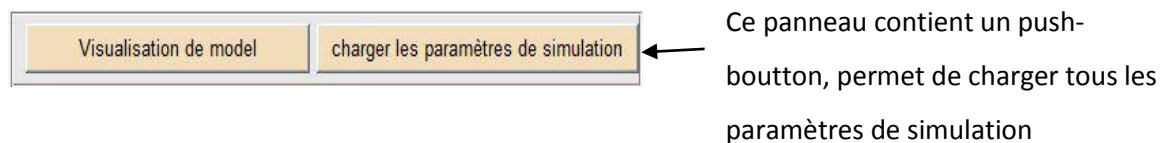


Figure II.12 présentation de sous panneau 6

B. 2^{ème} panneau

Le deuxième panneau est situé au milieu de l'interface

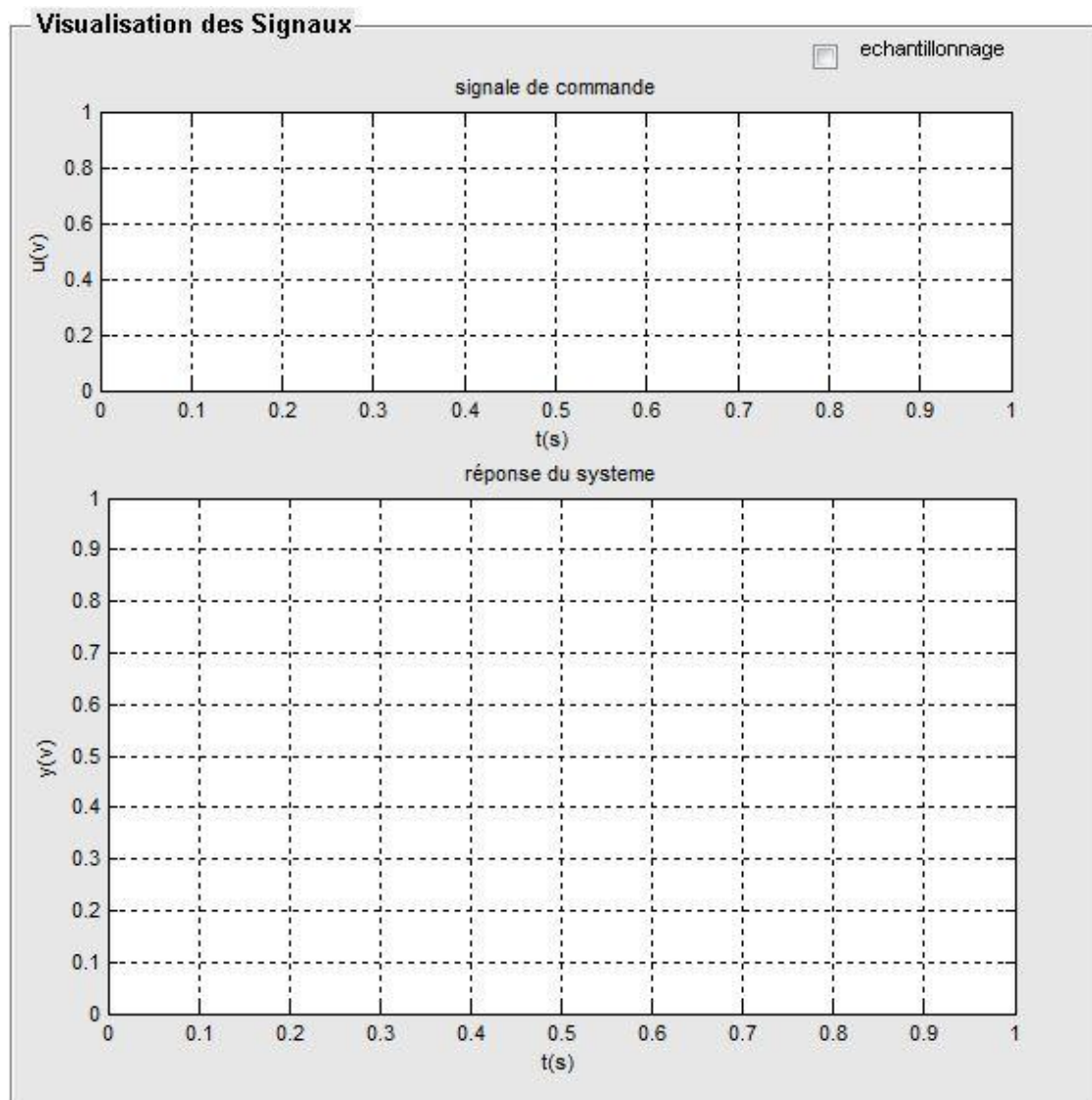


Figure II.13 présentation du 2^{ème} panneau

Ce panneau contient deux axes (axes1, axes2), permet de tracer le signal de commande et de sortie corrigée avec le régulateur RST.

Un Check-Box : permet de changer le mode de traçage normale vers un tracer échantillonné.

C. 3^{ème} panneau

Le troisième panneau est situé à droite de l'interface qui contient lui aussi trois sous panneaux :

The screenshot shows a vertical panel titled "Choix de systeme/Méthodes". It contains four sub-sections:

- Choix d'ordre de systeme**: A dropdown menu currently showing "troisieme ordre".
- Méthodes de Dimmensionnement**: Two dropdown menus. The first shows "Sans Integrateur" and the second shows "Sans simplification des zéros".
- Identification**: A checkbox that is checked.
- Paramètres Identifiées_MCC**: A section containing a button labeled "Identification", and two input fields labeled "Numérateur" and "Dénominateur".

Figure II.14 présentation du 3^{ème} panneau

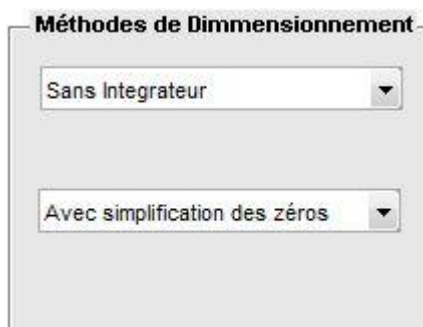
C.1 premier sous panneau

A close-up of the "Choix d'ordre du systeme" dropdown menu, showing the text "troisieme ordre" and a downward arrow.

← Ce panneau contient un popup-menu (liste des choix), permet de sélectionner l'ordre du système (du 1^{er} ordre jusqu'au 3^{ème} ordre)

Figure II.15 présentation de sous panneau 1

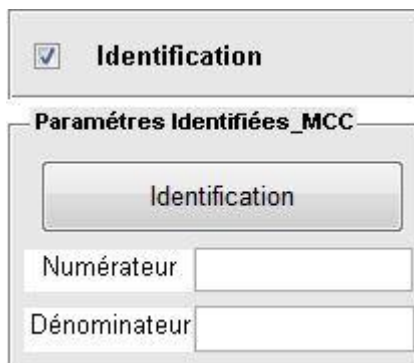
C.2 Deuxième sous panneau



Ce panneau contient deux popup-menu (liste des choix), permet de sélectionner la méthode pour le dimensionnement du régulateur RST en combinant entre le choix, avec ou sans simplifications des zéros du système à régler, avec ou sans intégrateur ; avec toutes les possibilités

Figure II.16 présentation de sous panneau 2

C.3 Troisième sous panneau



En coché pour basculer vers le mode identification

Ce panneau contient un push-bouton permet l'identification d'un système d'ordre un et l'afficher sur deux éditeurs un pour le numérateur et l'autre pour le dénominateur

Figure II.17 présentation de sous panneau 3

D. 4^{ème} panneau



Ce panneau contient trois push-boutton, Permet d'exécuter ces trois actions

Figure II.18 présentation du 4^{ème} panneau

Un bouton poussoir (Retour) : ce bouton permet de retourner aux valeurs par défaut de tous les objets de l'interface.

Un bouton poussoir (Effacer) : ce bouton permet d'effacer les axes

Un bouton poussoir (Quitter) : une boîte de dialogue apparaît quand on exécute cette action afin de décider de quitter l'interface ou bien de retourner.

E. Barre d'outils

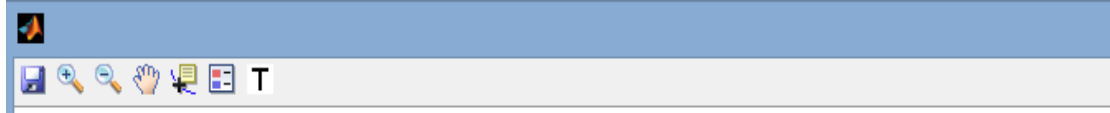


Figure II.19 La barre d'outils

- Bouton (zoom in) : pour agrandir le signal tracer sur l'un des axes placer dans l'interface.
- Bouton (zoom out) : pour faire revenir le zoom en arrière.
- Bouton (Grid on, Grid off) : pour maitre une grille si en le souhaite
- Bouton (enregistrer) : pour enregistrer les données dans un fichier d'extension (.mat).

II.7.3 Exemple d'application

Pour illustré l'efficacité de cette interface, en prend l'exemple de fonction de transfert :

a) Cas système avec intégrateur :

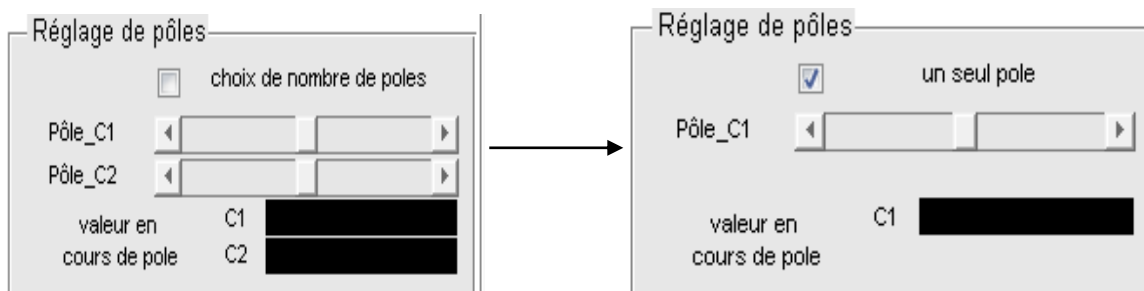
$$H(z) = \frac{z-0.5}{z^2-1.1*z+0.1} = \frac{1}{(z-1)} * \frac{z-0.5}{(z-0.1)} \quad (\text{II.1})$$

b) Cas système sans intégrateur :

$$H(z) = \frac{z-0.5}{z^2-1.1*z+0.24} \quad (\text{II.2})$$

1) Avec un seul pôle

Dans la partie qui suit en utilise uniquement un seul pôle.



a) Cas système avec intégrateur

L'utilisation de cette interface nous impose de suivre ces étapes :

- Choix d'ordre du système :
 - On a choisi le système d'ordre deux (deuxième ordre), (II.1)

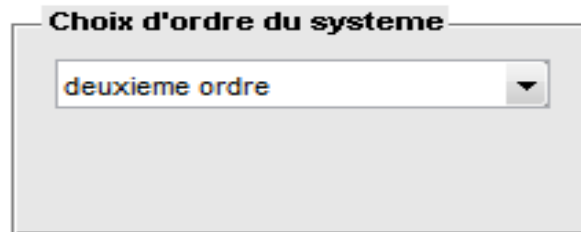


Figure II.20 L'ordre du système choisi

- Introduire les paramètres :
 - en tape les vecteurs
 - du numérateur [1 -0.5].
 - et dénominateur [1 -1.1 0.1].
 - par un slider on fixe la valeur de la consigne à '3'.
 - d'abord on va choisir par un check-box un seul pôle puis par un slider on fixe la valeur de pôle à '-0.463'.
 - En impose les paramètres de simulation
 - La période d'échantillonnage, '0.1'.
 - La valeur de la perturbation, '0.3'
 - Temps d'injection de la perturbation '5'
 - Temps de départ de la simulation '0'
 - Temps d'arrêt de la simulation '10'

Réglages

Consigne

Yref

valeur en cours de la consigne **3**

Réglage de pôles

☒ un seul pole

Pôle_C1

valeur en cours de pole C1 **-0.463076**

Paramètres du système

Numérateur

Dénominateur

paramètres de simulation

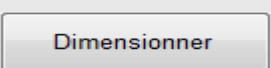
Période d'échantillonnage	0.1
Valeur de la perturbation	0.3
temp d'injection-perturbation	5
temp départ-simulation	0
temp d'arret-simulation	10

Figure II.21 Introduire les paramètres

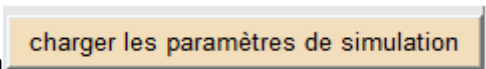
- Choix de la méthode
 - Sans intégrateur et sans simplification des zéros du système à régler.

