

الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي والبحث العلمي

UNIVERSITE BADJI MOKHTAR - ANNABA
BADJI MOKHTAR – ANNABA UNIVERSITY



جامعة باجي مختار – عنابة

Faculté : TECHNOLOGIE

Département : ELECTROTECHNIQUE

Domaine : SCIENCES ET TECHNOLOGIES

Filière : ELECTROTECHNIQUE

Spécialité : COMMANDE ELECTRIQUE

Mémoire

Présenté en vue de l'obtention du Diplôme de Master

Thème:

**ÉTUDES DES PERFORMANCES DU RÉGLAGE
AVANCÉ D'UN ENTRAÎNEMENT ÉLECTRIQUE**

Présenté par : *Bouziane Adel & Didi Zineddine*

Encadrant : *BAHI TAHAR*

Grade : *PROFESSEUR*

Badji Mokhtar-Annaba

Jury de Soutenance :

ZERZOURI Nora	MCB	Université Badji Mokhtar Annaba	Présidente
BAHI Tahar	PROF	Université Badji Mokhtar Annaba	Encadrant
MOHAMMEDI Moufid	MCA	Université Badji Mokhtar Annaba	Examineur

Année Universitaire : 2022/2023

DÉDICACE

Mes chers parents AMAR et HOURIYA

Aucun mot ne suffit pour vous remercier de chaque instant-vous pour tous vos encouragements. Je m'engage à continuer sur la voie tracée et à continuer de vous rendre fier. Qu'Allah vous accorde le bonheur, la santé et toutes les bonnes choses possibles. Je vous dédie ce travail avec un profond amour :

À mes sœurs YASMINE et HANINE, à chaque membre de ma famille

Et

Spécialement mon oncle ABDELOUAHAB, vraiment merci, pour votre patience, votre gentillesse, votre écoute et votre présence à mes côtés

dans ma vie.

À mes amis FARES, ABDERAHMAN, YUCEF, TEISSIR et

BADREDDINE

Merci pour votre soutien

ZINEDDINE

DÉDICACE

Avant tout , je tiens à remercier ALLAH , le TOUT PUISSANT

Avec l'expression de ma reconnaissance, je dédie ce modeste travail à ceux qui, quels que soient les termes embrassés, je n'arriverais jamais à leur exprimer mon amour sincère.

A mon Cher Père RABAH que ALLAH le protège , mon précieux offre du dieu, qui doit ma vie, ma réussite .

A l'âme de mon adorable mère ZEINEB qui a souffert sans me laisser souffrir, qui n'a jamais dit non à mes exigences et qui n'a épargné aucun effort pour me voir heureux.

A ma petite famille : MON EPOUSE qui n'a pas cessée de me conseiller, encourager et soutenir tout au long de mes études et A mes adorables petites filles ZEINEB et RAZANE et à mon nouveau fils Mohammed qui ont toujours su comment procurer la joie et le bonheur, Que Dieu les protège et leurs offre la chance et le bonheur.

Sans oublier mon binôme Didi Zinedine pour son soutien moral, sa patience et sa compréhension tout au long de la réalisation de ce projet de fin d'études. Et, aussi à mon ami : Mekahlia Youcef, de qui j'ai reçu tout le soutien et un bon accompagnement.

ADEL

REMERCIEMENT

Grace à dieu, nous avons terminé ce travail.

Avec tout le respect nous remercions notre encadreur Monsieur le
Professeur BAHY Tahar

Pour avoir accepté de diriger ce travail et puis par son esprit ouvert, ses
conseils et sa disponibilité pour nous aider à compléter ce travail.

Nous exprimons notre gratitude à tous les enseignants du département
d'électrotechnique.

En conclusion, nous adressons nos remerciements les plus sincères aux
membres du jury, qui font l'honneur d'accepter de faire partie de ce
jury.

Sommaire

Introduction générale.....	11
----------------------------	----

CHAPITRE 1

GÉNÉRALITÉ SUR LA MACHINE À COURANT CONTINU

1.1 Introduction.....	13
1.2 Machine à courant continu	13
1.3 Constitution de la MCC.....	14
1.4 Principe de fonctionnement.....	16
1.5 Types de machines à courant continu.....	16
1.6 Avantages et inconvénients de la MCC.....	17
1.6.1 Avantages.....	17
1.6.2 Inconvénients.....	18
1.7 Conclusion.....	18

CHAPITRE 2

MODÉLISATION ET IDENTIFICATION DE LA MACHINE À COURANT CONTINU

2.1 Introduction	20
2.2 Modélisation de la MCCAP.....	20
2.3 Identification des paramètres de la MCC.....	23
2.3.1 Méthode d'identification	24

2.4 Simulation et discussion du MCC en boucle ouvert.....	29
2.5 Synthèse du régulateur de vitesse.....	32
2.5.1 Régulateur PID.....	32
2.5.2 Synthèse	34
2.5.3 PID tuner.....	37
2.6 Simulation et discussion du MCC en boucle fermé.....	39
2.7 Conclusion.....	41

CHAPITRE 3

COMMANDE POLYNOMIALE RST

3.1 Introduction.....	43
3.2 Régulateur PID discret.....	45
3.3 Méthodes de discrétisation.....	46
3.3.1 Méthode d'approximation d'Euler vers l'arrière (dérivée).....	46
3.3.2 Méthode d'approximation Trapézoïdal	47
3.4 Principe de la commande RST.....	48
3.5 Relations entrées/ sorties	51
3.6 Synthèse du régulateur RST.....	53
3.6.1 Détermination des polynômes R et S.....	54
3.7 Stratégie de placement de pole robuste.....	59
3.8 Simulation et discussion du MCC en boucle fermée.....	61
3.9 Conclusion.....	63

CHAPITRE 4

COMMANDE DE VITESSE D'UN MOTEUR PAR MATLAB ET ARDUINO

4.1 Introduction.....	65
4.2 Composants utilisé.....	65
4.2.1 Carte Arduino.....	65
4.2.2 Hacheur double pont L298N.....	68
4.2.3 Moteur de petite puissance.....	70
4.2.4 Capteur de vitesse HC-020K.....	70
4.2.5 Matlab / Simulink.....	71
4.2.5.1 Simulink.....	71
4.2.5.2 Arduino Support package.....	71
4.3 Realisation.....	72
4.4 Révélées expérimentales	73
4.4.2 Essais en boucle ouverte (avec génératrice et changement de vitesse).....	73
4.4.3 Essais en boucle fermée (régulateur PI avec génératrice à vide).....	74
4.4.4 Essais en boucle fermée (régulateur PI avec génératrice et charge).....	74
4.4.5 Essais en boucle fermée (régulateur PI avec génératrice et variation de vitesse).....	75
4.4.6 Essais en boucle fermée (régulateur RST avec génératrice à vide).....	76
4.4.7 Essais en boucle fermée (régulateur RST avec perturbation).....	76
Conclusion générale	77
Références.....	78,79,80

TABLEAU DE FIGURES

CHAPITRE 1

<i>Figure 1.1</i>	Schémas blocs des modes de fonctionnement de la MCC.
<i>Figure 1.2</i>	Quadrants de fonctionnement.
<i>Figure 1.3</i>	Composants de la MCC.
<i>Figure 1.4</i>	Constitution d'une MCC.

CHAPITRE 2

<i>Figure 2.1</i>	Représentation schématique d'une MCC.
<i>Figure 2.2</i>	Schéma bloc du MCC.
<i>Figure 2.3</i>	Photo du MCC utilisé.
<i>Figure 2.4</i>	L'évolution de la vitesse en fonction du temps du moteur utilisé
<i>Figure 2.5</i>	Application Parameter Estimator
<i>Figure 2.6</i>	Création du nouvelle expérimente
<i>Figure 2.7</i>	Sélection des sorties mesurées et estimés.
<i>Figure 2.8</i>	Paramètres estimés positives.
<i>Figure 2.9</i>	Application d'estimation et obtention des résultats.
<i>Figure 2.10.A</i>	Résultats de simulation du modèle de la MCC en BO à vide.
<i>Figure 2.10.B</i>	Résultats de simulation du modèle de la MCC en BO en charge.
<i>Figure 2.11</i>	Démarrage et inversion de sens de rotation.
<i>Figure 2.12</i>	Structure du système en boucle fermée.
<i>Figure 2.13</i>	Sélection du régulateur PID et (tunning)
<i>Figure 2.14</i>	Configuration des valeurs désirée pour k_p et k_i .
<i>Figure 2.14.1</i>	Résultats de simulation de la MCC en boucle fermée à vide.
<i>Figure 2.14.2</i>	Résultats de simulation de la MCC en boucle fermée en charge.

CHAPITRE 3

<i>Figure 3.1</i>	Conception par modèles continus et discrets.
<i>Figure 3.2</i>	Temps de décalage vers l'avant.
<i>Figure 3.3</i>	Temps de décalage vers l'arrière.
<i>Figure 3.4</i>	L'intégrale de l'erreur sur l'intervalle (t-1) à (t).
<i>Figure 3.5</i>	Schéma bloc du système.
<i>Figure 3.6</i>	schéma bloc général pour la synthèse RST.
<i>Figure 3.7</i>	Représentation générale de l'RST.
<i>Figure 3.8</i>	Modèle de référence.
<i>Figure 3.9</i>	Structure de synthèse.
<i>Figure 3.10</i>	Distribution aléatoire des pôles.
<i>Fig 3.11</i>	Transformation des pôles instables dans la zone de stabilité.
<i>Figure 3.12</i>	Projection des pôles peu amortie sur ($\lambda = 0.86$).
<i>Fig 3.13</i>	Projection des pôles lents sur la verticale p_c
<i>Figure 3.14</i>	Projection des pôles sur la verticale p_f .
<i>Figure 3.15</i>	Résultats de simulation en boucle fermée (régulateur RST)

CHAPITRE 4 :

<i>Figure 4.1 :</i>	Architecture du Arduino méga.
<i>Figure 4.2 :</i>	Rapport cyclique (α).
<i>Figure 4.3</i>	Hacheur double pont L298N.
<i>Figure 4.4.</i>	Moteur utilisé dans la réalisation.
<i>Figure 4.5</i>	Capteur de vitesse HC-020K.
<i>Figure 4.6</i>	Réalisation du projet
<i>Figure 4.7</i>	Résultats de simulation en boucle ouverte à vide.
<i>Figure 4.8</i>	Démarrage et changement de vitesse à vide.
<i>Figure 4.9</i>	Réglage de vitesse avec un PI (à vide).
<i>Figure 4.10</i>	Réglage de vitesse avec un PI (en charge).
<i>Fig 4.11</i>	Variation de vitesse avec un PI.
<i>Fig 4.12</i>	Réglage de vitesse avec un RST (à vide).
<i>Figure 4.13</i>	Réglage de vitesse avec un RST (en charge).

- Tableau 2.1 Résultats de la mesure de la résistance.....25
- Tableau 2.2 Valeurs identifiées du moteur29

ملخص

تستخدم وحدات القيادة الكهربائية ذات السرعة المتغيرة بشكل واسع في العديد من المجالات. يركز هذا العمل، الذي تم إجراؤه في إطار مشروع نهاية درجة الماجستير، على دراسة التحكم في محرك التيار المستمر ذو المغناطيس الدائم ذو القوة الصغيرة. يتعلق الأمر بدراسة نظرية وعملية للتحكم في السرعة باستخدام المتحكمات المتعددة الحدود "RST". تم إنشاء نموذج أولي باستخدام لوحة Arduino Mega 2560 ومفتاح قطع L298N. تم إجراء محاكاة واختبارات تجريبية ومناقشة النتائج.

الكلمات المفتاحية:

نموذج تجريبي ، منظم RST ، سرعة متغيرة ، محرك التيار المستمر ذو مغناطيس دائم،

ABSTRACT

Variable speed electric drives are widely used in several fields. This work, carried out as part of the end of master's degree project, focuses on the study of the control of a small power permanent magnet DC motor. It involves a theoretical and practical study of speed control using "RST" polynomial controllers. A prototype was created using an Arduino Mega 2560 board and an L298N chopper; simulations and experimental tests were carried out and the results discussed.

Keywords:

Variable speed, permanent magnet DC motor, RST Regulator, Experimental model.

RÉSUMÉ

Les entraînements électriques fonctionnant à vitesse variable sont très employés dans plusieurs domaines. Ce travail réalisé dans le cadre du projet de fin d'étude master porte sur l'étude de la commande d'un moteur à courant continu à aimants permanents de petite puissance. Il s'agit d'une étude théorique et pratique de la commande en vitesse en utilisant des contrôleurs polynomiale "RST". Un prototype est réalisé à base d'Une carte Arduino Mega 2560 et un hacheur L298N à des simulations et des essais expérimentaux ont été effectués et les résultats discutés.

Mots clés :

Vitesse variable, moteur à courant continu à aimants permanent, Régulateur RST, Maquette expérimentale.

INTRODUCTION GÉNÉRALE

L'avancée technologique contemporaine a favorisé l'évolution des sciences et a imposé la nécessité d'explorer des domaines théoriques toujours plus sophistiqués. Parmi ces disciplines scientifiques en pleine croissance, la commande électrique des systèmes, qui intègre rapidement les avancées technologiques, se distingue. Aujourd'hui, elle est considérée comme un domaine essentiel des sciences de l'ingénieur, englobant la modélisation, l'analyse, l'identification, le contrôle et la régulation des systèmes dynamiques dans le secteur industriel.

Dans une grande partie des processus industriels, les machines électriques jouent un rôle primordial. La vitesse de rotation de ces machines est, en effet, un élément clé dans de nombreuses applications industrielles. Dans le cas des machines à courant continu à aimants permanents, il est possible de moduler facilement la vitesse de rotation, ce qui nécessite souvent l'application d'une loi de commande. La commande robuste, basée sur le régulateur RST, est une méthode éprouvée qui donne d'excellents résultats en améliorant la rapidité, en rejetant les perturbations et en filtrant, grâce à l'action du régulateur RST.

L'ambition de ce travail est d'implémenter une méthode de contrôle sur un moteur à courant continu à aimants permanents, précisément la commande RST robuste, en utilisant une approche de placement de pôles robuste et MatLab. Notre tâche consiste à intégrer cette commande sur une carte Arduino Mega pour contrôler un moteur à courant continu de faible puissance.

L'objectif du premier chapitre est de présenter les notions générales de la machine à courant continu.

Dans le deuxième chapitre on s'intéresse à la modélisation et l'identification du MCC, puis, la synthèse du régulateur PI et finalement une simulation en boucle ouverte et en boucle fermée.

Le troisième chapitre s'intéresse à la discrétisation d'un système continu et la commande polynomiale RST et nous avons conclu avec la simulation.

La réalisation de notre travail sera dans le quatrième chapitre, faire une application réelle du circuit de régulation de vitesse à l'aide du MatLab et l'Arduino avec l'application du régulateur PI et le régulateur RST. Et on terminera ce travail par une conclusion générale.

CHAPITRE 1

GÉNÉRALITÉ SUR LA MACHINE À COURANT CONTINU

CHAPITRE 1

GÉNÉRALITÉ SUR LA MACHINE À COURANT CONTINU

1.1 Introduction

Ce chapitre présente, principalement, les notions de base sur la construction et le réglage de la machine à courant continu (MCC). Par ailleurs, nous évoquons son utilisation dans le domaine industriel et particulièrement son fonctionnement à vitesse variable. Puis, nous résumerons ses avantages et inconvénients et à la fin du chapitre une conclusion sera donnée.

1.2 Machine à courant continu

La machine à courant continu est un convertisseur électromécanique d'énergie totalement réversible [1]. Il permet de transformer l'énergie électrique continue en énergie mécanique pour entrainer une charge. Elle est réversible, c'est-à-dire que la constitution d'une génératrice à courant continu (GCC) est identique à celle d'un moteur à courant continu (MCC). La figure 1.1 présente les deux (2) schémas bloc relatifs aux modes de fonctionnement en génératrice et en moteur.

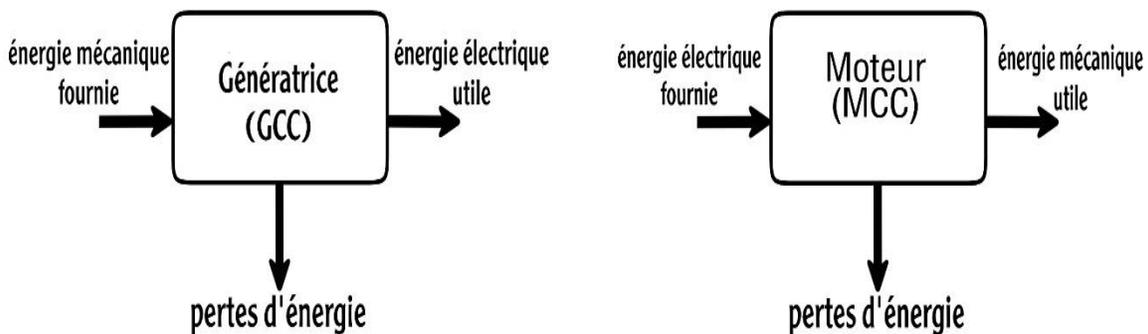


Fig 1.1 Schémas blocs des modes de fonctionnement de la MCC

Ces différents modes de fonctionnement peuvent être représentés dans les quatre (4) quadrants du plan couple en fonction de la vitesse ($C = f(\Omega)$) comme le montre la figure 1.2.

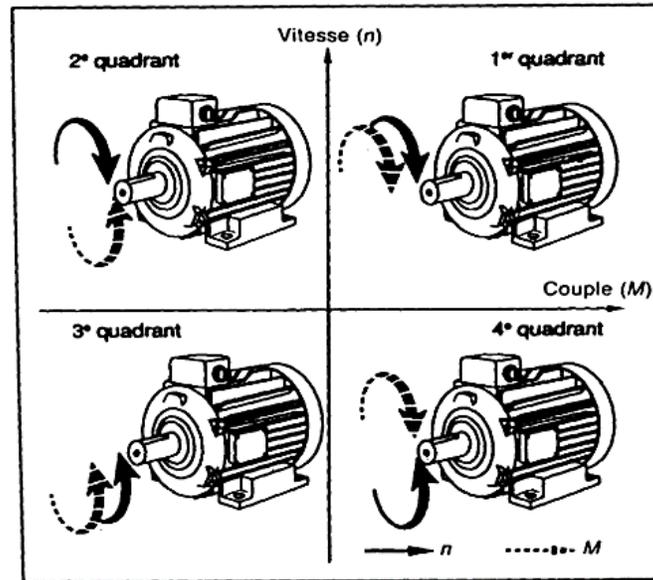


Fig 1.2 Quadrants de fonctionnement

Ainsi, deux quadrants (1 et 3) pour le régime moteur et les deux autres quadrants (2 et 4) pour le régime générateur.

1.3 Constitution de la MCC

Du point de vue construction, MCC comporte deux (2) circuits magnétiques appelés stator (inducteur) et rotor (induit) ou son principe de fonctionnement est :

Un courant électrique parcourt les conducteurs du rotor et lorsque ces derniers sont balayés par le champ magnétique (B) créée par l'inducteur. Cependant, le champ magnétique est soit créée par un inducteur bobiné alimenté en courant continu ou par des aimants permanents qui sont fabriqués en ferrites dans la plupart des cas [2].

La figure 1.3 montre un inducteur à aimants permanents (figure 1.3.a) , l'induit (figure 1.3.b) , le collecteur (figure 1.3.c), et modèle des balais et porte balais (figure 1.3.d).