Méthodes de Dimmensionnement

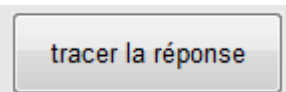
Figure II.22 choix de la méthode

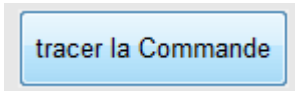
- on appuie sur le bouton  , les paramètres du régulateur RST calculés sont :

Paramètres du Régulateur RST	
s0	-0.109615
s1	1.18654
s2	0
s3	0
r0	-0.548077
r1	0
t0	0
t1	1.07692
t2	0
t3	0

- on appuie sur le bouton  , pour affecté tous les paramètres et les résultats vers le model Simulink qui correspond aux choix effectués.

- Pour visualiser les différents signaux on appuie sur le bouton :

-  , pour lancer la simulation et tracer la réponse du système.

- Puis sur  pour lancer la simulation et tracer le signal de commande.

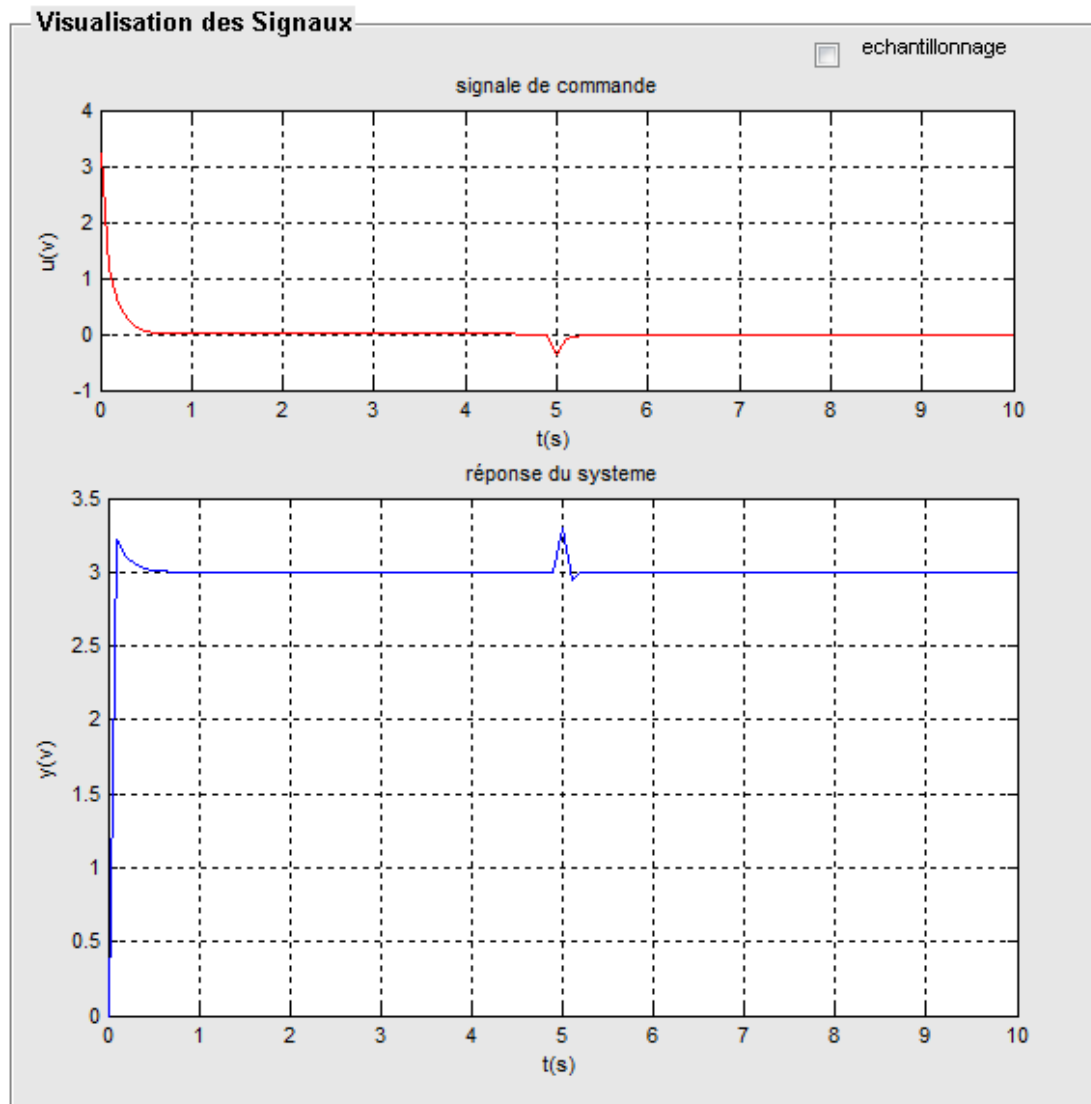


Figure II.23 les signaux, commande et réponse du système, sans simplification des zéros du système à régler et sans intégrateur, Axes

Plus de détails

Sur le bouton pour tracer, le signal de commande, le signal de référence et la réponse du système sur la même figure.

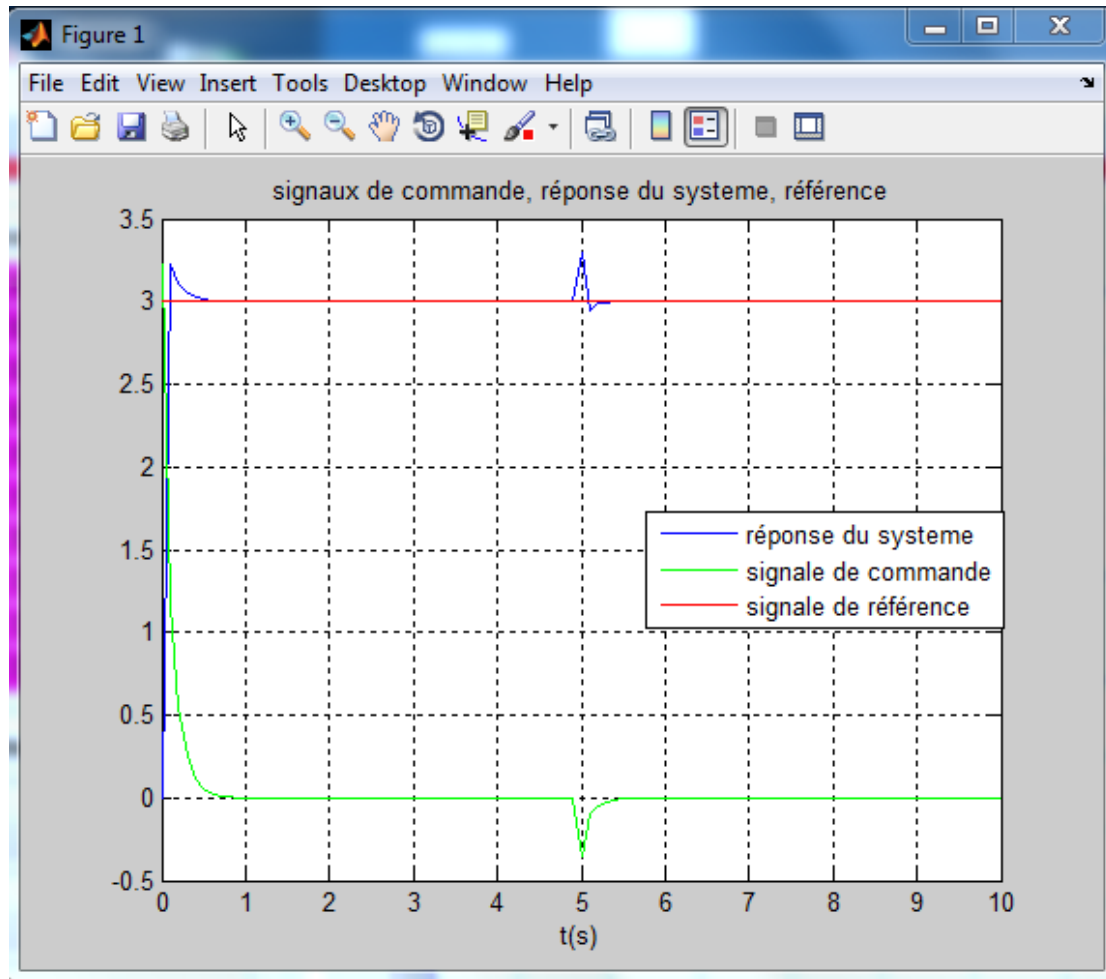


Figure II.24 les signaux de commande, réponse du système et la référence, sans simplification des zéros du système à régler et sans intégrateur, figure

Dans ce cas la réponse du système est rapide et suit la référence mais avec un dépassement, $D = 8\%$

En change le pôle $c_1 = -0.6$

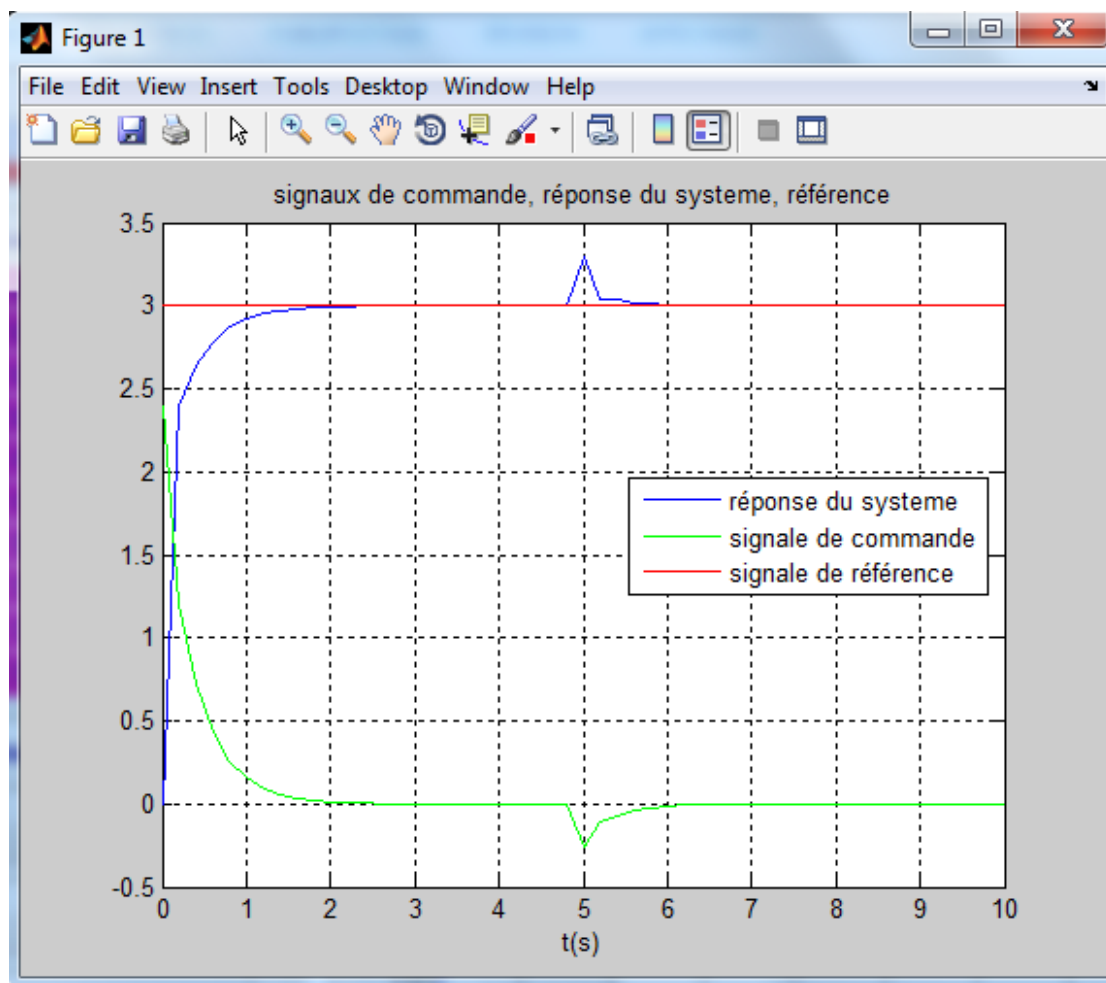


Figure II.25 les signaux de commande, réponse du système et la référence, sans simplification des zéros du système à régler et sans intégrateur, figure, pour $c_1 = -0.6$

Après changement de la valeur de pôle, la (figure II.25) montre que la réponse du système est sans dépassement mais avec une réponse un peu lente par rapport à celle de la (figure II.24)

Maintenant, pour les mêmes paramètres et avec simplifications des zéros du système à régler et sans intégrateur, on dimensionne le régulateur RST pour le même système (II.1), puis en visualise les différents signaux.

En prend : $c_1 = -0.463$

Les paramètres du régulateur RST calculés sont :

Paramètres du Régulateur RST	
s0	-0.1
s1	0.638462
s2	0
s3	0
r0	-0.5
r1	0
t0	0
t1	0.538462
t2	0
t3	0

La (Figure II.26) montre les signaux de commande et de réponse du système, avec simplification des zéros du système à régler et sans intégrateur :

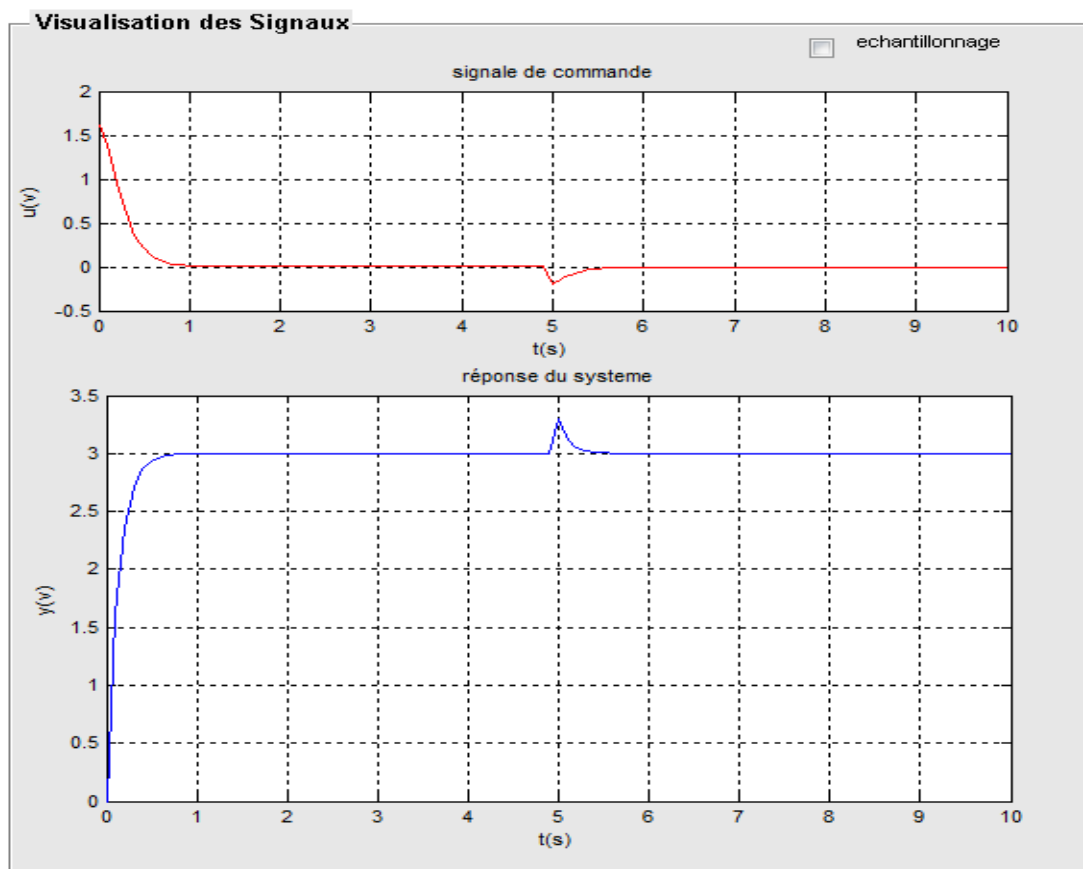


Figure II.26 les signaux de commande et de réponse du système, avec simplification des zéros du système à régler et sans intégrateur, Axes

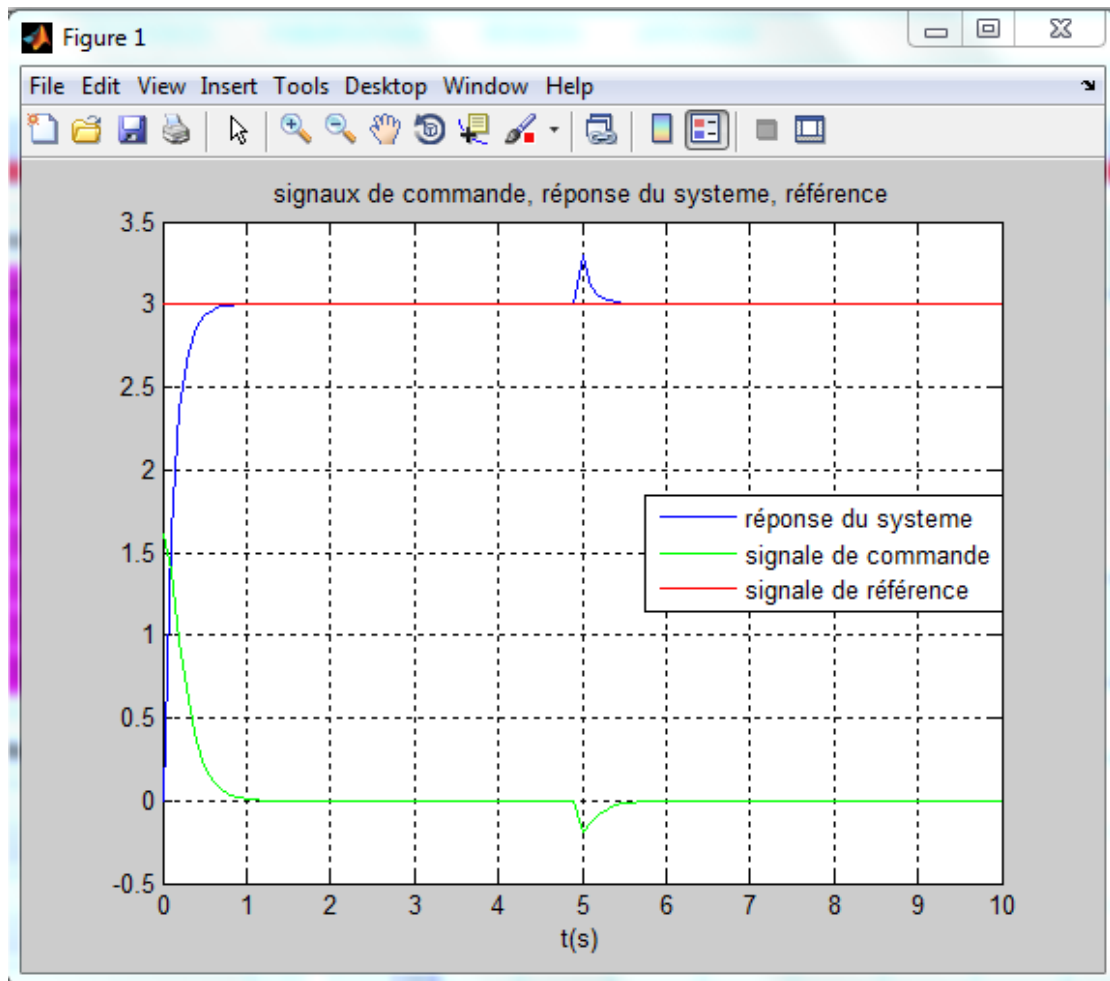


Figure II.27 les signaux de commande, réponse du système et la référence, avec simplification des zéros du système à régler et sans intégrateur, figure

La (figure II.27) montre que la réponse du système est sans dépassement

$D = 0\%$, Contrairement à celle de la (figure II.25) où il n'y a pas de simplification des zéros du système à régler la réponse du système possède un dépassement 'D' vaut 8%

La poursuite entre la référence et la réponse du système est vérifié.

b) Cas système sans intégrateur :

Maintenant en va insérer l'intégrateur pour le calcul des paramètres du régulateur RST et en prend le système d'ordre deux (II.2), les mêmes paramètres pris au pare avant pour la simulation et le pôle reste toujours $c1 = -0.463$:

Les résultats de dimensionnement du régulateur RST sont :

Paramètres du Régulateur RST	
s0	0.24
s1	-1.34
s2	1.63846
s3	0
r0	-0.5
r1	0
t0	0
t1	0
t2	0.538462
t3	0

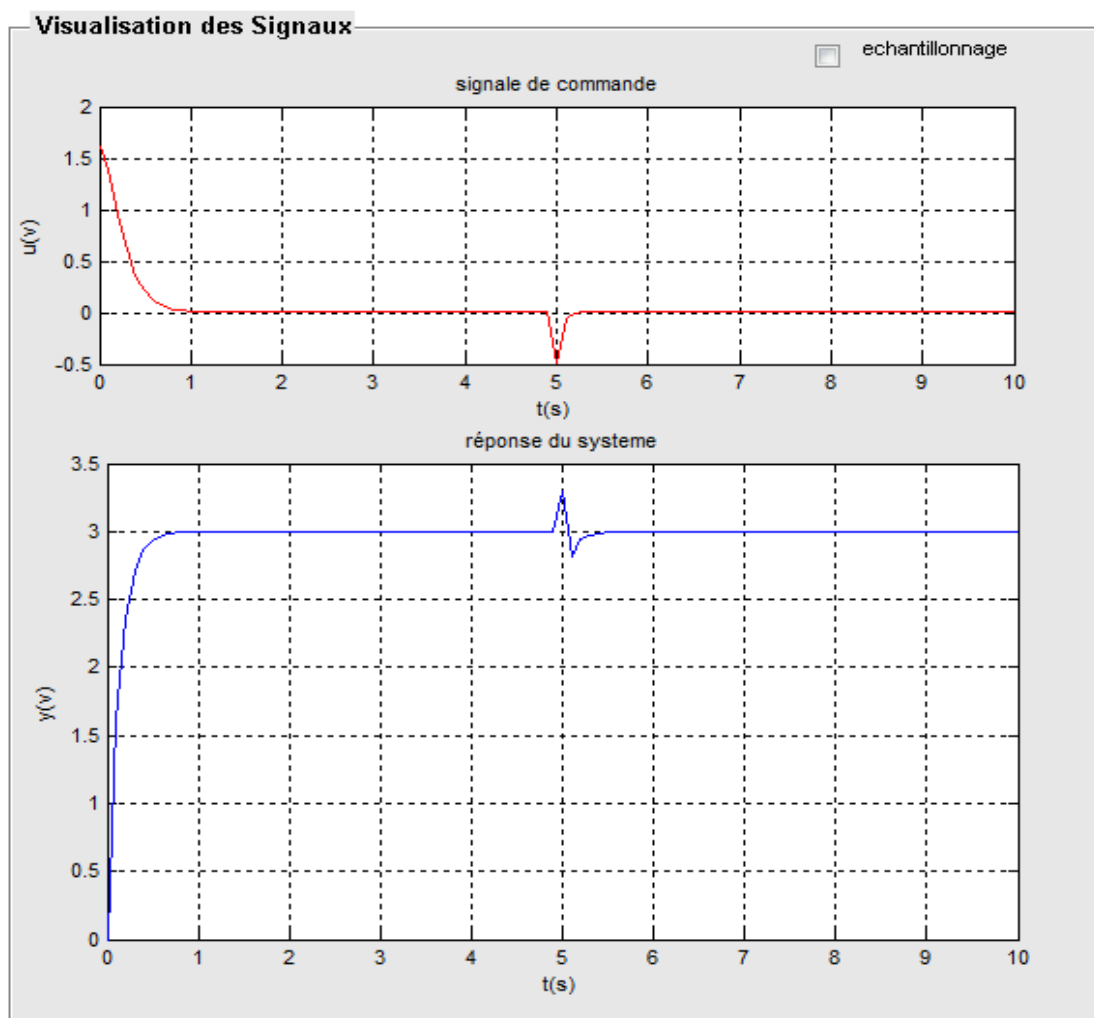


Figure II.28 les signaux de commande et de réponse du système, avec simplification des zéros du système à régler et avec intégrateur