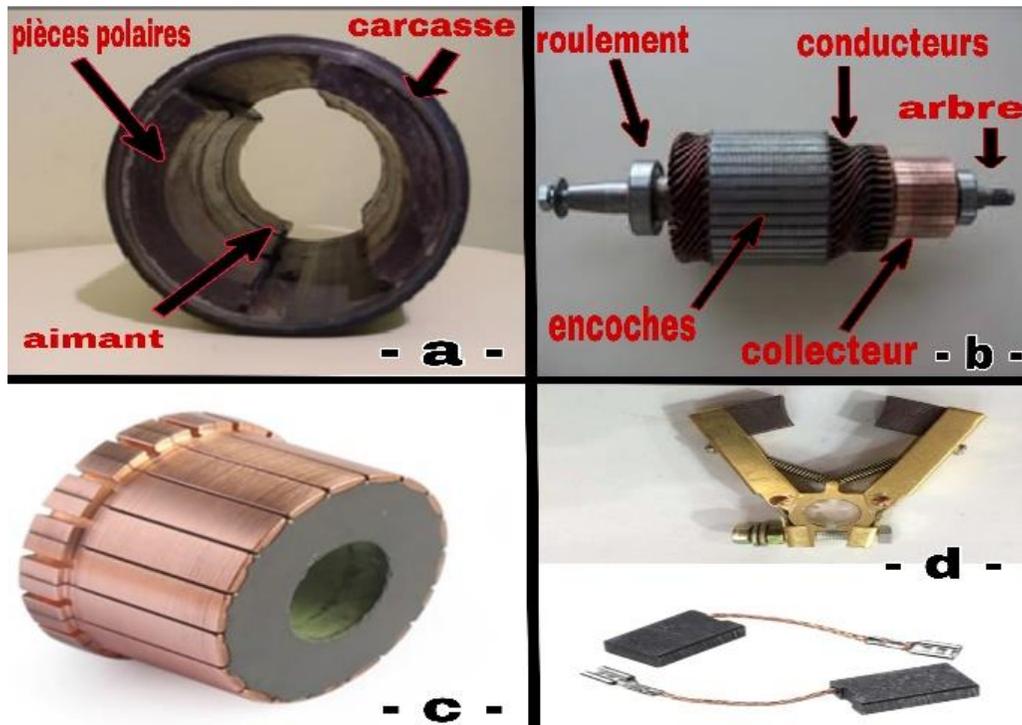


Fig 1.3 Composants de la MCC

Concernant l'inducteur, il est formé essentiellement de la carcasse (culasse) qui renferme le champ magnétique, les pièces polaires en tôles feuilletées qui produisent le champ d'induction et l'enroulement d'excitation qui sert à la création du champ magnétique. Et, la partie mobile comporte le collecteur qui a deux rôles, l'adaptation de la fréquence des courants rotoriques à la vitesse de rotation de la machine (redresseur mécanique) et l'inversion du courant dans un conducteur du rotor en même temps que l'inversion de la f.é.m. correspond par ailleurs les balais et les portes balais qui frottent sur la surface du collecteur sont faits en graphite ou en charbon. Le dispositif collecteur/ balais permet donc de faire circuler un courant dans l'induit.

La figure 1.4 expose les différentes pièces nécessaires pour l'assemblage d'un MCC [3].

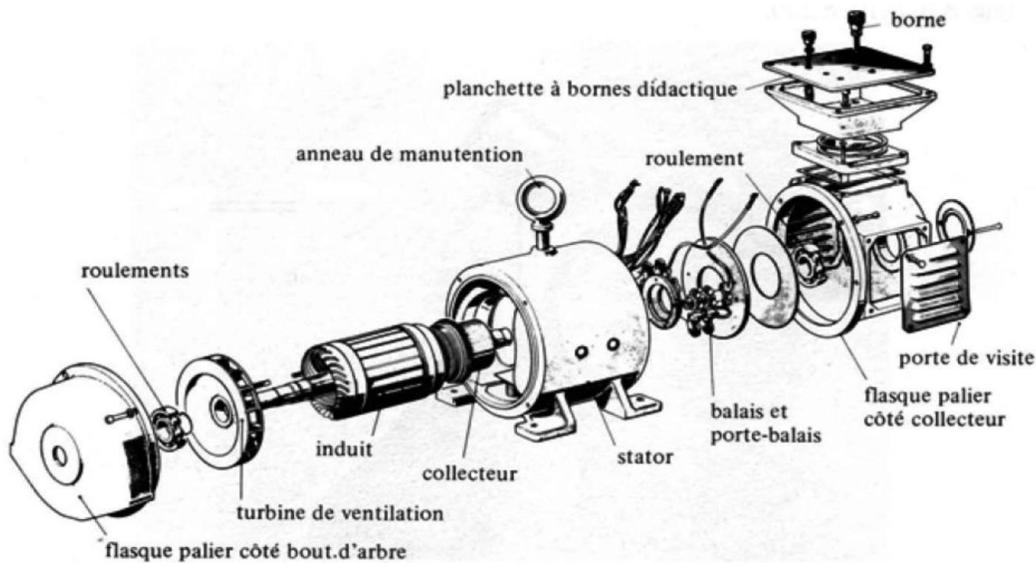


Fig 1.4 Constitution d'une MCC

1.4 Principe de fonctionnement

Lorsque le bobinage de l'inducteur est alimenté, un champ magnétique (flux de champ) est généré dans une direction Nord-Sud, selon le même principe qu'un aimant permanent, qui intercepte le bobinage du rotor. Alors, selon la loi de Laplace, tout conducteur traversé par un courant et placé dans un champ magnétique subit une force. Les conducteurs d'induit placés de part et d'autre de l'axe du balai (fil neutre) sont soumis à une force égale mais opposée (F) qui, en générant un couple moteur, fait tourner l'induit.

1.5 Types de machines à courant continu

Selon la connexion de circuit d'excitation de l'inducteur à celui de l'induit, on distingue ces types de MCC [4] :

Machine à excitation série : les enroulements du stator sont connectés en série aux enroulements du rotor donc tous deux parcourus par un même courant (I_a) et alimentée une tension (V_a).

Machine à excitation shunt (parallèle) : une machine à excitation shunt à l'enroulement de champ connecté en parallèle avec l'enroulement d'induit.

Machine à excitation composée : de telle machine a son enroulement de champ composé d'un enroulement en série et d'un enroulement shunt connectés en parallèle. Cela permet au moteur de combiner les avantages des deux types de machines (à excitation série et à excitation shunt) .

Machine à excitation séparée : ce type de machine à ses enroulements de champ alimentés par une alimentation indépendante de celle de l'induit.

Machine à aimants permanents : On distingue aussi le type de MCC à aimants permanents. Dans ce modèle, le circuit bobiné de l'inducteur est remplacé par des aimants permanents. Notons déjà que c'est ce type de moteur qui sera considéré dans la suite de ce mémoire.

1.6 Avantages et inconvénients de la MCC

Les machines à courant continu présentent certains avantages et inconvénients [5].

1.6.1 Avantages

La machine à courant continu offre de nombreux avantages comme :

Excellent contrôle de la vitesse : les MCC ont une bonne régulation de vitesse, de sorte qu'elles peuvent maintenir une vitesse constante même lorsque la charge change. Cela la rend populaire pour les applications qui nécessitent un contrôle précis de la vitesse.

Couple de démarrage élevé : les MCC peuvent produire un couple de démarrage élevé, ce qui les rend idéales pour les applications nécessitant un couple de démarrage élevé.

Contrôle de la vitesse et du couple : un MCC peut être contrôlé simplement en ajustant le courant dans l'enroulement de champ ou en modifiant la tension appliquée à l'induit, permettant un contrôle précis de la vitesse et du couple.

1.6.2 Inconvénients

Les MCC nécessitent un entretien régulier pour maintenir leurs performances, tels que le remplacement des balais usés et des inspections régulières du collecteur.

- ❖ **Coût élevé :** les MCC sont généralement plus complexes à concevoir et nécessitent des composants tels que des commutateurs et des enroulements de champ, ce qui les rend plus coûteux à fabriquer que les machines à courant alternatif (AC) ;
- ❖ **Génération de bruit :** les MCC génèrent plus de bruit que les machines à courant alternatif car les balais frottent contre le collecteur ;
- ❖ **Perte d'efficacité à haute vitesse :** les MCC ont tendance à être moins efficaces à des vitesses élevées en raison des pertes par frottement et des effets des courants de Foucault.

1.7 Conclusion

La supériorité des machines à courant continu réside dans le fait qu'elles se prêtent facilement à un contrôle souple, simple, optimale, continu et presque instantané de leur vitesse. Elles sont naturellement découplées c'est-à-dire qu'on peut contrôler le couple sans influencer sur le flux et vice versa. Malgré leurs lacunes, les machines à courant continu ont été et demeurent largement utilisées dans diverses applications industrielles en raison de leurs avantages en matière de contrôle de la vitesse et du couple, de leur capacité à produire un couple de démarrage élevé et de leur fiabilité. Elles sont très sollicitées pour les entraînements électriques fonctionnant à vitesse variable.

CHAPITRE 2

MODÉLISATION ET IDENTIFICATION DE LA MACHINE À COURANT CONTINU

CHAPITRE 2

MODÉLISATION ET IDENTIFICATION DE LA MACHINE À COURANT CONTINU

2.1 Introduction

Ce chapitre est réservé à la présentation du moteur à courant continu à aimants permanents (MCCAP) qui est l'objet de réglage de notre étude. Cette étape qui est connue sous l'appellation de la modélisation de la machine permet de présenter la MCCAP par un ensemble d'expressions mathématiques déduites de son comportement physique. Puis, en deuxième étape du chapitre, on procède à l'identification des paramètres de la MCCAP et ce en adaptant une méthode basée sur le calcul et expérimentation. On complète le chapitre par l'étude du comportement du moteur en boucle ouverte, la synthèse du régulateur de type proportionnel - intégral et finalement l'étude en boucle fermée par le réglage de vitesse.

2.2 Modélisation de la MCCAP

La modélisation de ces machines est essentielle pour comprendre leur comportement dans différentes conditions de fonctionnement et pour concevoir des systèmes de contrôle efficaces [7]. La modélisation sert de base pour concevoir des systèmes de contrôle, simuler leurs comportements en régime statique et dynamique et étudier leurs performances.

Cette tâche peut être divisées en deux (2) parties principales :

La modélisation électrique décrit les relations entre les grandeurs électriques de la machine comme tension (U) et courant (I) et la modélisation mécanique représente le mouvement mécanique de la machine, notamment, la vitesse rotorique (Ω) et le couple électromagnétique (C_{em}) que peut développer la machine.

D'une façon globale, une MCC peut être représenté schématique comme le montre la figure 2.1

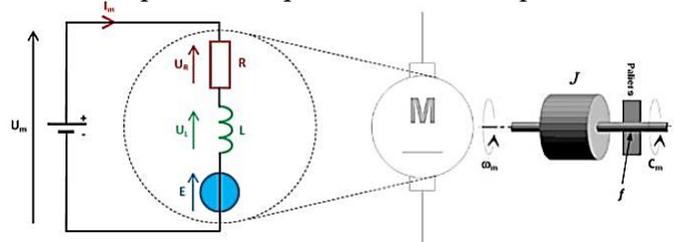


Fig 2.1 Représentation schématique d'une MCC [6]

Ainsi, on peut déduire les relations suivantes :

- **Partie électrique**

$$U_{ind} = U_r + U_L + E \quad (2.1)$$

Soit,

$$U_{ind} = R_{ind} \cdot I_{ind} + L_{ind} \frac{dI_{ind}}{dt} + E \quad (2.2)$$

Avec,

U : tension d'alimentation d'induit (V) ;

E : force contre-électromotrice (V) ;

R : résistance d'induit (Ohm) ;

I : courant d'induit (A) ;

L_{ind} : inductance d'induit (H) .

Par ailleurs, la force contre-électromotrice est donnée par l'expression suivante :

$$E = k \cdot \Phi_{ext} \cdot \Omega \quad (2.3)$$

Où,

k : coefficient de construction de la machine ;

Φ_{ext} : flux d'excitation (constant dans le cas d'un inducteur à aimants permanents) ;

Ω : vitesse de rotation (rad/s) ;

Cette expression est aussi appelée « caractéristiques électriques d'un moteur à courant continu ».

- **Partie mécanique**

Elle est exprimée par l'équation dynamique dite de mouvement représentée ci-dessous :

$$J \cdot \frac{d\Omega}{dt} = C_{em} - C_r - k_f \cdot \Omega \quad (2.4)$$

Avec,

J : moment d'inertie (Kg.m^2) ;

C_{em} : couple électromagnétique (Nm) ;

C_r : couple résistant du à la charge (Nm) ;

k_f : coefficient de frottement crée par la rotation du moteur ;

Ω : vitesse de rotation (rad/s) .

Le couple électromagnétique (C_{em}) s'exprime par la relation suivante :

$$C_{em} = k \Phi_{ext} I_{ind} \quad (2.5)$$

En considérant $\Phi_{ext} = \text{constante}$ (cas de MCC à aimants permanents), on peut écrire

$$C_{em} = K I_{ind} \quad (2.6)$$

Avec $K = k \Phi_{ext}$

En utilisant la transformée de Laplace [6], on obtient :

$$\text{De (2.2), on a} \quad U(S) = R_{ind} . I_{ind} (S) + L_{ind} . S . I_{ind} (S) + E (S) \quad (2.7)$$

$$\text{De (2.3), on a} \quad J . S . \Omega (S) = C_{em} (S) - C_r - k_f \quad (2.8)$$

Avec,

$$E(S) = K . \Omega (S) \quad (2.9)$$

D'où,

$$C_{em} (S) = K . I_{ind}(S) \quad (2.10)$$

$$C_{em} (S) - C_r = J . S . \Omega (S) + k_f . \Omega \quad (2.11)$$

En tenant compte des relations sous forme de Laplace définies précédemment , on peut représenter le schéma bloc générale du MCC montré par la figure 2.2

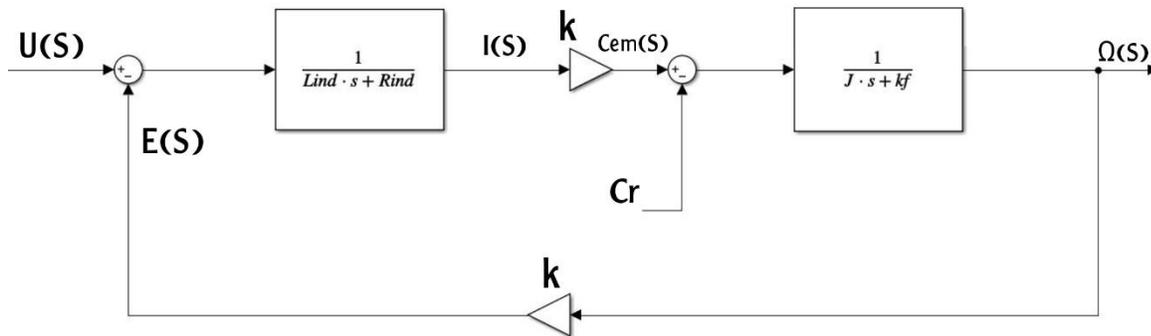


Fig 2.2 Schéma bloc du MCC

Ce schéma montre les deux parties du MCC : la partie électrique et la partie mécanique. Elles dépendent des paramètres (R_{ind} , L_{ind} , J , K et k_f) qui sont des données propres de la MCC. Cependant, conformément à notre sujet de mémoire qui nécessite la connaissance des valeurs de ces paramètres pour pouvoir analyser significativement, les performances du contrôle de vitesse du moteur lors de la réalisation expérimentale, on a procédé à l'identification des paramètres du moteur comme le présente les paragraphes suivants :

2.3 Identification des paramètres de la MCC

Concernant ce travail, nous avons utilisé une MCC à aimants permanents de petite puissance en vue de lui appliquer des techniques de commande de vitesse et les valider expérimentalement. La figure 2.3 montre une photo de la MCC à aimants permanents utilisé dans la partie réalisation.



Fig 2.3 Photo du MCC utilisé

2.3.1 Méthode d'identification

Cette partie expose la technique d'identification des paramètres du moteur utilisé. Rappelons, que plusieurs méthodes d'identification ont été développées et utilisées à cette fin [7]. En ce qui nous concerne, nous sommes inspirés du paragraphe de modélisation et il s'est avéré qu'on besoins des valeurs des paramètres : résistance rotorique (R_{ind}), inductance rotorique (L_{ind}), moment d'inertie (J), le coefficient de construction de la machine (K) et coefficient de frottement (k_f). Ceci étant, il nous a été facile à déterminer la valeur du R_{ind} expérimentalement [8]. Après avoir effectué les cinq (5) mesures de la résistance d'induit (voir tableau 2.1).

Mesure de la résistance (R_{ind}): nous avons mesuré plusieurs fois et à des temps séparés la résistance à l'aide du multimètre. Les résultats mesurés sont représentés dans le tableau suivant :

N^0 de la mesure	1	2	3	4	5
Valeurs, (en ohm)	395.4	394.3	395.7	395	394.6

Tableau 2.1 Résultats de la mesure de la résistance

On a déterminé la valeur moyenne correspondante à $R_{ind} = 395 \Omega$.

Concernant le reste des paramètres, nous avons profité du ToolBox de MatLab et précisément l'application « *Parameter Estimator* » pour les déterminer [9]. Toutefois, il est essentiel pour cette application de lui introduire en données numériques l'évolution en boucle ouverte de la vitesse en fonction du temps. Ces deux grandeurs mesurées seront utilisées comme deux (2) variables de données comme l'explique et l'expose la partie ci-dessous. Pour commencer, la figure 2.4 présente l'évolution de la vitesse en fonction du temps que nous avons enregistré en utilisant le moteur que nous avons utilisé.

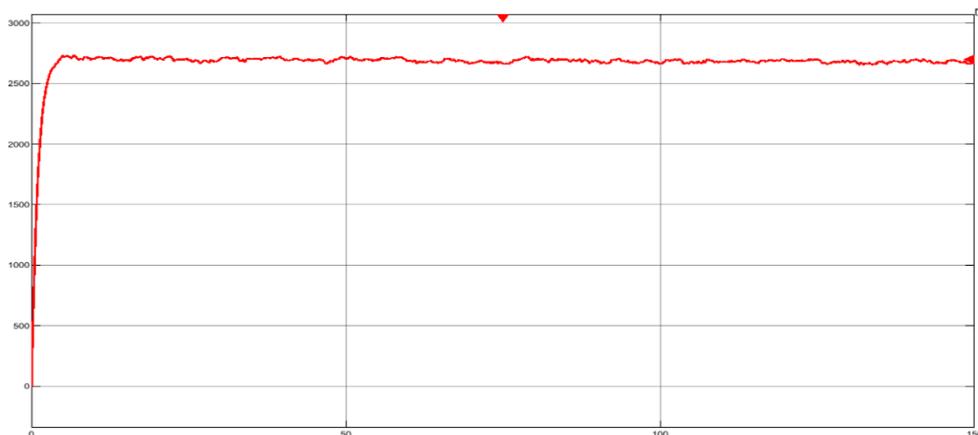


Fig 2.4 L'évolution de la vitesse en fonction du temps du moteur utilisé

PARAMETER ESTIMATOR

L'application « *Parameter Estimator* » du logiciel MatLab a pour rôle d'identifier et d'estimer avec une grande précision les paramètres d'un système à l'aide de données mesurées. Pour cela, il suffit d'importer et prétraiter les données mesurées, trouver les paramètres les plus influents à identifier puis exécuter l'application et surveiller la progression de l'identification afin de finalement valider les résultats [10].

Cette application de MatLab utilise des algorithmes d'optimisation. Cependant, l'algorithme qui sera utilisé dépend des options que nous sélectionnons dans l'application. Parmi ces algorithmes, on y trouve :

- ❖ **Algorithme de Levenberg-Marquardt** : cet algorithme est une modification de l'Algorithme de Gauss-Newton pour les problèmes de moindres carrés non linéaires. Il introduit un facteur d'amortissement, qui le rend plus robuste face à de mauvaises estimations initiales des paramètres.
- ❖ **Algorithme Trust-Region** : c'est un autre algorithme pour les problèmes de moindres carrés non linéaires. Il fonctionne en définissant une région autour de l'estimation actuelle des paramètres et ne recherche qu'à l'intérieur de cette région. La taille de la région est ajustée en fonction de la qualité de l'ajustement du modèle aux données.
- ❖ **Algorithme Sequential Quadratic Programming (SQP)** : c'est une méthode itérative pour les problèmes d'optimisation contraints. À chaque itération, il résout un sous-problème de programmation quadratique pour trouver une direction de pas, puis utilise une recherche de ligne pour trouver une longueur de pas.