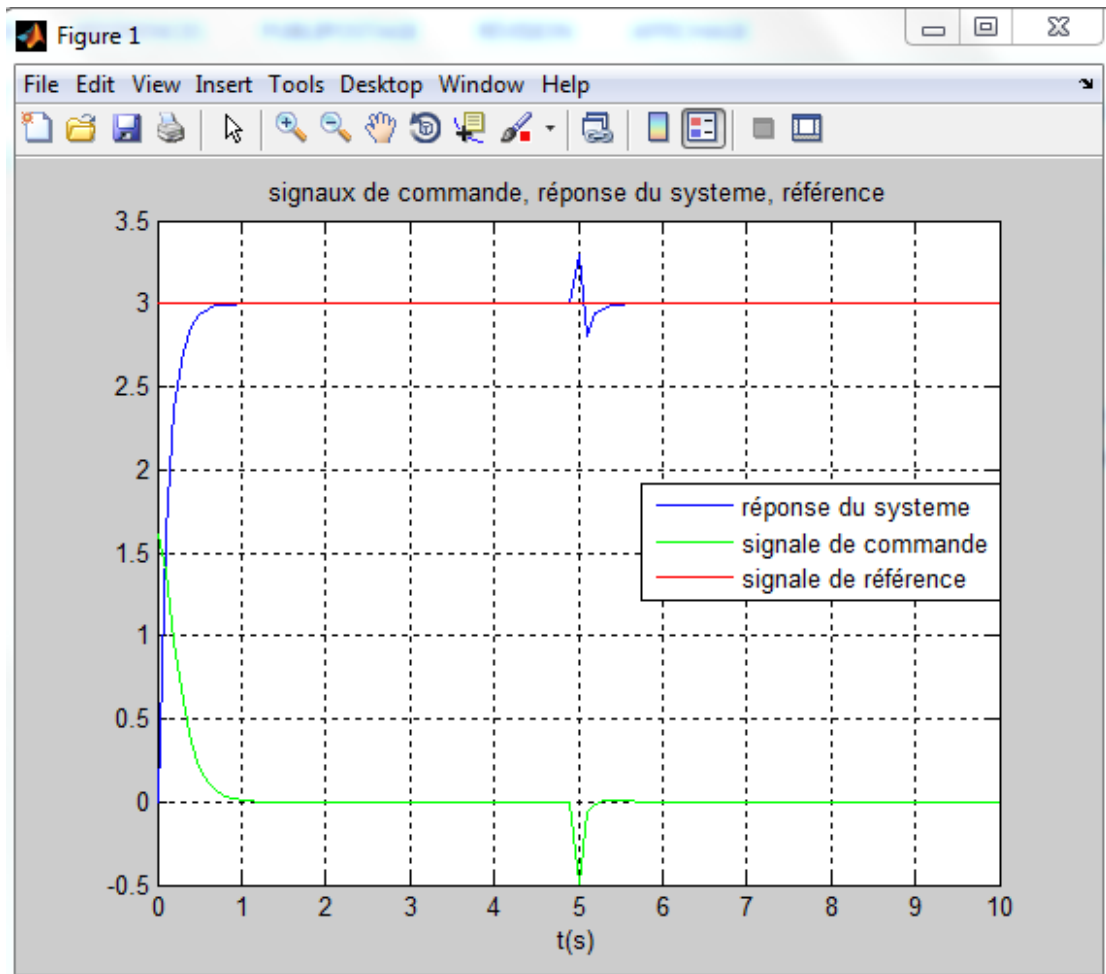


Figure II.29 les signaux de commande, réponse du système et la référence, avec simplification des zéros du système à régler et avec intégrateur

D'après la (figure II.29), on constate que la réponse est acceptable, et le signal de commande réagit aux changements de l'allure de signal de sortie afin de maintenir la poursuite et la rejection de la perturbation.

En effet, le dimensionnement du régulateur RST pour un système ne possède pas un intégrateur, nous laisse pas le choix de ne pas l'insérer pour le calcul des paramètres de ce dernier, afin de rejeter les perturbations qui influence sur la réponse du système.

2) Avec deux pôles

Cette fois on dimensionne le régulateur RST sans intégrateur et sans simplification des zéros du système à régler, en prend le système (II.1).

On impose $c_1=c_2= - 0.48$

Et pour faire une comparaison on dimensionne le régulateur RST avec un seul pôle $c_1 = -0.48$

Les paramètres du régulateur RST est :

Pour $c_1 = c_2 = -0.48$

Paramètres du Régulateur RST	
s0	-0.1002
s1	0.641
s2	0
s3	0
r0	-0.501
r1	0
t0	0
t1	0.5408
t2	0
t3	0

Pour $c_1 = -0.48$

Paramètres du Régulateur RST	
s0	-0.105
s1	1.145
s2	0
s3	0
r0	-0.525
r1	0
t0	0
t1	1.04
t2	0
t3	0

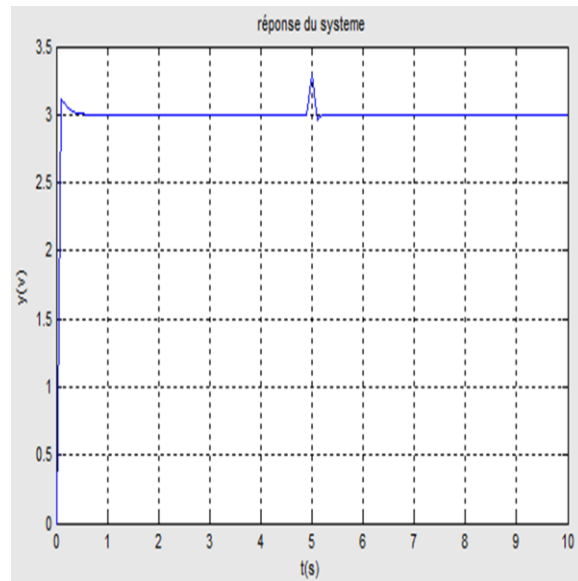
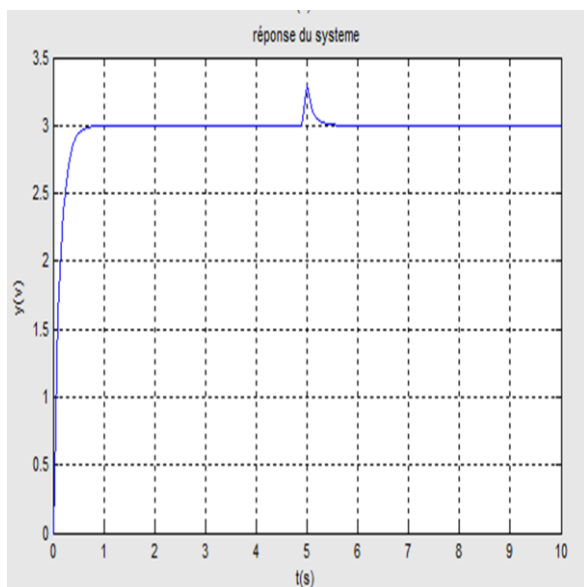


Figure II.30 la réponse du système pour un seul pôle et le pôle multiple

La (figure II.30) montre que le pole multiple $c_1 = c_2 = -0.48$ à permet d'éviter le dépassement mais la réponse du système est un peu lente par rapport à la réponse de système avec un seul pôle $c_1 = -0.48$

II.8 Conclusion

Dans ce chapitre on a commencé par présenter GUIDE Matlab d'une manière générale pour avoir une idée de son fonctionnement et de se familiariser avec lui, ensuite et à base de ces notions on a proposé une solution à notre cahier des charges on réalisant une interface graphique qui répond aux exigences demandé ce qui nous a permis de se familiariser avec Matlab GUIDE et avoir une idée bien claire sur ces fonctions.

L'effet principal d'un intégrateur est de rejeter des perturbations, l'intégrateur dans un régulateur RST n'élimine pas forcément les erreurs permanentes d'asservissement.

L'introduction d'un terme intégral dans un système affecte généralement la réponse dynamique et permanente du montage en asservissement, même après retouche des paramètres du régulateur sans intégrateur.

Parfois, il est nécessaire de simplifier les zéros du système à régler afin d'obtenir un arrangement nécessaire des pôles et zéros sur l'axe réel évitant le dépassement.

Chapitre III

Application de la commande d'un moteur à courant continu

III-1 Introduction

Ce chapitre est consacré pour la partie pratique en exploitons de nombreux points essentiel sur la description des méthodes RST et à l'aide d'une interface graphique qui va permettre de dimensionnée ce régulateur.

Matlab contient un 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 control des systèmes. Le prototypage passe alors par trois étapes :

- Construire le système de commande en utilisant les blocs de Simulink
- Simuler le système pour voir les résultats dans différents scénarios.
- Exécuter le modèle en temps réel à travers la carte DS1104.

En effet, en commence par l'identification du la fonction de transfert du moteur à courant continu du laboratoire en utilisons la carte DS1104

Après identification en passe au dimensionnement du régulateur RST pour les différents cas indiquer dans le chapitre I et à l'aide de l'interface réaliser afin de visualiser les signaux de commande, de référence et de la réponse du système en boucle fermer, et en exploitons le logiciel Control Desk de la carte DS1104 qui va nous permettre de suivies nos essais en temps réel (développement et implémentation des solutions pour le contrôle des systèmes en temps réel.)

L'interprétation des résultats afin d'évaluer les performances du système en fonction de Régulateur dimensionné introduit dans ce procédé.

III.2 Carte DSPACE RTI1104

La figure suivante représente la carte DS1104



Figure III.1 Carte DS1104

III.2.1 Création d'un model en temps réel dans Simulink/DSPACE RTI1104

Un des meilleurs atouts de DS1104 est la facilité avec laquelle on peut construire des applications en temps réel. Le temps entre la conception des instructions numériques pour le DSP et l'exécution effective de l'application dépend seulement de la façon dont l'ordinateur peut compiler rapidement le code initial. Fondamentalement une application en temps réel peut être créée par deux méthodes :

- en utilisant Matlab/Simulink pour établir le modèle et produire automatiquement le code en C et le télécharger dans la mémoire du DSP,
- en programmant le codage en C manuellement et en compilant le modèle dans le code du DSP

On en convient volontiers que la première méthode est la plus rapide, surtout pour les systèmes qui ne présentent pas d'exigences particulières.

Pour avoir la librairie RTI1104 il suffit de taper dans l'espace de travail Matlab la commande 'rit'.

Une fois que l'on a achevé le modèle de Simulink qu'on veut exécuter en temps réel, la seule commande exigée est RTW Build sous le menu Tools dans Simulink. Une fois que la commande est exécutée, le logiciel de dSPACE crée un fichier objet (*.obj), le télécharge sur la carte DS1104 et commence automatiquement l'exécution matérielle. Cependant, il y a

quelques arrangements importants qu'on doit faire avant de "transporter" le modèle dans le monde du temps réel. Le modèle développé pour la simulation doit être connecté à des dispositifs externes (générateurs de signaux et oscilloscopes). Puisque ces dispositifs sont physiquement des générateurs / récepteurs de signaux partant ou provenant du DSP, on doit faire passer ces signaux par l'intermédiaire des canaux d'entrée-sortie analogiques, situés sur le boîtier de connexion de la carte.

Les signaux entrant ou sortant analogique sont pris avec la conversion suivante 10V sur les bornes ADC ou DAC du boîtier de connexion vaut 1V dans l'espace Simulink. Le canal d'entrée analogique ADC prend le signal avec un rapport de 1/10. Ceci signifie que 10V à l'entrée sera lu comme 1V dans le modèle. Il doit être multiplié par 10 dans le modèle pour retrouver l'amplitude réelle. A la sortie, le signal doit être divisé par 10 avant d'être connecté au canal. Ainsi, deux blocs de gain, de la bibliothèque de maths, seront exigés correctement pour lire et écrire les valeurs provenant et partant des voies analogiques. La communication avec les canaux d'entrée-sortie est assurée par l'intermédiaire de deux blocs de dSPACE qui se trouvent dans la bibliothèque de dSPACE RTI1104.

III.2.2 configuration des paramètres de simulation du modèle

En doit reconfigurés Les paramètres de simulation du modèle avant l'exécution en temps réel comme suivant :

- Mettre 'inf.' dans Stop Time (le système sera exécuté continuellement).
- Dans solver prendre Fixed step et le pas d'échantillonnage convenable.
- Dans Advanced/Simulations paramètres mettre 'off' dans Block Réduction.
- Dans la fenêtre de Real Time Workshop, vérifier que en face de : System Target file on a bien : rti 1104.tlc, sinon cliquer sur l'icône Browser pour le choisir.
- Ensuite, dans le menu Tools, choisir Real - Time Workshop, Build Model, ou, au clavier, appuyer sur CTRL+B.

III-3 Identification de la fonction de transfert du moteur à courant continu

Pour avoir la fonction de transfert de notre système (moteur à courant continu), on applique à l'entrée du système une tension à amplitude bien déterminée, puis on prélève les deux signaux d'attaque et de réponse afin de pouvoir identifier la fonction de transfert de notre système.

Pour cela en a le choix entre deux méthodes :

- ✓ La première est la méthode classique qui consiste à tracer la tangente de la réponse en boucle ouverte et faire la projection du point d'intersection entre la tangente et la valeur d'amplitude de la réponse du système en régime permanent pour déterminer la constante de temps τ_r , pour le gain il suffit de faire le rapport entre la valeur finale du signal et la valeur de l'échelon d'attaque.
- ✓ La seconde est d'utiliser la méthode de moindre carrée récurrentes

On a utilisé la deuxième méthode pour l'identification du système (méthode moindres carrés récurrentes) à l'aide de la carte Dspace1104 et son logiciel Desk.

Et après avoir appliqué à l'entrée du notre système un échelon de 1V

On trouve :

$$H(z) = \frac{0.11}{z-0.6} \quad (\text{III.1})$$

III.4 Application

Après avoir identifié la fonction de transfert de notre système (III.1), à l'aide de l'interface graphique on va dimensionner le Régulateur RST pour se système.

Suite que le système est d'ordre un et ne possède pas un intégrateur, il suffit d'utiliser un seule pôle et l'insertion d'un intégrateur pour le calcul des paramètres du régulateur RST

On doit varier le pôle jusqu'à ce qu'on obtienne une bonne réponse du système avec un signal de commande acceptable.

-  En lance l'application.
-  Choisir, identification.
-  Choisir, système premier ordre.
-  Choisir, avec intégrateur.
-  Choisir, un seule pole.
-  Les paramètres de la simulation.
-  La consigne (référence).

- ✚ En click sur le bouton 'identification'.
- ✚ En click sur le bouton 'Dimensionner'.
- ✚ En click sur le bouton 'charger les paramètres de simulation'.
- ✚ Puis sur tracer la réponse et la commande.
- ✚ Comme en va visualiser tous les signaux (réponse du système, commande et la référence) sur une figure, en click sur le bouton 'plus de détails'.

Il est possible de faire l'identification du système avec l'interface graphique

Et pour cela il suffit d'actionnée sur le bouton identification puis l'affichage immédiat des paramètres du système identifiée avec la méthode des moindres carrées récurrentes :

$$H(z) = \frac{0.11013}{z - 0.59995} \quad (\text{III.2})$$

On voit que le système (II.1) d'un moteur à courant continu identifiée à l'aide du logiciel Control Desk et le même à celui identifiée avec l'interface graphique avec la même méthode

Maintenant on dimensionne le régulateur RST adéquat pour le système (II.1), et de vu qu'il ne possède pas un intégrateur on va l'insérer pour le calcul des paramètres du régulateur pour un pôle $c1 = -0.4$ obtenu après plusieurs essais dans le but d'avoir une réponse rapide avec rejet de perturbation dans un temps très petit.

On obtient :

Paramètres du Régulateur RST	
s0	-5.45345
s1	10.9073
s2	0
s3	0
r0	0
r1	0
t0	0
t1	5.4539
t2	0
t3	0

La (figure III.2) montre Les signaux, de commande, référence et réponse du système corriger avec le régulateur RST en présence de la perturbation.

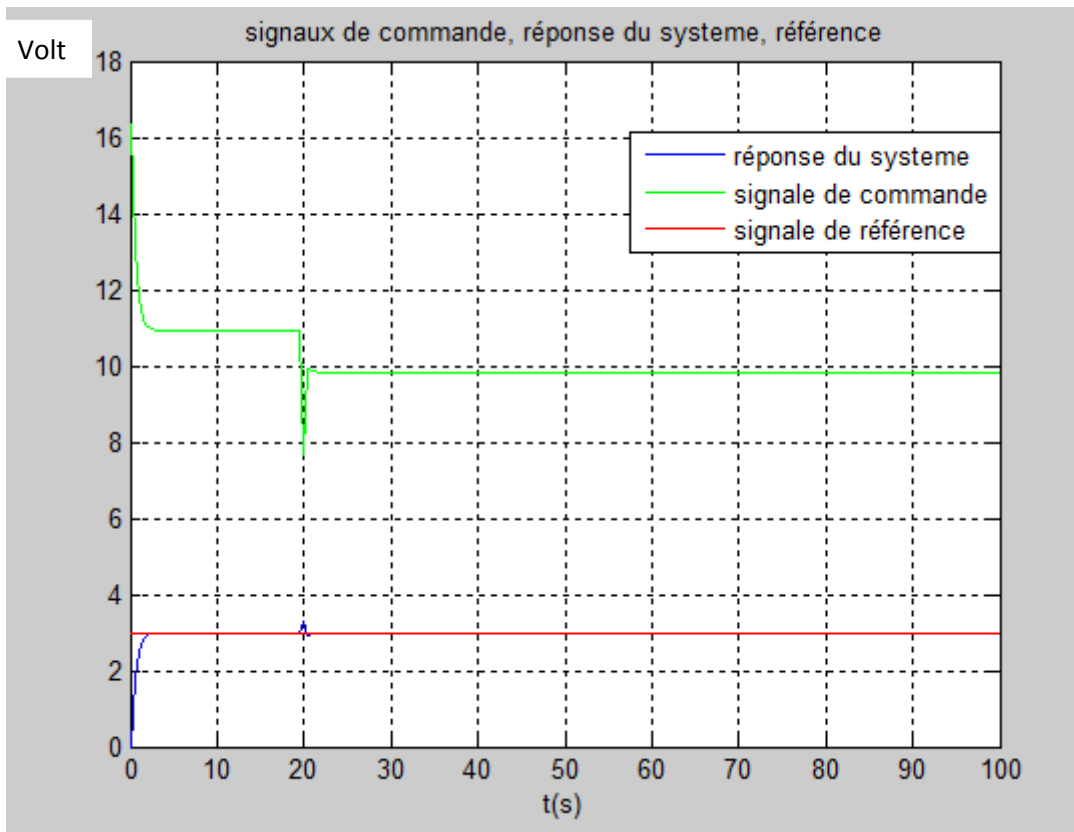


Figure III.2 signaux de commande, de référence, réponse de système pour système 1^{er} ordre corriger avec le régulateur RST avec intégrateur

III.5 Essai pratique

Après l'exécution du programme en temps réel, le plus important qui reste à avoir est l'interaction avec le système. On doit visualiser, modifier et analyser les variables. Pour cela, dSPACE a sa propre interface utilisateur graphique, appelée CONTROL DESK représenter dans la (figureIII.3).

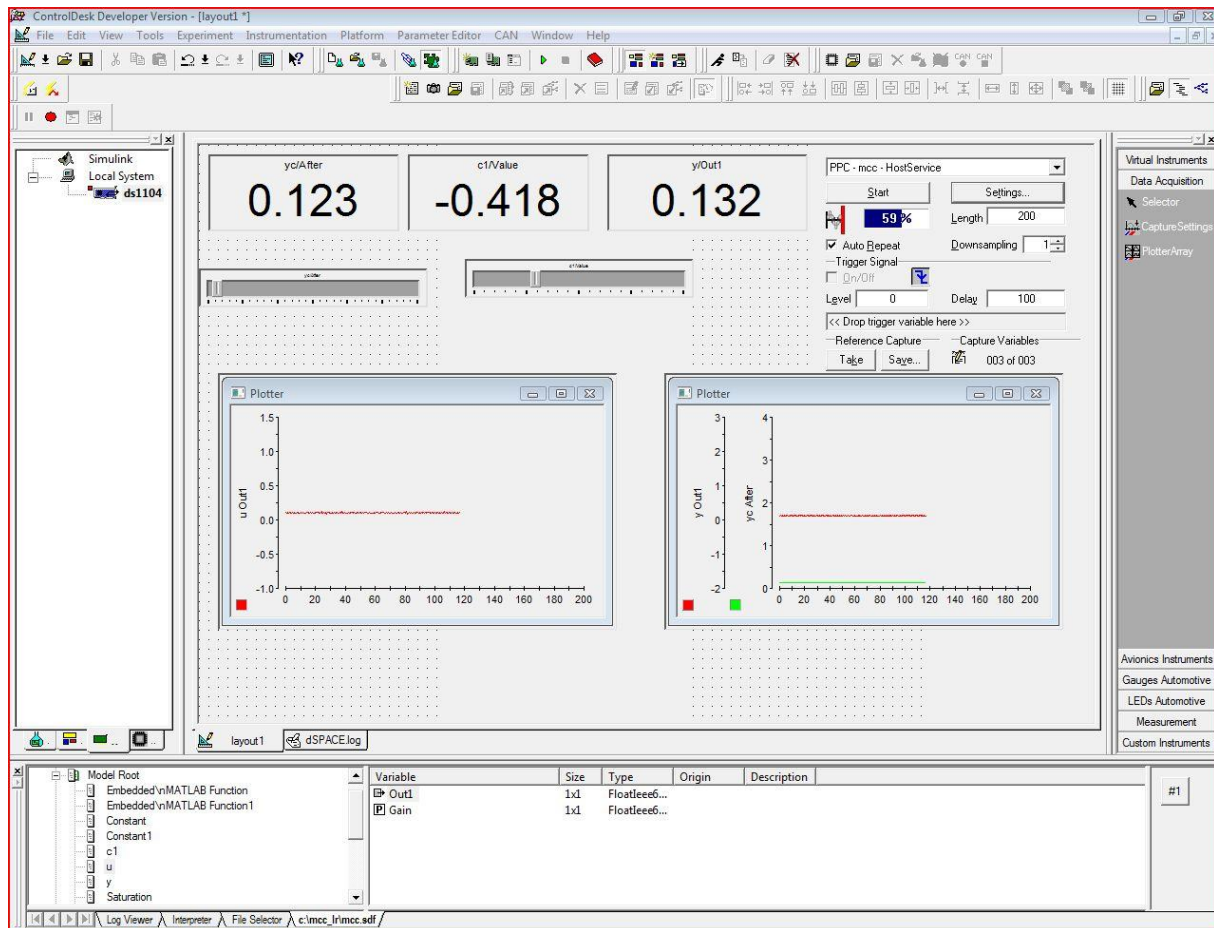


Figure III.3 interface utilisateur Control Desk

Cette interface est mené de :

- Deux sliders un pour la référence et l'autre pour le pôle
- Trois afficheurs (Displays) pour l'affichage des valeurs (référence, sortie, le pôle)
- Deux axes (plotter array) un pour visualiser le signal de commande et l'autre pour la référence et la réponse du système.
- Capture setting pour capter les différents signaux

La vitesse du moteur à courant continu est représentée comme suivante :

$$y_c = 1V \longrightarrow y = 1000\text{tr/min}$$

Après plusieurs essais, on a obtenu les signaux suivants :