Ces algorithmes sont bien adaptés à l'estimation de paramètres car ils peuvent gérer des modèles non linéaires et sont conçus pour minimiser la somme des carrés, qui est la fonction de coût habituelle dans les problèmes d'estimation de paramètres. Cependant, les performances de ces algorithmes peuvent dépendre grandement des spécificités du problème, comme la qualité de l'estimation initiale des paramètres, la complexité du modèle, et la quantité et la qualité des données.

Alors, en ce qui concerne notre travail, nous avons enregistré les valeurs mesurées de la vitesse et le temps de l'expérimentation réalisé sur le moteur qu'allons utiliser dans partie expérimentale. On définit les valeurs du temps et vitesse mesurée comme deux (2) variables (A) et (B), soit :

A = temps (: , 1)
B = Vitesse (: , 2)

Nous ouvrons la fenêtre des applications dans MatLab et sélectionnons l'application (estimateur de paramètres) comme le montre la figure suivante :



Fig 2.5 Application Parameter Estimator

Après cela, on sélectionne l'icône [*nouvelle expérience*] alors une demande pour les sorties mesurées et simulées ainsi que les paramètres que nous désirons identifier (figure 2.6 et 2.7).

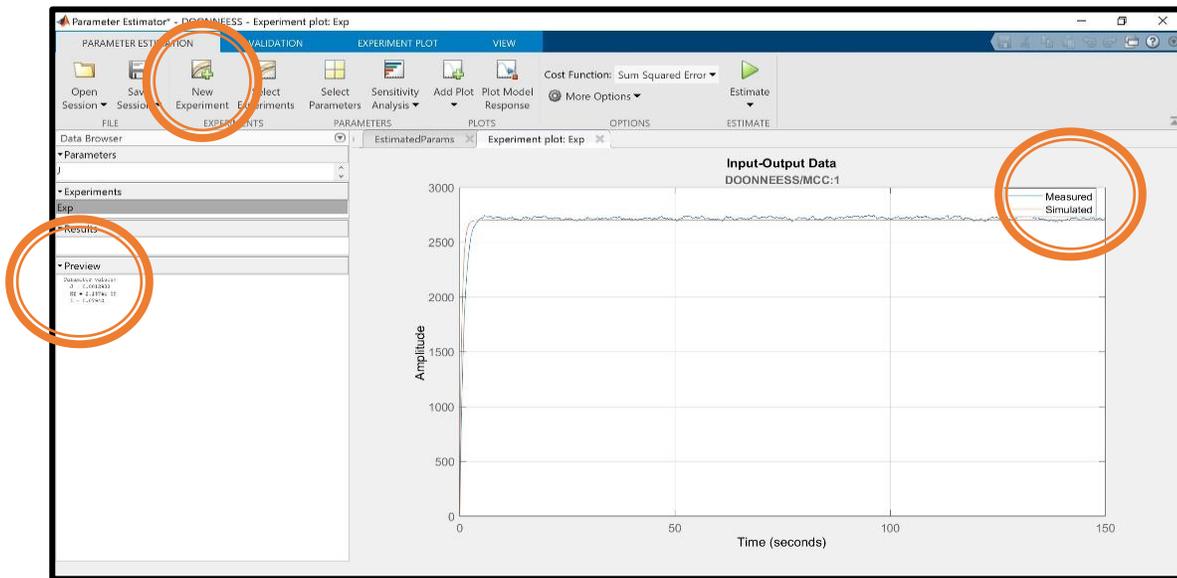


Fig 2.6 Création du nouvelle expérience

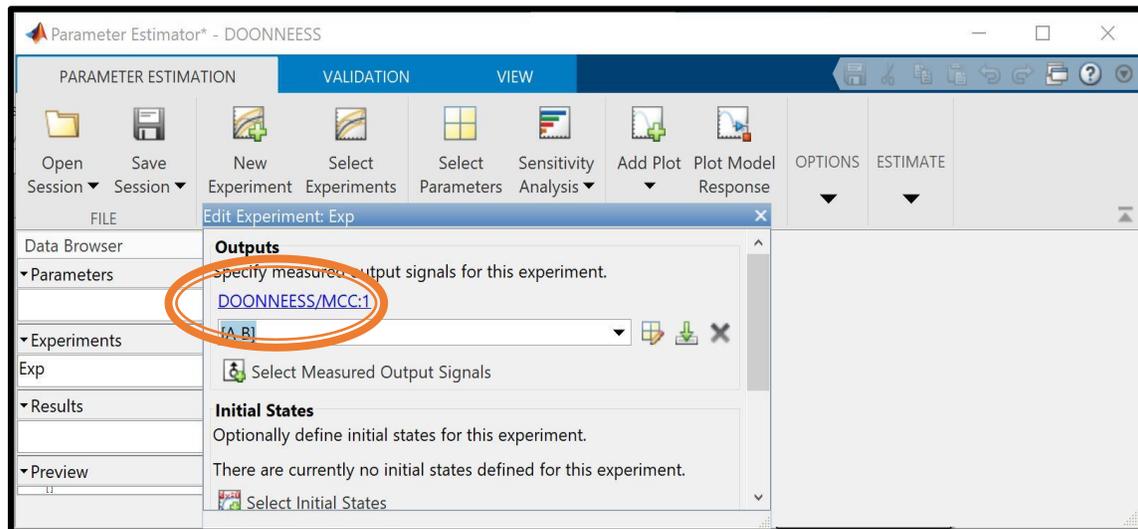


Fig 2.7 Sélection des sorties mesurées et estimées

Dans « *outputs* » nous sélectionnons la sortie simulée et entrons les valeurs mesurées (A, B), on s’assure que les paramètres estimés sont totalement positifs comme la figure 2.8.



Fig 2.8 Paramètres estimés positifs

A cette étape, on commence l'opération de l'identification en cliquant sur [estimer] et attendons que le processus se termine. Après cela, les paramètres seront affichés dans les résultats sont ceux estimés. Notons également que la courbe simulée aura à la fin de l'opération, la même forme que celle mesurée comme le montre la figure ci-dessous :

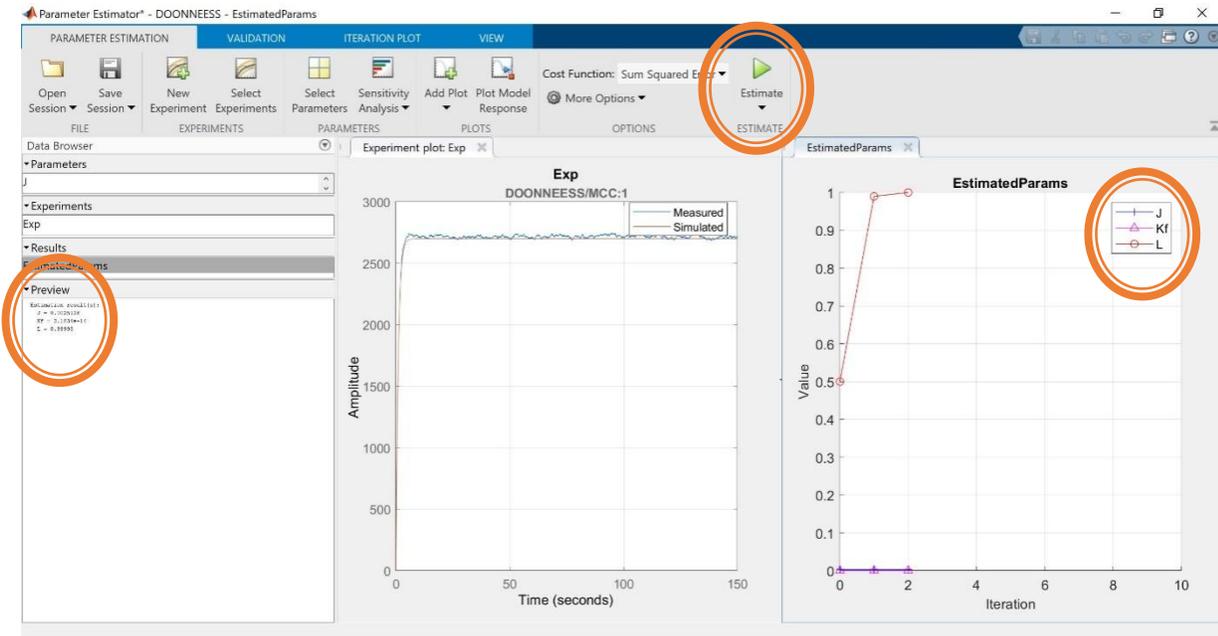


Fig 2.9 Application d'estimation et obtention des résultats

Finalement, à l'issue de l'identification du moteur, les valeurs obtenue des paramètres sont consignées dans le tableau 2.2

R_{ind}	L_{ind}	J	k_f
395	0.9998	0.0025126	0.000000008103

Tableau 2.2 Valeurs identifiées du moteur

2.4 Simulation et discussion du MCC en boucle ouverte

Ce paragraphe comprend les simulations réalisées sous MatLab / Simulink. A cet effet, nous avons commencé à tester le modèle du moteur sous une tension d'alimentation et couple variables. Les figures 2.10.A et 2.10.B montre les résultats de simulation.

Simulation à vide

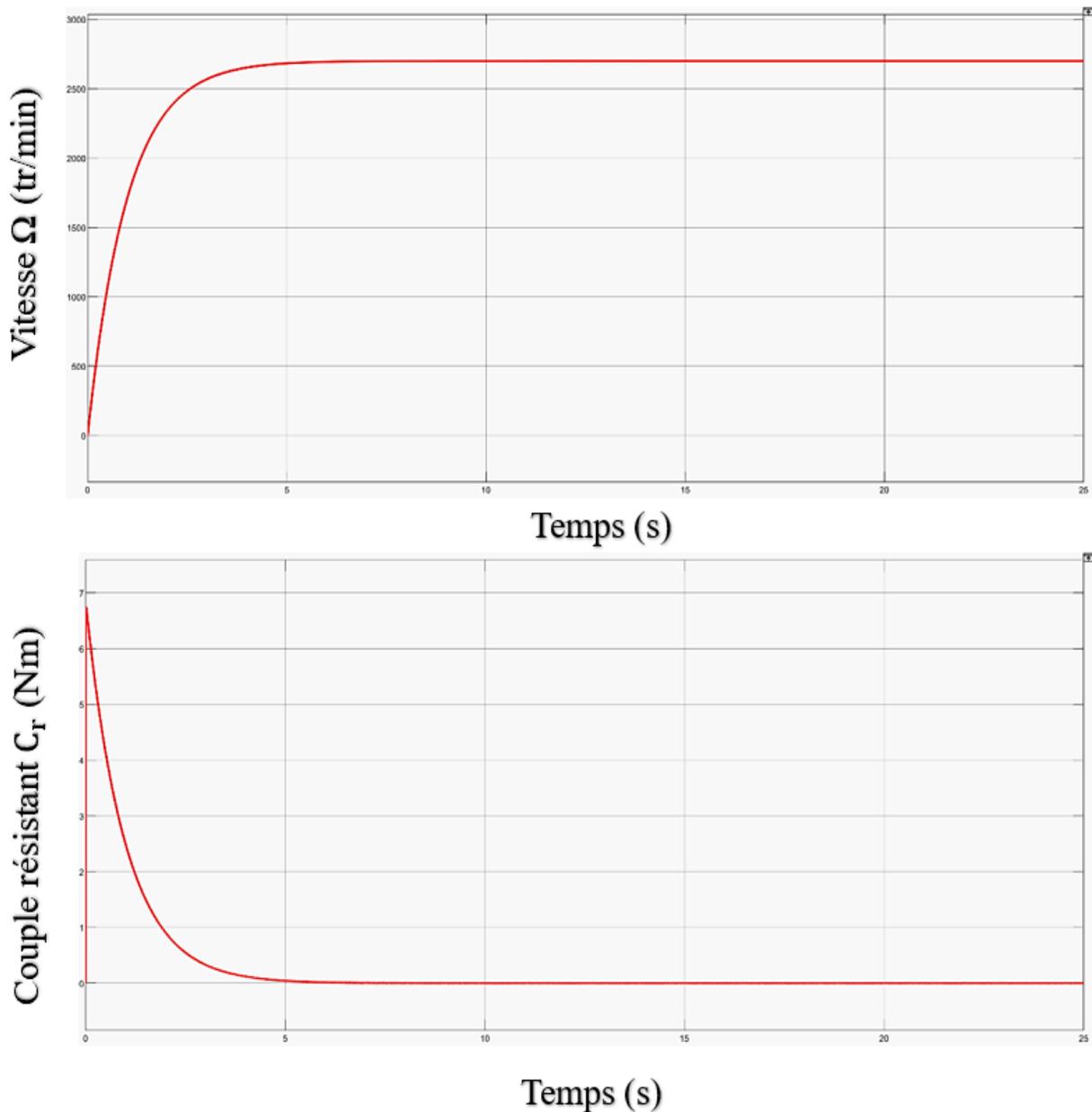


Fig 2.10.A Résultats de simulation du modèle de la MCC en BO à vide

✚ Simulation en charge

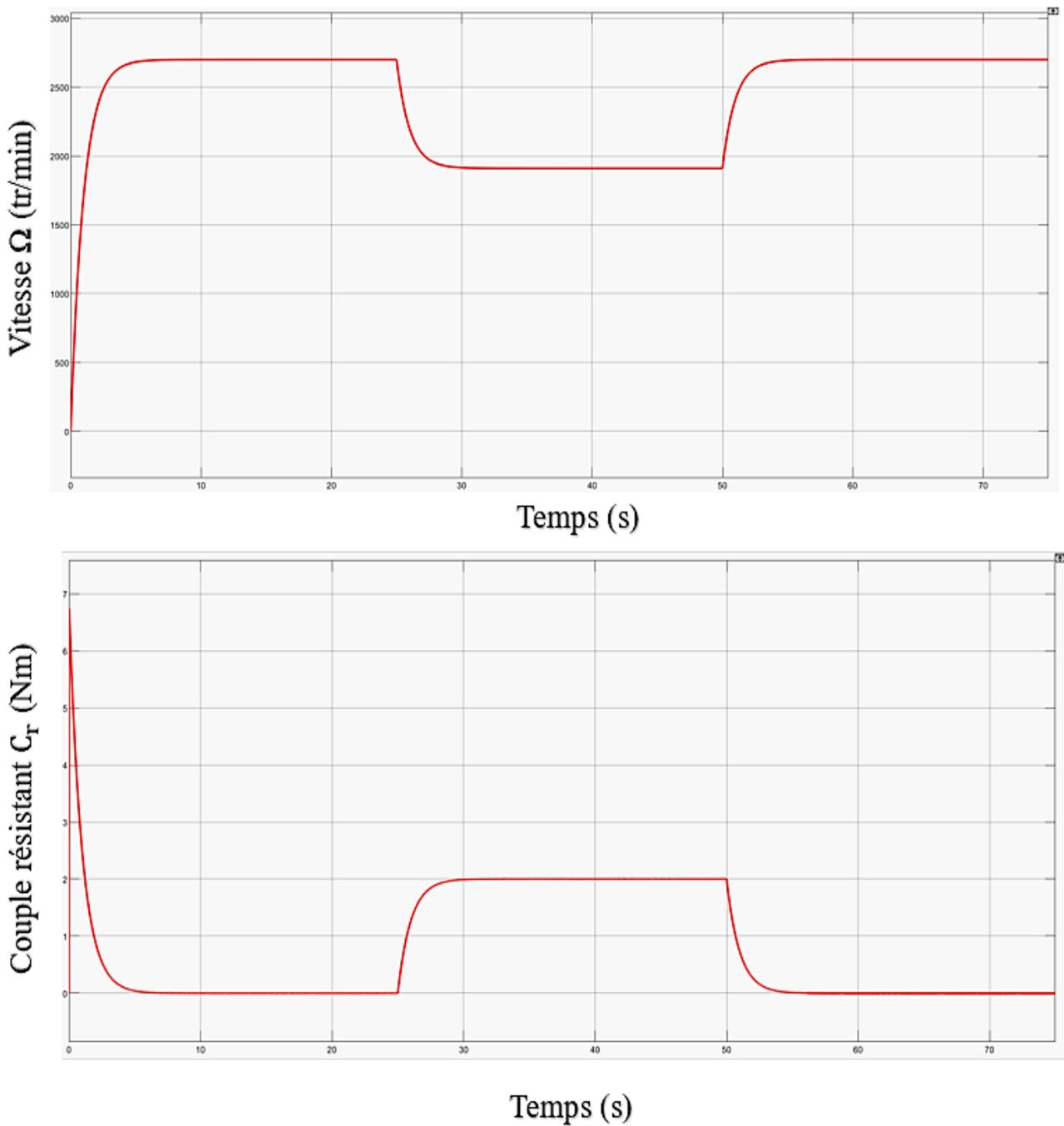


Fig 2.10.B Résultats de simulation du modèle de la MCC en BO en charge.

Les figures précédentes représentent les résultats de simulation du MCC en boucle ouverte à vide et en charge, définis par la vitesse, le courant et le couple. Pendant le régime transitoire, on peut observer que la vitesse est d'une forme apériodique avant de se stabiliser à la vitesse désirée en régime permanent. Il est important de noter que la réponse du courant est la même que la réponse du couple à vide, car le couple est l'image du courant selon la relation $C_e = K.I$. Alors, pendant le démarrage, il est bien remarqué que les deux expriment par un pic fort puis il se stabilise à titre d'exemple un couple de charge de deux (2) N.m à partir de la 25^{ème} seconde, on remarque que la valeur du couple augmente et se stabilise à cette valeur, puis quand dans le cas d'annulation de la charge, la valeur couple diminue et revient à sa valeur initiale aussi.

Par la suite nous avons testé par simulation le comportement du moteur sous l'effet du changement du sens de rotation (inversion) et dont la simulation sont montées par la figure 2.11

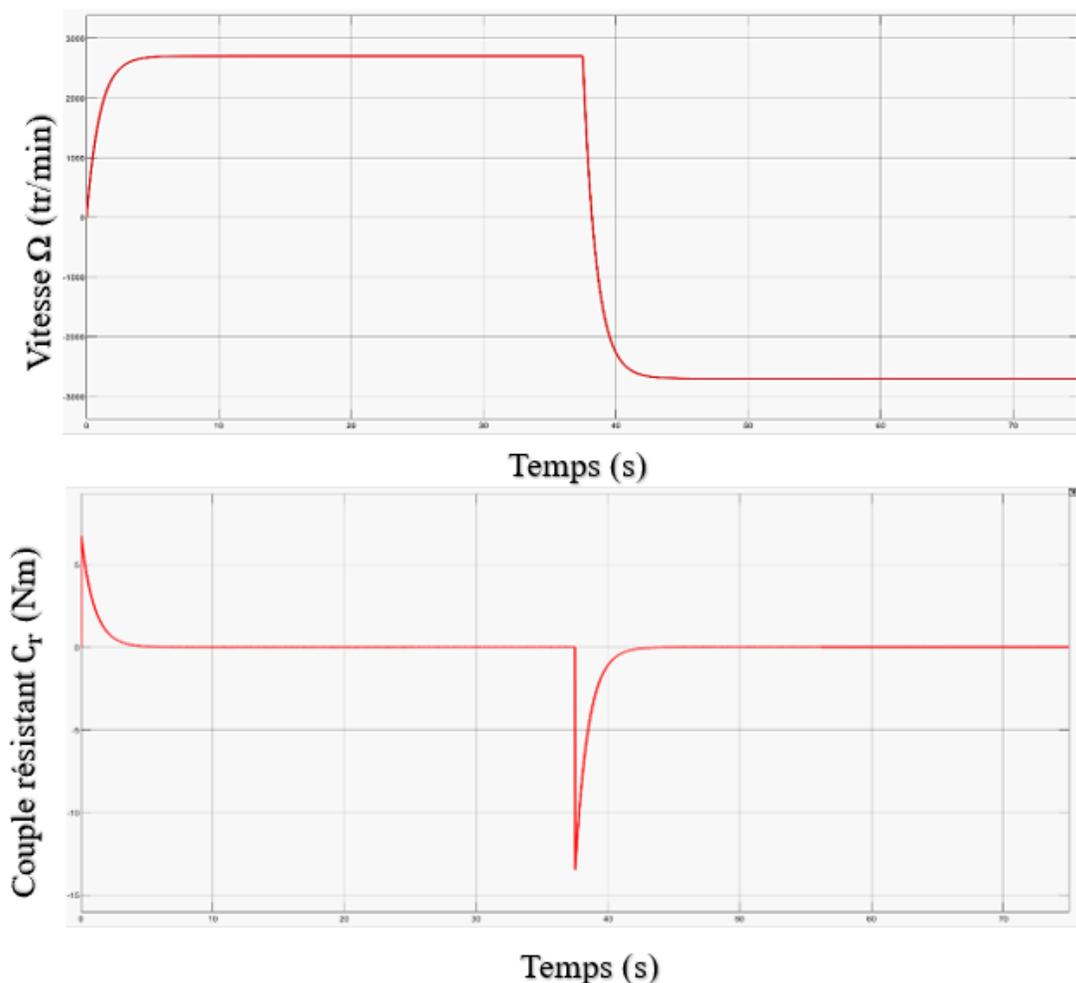


Fig 2.11 Démarrage et inversion de sens de rotation

L'analyse de ces résultats de simulation, nous fait remarqué que le sens de rotation s'inverse dans l'instant ($t = 37.5$ s), la vitesse s'arrête puis recommence mais dans l'autre sens. Pour le courant, sa valeur devient le double de sa valeur initiale dans le démarrage dans la période d'inversement revient à sa valeur initiale.