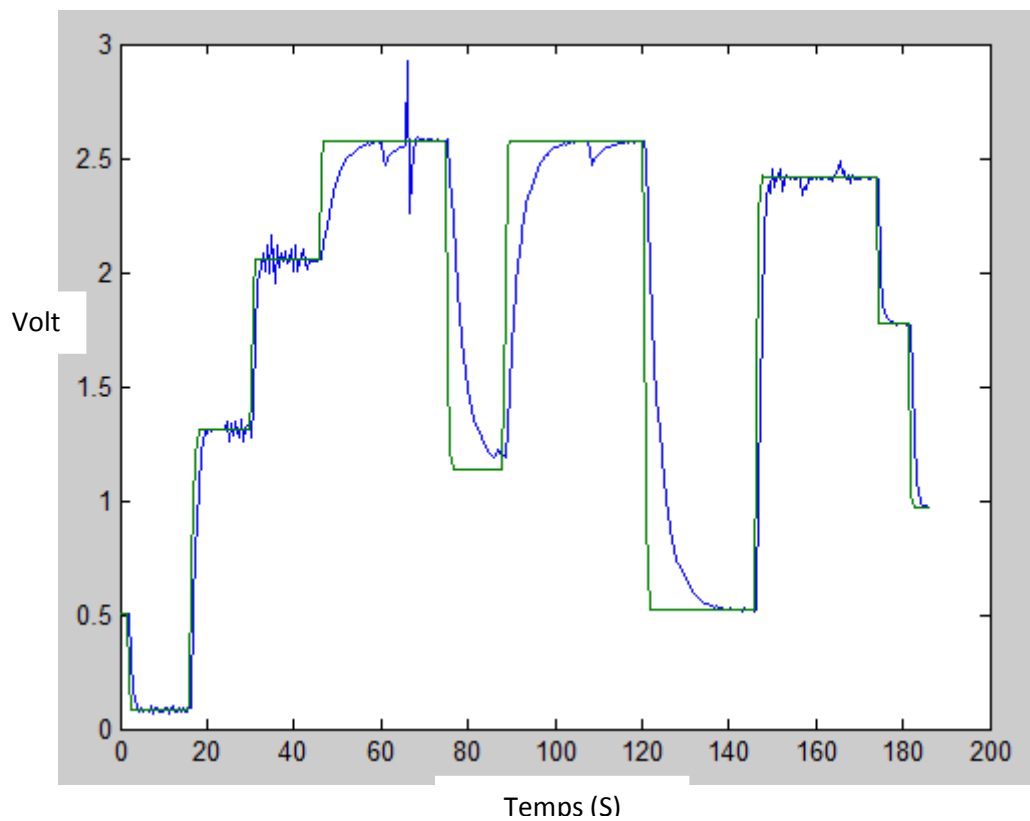


Figure III.4 signaux de sortie et de la référence essai pratique

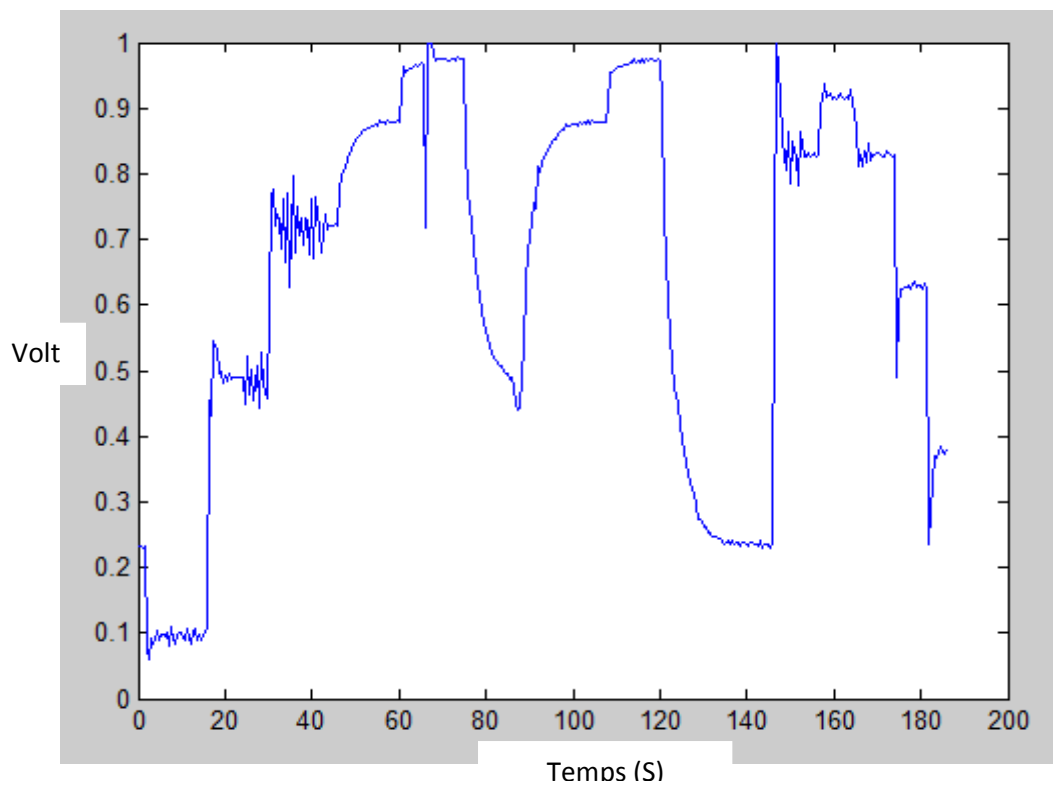


Figure III.5 signal de commande essai pratique

La figure (III.4) montre que la réponse du système suit la référence, aussi la présence de la charge qui est une perturbation pour notre système est rejetée grâce au signal de commande représenté dans la figure (III.5)

En effet la dynamique du système est affectée par la valeur de pôle :

- Pour un pôle proche de -1 , on remarque que la réponse du système est un peu lente.
- Pour un pôle $c_1 = -0.4$, la réponse du système est rapide avec un dépassement.

III.6 Conclusion

Ce présent chapitre nous a permis de voir un peu l'utilité de l'interface pour le dimensionnement du régulateur RST.

Le contrôle du système à temps réel avec la carte Dspace Rti 1104 et son logiciel Control Desk nous a permis d'analyser, modifier et visualiser les variables (référence, sortie, commande, pôle) pour commander la vitesse du moteur à courant continu, après son identification

La commande RST effectuée :

- La poursuite
- La régulation

La dynamique du système dépend de la valeur de pôle.

Conclusion générale

Conclusion générale

Afin de montrer l'intérêt d'un régulateur numérique de type RST nous avons décrit les méthodes de synthèse de ce dernier et comment dimensionner ce type de régulateurs avec une interface graphique, tout au long de notre travail notre souci constant a été de rechercher à travers les étapes de construction de notre projet les moyens les plus à même de rendre l'utilisation de ce logiciel la plus simple possible avec en perspective une performance satisfaisante. Cette étude montre combien un logiciel est indispensable surtout quand il peut répondre aux besoins des utilisateurs, en leur permettant de visualiser et imprimer les différents résultats nécessaires pour argumenter leur étude, et de faciliter ainsi dans une large mesure la compréhension de la théorie des systèmes, durant notre travail, on a constaté que le régulateur numérique RST :

- ✚ A la stratégie de dimensionnement basé sur le placement des pôles sur un modèle à poursuivre choisit par l'utilisateur.
- ✚ Il est dit « poursuite, régulation à objectifs indépendants » suite à l'intérêt de sa commande numérique, contrairement à beaucoup d'autres correcteurs numériques tel le classique PID.
- ✚ La simplification des zéros du système à régler provoque parfois un arrangement des pôles et zéros.
- ✚ Le bon rejet des perturbations.

On a vu aussi l'intérêt d'utiliser la carte Dspace Rti1104 et son logiciel Control Desk pour le contrôle des systèmes à temps réel.

La régulation numérique reste une méthode très importante dans l'étude des procédés ce qui élargit son application dans plusieurs domaines à savoir l'électronique, la physique, la chimie et la mécanique et d'autres disciplines.

Références

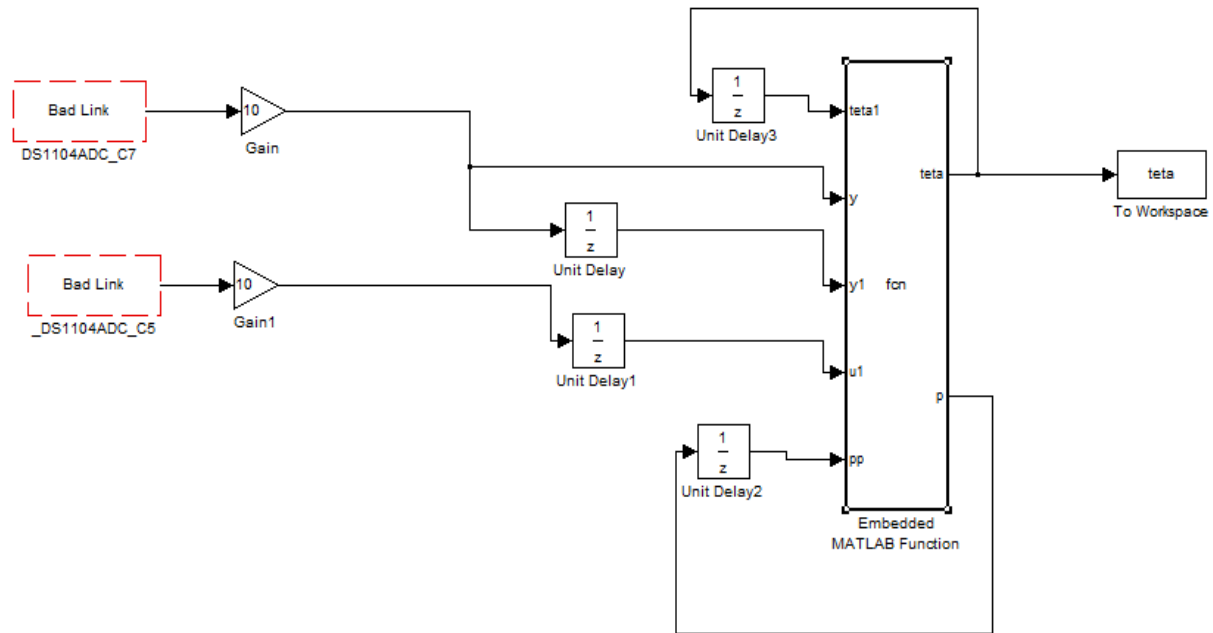
Bibliographiques

REFERENCES BIBLIOGRAPHIQUE

[1]	P.Borne, G.Dauohin-Tanguy, J.-P.Richard, F.Rotella, I.Zambettakis, Automatique, « Analyse et Régulation des processus industriels », Tome 2 Régulation numérique, Edition TECHNIP
[2]	Moussa BENJEDIA « synthèse d'algorithme de commande sans capteur de moteur pas à pas et implantation sur architecture programmable », thèse pour obtenir le grade de docteur de l'université de FRANCHE-COMTE, 16 novembre 2007
[3]	« Introduction à MATLAB et Simulink », Hoang Le-Huy, Professeur Département de génie électrique et de génie informatique Université Laval, Québec, CANADA, Septembre 1998
[4]	ROLAND LONGCHAMP « Commande numérique de systèmes dynamiques », première édition, presses polytechniques et universitaires romandes, 1995
[5]	Raymond KONN « Commande Analogique et Numérique des systèmes », édition ellipse
[6]	Fichier polenz.sq, Disponible sur le site : http:// homepage.mac.com/jpchemla
[7]	M. Cougnon JL Cours, «synthèse des correcteurs par placement des pôles »
[8]	Jérôme Briot « Introduction à la programmation des interfaces graphiques », Date de publication : 11/06/2007, dernière mise à jour 20/11/2012 ; disponible sur le site : http://briot-jerome.developpez.com/matlab/tutoriels/
[09]	Adrian Biran et Moshe Breiner, « MATLAB pour l'ingénieur », version 6 et 7, 2004 Pearson Education France
[10]	Département : Automatique, « MATLAB – SIMULINK Applications a l'Automatique »École des HAUTES ÉTUDES INDUSTRIELLES, Abdel AITOUCHE, Patrick DEBAY, Fabrice VIENNE
[11]	« A Guide to MATLAB » for Beginners and experienced users Brian R. Hunt, Ronald L. Lipsman, Jonathan M. Rosenberg, CAMBRIDGE University Press, 1995.

ANNEXES

Identification du système premier ordre

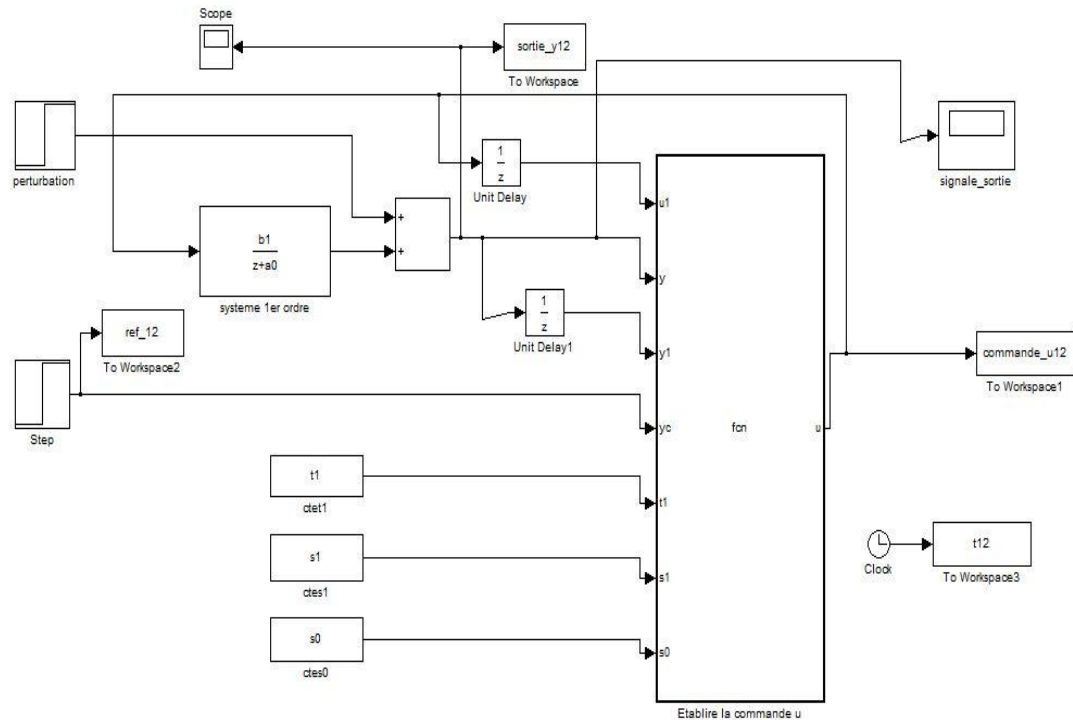


Algorithme de la méthode des moindres carrée récurrente

```
function [teta,p] = fcn(teta1,y,y1,u1,pp)
i=eye(2)
fi=[-y1;u1]
k=(pp*fi)/(1+fi'*pp*fi)
teta=teta1+k*(y-fi'*teta1)
p=(i-k*fi')*pp
```