2.5 Synthèse du régulateur de vitesse

La régulation de la vitesse du moteur à courant continu est réalisée en contrôlant la tension d'alimentation. Pour réguler la vitesse, elle doit être mesurée avec un capteur de vitesse tel qu'un tachymètre (pour le cas de ce travail, nous avons utilisé dans la partie expérimentale un capteur de vitesse *HC-020K*).

La vitesse mesurée est ensuite comparée à la consigne de vitesse souhaitée pour déterminer l'erreur de régulation. Une fois l'erreur calculée, un signal de commande est généré pour commander un hacheur qui peut ajuster la tension d'alimentation à sa sortie pour alimenter le MCC.

La régulation de la vitesse des moteurs à courant continu est généralement réalisée à l'aide d'un système de contrôle en boucle fermée qui surveille en permanence la vitesse et ajuste la tension d'alimentation pour maintenir la vitesse souhaitée. Les systèmes de contrôle en boucle fermée peuvent être mis en œuvre à l'aide de différents algorithmes, tels qu'un contrôleur Proportionnel-Intégral-dérivé (PID).

2.5.1 Régulateur PID

Un régulateur proportionnel-intégral-dérivé (PID) est un algorithme couramment utilisé dans les systèmes de contrôle de vitesse, il utilise trois termes [11]

- ✚ Le terme proportionnel (P) ;
- ✚ Le terme intégral (I) ;
- ✚ Le terme dérivé (D) ;

Ces termes sont utilisés pour régler la tension d'alimentation du moteur et maintenir la vitesse souhaitée.

- ❖ **Terme proportionnel (P) :** Ce terme est calculé en multipliant l'erreur de régulation (différence entre la vitesse mesurée et la vitesse de consigne) par le facteur de proportionnalité k_p , il ajuste la tension d'alimentation en fonction de l'erreur de régulation du courant, donc plus l'erreur est grande, plus l'ajustement est important.

La description temporelle du régulateur proportionnel est alors la suivante : en tant que :

$$U(t) = k_p \cdot \varepsilon(t) \quad (2.10)$$

Avec,

- + U(t) : signal de commande ;
- + k_p : facteur de proportionnalité ;
- + $\varepsilon(t)$: l'erreur de régulation .

Ce qu'en transformée de Laplace donne :

$$U(S) = k_p \cdot \varepsilon(S) \quad (2.11)$$

➤ **Avantages:**

- Correction de manière rapide ;
- Permet de vaincre les grandes inerties du système ;
- Augmentation du gain.

➤ **Inconvénients :**

- Amplification dans toute la bande de fréquence ;
- Ne contrôle pas la précision.

❖ **Terme intégral (I) :** est calculé en intégrant l'erreur de commande sur une période de temps et en multipliant le résultat par le facteur intégral k_i . Le terme intégral ajuste la tension d'alimentation en fonction de l'erreur de régulation accumulée au fil du temps, donc si l'erreur persiste pendant une longue période, l'ajustement sera important.

Comme pour la composante proportionnelle, une division trop importante de la composante (I) peut causer une instabilité dans la boucle de régulation. Il est également important de trouver un compromis entre stabilité et vitesse lors de la mise en place.

Donc la description temporelle du régulateur intégrateur est alors la suivante :

$$U(t) = \frac{1}{T_i} \int_0^t \varepsilon(t) dt \quad (2.12)$$

Avec,

T_i : constante d'intégration.

Ce qu'en transformée de Laplace donne :

$$U(S) = k_i \cdot \frac{\varepsilon(S)}{S} \quad (2.13)$$

❖ **Terme dérivé (D) :** L'action dérivée va ainsi intervenir uniquement sur la variation de l'erreur ce qui augmente la rapidité du système (diminution des temps de réponse) et permet aussi d'augmenter la stabilité du système). L'annulation de cette action en régime statique impose donc de ne jamais l'utiliser seule et n'exerce qu'un complément à l'action proportionnelle.

Donc, la description temporelle du régulateur dérivateur est donnée par :

$$U(t) = k_d \cdot \frac{d\varepsilon(t)}{dt} \tag{2.14}$$

Ce qu'en transformée de Laplace donne :

$$U(S) = k_d \cdot S \cdot \varepsilon(S) \tag{2.15}$$

D'après les équations (2.11), (2.13), (2.15), on obtient la forme générale d'un régulateur PID.

Soit :

$$\begin{cases} U(S) = k_p \cdot \varepsilon(S) + k_i \cdot \frac{\varepsilon(S)}{S} + k_d \cdot S \cdot \varepsilon(S) \\ U(S) = \varepsilon(S) \left(k_p + \frac{k_i}{S} + k_d \cdot S \right) \end{cases} \tag{2.16}$$

2.5.2 Synthèse

Dans cette partie , nous avons présenté la régulation du modèle du moteur avec un régulateur PID pour obtenir les performances désirées , c'est-à-dire le système est on boucle fermée comme le montre la figure ci-dessous :

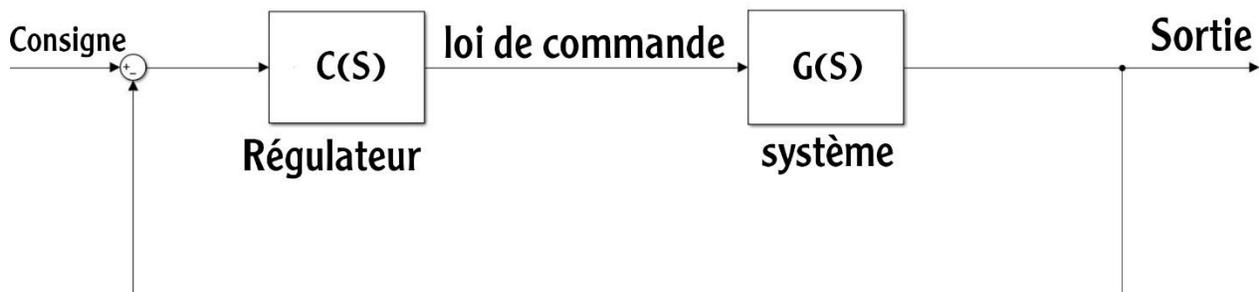


Fig 2.12 Structure du système en boucle fermée

La fonction de transfert en boucle fermée de ce système est donnée par :

$$H(p) = \frac{C(S) \cdot G(S)}{1 + C(S) \cdot G(S)} \quad (2.17)$$

Où,

C(S) : Fonction de transfert de régulateur (cas d'un régulateur PI) ;

G(S) : fonction de transfert de l'objet réglé.

D'après l'équation (2.17) on définit que :

$$\left\{ \begin{array}{l} C(S) = k_p + \frac{ki}{s} = \frac{Kp S + ki}{s} \\ G(S) = \frac{K}{(LS+R)(JS+k_f) K^2} \end{array} \right. \quad (2.18) (2.19)$$

En remplace les équations (2.18) et (2.19) dans l'équation (2.17) :

$$H(S) = \frac{\frac{KpS+Ki}{s} \cdot \frac{k}{(LS+R)(JS+k_f) K^2}}{1 + \frac{KpS+Ki}{s} \cdot \frac{k}{(LS+R)(JS+k_f) K^2}} \quad (2.20)$$

$$H(s) = \frac{k (KpS+Ki)}{s((LS+R)(JS+k_f)+K^2)+k(KpS+Ki)} \quad (2.21)$$

En négligée l'inductance (L) :

$$H(s) = \frac{k (KpS + Ki)}{S(RJS + Rk_f + K^2) + k.KpS + kKi} \quad (2.22)$$

$$H(s) = \frac{Ke (KpS + Ki)}{RJ(S^2 + \frac{Rk_f + Ke^2 + KeKp}{RJ} S + \frac{KeKi}{RJ})} \quad (2.23)$$

On peut écrire le dénominateur de l'équation (2.23) comme une équation de deuxième ordre pour déterminer les coefficients k_p et k_i :

$$S^2 + \frac{Rk_f + K^2 + K.Kp}{RJ} S + \frac{K.Ki}{RJ} = S^2 + 2 \zeta \omega_0 S + \omega_0^2 \quad (2.24)$$

Avec

ω_0 : pulsation naturelle ;

ζ : coefficient d'amortissement.

Donc, on peut déduire les valeurs de k_p et k_i comme suit :

$$\left\{ \begin{array}{l} k_p = \frac{2RJ\zeta\omega_0 - RKf - K^2}{K} \\ k_i = \frac{\omega_0^2 RJ}{K} \end{array} \right. \quad (2.25)$$

Pour un système du second ordre sous-amorti, le temps de montée (Tr) peut être approximativement lié à (ω_0) par la formule :

$$Tr = \frac{\pi}{\omega_0 \sqrt{1 - \zeta^2}} \quad (2.26)$$

Pour un temps de montée de 2 seconde et un coefficient d’amortissement de 0,7, on obtient $\omega_0 \approx 2.199554$ rad/s. Le dépassement maximal (M_p) peut être approximativement lié à coefficient d’amortissement (ζ) par l’équation (2.23).

$$M_p = e^{\frac{-\pi\zeta}{\sqrt{1-\zeta^2}}} \tag{2.27}$$

Alors, pour un dépassement maximal de 5%, on obtient $\zeta \approx 0,7$.

Après le calcul, en déduit les valeurs de k_p et k_i comme suit :

$$\begin{cases} k_p = 2.05623 \\ k_i = 4.801641 \end{cases}$$

2.5.3 PID tuner

Il existe une autre méthode pour configurer les coefficients k_p et k_i afin d’obtenir une performance optimale du système commandé. On peut trouver cette méthode dans MatLab, connue sous l’appellation ‘PID tuner’. C’est une fonction Matlab qui permet de concevoir et d’ajuster des contrôleurs PID pour différents systèmes. Aussi, C’est une interface graphique conviviale qui permet à l'utilisateur de régler les paramètres PID de manière interactive en utilisant des analyses temporelles et fréquentielles[12]

Nous sélectionnons le bloc (PID Controller) dans Matlab comme le montre la figure suivante :

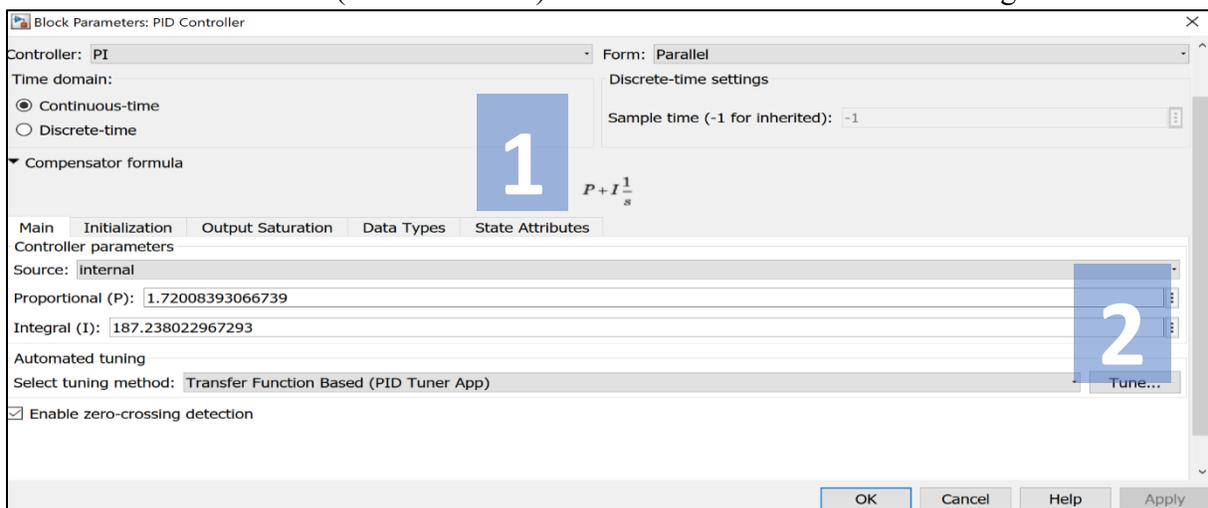


Fig 2.13 Sélection du régulateur PID puis la fonction « Tune»

Dans cette partie, nous présenterons les étapes du réglage d'un contrôleur PI, l'interface et les différents paramètres avec cette méthode :

Tout d'abord, nous pouvons choisir le type de régulateur utilisée (P, PI, PID) (1), après nous cliquons sur l'icône apparente (tune) (2). On peut optimiser la réponse temporelle du système et son comportement transitoire.

La réponse temporelle se réfère à la rapidité avec laquelle le système atteint sa valeur de consigne après une perturbation, tandis que le comportement transitoire décrit comment la sortie varie en réponse à une perturbation ou un changement de la consigne (3).

Nous comparerons entre la réponse du bloc (block response) et la réponse réglée (tuned response) et ajusterons la réponse optimale (4). Lorsque le processus de réglage est terminé, nous cliquons sur le bouton de mise à jour (update block) pour déplacer les paramètres vers le bloc PI dans Simulink. (5) comme le montre la figure ci-dessous :

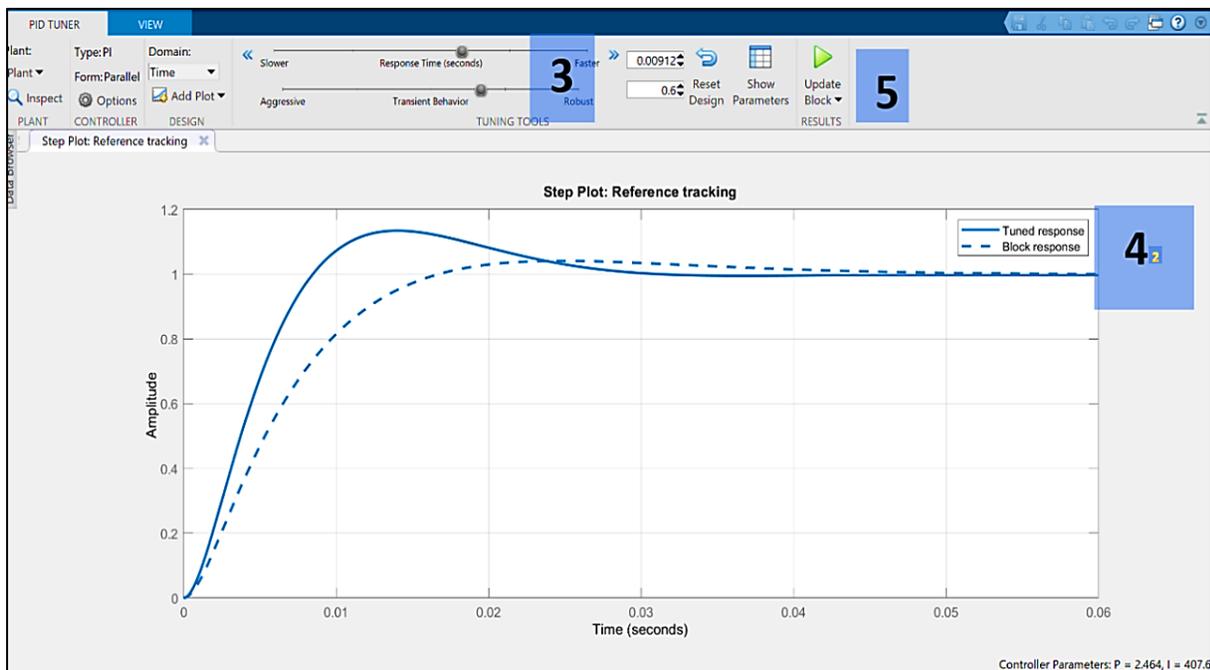


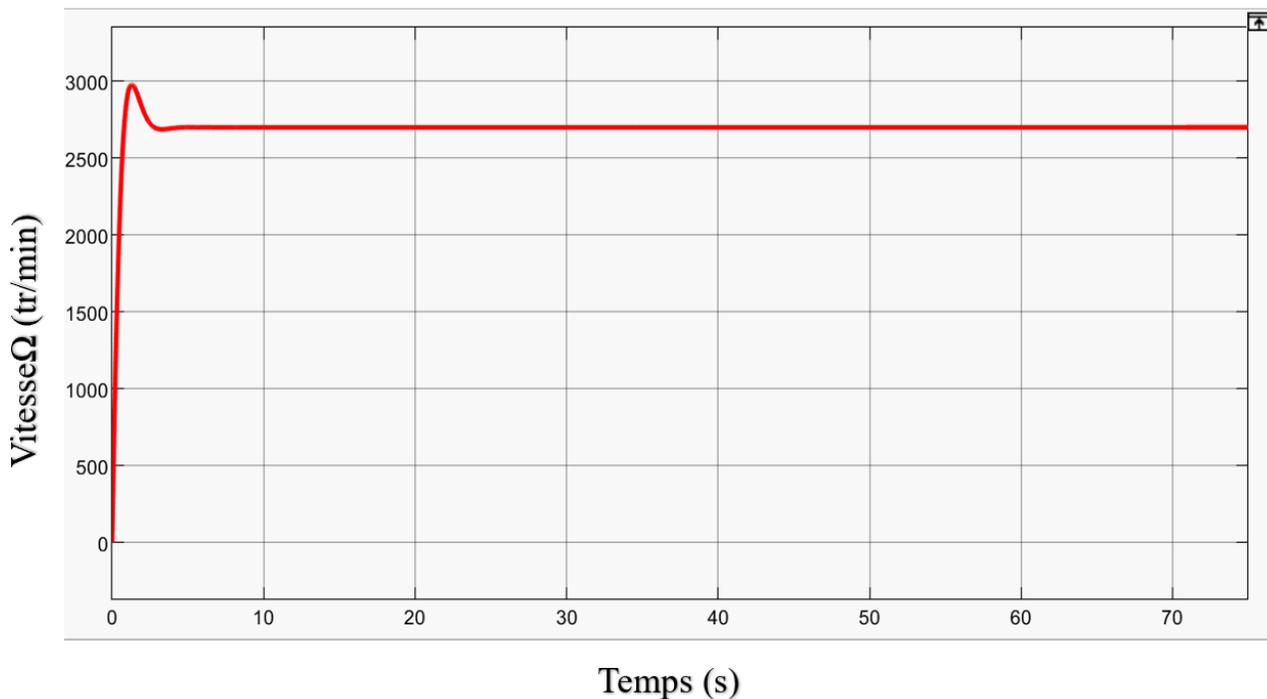
Fig 2.14 Configuration des valeurs désirée pour k_p et k_i

2.6 Simulation et discussion du MCC en boucle fermée

À la suite de l'utilisation de l'application "Tunning" du MatLab nous avons ajusté les paramètres selon le choix prédéfini et désirée de nos performances. Ces paramètres ont été introduit dans notre schéma de simulation et ceci nous a permis de visualiser les évolutions de vitesse, courant et du couple pour une fonctionnement à vide, comme le montre la figure 2.4.A. De même, le comportement, lors de l'application d'une charge de 2 Nm appliqué à l'instant 25 seconde est montré par la figure 2.4.B. On constate que la régulation rejette l'effet de perturbation et fait revenue la vitesse à sa valeur de référence.