Schémas de simulations

Système première ordre avec intégrateur



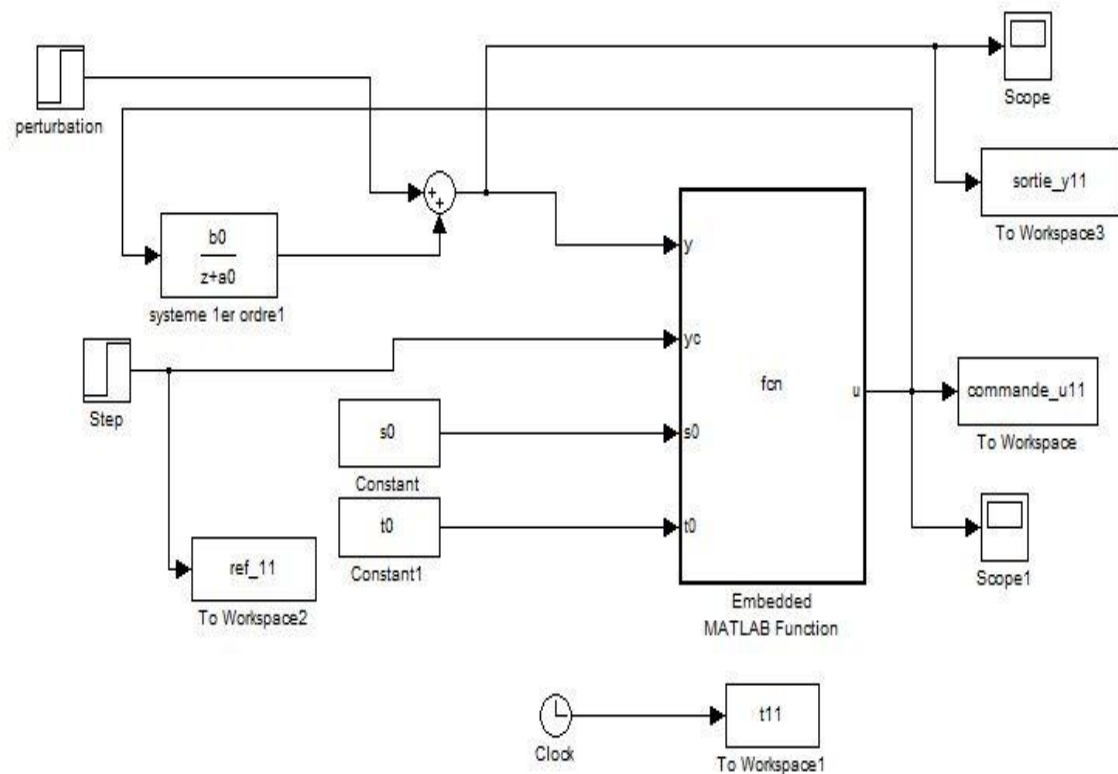
Paramètres du régulateur RST Avec un seul pôle

$$\begin{aligned} s0 &= a0/b0 \\ s1 &= (-a0+1+c1)/b0 \\ t1 &= (1+c1)/b0 \end{aligned}$$

La loi de commande

$$u[k] = u[k-1] + t1*yc[k] - s1*y[k] - s0*y[k-1]$$

Système première ordre sans intégrateur



Paramètres du Régulateur RST

Avec un seul pôle

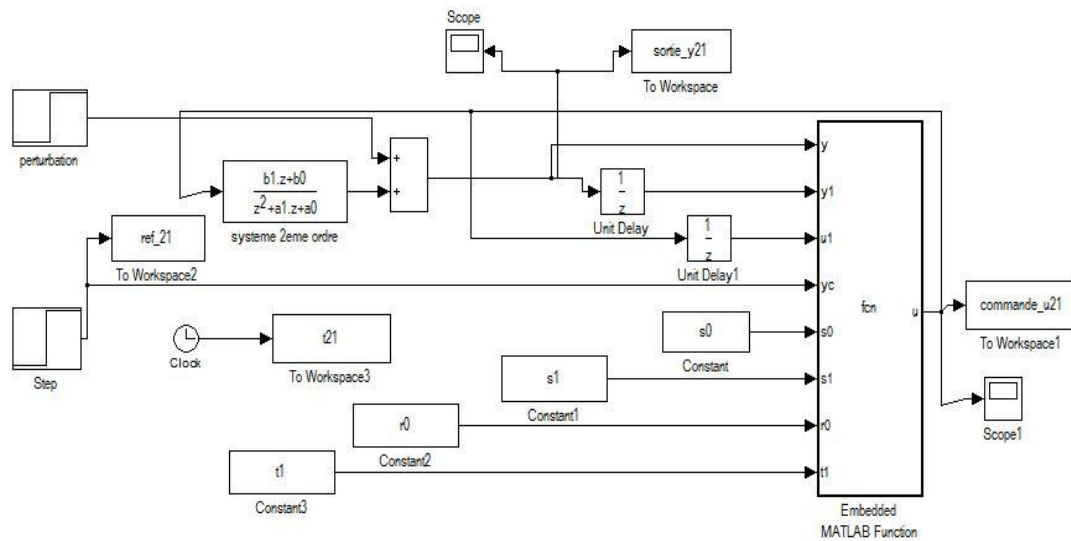
$$s0 = (c1-a0)/b0$$

$$t0 = (1+c1)/b0$$

La loi de commande

$$u[k] = t0*yc[k]-s0*y[k]$$

Système 2^{ème} ordre Sans intégrateur sans simplification des zéros



Paramètres du Régulateur RST

Avec un seul pôle

```

M = [0 b1 1;b1 b0 a1;b0 0 a0]
X = [c1-a1;-a0;0]
rst = inv(M)*X
s0 = rst(1)
s1 = rst(2)
r0 = rst(3)
t1 = (1+c1)/(b0+b1)
    
```

Avec deux pôles

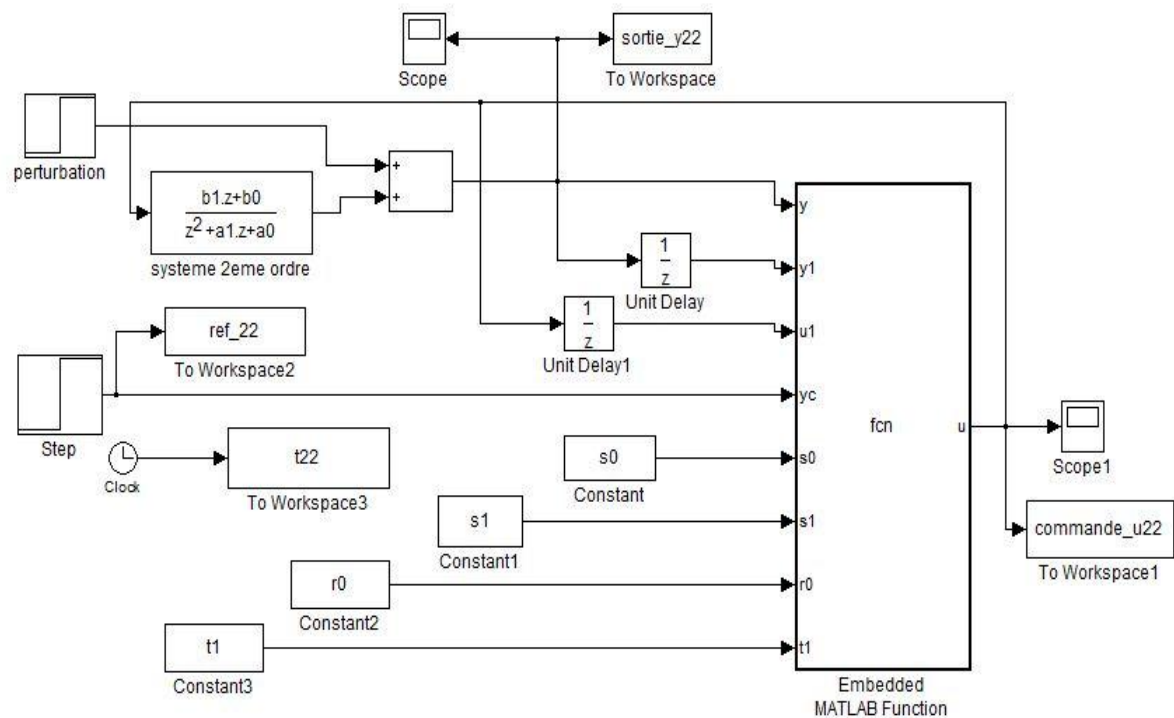
```

M = [0 b1 1;b1 b0 a1;b0 0 a0]
X = [(c1+c2-a1);c1*c2-a0;0]
rst = inv(M)*X
s0 = rst(1)
s1 = rst(2)
r0 = rst(3)
t1 = ((1+c1)*(1+c2))/(b0+b1)
    
```

La loi de commande

$$u[k] = -r0*u[k-1] + t1*yc[k] - s1*y[k] - s0*y[k-1]$$

Système 2^{ème} ordre sans intégrateur et avec simplification des zéros



Paramètres du Régulateur RST

Avec un seul pôle

$$\begin{aligned} r0 &= b0/b1 \\ s1 &= (c1-a1)/b1 \\ s0 &= -a0/b1 \\ t1 &= (1+c1)/b1 \end{aligned}$$

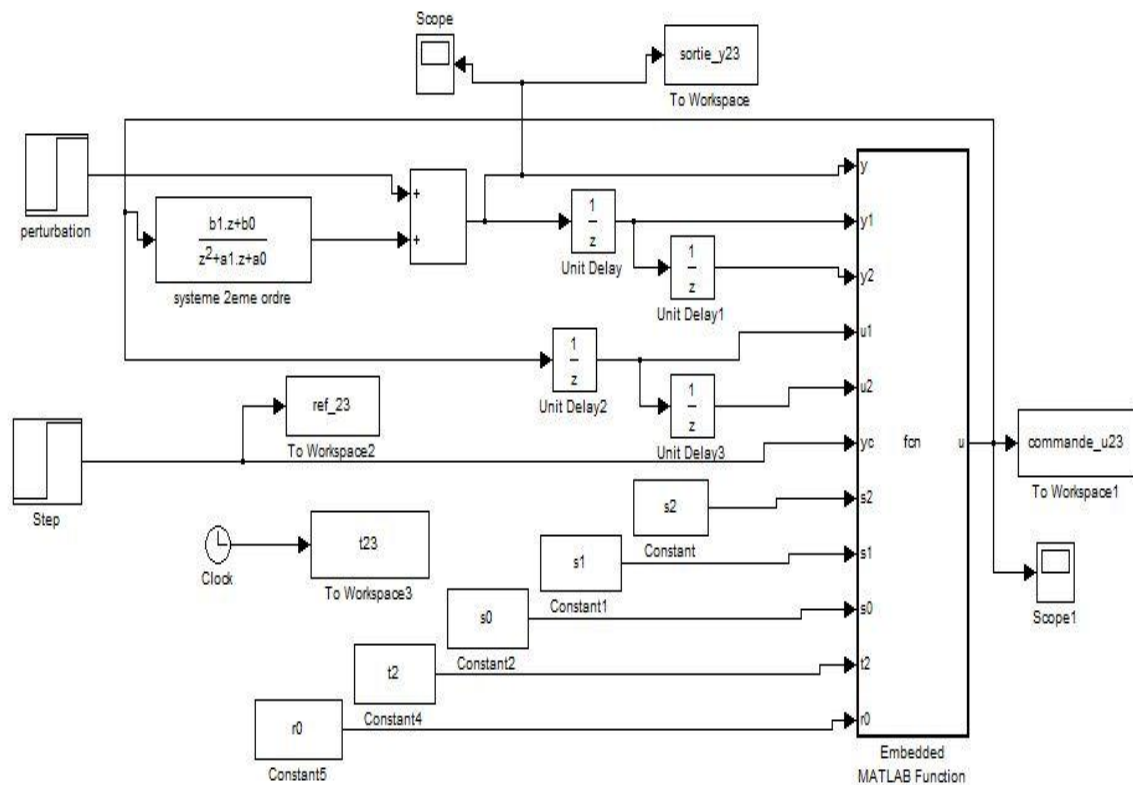
Avec deux pôles

$$\begin{aligned} s1 &= (c1+c2-a1)/b1 \\ s0 &= (c1*c2-a0)/b1 \\ r0 &= b0/b1 \\ t1 &= ((1+c1)*(1+c2))/b1 \end{aligned}$$

La loi de commande

$$u[k] = -r0*u[k-1] + t1*yc[k] - s1*y[k] - s0*y[k-1]$$

Système 2^{ème} ordre avec intégrateur et avec simplification des zéros



Paramètres du Régulateur RST

Avec un seul pôle

$$\begin{aligned} t2 &= (1+c1)/b1 \\ r0 &= b0/b1 \\ s2 &= (c1-a1+1)/b1 \\ s1 &= (a1-a0)/b1 \\ s0 &= a0/b1 \end{aligned}$$

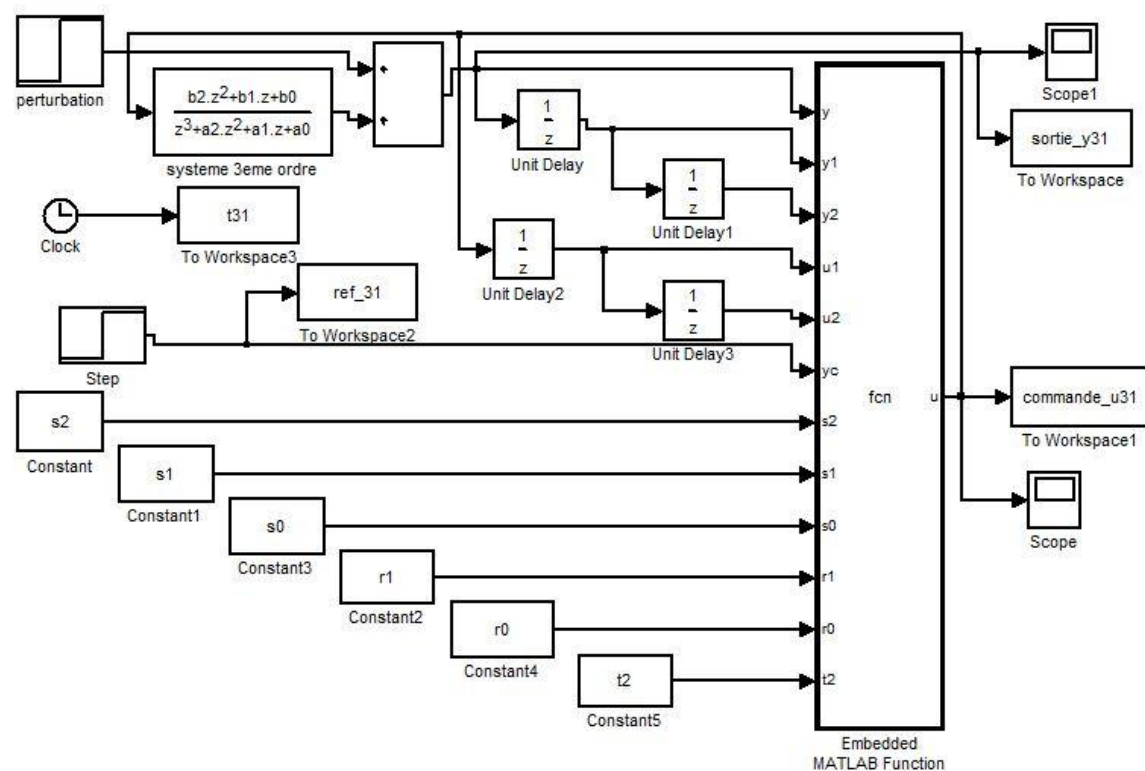
Avec deux pôles

$$\begin{aligned} t2 &= (1+c1)*(1+c2)/b1 \\ r0 &= b0/b1 \\ s2 &= (c1+c2-a1+1)/b1 \\ s1 &= (a1-a0+c1*c2)/b1 \\ s0 &= a0/b1 \end{aligned}$$

La loi de commande

$$u[k] = (1-r0)*u[k-1]+r0*u[k-2]+t2*yc[k]-s2*y[k]-s1*y[k-1]-s0*y[k-2]$$

Système 3^{eme} ordre sans intégrateur et sans simplification des zéros



Paramètres du Régulateur RST

Avec un seul pôle

$$M = \begin{bmatrix} b_2 & 0 & 0 & 1 & 0; & b_1 & b_2 & 0 & a_2 & 1; & b_0 & b_1 & b_2 & a_1 & a_2; & 0 & b_0 & b_1 & a_0 & a_1; & 0 & 0 & b_0 & 0 & a_0 \end{bmatrix}$$

$$X = \begin{bmatrix} (c_1 - a_2); & -a_1; & -a_0; & 0; & 0 \end{bmatrix}$$

$$\text{rst} = \text{inv}(M) * X$$

$$s_2 = \text{rst}(1); s_1 = \text{rst}(2); s_0 = \text{rst}(3)$$

$$r_1 = \text{rst}(4)$$

$$r_0 = \text{rst}(5)$$

$$t_2 = (1 + c_1) / (b_0 + b_1 + b_2)$$

Avec deux pôles

$$M = \begin{bmatrix} 1 & 0 & b_2 & 0 & 0; & a_2 & 1 & b_1 & b_2 & 0; & a_2 & b_0 & b_1 & b_2; & a_0 & a_1 & 0 & b_0 & b_1; & 0 & a_0 & 0 & 0 & b_0 \end{bmatrix}$$

$$X = \begin{bmatrix} (c_1 + c_2) - a_2; & (c_1 * c_2) - a_1; & -a_0; & 0; & 0 \end{bmatrix}$$

$$\text{rst} = \text{inv}(M) * X$$

$$s_2 = \text{rst}(3); s_1 = \text{rst}(4); s_0 = \text{rst}(5)$$

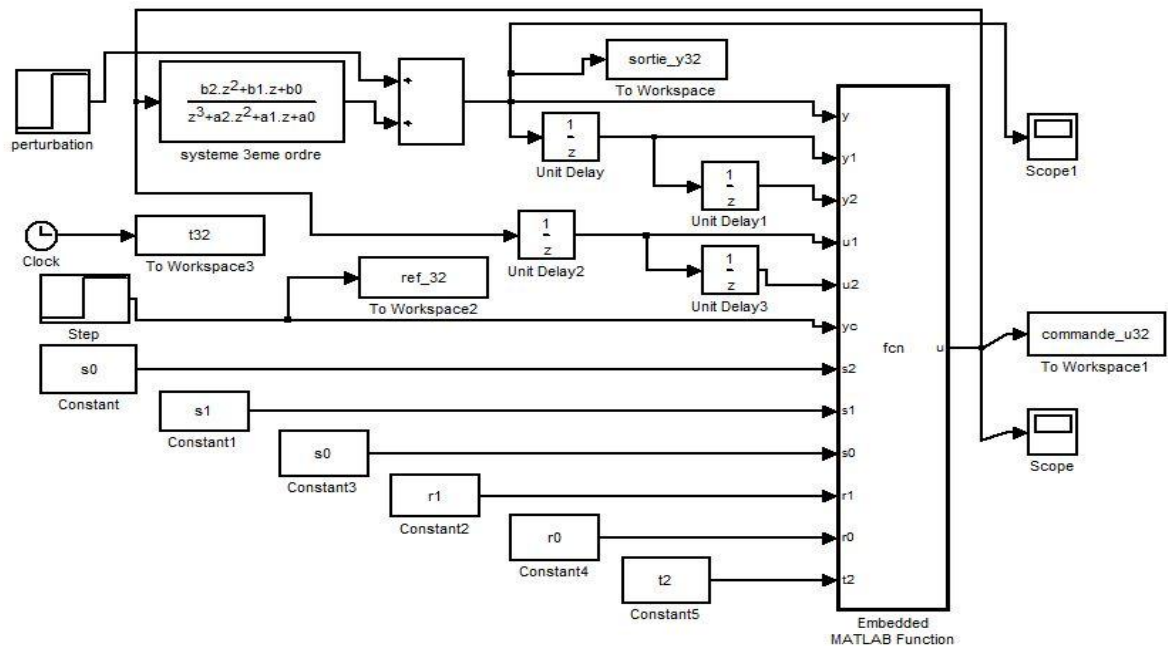
$$r_1 = \text{rst}(1); r_0 = \text{rst}(2)$$

$$t_2 = (1 + c_1) * (1 + c_2) / (b_0 + b_1 + b_2)$$

La loi de commande

$$u[k] = -r_1 * u[k-1] - r_0 * u[k-2] + t_2 * y_c[k] - s_2 * y[k] - s_1 * y[k-1] - s_0 * y[k-2]$$

Système 3^{ème} ordre sans intégrateur avec simplification des zéros



Paramètres du Régulateur RST

Avec un seul pôle

$$\begin{aligned} t2 &= (1+c1)/(b2) \\ s2 &= (c1-a2)/b2 \\ s1 &= -a1/b2 \\ s0 &= -a0/b2 \\ r1 &= b1/b2 \\ r0 &= b0/b2 \end{aligned}$$

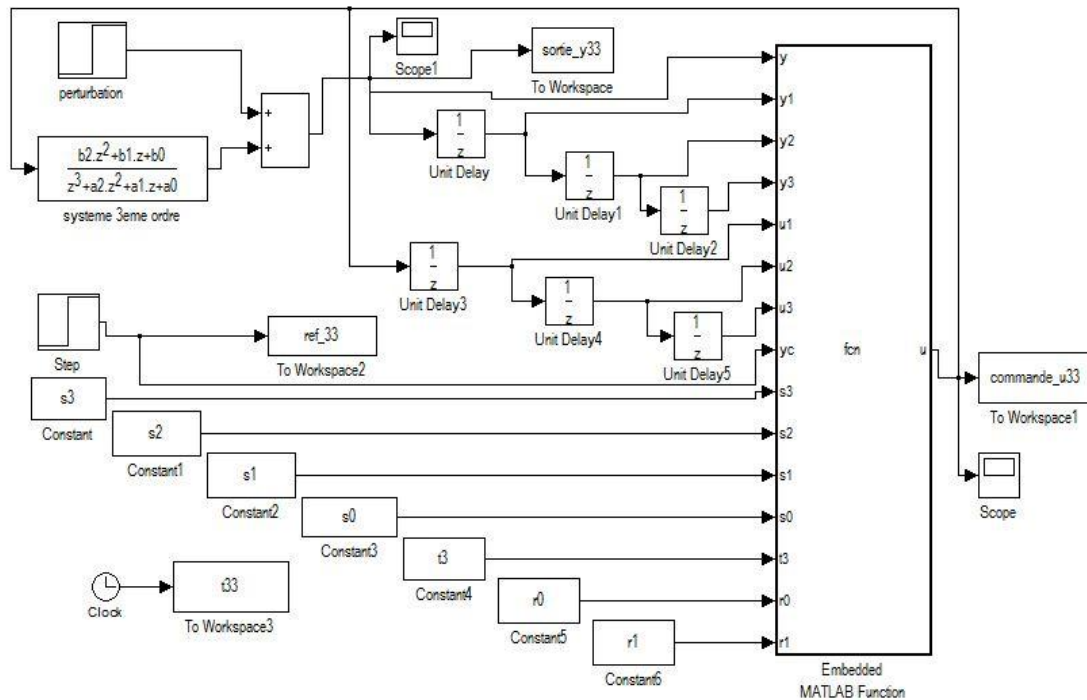
Avec deux pôles

$$\begin{aligned} t2 &= (1+c1)*(1+c2)/b2 \\ s2 &= (c1+c2-a2)/b2 \\ s1 &= ((c1*c2)-a1)/b2 \\ s0 &= -a0/b2 \\ r1 &= b1/b2 \\ r0 &= b0/b2 \end{aligned}$$

La loi de commande

$$u[k] = -r1*u1[k-1]-r0*u[k-2]+t2*yc[k]-s2*y[k]-s1*y[k-1]-s0*y[k-2]$$

Système 3^{eme} ordre avec intégrateur avec simplification des zéros



Paramètres du Régulateur RST

Avec un seul pôle

$$\begin{aligned} s3 &= (c1-a2+1)/b2 \\ s2 &= (a2-a1)/b2 \\ s1 &= (a1-a0)/b2 \\ s0 &= a0/b2 \\ t3 &= (1+c1)/b2 \\ r1 &= b1/b2 \\ r0 &= b0/b2 \end{aligned}$$

Avec deux pôles

$$\begin{aligned} s3 &= ((c1+c2)-a2+1)/b2 \\ s2 &= ((c1*c2)+a2-a1)/b2 \\ s1 &= (a1-a0)/b2 \\ s0 &= a0/b2 \\ t3 &= (1+c1)*(1+c2)/b2 \\ r1 &= b1/b2 \\ r0 &= b0/b2 \end{aligned}$$

La loi de commande

$$u[k] = (1-r1)*u[k-1] + (r1-r0)*u[k-2] + r0*u[k-3] + t3*yc[k] - s3*y[k] - s2*y[k-1] - s1*y[k-2] - s0*y[k-3]$$

Résumé

On a partagé notre travail en trois parties suivantes :

Le premier chapitre est consacré à la description des méthodes de synthèse du régulateur RST et de déterminer les algorithmes pour le dimensionnement.

Dans le deuxième chapitre, on présente l'outil de programmation des interfaces GUIDE MATLAB ensuite l'interface graphique réalisée.

Le dernier chapitre consacré pour la partie pratique a objectif d'identifier notre système qui est un moteur à courant continue afin d'élaborer la commande RST par l'exploitation de l'interface graphique et la carte Dspace Rti1104 et son logiciel control Desk.