Dans la partie de simulation précédente, nous avons tester le modèle du moteur en boucle ouverte, donc maintenant pour assurer la meilleure performance et obtenir les résultats désirés on peut le testé à l'aide d'un régulateur PI. La figure 2.14 montre les résultats de simulation en boucle fermée.

Simulation à vide



1

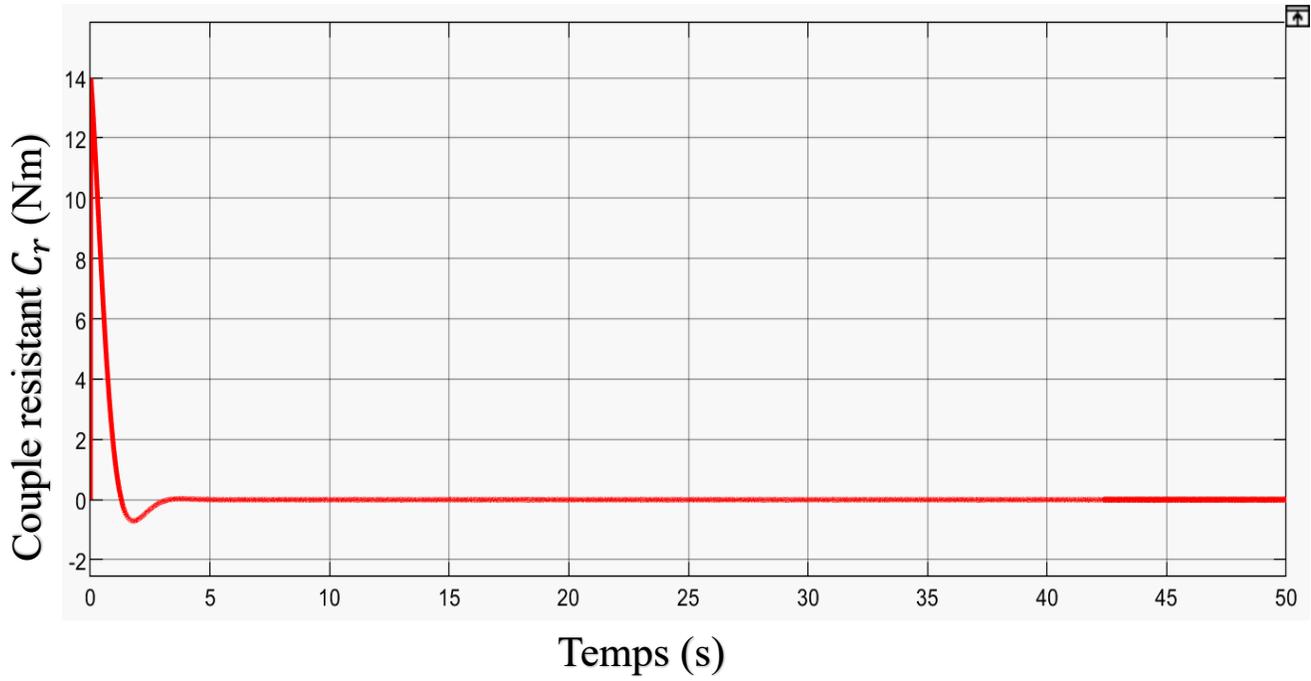
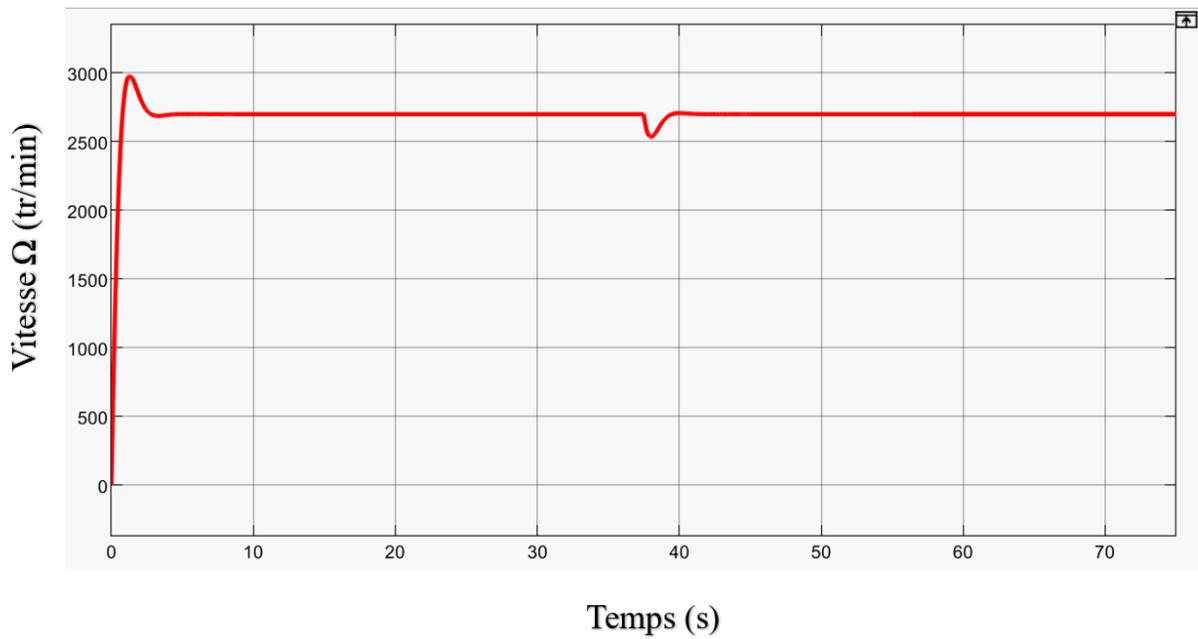


Fig 2.14.1 Résultats de simulation de la MCC en boucle fermée à vide.

 Simulation en charge



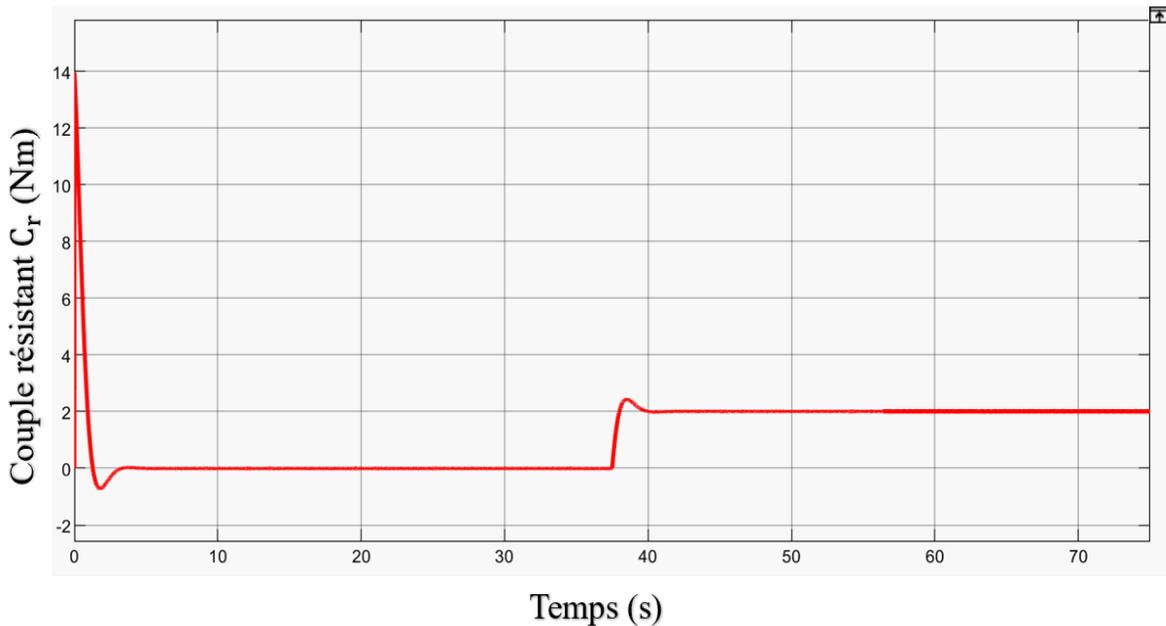


Fig 2.14.2 Résultats de simulation de la MCC en boucle fermée en charge.

2.7 Conclusion

Ce chapitre a regroupé des parties fondamentales de la réalisation de ce travail, notamment, la partie modélisation qui parmi la mise en équation du modèle du moteur à courant continu à aimant permanent, puis la partie théorique qui a réservé pour l'identification des paramètres de la machine en vue de l'utiliser dans la partie expérimentale. Enfin, pour un fonctionnement de vitesse, la méthode par compensation des pôles a été utilisée pour déterminer les valeurs des paramètres du moteur grâce auxquelles l'analyse du comportement a été discuté.

Cependant, en vue d'améliorer les performances de réglage de vitesse, le régulateur PI sera remplacé par un régulateur RST dans le prochain chapitre.

CHAPITRE 3

COMMANDE POLYNOMIALE RST

CHAPITRE 3 COMMANDE POLYNOMIALE RST

3.1 Introduction

Conformément au cahier de charge de ce travail, ce chapitre est réservé aux notions sur la commande polynômiale (RST) et la technique de sa synthèse en vue de son application pour le contrôle de vitesse du MCCAP.

La méthode de régulation RST est mise en œuvre avec l'objectif d'optimiser les performances du contrôle. Son fondement repose sur la résolution de l'équation de Bézout qui mène à l'identification des polynômes R, S et T [13]. Ce processus permet de limiter considérablement les effets perturbateurs et d'obtenir d'excellentes performances de compensation tout en assurant une robustesse remarquable.

La discrétisation est par définition une technique qui permet de convertir des données continues en données discrètes, c.à.d. à échantillonner une fonction continue à des intervalles réguliers [14]. Ceci, simplifie la manipulation et le traitement de données et d'effectuer des calculs numériques.

Une des applications courantes de la discrétisation est La transformation en Z qu'est une technique de traitement de signaux numériques utilisée pour la discrétisation.

La figure (3.1) représente la conception par modèles du systèmes continus et discrets.

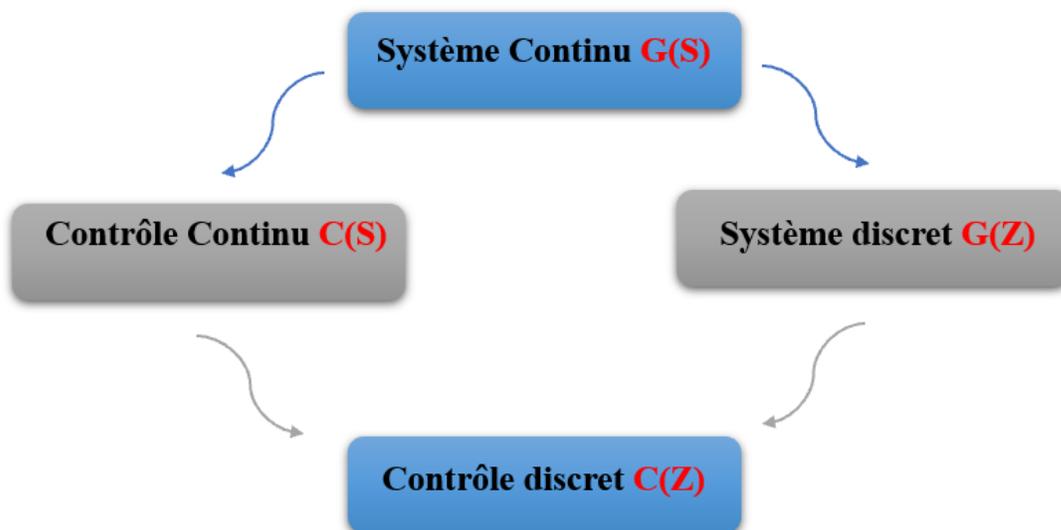


Fig 3.1 Conception par modèles continus et discrets

Système continu : un système où le temps et la variable varient de manière continue.

Système discret : un système où le temps et la variable évoluent de manière discrète, généralement à des intervalles de temps réguliers.

- **Temps de décalage vers l'avant**

Il consiste à anticiper et à compenser les perturbations futures en incorporant des prédictions dans le système de contrôle et annuler de manière proactive les effets des perturbations avant que la sortie du système soit effectuée.

Il est exprimé par Z et donne un changement vers l'instant $(t+1)$, formulé par l'équation suivante:

$$Ze(t) = e(t+1) \quad (3.1)$$

La figure 3.2 représente le temps de décalage vers l'avant :

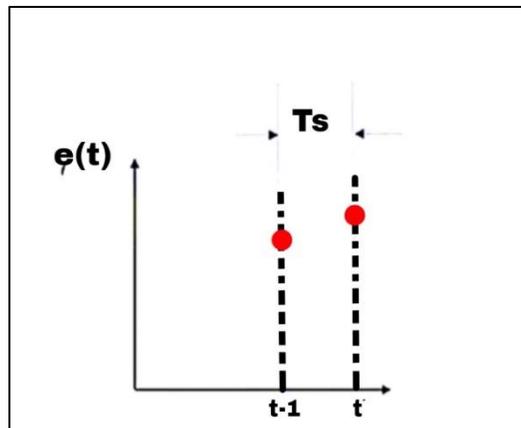


Fig 3.2 Temps de décalage vers l'avant

- **Temps de décalage vers l'arrière**

Ce temps consiste à une rétroaction en temps réel de la sortie du système pour ajuster le signal de contrôle. Aussi, il détecte les erreurs entre la sortie réelle et la consigne souhaitée, et applique des corrections pour réduire ces erreurs. Quand les valeurs passées du signal échantillonné existent, il est normal d'utiliser z^{-1} où z^{-1} est le changement vers l'instant $(t-1)$, qui est donné par :

$$z^{-1} e(t) = e(t-1) \quad (3.2)$$

La figure 3.3 représente le temps de décalage vers l'arrière :

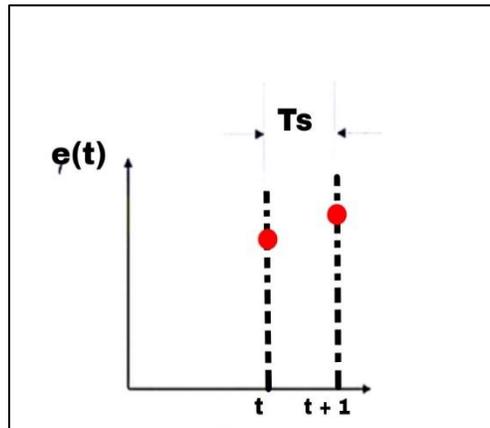


Fig 3.3 Temps de décalage vers l'arrière.

3.2 Régulateur PID discret

Le PID discret utilise des temps d'échantillonnage réguliers pour effectuer des calculs de contrôle. Toutefois, à la différence du PID analogique, chaque terme du PID discret utilise une formule différente :

- Le terme proportionnel est calculé en multipliant l'erreur de régulation par le facteur de proportionnalité K_p , similaire au PID analogique.
- Le terme intégral est calculé en utilisant la somme discrète de l'erreur de contrôle sur la période d'échantillonnage et en multipliant le résultat par le facteur intégral K_i .
- Le terme dérivé est calculé en utilisant la différence entre l'erreur de contrôle actuelle et l'erreur de l'échantillon précédent, multipliée par le facteur dérivé K_d et divisée par la période d'échantillonnage.

3.3 Méthodes de discrétisation

3.3.1 Méthode d'approximation d'Euler vers l'arrière (dérivée)

La méthode d'approximation d'Euler vers l'arrière est largement utilisée dans la discrétisation des systèmes pour approximer les dérivées temporelles. En discrétisant un système continu en un ensemble de pas de temps finis, cette méthode permet de calculer les valeurs futures en utilisant les valeurs précédentes aussi pour réduire les erreurs d'approximation accumulées et les valeurs des variables du système peuvent être calculées pas à pas.

La dérivée approchée est la pente moyenne entre $y(t)$ et $y(t-1)$, donné par l'équation suivante :

$$\frac{de}{dt} = \frac{(1-(t-1))e(t)}{Ts} = \frac{e(t) - e(t-1)}{Ts} = \frac{(e(t) - z^{-1}e(t))}{Ts} \quad (3.3)$$

Où,

z^{-1} : retard d'un temps d'échantillonnage ;

T_s : temps d'échantillonnage.

Donc, la forme de la régulateur PD :

$$U(S) = K_p \cdot e(S) + K_d \cdot S \cdot e(S) \quad (3.4)$$

Et, dans le domaine temporel est la suivante :

$$U(t) = K_p \cdot e(t) + K_d \cdot \frac{de(t)}{dt} \quad (3.5)$$

En remplaçant l'équation (3.3) dans l'équation (3.5), on aura :

$$U(t) = K_p \cdot e(t) + K_d \frac{e(t) - e(t-1)}{Ts} \quad (3.6)$$

$$U(t) = K_p \cdot e(t) + \frac{K_d}{Ts} e(t) - + \frac{K_d}{Ts} e(t-1) \quad (3.7)$$

Finalement, la forme d'un régulateur PD discret est :

$$U(t) = \left(K_p + \frac{K_d}{Ts}\right) e(t) - \left(\frac{K_d}{Ts}\right) e(t-1) \quad (3.8)$$

3.3.2 Règle d'approximation trapézoïdal (intégral)

La règle d'approximation trapézoïdale est une méthode couramment utilisée pour approximer l'intégrale d'une fonction continue sur un intervalle donné. En discrétisant l'intervalle en segments plus petits, cette méthode divise l'aire sous la courbe en trapèzes et calcule une approximation en utilisant la somme des aires de ces trapèzes. La zone entre les échantillons est l'intégrale de l'erreur sur l'intervalle (t-1) à (t). Elle sera approximée par :

$$\frac{T_s}{2} e(t) - e(t-1) \quad (3.9)$$

Où,

$$\frac{T_s}{2} (1 + z^{-1}) e(t) \quad (3.10)$$

La forme générale est donnée par l'équation suivante :

$$\frac{T_s}{2} (e(t) + 2e(t-1) + e(t-2)) \quad (3.11)$$

La figure 3.4 représente l'intégrale de l'erreur sur l'intervalle (t-1) à (t) :

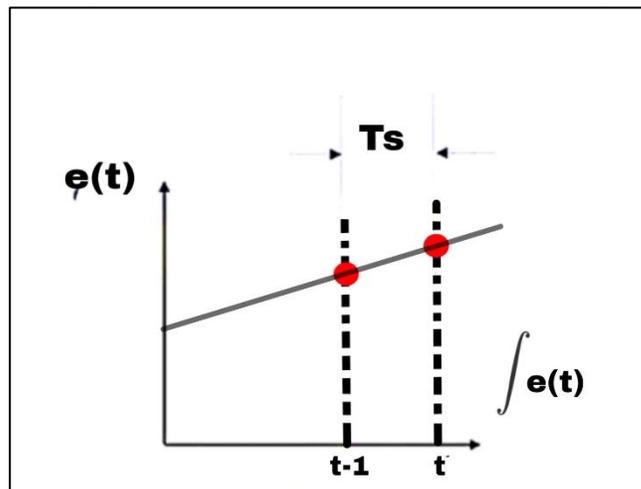


Fig 3.4 L'intégrale de l'erreur sur l'intervalle (t-1) à (t)

Donc, d'après les équations (2.9) et (2.10) on obtient la forme de la régulateur PI :

$$U(S) = K_p \cdot e(S) + K_i \cdot \frac{e(S)}{S} \quad (3.12)$$

$$U(S) = \left(K_p + \frac{K_i}{S} \right) e(s)$$

D'où dans le domaine temporel :

$$U(t) = K_p \cdot e(t) + K_i \int e(t) dt \quad (3.13)$$

En remplaçant l'équation (3.11) dans la (3.13), on aura :

$$U(t) = K_p \cdot e(t) + K_i \frac{T_s}{2} (e(t) + 2e(t-1) + e(t-2)) \quad (3.14)$$

Par conséquent, les trois (3) termes P.I.D C (z^{-1}) en temps discret s'écrivent la forme d'un régulateur PID discret :

$$U(t) = K_p \cdot e(t) + K_i \frac{T_s}{2} (e(t) + 2e(t-1) + e(t-2)) + K_d \frac{e(t) - e(t-1)}{T_s} \quad (3.15)$$

3.4 Principe de la commande RST :

La structure de régulation RST est principalement liée aux objectifs définis par le cahier de charge et la qualité du modèle de l'objet à régler. Par principe, il doit :

- Assurer la stabilité de la boucle de régulation ;
- Éliminer ou déminer les perturbations ;
- Suivre précisément le signal de consigne imposé.

En termes du développement de cette commande, soit l'analyse du système représenté par la fonction de transfert dans le domaine de (z^{-1}) suivante :

$$H(z^{-1}) = \frac{z^{-d} \cdot B(z^{-1})}{A(z^{-1})} \quad (3.16)$$

Où $B(z^{-1})$ et $A(z^{-1})$ sont respectivement les polynômes du numérateur et du dénominateur de la fonction de transfert du système. En considérons le degré du polynôme A est strictement plus grand que celui du polynôme numérateur B . Par ailleurs la variable z^{-1} est l'opérateur de décalage arrière [15]. Cependant dans le domaine z , un retard (d) de d'échantillons est représenté par z^{-d} . Ainsi, z^{-d} décale le signal d'échantillons dans le passé.

La figure (3.5) montre la schéma bloc de ce système :

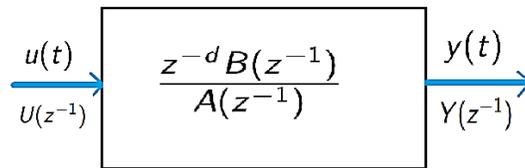


Fig 3.5 Schéma bloc du système

La figure 3.6 présente le schéma bloc général pour la synthèse RST où R, S et T sont des polynômes exprimés en fonction en la variable (z^{-1}). Chaque polynôme joue un rôle distinct dans le fonctionnement du système de commande:

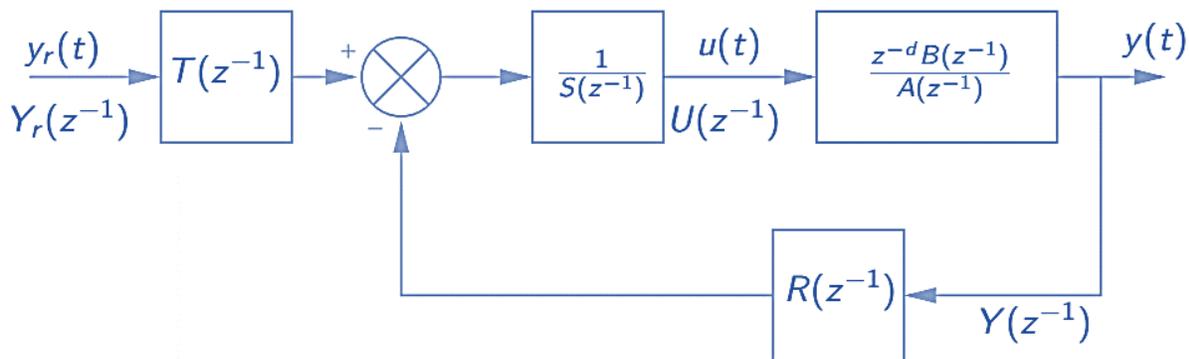


Fig 3.6 Schéma bloc général pour la synthèse RST

- Le polynôme $R(z^{-1})$ est responsable de la rétroaction de la sortie du système. Il définit comment le système répond aux variations de la sortie et contribue principalement à la stabilité globale du système et à la minimisation des erreurs de régulation statique.
- Quand $S(z^{-1})$ est associé à la rétroaction d'erreur dans le système de commande. Il contribue à la dynamique de réponse du système face à une erreur entre la sortie attendue et la sortie réelle , en vue d'amélioration la réponse transitoire du système .
- Enfin $T(z^{-1})$ est lié à l'action directe sur la commande du système pour déterminer comment le système répond aux variations des valeurs d'entrée et est généralement utilisé pour améliorer le suivi de la référence, il est aussi introduit un degré de liberté supplémentaire qui va permettre de faire de la poursuite .

La fonction de transfère en boucle fermée du système est :

$$Y(z^{-1}) = U(z^{-1}) \cdot H(z^{-1}) = \left(Y_r(z^{-1}) \frac{T(z^{-1})}{S(z^{-1})} - Y(z^{-1}) \frac{R(z^{-1})}{S(z^{-1})} \right) \cdot H(z^{-1}) \quad (3.17)$$

$$S(z^{-1}) \cdot U(z^{-1}) = Y_r(z^{-1}) \cdot T(z^{-1}) - Y(z^{-1}) \cdot R(z^{-1}) \quad (3.18)$$

D'après l'équation (3.17), on peut remplacer $U(z^{-1})$ par $\frac{Y(z^{-1})}{H(z^{-1})}$ dans l'équation (3.18) :

$$S(z^{-1}) \cdot \frac{Y(z^{-1})}{H(z^{-1})} = Y_r(z^{-1}) \cdot T(z^{-1}) - Y(z^{-1}) \cdot R(z^{-1}) \quad (3.18)$$

En multipliant les deux côtes de l'équation (3.18) par $H(z^{-1})$, nous obtenons :

$$S(z^{-1}) \cdot Y(z^{-1}) = [Y_r(z^{-1}) \cdot T(z^{-1})] \cdot H(z^{-1}) - [Y(z^{-1}) \cdot R(z^{-1})] \cdot H(z^{-1}) \quad (3.19)$$

Cette dernière équation peut s'exprimer par :

$$[(S(z^{-1}) + R(z^{-1})) \cdot H(z^{-1})] \cdot Y(z^{-1}) = [T(z^{-1}) \cdot H(z^{-1})] \cdot Y_r(z^{-1}) \quad (3.20)$$

Finalement, nous pouvons diviser par $[(S(z^{-1}) + R(z^{-1})) \cdot H(z^{-1})]$ premièrement et après par $Y_r(z^{-1})$ pour obtenir la fonction de transfert suivante :

$$\frac{Y(z^{-1})}{Y_r(z^{-1})} = \frac{T(z^{-1}) \cdot H(z^{-1})}{S(z^{-1}) + H(z^{-1}) \cdot R(z^{-1})} = \frac{T(z^{-1}) \cdot z^{-d} B(z^{-1})}{A(z^{-1}) S(z^{-1}) + z^{-d} B(z^{-1}) \cdot R(z^{-1})} \quad (3.21)$$

Avec

$$P(z^{-1}) = A(z^{-1}) \cdot S(z^{-1}) + z^{-d} B(z^{-1}) \cdot R(z^{-1}) \quad (3.22)$$

Où, le polynôme $P(z^{-1})$ définit les pôles du système en boucle fermée.

Soit, le schéma bloc de l'RST :

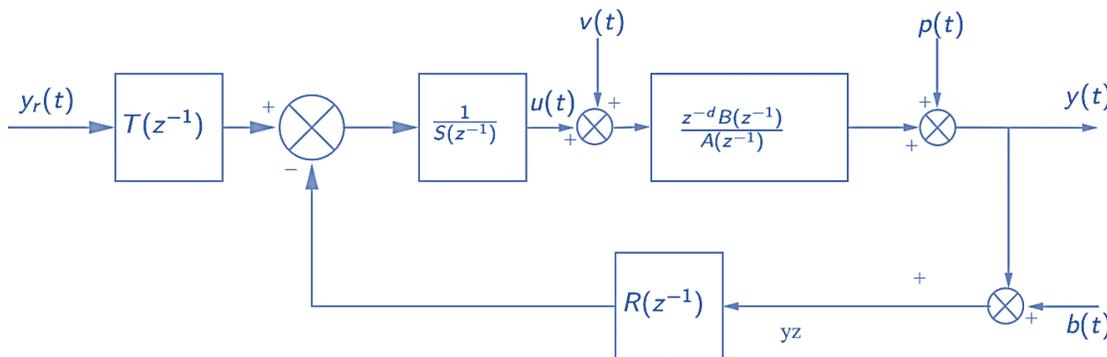


Fig 3.7 Représentation générale de l'RST

Cette représentation est, essentiellement, composée de divers facteurs clés qui jouent un rôle crucial dans le processus de régulation et de commande.

- $Y_r(t)$ est la référence ou la consigne, qui est l'objectif de sortie visé par le système. C'est la valeur idéale que le système vise à accomplir.
- $Y(t)$ est la valeur de sortie actuelle du système, la réponse effective du système par rapport à la consigne et aux différentes entrées.
- $U(t)$ est la commande du système, qui est l'entrée du régulateur RST. Cette commande est dérivée de la différence entre la consigne $Y_r(t)$ et la sortie actuelle $Y(t)$.
- $V(t)$ représente une perturbation qui affecte la commande du système, modifiant ainsi l'entrée du système.
- $P(t)$ est une perturbation qui influence directement la sortie du système, altérant ainsi la valeur de sortie.
- $b(t)$ représente le bruit de mesure qui peut être introduit lors de la détermination de la sortie du système. Il s'agit généralement d'un bruit de haute fréquence qui peut perturber la précision des mesures réalisées.

Ces différents signaux interagissent ensemble au sein du système afin de produire la sortie $Y(t)$. Le but du régulateur RST est d'aligner cette sortie avec la consigne $Y_r(t)$ en dépit des diverses perturbations et bruits qui peuvent éventuellement se manifester.

3.5 Relations entrées / sorties

Les fonctions de transfert suivantes sont des représentations de la relation entre différents signaux d'entrée et de sortie dans un système avec un régulateur RST[13]. Les signaux d'entrée sont généralement des perturbations ou des consignes, et les signaux de sortie sont les réponses du système à ces entrées.

$$\bullet \quad S_{p \rightarrow y} = AS / (AS + \bar{B}R)$$

Cette fonction de transfert représente la relation entre la perturbation de sortie (S_p) et la sortie du système (Y), en tant que le terme (AS) représente l'effet de la perturbation de sortie sur la sortie du système et le dénominateur ($AS + \bar{B}R$) représente l'effet total sur la sortie du système à la fois de la perturbation de sortie et de la commande du système, on obtient :

$$Y(z^{-1}) = \frac{p(z^{-1})}{R(z^{-1}) + H(z^{-1}) S(z^{-1})} \quad (3.23)$$

Avec,

$p(z^{-1})$: l'effet de la perturbation de sortie.

- $S_{r \rightarrow y} = \bar{B}T / (AS + \bar{B}R)$

C'est la représentation de la relation entre la consigne (S_r) et la sortie du système (Y) ou le terme ($\bar{B}T$) représente l'effet de la consigne sur la sortie du système et le dénominateur $(AS + \bar{B}R)$ représente l'effet total sur la sortie du système. Pour cela on a :

$$Y(z^{-1}) = \frac{T(z^{-1})S(z^{-1})}{R(z^{-1}) + H(z^{-1})S(z^{-1})} \quad (3.24)$$

- $S_{p \rightarrow u} = -AR / (AS + \bar{B}R)$

C'est la relation entre la perturbation agissant au niveau de la sortie (S_p) et la commande du système (u). Le terme (-AR) met en lumière l'effet contraire (effet négative) que la perturbation de sortie impose sur l'action du système. Le dénominateur $(AS + \bar{B}R)$ représente l'influence totale sur la commande du système, prenant en compte à la fois l'effet de la perturbation de sortie et l'effet du retour d'information de la sortie du système.

$$U(z^{-1}) = \frac{-P(z^{-1})R(z^{-1})}{R(z^{-1}) + H(z^{-1})S(z^{-1})} \quad (3.25)$$

- $S_{b \rightarrow y} = -\bar{B}R / (AS + \bar{B}R)$.

Cette fonction de transfert représente la relation entre le bruit de mesure (S_b) et la sortie du système (y). Le terme ($-\bar{B}R$) représente l'effet négatif du bruit de mesure sur la sortie du système et le dénominateur $(AS + \bar{B}R)$ représente l'effet total sur la sortie du système, y compris l'effet du bruit de mesure et l'effet de la rétroaction de la sortie du système sur la commande du système. à partir de là, on trouve :

$$Y(z^{-1}) = \frac{B(z^{-1})H(z^{-1})S(z^{-1})}{(R(z^{-1}) + H(z^{-1})S(z^{-1}))} \quad (3.26)$$

Chacune de ces fonctions de transfert donne une image différente de la manière dont le régulateur RST et le système réagissent à différents signaux d'entrée, et elles sont toutes utiles pour comprendre la dynamique globale du système.

Avec,

$$\bar{B} = z^{-d}B(z^{-1})$$

Note : l'influence d'une perturbation est additive sur la sortie du procédé réel.

3.6 Synthèse du régulateur RST

Le RST est une solution attractive en ce comparant aux régulateurs de type Proportionnel – Intégral. Il offre une gestion optimisée du compromis entre vitesse et performance. Les polynômes R, S et T ont des degrés déterminés en fonction de celui des fonctions de transfert précédemment définies.

L'objectif principal de la conception d'un régulateur RST réside dans l'atténuation de l'influence des perturbations sur la sortie du système, au moins pour des périodes spécifiques. Ce processus de sélection des polynômes R, S et T est orienté par plusieurs critères, dont [16]:

- Détermination de la dynamique de rejet des perturbations : se réfère à la vitesse et à l'efficacité avec lesquelles le système peut neutraliser les effets de perturbations inattendues ;
- Temps de réponse : l'évaluation de la durée nécessaire pour que le système atteigne un nouvel état stable suite à une perturbation ou à un changement dans la consigne. Un temps de réponse plus court est généralement préférable, car il permet au système de s'adapter plus rapidement à de nouvelles conditions ;
- Dépassement : ce critère mesure à quel point la sortie du système dépasse sa valeur finale désirée en réponse à une perturbation ou à un changement dans la consigne. Un dépassement minimal est souhaitable pour éviter les fluctuations excessives dans la sortie du système.

3.6.1 Détermination des polynômes R et S

Le choix du dénominateur $D(z)$ de la fonction de transfert du système bouclé impose à R et S de vérifier l'équation Bézout suivante :

$$P(z^{-1}) = A(z^{-1}).S(z^{-1}) + z^{-d}B(z^{-1}).R(z^{-1}) \quad (3.27)$$

Il faut choisir les pôles de $D(z)$ afin d'assurer un rejet de perturbation satisfaisant (en terme de rapidité et de dépassement) car la dynamique de la régulation et du rejet de perturbation est fixée par ces pôles .

a) Résolution de l'équation de BÉZOUT

Cette équation est une équation diophantienne, ce qui signifie que nous cherchons des solutions dans un ensemble discret. Elle possède une solution unique, si est seulement si les relations (3.28) sont vérifiées .

$$\left\{ \begin{array}{l} n_p \leq n_A + n_B + d - 1 \\ n_S = n_B + d - 1 \\ n_R = n_A - 1 \end{array} \right. \quad (3.28)$$

Avec,

n_p : degré du polynôme $P(z^{-1})$.

n_S : degré du polynôme $S(z^{-1})$.

n_R : degré du polynôme $R(z^{-1})$.

n_A : degré du dénominateur $A(z^{-1})$.

n_B : degré du numérateur $B(z^{-1})$.

Ou, les polynômes S et R sont définies par :

$$\left\{ \begin{array}{l} S(z^{-1}) = S_0 + S_1(z^{-1}) + \dots + S_{n_S}(z^{-n_S}) \\ R(z^{-1}) = 1 + R_0 + R_1(z^{-1}) + \dots + R_{n_R}(z^{-n_R}) \end{array} \right. \quad (3.29)$$

Pour résoudre cette équation, dont les variables inconnues sont les coefficients de R et S, nous transformons cette équation en une forme matricielle suivante :

$$\left\{ \begin{array}{l} Mx = p \\ x = [1 \ S_1 \ S_2 \ \dots \ S_{n_S} \ R_0 \ R_1 \ \dots \ R_{n_R}]^T \\ P = [1 \ P_0 \ P_1 \ \dots \ P_{n_p} \ 0 \ 0 \ \dots]^T \end{array} \right. \quad (3.30)$$

Remarque :

Dans le cas où A et B ne partagent aucun facteur commun, la matrice M est inversible, c'est-à-dire que son déterminant n'est pas égal à zéro.

On peut aussi déduire directement les coefficients des polynômes R et S, lorsque le système de Sylvester (3.31) est trouvé, qui est représentée par la formule suivante [17]:

$$\begin{bmatrix} a_n & 0 & \dots & 0 & b_n & 0 & \dots & 0 \\ \cdot & a_{n-1} & & a_n & b_{n-1} & b_n & \cdot & \\ \cdot & a_{n-1} & & \cdot & \cdot & b_{n-1} & 0 & \\ \cdot & \cdot & & a_n & \cdot & \cdot & b_n & \\ \cdot & \cdot & & a_{n-1} & b_0 & \cdot & b_{n-1} & \\ a_0 & a_0 & \cdot & \cdot & 0 & b_0 & \cdot & \\ 0 & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \\ \cdot & \\ 0 & 0 & a_0 & \cdot & \cdot & \cdot & \cdot & \\ 0 & 0 & 0 & 0 & 0 & \cdot & b_n & \end{bmatrix} \cdot \begin{bmatrix} S_n \\ S_{n+1} \\ \cdot \\ S_{ns} \\ R_n \\ R_{n-1} \\ \cdot \\ \cdot \\ \cdot \\ R_0 \end{bmatrix} = \begin{bmatrix} P_{2n+1} \\ p_{2n} \\ p_{n-1} \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ p_1 \\ p_0 \end{bmatrix} \quad (3.31)$$

Ce système est développé d'après un système linéaire en fonction du degré du polynôme $S(z^{-1})$:

$$\left\{ \begin{array}{l} B(S) = b_{(n-1)}S^{(n-1)} + b_{(n-2)}S^{(n-2)} + \dots + b_1S^1 + b_0 \\ A(S) = a_{(n)}S^{(n)} + a_{(n-1)}S^{(n-1)} + a_{(n-2)}S^{(n-2)} \dots + a_1S^1 + a_0 \\ R(S) = r_{(n)}S^{(n)} + r_{(n-1)}S^{(n-1)} + r_{(n-2)}S^{(n-2)} \dots + r_1S^1 + r_0 \end{array} \right. \quad (3.32)$$

b) Choix des parties fixes

Le choix des parties fixes dans les polynômes S et R peut être motivé par plusieurs raisons. Ils peuvent servir à respecter certaines spécifications du système à contrôler, ou bien à imposer des contraintes sur le comportement du système en réponse à certaines entrées.

A titre de référence, le rejet de perturbation polynômiales imposent :

$$S(z^{-1}) = (1 - z^{-1})^q S^* (z^{-1}) \quad (3.33)$$

Dans l'équation (3.33) , la partie $[(1 - z^{-1})^q]$ est utilisée pour introduire des zéros à l'origine dans le polynôme $S(z^{-1})$.L'équation décrit une manière de formuler le polynôme (S) dans le contexte du régulateur RST pour qu'il contienne (q) zéros à $z=1$ (équivalent à un retard unitaire). C'est souvent souhaitable dans la conception de systèmes de contrôle, car cela peut aider à rejeter les perturbations constantes (des perturbations qui ont une transformée de Laplace en $z=1$).

Le terme (q) dans l'équation représente le nombre de fois que cette opération est effectuée, c'est-à-dire le nombre de zéros à $z = 1$ dans le polynôme (S) . $S * (z^{-1})$ est généralement un polynôme sans zéros à $z = 1$, il est multiplié par $(1 - z^{-1})^q$ pour introduire les zéros à l'origine souhaités.

De la même manière, $R(z^{-1})$ peut contenir des parties fixes. On pose ainsi :

$$\begin{cases} S(z^{-1}) = S * (z^{-1})H_S(z^{-1}) \\ R(z^{-1}) = R * (z^{-1})H_R(z^{-1}) \end{cases} \quad (3.34)$$

Où, $H_R(z^{-1})$ et $H_S(z^{-1})$ sont les parties fixes du polynômes $R(z^{-1})$ et $S(z^{-1})$ respectivement.

Donc, la fonction de transfère en boucle fermée du système sera :

$$\frac{T(z^{-1}). z^{-d}B(z^{-1})}{A(z^{-1})S*(z^{-1})H_S(z^{-1})+z^{-d} B(z^{-1}).R*(z^{-1})H_R(z^{-1})} \quad (3.35)$$

Alors,on doit résoudre l'équation suivante :

$$P(z^{-1}) = A(z^{-1}).S * (z^{-1})H_S(z^{-1}) + z^{-d}B(z^{-1}).R * (z^{-1})H_R(z^{-1}) \quad (3.36)$$

$$\begin{cases} n_p \leq n_A + n_{H_S} + n_B + n_{H_R} + d - 1 \\ n_{S*} = n_B + n_{H_R} + d - 1 \\ n_R = n_A + n_{H_S} - 1 \end{cases} \quad (3.36)$$

c) Détermination du polynôme T

Lors de l'instauration d'une rétroaction en boucle fermée, l'objectif test d'assurer que la sortie du système suit la référence. Cette trajectoire de référence spécifiée est caractérisée comme la sortie d'un dispositif nommé modèle de référence.

La figure 3.8 présente le modèle de référence :

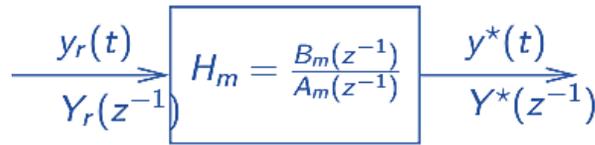


Fig 3.8 Modèle de référence

Habituellement, les paramètres $B_m(z^{-1})$ et $A_m(z^{-1})$ sont définis en fonction des performances attendues de la trajectoire, telles que le temps de réponse et le dépassement. Ces spécifications de performance influencent la sélection appropriée de ces paramètres dans le modèle de référence. L'objectif est alors de déterminer la manière dont on peut suivre la trajectoire désirée [13].

$$Y^*(z^{-1}) = \frac{B_m(z^{-1})}{A_m(z^{-1})} Y_r \tag{3.37}$$

Dans ce contexte, $T(z^{-1})$ est probablement le précompensateur qui est utilisé pour modifier la réponse du système $P(z^{-1})$ avant l'application du régulateur .

Le précompensateur $G(z^{-1})$ est introduit ici pour assurer :

- **Un gain unitaire entre Y_r et Y^* :** cela signifie que si vous souhaitez que le système atteigne une certaine valeur Y_r , alors le système doit effectivement atteindre cette valeur.
- **Compensation des pôles de $P(z^{-1})$:** cela signifie que vous souhaitez que le précompensateur $G(z^{-1})$ annule les pôles du système $P(z^{-1})$ qui ont une influence significative sur la stabilité et la réponse transitoire du système. En annulant ces pôles, le précompensateur $G(z^{-1})$ peut aider à stabiliser le système et à améliorer sa réponse.

Donc,

$$T(z^{-1}) = G(z^{-1}). P(z^{-1}) \tag{3.38}$$

Avec,

$$G(z^{-1}) = \begin{cases} \frac{1}{B(1)} & \text{Si } B(1) \neq 1 \\ 1 & \end{cases} \tag{3.39}$$

Dans l'équation (3.39) montre comment le précompensateur $G(z^{-1})$ est conçu. Si $B(1)$ (qui pourrait représenter le premier coefficient d'un polynôme dans une représentation en retard de z) n'est pas égal à 1, alors $G(z^{-1})$ est défini comme l'inverse de $B(1)$, cela pourrait être utilisé pour normaliser le gain du système.

Si $B(1)$ est égal à 1, alors $G(z^{-1})$ est simplement défini comme 1, ce qui signifie que le précompensateur n'a pas d'effet sur le système. La relation entre $B(1)$ et $B(z^{-1})$ est donc que $B(1)$ est une évaluation spécifique du polynôme $B(z^{-1})$.

Dans le cas d'un gain unitaire l'équation reste la même sinon dans le cas contraire on peut déduire l'équation suivante :

$$Y^*(z^{-1}) = \frac{B(z^{-1})}{B(1)} \frac{B_m(z^{-1})}{A_m(z^{-1})} Y_r \tag{3.40}$$

La figure 3.9 présente la structure de synthèse RST

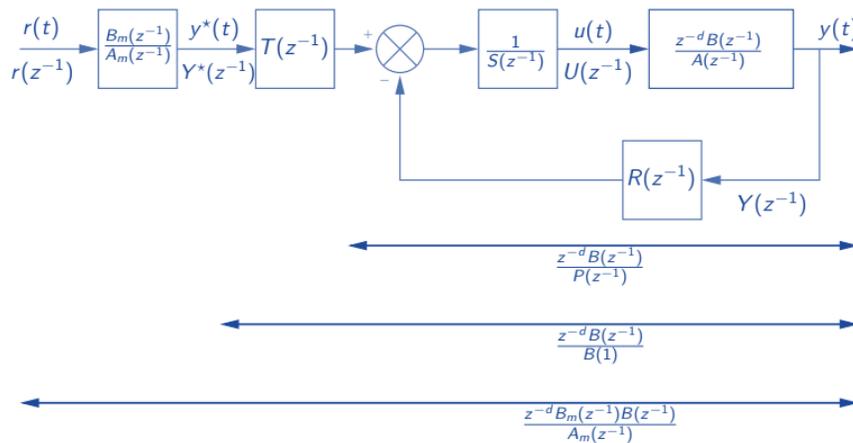


Fig 3.9 Structure de synthèse

Le polynôme $P(z^{-1})$ qui est représenté par le dénominateur $D(s)$, est utilisé pour réaliser un compromis entre performance et robustesse qui est factorisé dans le produit de deux polynômes :

$$D(S) = C(S) \cdot F(S) \tag{3.41}$$

Avec,

$C(S)$: pôles de commande ;

$F(S)$: pôles de filtrage .

Dans l'objectif de réaliser cette factorisation, deux paramètres de synthèse sont employés : Un horizon de commande (T_c) et un horizon de filtrage (T_f) ou le pôle de polynôme de commande est défini par :

$$p_c = -\frac{1}{T_c} \tag{3.42}$$

Et le pôle de polynôme de filtrage est donnée par :

$$p_f = -\frac{1}{T_f} \tag{3.43}$$

3.7 Stratégie de placement de pôles robuste

On effectue les opérations disposées ci-dessous afin de déduire les pôles en boucle ouverte :

- Par rapport à l'axe imaginaire, les pôles instables sont remplacés par des pôles symétriques .
- Les arbres complexes humides sont réduits au niveau d'amortissement spécifié.
- Les électrodes très lentes à droite de la verticale p_c sont ramenées dans cette position verticale.
- Les pôles de polynôme p_f déduire selon la même procédure mais seulement, on ajoute un pôle $p_0 = 0$ pour obtenir $n+1$ pôles pour le polynôme $(f(p))$.

La figure 3.10 montre une distribution aléatoire des pôles :

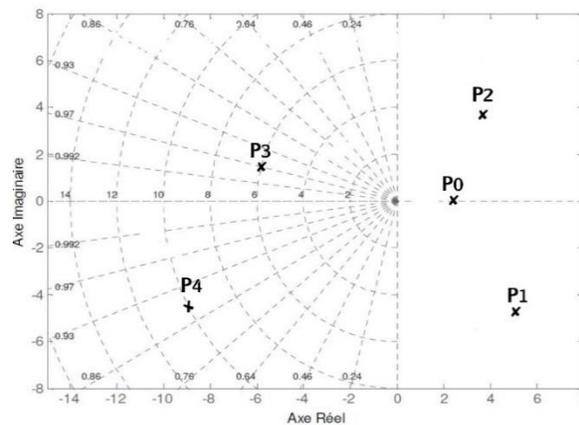


Fig 3.10 Distribution aléatoire des pôles.

Dans la figure 3.11, les pôles dans la zone instable (p_0, p_1 et p_2) sont remplacés par leurs symétriques par rapport à l'axe imaginaire.

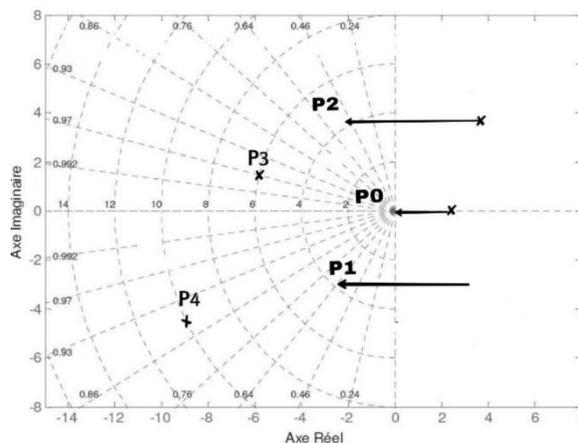


Fig 3.11 Transformation des pôles instables dans la zone de stabilité.

Dans la figure 3.12, les pôles dans la zone stable peu amortie (p_1 et p_2) sont ramenés à l'amortissement arbitraire ($\lambda = 0.86$).

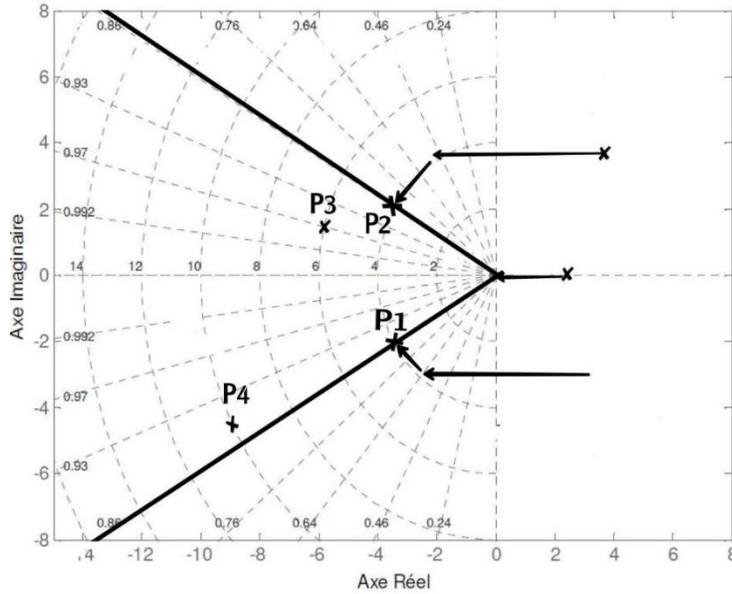


Fig 3.12 Projection des pôles peu amortie sur ($\lambda = 0.86$)

Dans la figure 3.13, les pôles dans la zone lente amortie (p_0, p_1, p_2 et p_3) à droite de la verticale (horizon de commande) sont ramenés à cette verticale,

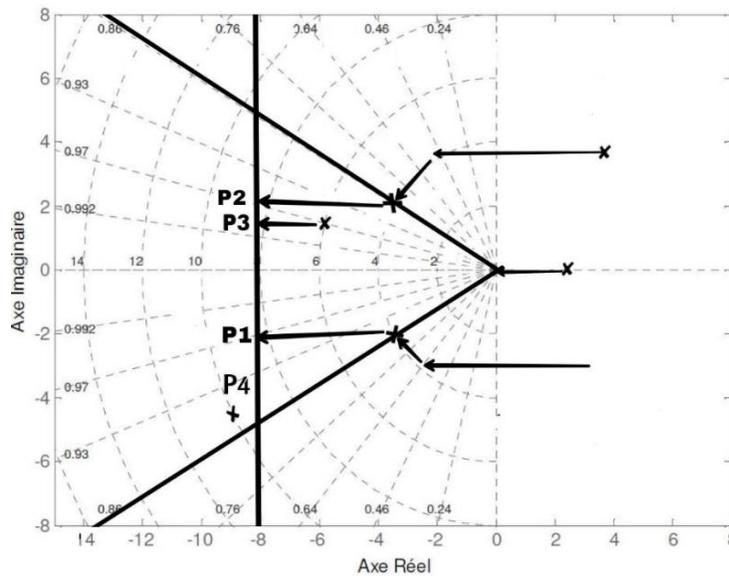


Fig 3.13 Projection des pôles lents sur la verticale p_c

Les pôles dans la zone p_0, p_1, p_2, p_3 et p_4 à droite de la verticale (horizon de filtrage) sont ramenés à cette verticale, $p_5=0$ s'ajoute à fin d'obtenir $n+1$ pôles pour le polynôme $f(S)$ comme il est montré dans la figure 3.14 :

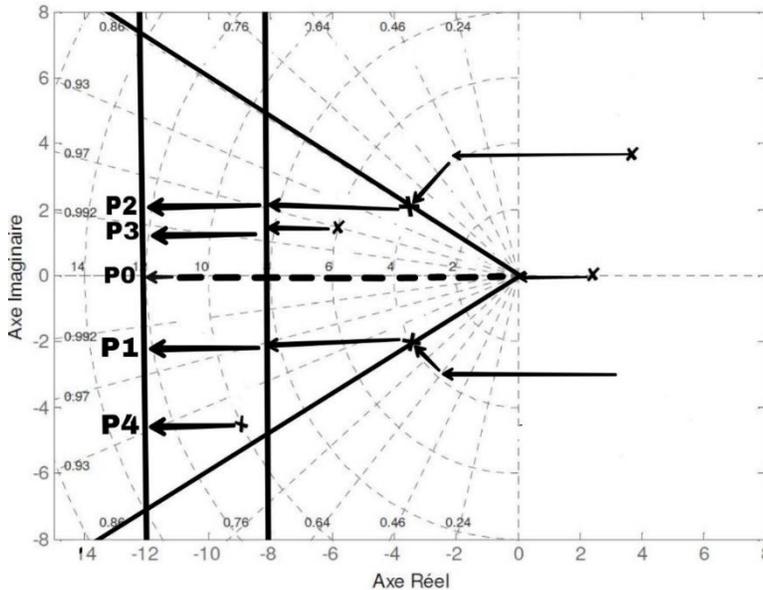


Fig 3.14 Projection des pôles sur la verticale p_f

La stratégie de placement de pôles robuste et le réglage de la loi de commande se dépend aux choix des horizons de filtrage (p_f) et commande (p_c) .

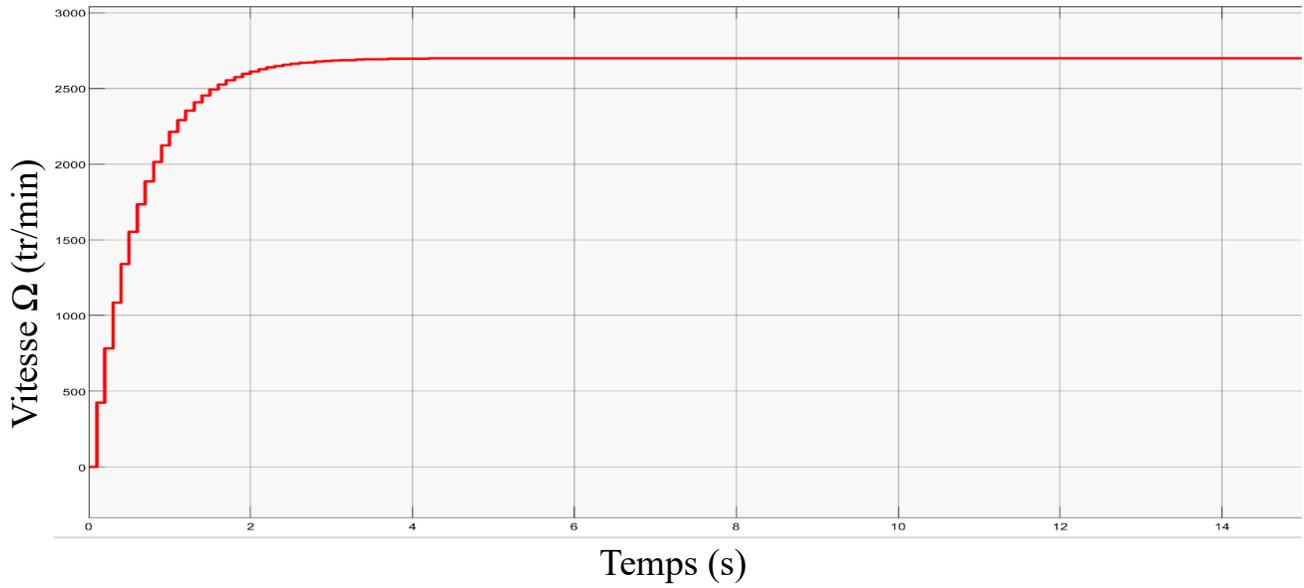
3.8 Simulation et discussion du MCC en boucle fermée

Dans la partie de simulation précédente, nous avons testé le modèle du moteur en boucle fermée à l'aide d'un régulateur PI, donc maintenant on peut remplacer ce régulateur avec un régulateur RST. On considère que la fonction de transfert discrète du système est :

$$H_d = \frac{0.09376 z + 0.002323}{z^2 - 0.9039 z} \tag{3.42}$$

Cette fonction de transfert discrète est testée par un fonctionnement à vide et avec application d'une charge à l'instant 10 seconde, comme le montre la figure 2.15 :

 *Simulation à vide*



 *Simulation avec perturbation*

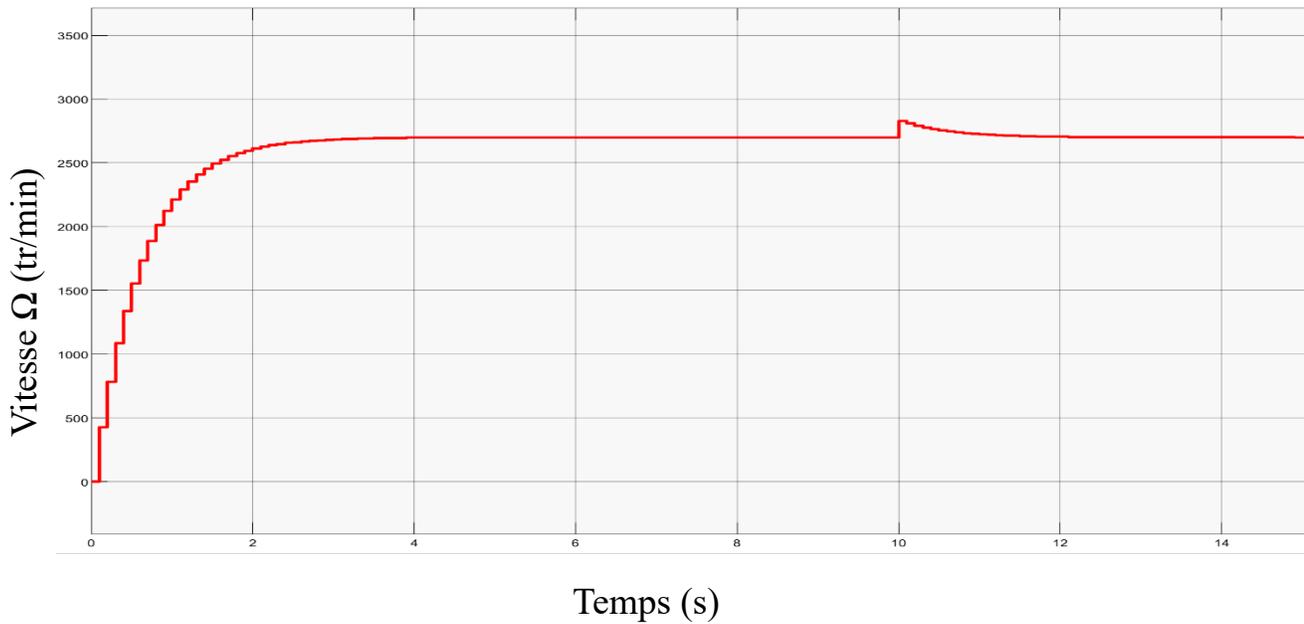


Fig 3.15 Résultats de simulation de la MCC en boucle fermée (régulateur RST)

3.9 Conclusion

Au terme de ce Chapitre 3, nous avons approfondi notre compréhension de concepts importants tels que la discrétisation, le régulateur PID discret et la commande RST, tous essentiels à notre projet de contrôle de moteur à courant continu.

Nous avons débuté avec la discrétisation, une méthode incontournable qui facilite la transition entre un modèle en temps continu à un modèle en temps discret, ensuite, nous avons exploré le régulateur PID discret, qui offre un équilibre judicieux entre une bonne réponse transitoire et une faible erreur en régime permanent, ce qui est crucial pour le contrôle précis de notre système.

Enfin, nous avons introduit la commande RST, une méthode avancée de contrôle qui promet d'améliorer le comportement dynamique de notre moteur. Son exploration approfondie a mis en évidence son potentiel considérable pour optimiser les performances du moteur.

Le chapitre 4 se concentrera sur l'application pratique de ces techniques théoriques. Nous procéderons à l'implémentation des techniques étudiées et réaliserons des tests expérimentaux pour évaluer les performances de notre système de contrôle. Ce travail pratique permettra de valider notre approche théorique, de mettre en évidence les avantages et les inconvénients de chaque méthode, et ultimement, de perfectionner le fonctionnement de notre moteur à courant continu.

CHAPITRE 4

COMMANDE DE VITESSE D'UN MOTEUR PAR MATLAB ET ARDUINO

CHAPITRE 4

COMMANDE DE VITESSE D'UN MOTEUR PAR MATLAB ET ARDUINO

4.1 Introduction

Ce chapitre est consacré à l'implémentation pratique d'un circuit de contrôle de vitesse pour un petit moteur à courant continu à aimants permanents en utilisant le logiciel MatLab et la carte Arduino de type MEGA 2560.

Il décrit la partie réalisation effectuée en évoquant des références et rôles de chaque composante utilisée. À la fin, des essais de testes sont montrés et interprétés.

4.2 Composants utilisés

4.2.1 Carte Arduino

La carte Arduino est un dispositif électronique compact, doté d'un microcontrôleur capable d'analyser et de générer des signaux électriques pour accomplir différentes fonctions. Grâce à l'addition de capteurs connectés à ses broches d'entrée, la carte Arduino peut percevoir son environnement. Il est alors possible de recueillir et de traiter ces informations en programmant l'Arduino à l'aide du logiciel de développement Arduino IDE, qui est gratuit. Ensuite, on peut contrôler les sorties pour influencer d'autres éléments tels que des LED, des afficheurs, des moteurs et plus encore[18].

A) Types de cartes Arduino

Il existe de nombreux modèles de cartes Arduino, incluant les versions originales et les versions dérivées, qui sont compatibles avec Arduino. Les modèles les plus fréquemment utilisés dans le domaine des systèmes embarqués incluent : la UNO, la LEONARDO, la DUE, la MEGA et son amélioration, la MEGA 2560. Nous avons choisi la MEGA 2560 pour cette réalisation, en raison de ses performances supérieures, qui sont adéquates pour la gestion de notre système.

B) Les caractéristiques techniques de la carte Arduino Mega 2560

La carte Arduino Mega 2560 est fondée sur un Microcontrôleur Atmel ATmega2560 à 16 Mhz. Les caractéristiques de cette carte sont les suivantes [19] :

- Elle possède 54 broches numériques, dont 15 peuvent servir à la modulation de largeur d'impulsion (PWM). Ces broches peuvent être configurées comme entrées ou sorties à l'aide du code IDE Arduino.
- Elle est équipée de 16 broches d'entrée analogiques.
- Elle dispose de 4 broches UART, USB, RX et TX pour la communication, ainsi que des interfaces TWI et SPI (port série matériel).
- Les broches d'alimentation sont de 5v, ce qui permet d'alimenter des projets tant que la carte est alimentée entre 7 et 12v ou par l'USB 5v. La broche 3v3 peut délivrer une tension de 3.3 volts.
- Les broches GND peuvent être utilisées pour connecter notre projet à la terre. Quant à la broche IOREF, elle est utilisée pour fournir la tension de référence avec laquelle le microcontrôleur fonctionne.
- Chaque broche d'E/S peut délivrer un courant de 40 mA en continu.
- La broche 3v3 peut délivrer un courant de 50 mA.
- Elle est dotée d'un quartz de 16Mhz.
- Elle possède une connexion USB.
- Elle est pourvue d'un connecteur d'alimentation jack.
- Elle est équipée d'un connecteur ICSP (pour la programmation "in-circuit") et d'un bouton de réinitialisation.

La carte contient tout ce qui est nécessaire pour le fonctionnement du microcontrôleur. Pour la mettre en service, il suffit de la connecter à un ordinateur via un câble USB. Bien qu'elle puisse

aussi être alimentée par un adaptateur secteur ou une pile, ce n'est pas indispensable car l'alimentation peut être fournie par le port USB.

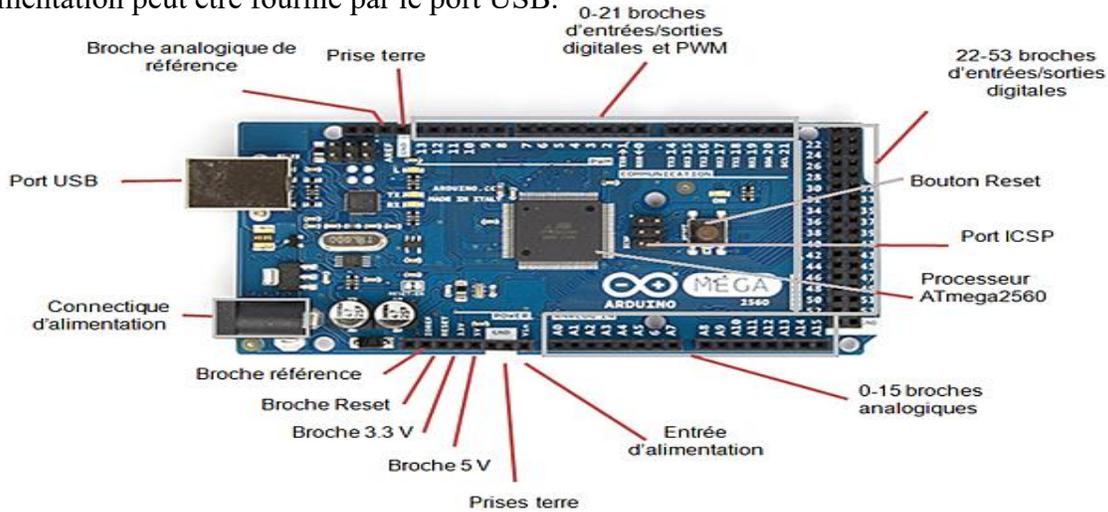


Fig 4.1 Architecture du Arduino Mega

Les composants essentiels de la carte Arduino comprennent :

- ❖ **Microcontrôleur** : Un microcontrôleur est une variante compacte de processeur informatique reliée à des entrées et sorties numériques (représentées par les valeurs 0 ou 1) ou analogiques (caractérisées par une tension variable). Sa fonction principale est de stocker et d'exécuter un programme spécifique qui interprète les signaux d'entrée afin d'activer les sorties correspondantes. La programmation d'un microcontrôleur s'effectue généralement via un ordinateur où un programme est élaboré pour établir les instructions à suivre par le microcontrôleur.
- ❖ **Technique de la PWM (modulation de largeur d'impulsion) [20]** : Pour un moteur à courant continu, il existe une corrélation directe entre la tension d'alimentation et sa vitesse. Plus la tension est élevée, plus la vitesse de rotation est grande. Le microcontrôleur peut être utilisé pour générer des ondes carrées de rapports cycliques variables. Cela délivre en sortie de la carte Arduino une tension moyenne comprise entre 0 et 5V. Grâce à la technique de la PWM, nous pouvons générer une tension à travers des impulsions de diverses intensités. Cette modulation de largeur d'impulsion permet de produire une tension via des impulsions de forces variables. Ce signal contrôle un transistor ou un relais, qui à son tour commande un moteur. Par conséquent, le moteur est alimenté de façon intermittente par des impulsions PWM. Cela modifie la vitesse du moteur en ajustant le

rapport cyclique (α). Plus ce dernier est faible (c'est-à-dire lorsque l'état LOW est plus long que l'état HIGH), plus le moteur tourne lentement. À l'inverse, plus le rapport cyclique est élevé (c'est-à-dire lorsque l'état HIGH dure plus longtemps que l'état LOW), plus le moteur tourne rapidement.

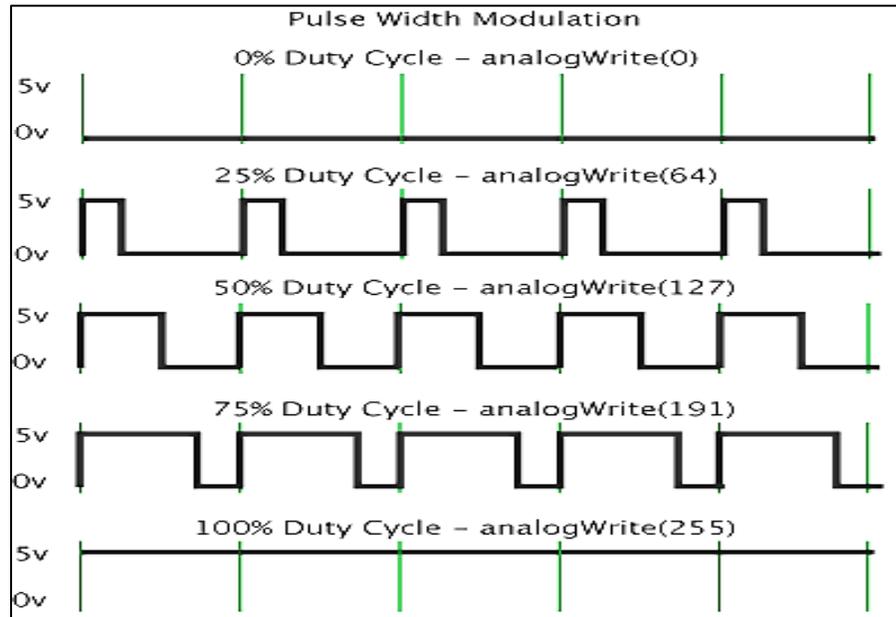


Fig 4.2 Rapport cyclique (α)

4.2.2 Hacheur double pont L298N

Le L298N est un double pont-H conçu pour le contrôle de moteurs à courant continu (H-Bridge Motor Driver). Il est efficace pour piloter des charges inductives telles que des relais, solénoïdes, moteurs à courant continu et moteurs pas-à-pas bipolaires (4 fils, 2 phases). Ces deux types de moteurs peuvent être contrôlés aussi bien en termes de vitesse (via la modulation de largeur d'impulsion, ou PWM) que de direction. Toutes les sorties de puissance sont protégées par des diodes de roue libre. Le L298N est conçu pour résister à des tensions élevées et des courants importants, tout en offrant une interface de commande logique TTL (Transistor-Transistor Logic), qui utilise des tensions basses et des courants faibles, ce qui le rend idéal pour être piloté par un microcontrôleur. La figure 4.3 présente le Hacheur double pont L298N

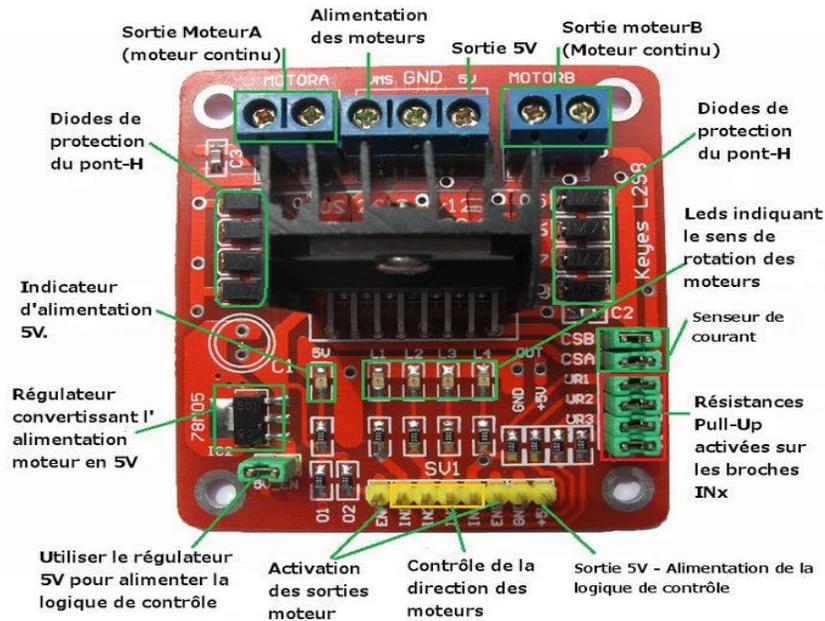


Fig 4.3 Hacheur double pont L298N

En jaune, on retrouve les entrées de commande : ENA, In1, In2 pour le moteur A et ENB, In3, In4 pour le moteur B. Il y a également une broche de masse et une sortie de 5V.

Sur le côté droit en vert, on aperçoit différents cavaliers permettant d'activer des résistances de pull-down (qui forcent une entrée/sortie à l'état bas) et de câbler la mesure du courant de sortie des ponts.

En bleu, on trouve les bornes pour connecter les moteurs A et B (respectivement à gauche et à droite), ainsi qu'au milieu, un bornier pour apporter l'alimentation de puissance (et une entrée ou sortie) de 5V.

Au centre, on identifie le L298, entouré de chaque côté (à gauche et à droite) par les diodes de roue libre associées à chaque moteur.

Une précision est nécessaire concernant ce Shield : la carte intègre un régulateur 5V (le petit bloc noir en bas à gauche, marqué 78M05). Ce régulateur peut être utilisé ou non (sa mise en marche est contrôlée par le cavalier vert situé juste à côté). Si on décide de le laisser activé, il fournira l'alimentation nécessaire pour la logique du L298. En revanche, si on le désactive, on devra lui fournir les 5V nécessaires pour la logique. Dans tous les cas, il est indispensable de relier les masses de puissance et de logique entre l'Arduino et le Shield afin de garantir un référentiel commun. On commande directement le pont en H. Par conséquent, on a besoin de trois broches par moteur, deux pour gérer la direction et le frein, et une (PWM) pour la vitesse.

4.2.3 Moteur de petite puissance

Le moteur utilisé dans notre réalisation est un moteur à courant continu à aimant permanent qu'on peut alimenter avec 12 V comme le montre la figure 4.4



Fig 4.4 Moteur utilisé dans la réalisation

4.2.4 Capteur de vitesse HC-020K

Le HC-020K est un capteur qui peut effectuer des mesures avec une résolution élevée et un temps de réponse court, il permet de calculer la vitesse et de déterminer la position des roues. Le diamètre du trou pour l'axe est de 4 mm (). La figure 4.5 présente la photo du capteur utilisé :



Fig 4.5 Capteur de vitesse HC-020K

4.2.5 MatLab / Simulink

Matlab, qui signifie "Matrix Laboratory", est un langage de programmation et un environnement de développement conçu par MathWorks. Il est particulièrement apprécié pour les calculs numériques, l'analyse de données, l'algèbre linéaire, le traitement du signal et de l'image, et les visualisations graphiques. Avec ses boîtes à outils spécialisées, il est largement utilisé dans les domaines scientifiques et d'ingénierie pour les simulations et la modélisation [22].

4.2.5.1 Simulink

Est un environnement de simulation et de modélisation graphique qui permet aux utilisateurs de concevoir, de simuler, de tester et de raffiner des systèmes dynamiques et embarqués. Utilisant une interface glisser-déposer, Simulink facilite la représentation graphique de systèmes linéaires et non linéaires et la simulation de leur comportement dans le temps [23].

4.2.5.2 Arduino Support Package

Le support package Arduino est un ensemble de logiciels, de bibliothèques et de ressources qui permettent d'intégrer et de programmer des cartes Arduino avec divers environnements de développement (IDE) tels que MatLab/Simulink. Il fournit une interface conviviale et simplifiée pour interagir avec les broches d'entrée/sortie, les capteurs et les actionneurs d'une carte Arduino. Grâce à la, nous pouvons développer des algorithmes personnalisés, capturer des données en temps réel à partir des capteurs, contrôler des actionneurs et créer des applications interactives en utilisant les fonctionnalités de MatLab/Simulink. Cela permet une intégration facile entre les fonctionnalités avancées de MatLab/Simulink et la simplicité et la flexibilité des cartes Arduino [24].

4.3 Réalisation

Cette étude présente la réalisation d'un projet impliquant l'utilisation d'un système intégré Arduino Mega pour la commande de vitesse d'un moteur à courant continu. La plateforme matérielle utilisée est constituée d'un Arduino Mega, d'un capteur de vitesse HC-020K et d'un hacheur L298N. L'objectif de cette partie est d'exploiter ces composants pour commander efficacement la vitesse du moteur à courant continu.

La première partie consiste en simulation dans MatLab du moteur à courant continu de petite puissance en boucle ouverte, à vide et en charge. Ces simulations permettent d'analyser le comportement du système sous différentes conditions de charge et de fournir une base pour la conception des régulateurs. Dans la seconde partie, deux types de régulateurs - un régulateur Proportionnel-Intégral (PI) et un régulateur RST - sont mis en œuvre pour contrôler la vitesse du moteur. Ces régulateurs sont conçus, simulés et optimisés dans MatLab, avec une mise en œuvre finale sur le système Arduino.

L'interaction entre le logiciel (MatLab) et le matériel (Arduino, capteur de vitesse, hacheur) est examinée en détail, avec une attention particulière portée à la précision, l'efficacité et la fiabilité du système de commande de vitesse. La figure suivante présente une photo de la maquette développée dans notre travail. Les composants illustrés ont été tous décrits dans les parties précédentes de ce mémoire.

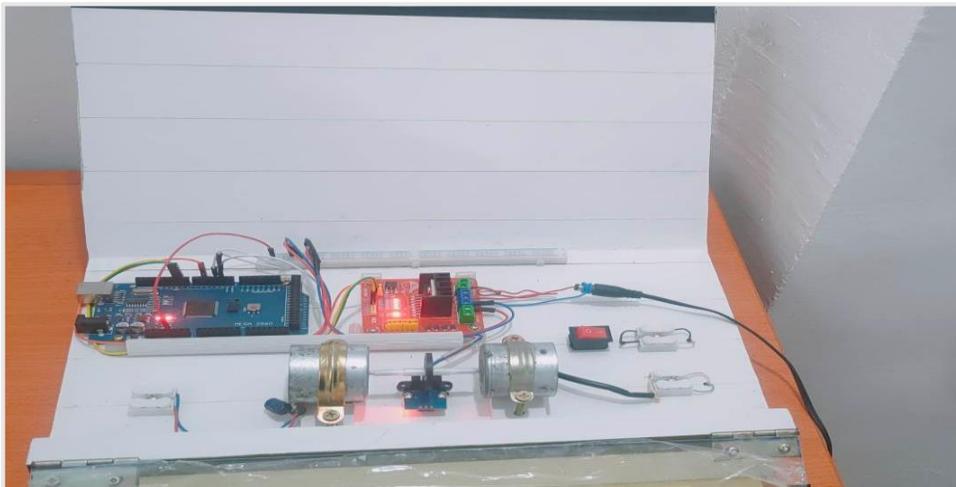


Fig 4.6 Maquette développée

4.4 Révélées expérimentales

4.4.1 Essais en boucle ouverte (moteur avec génératrice)

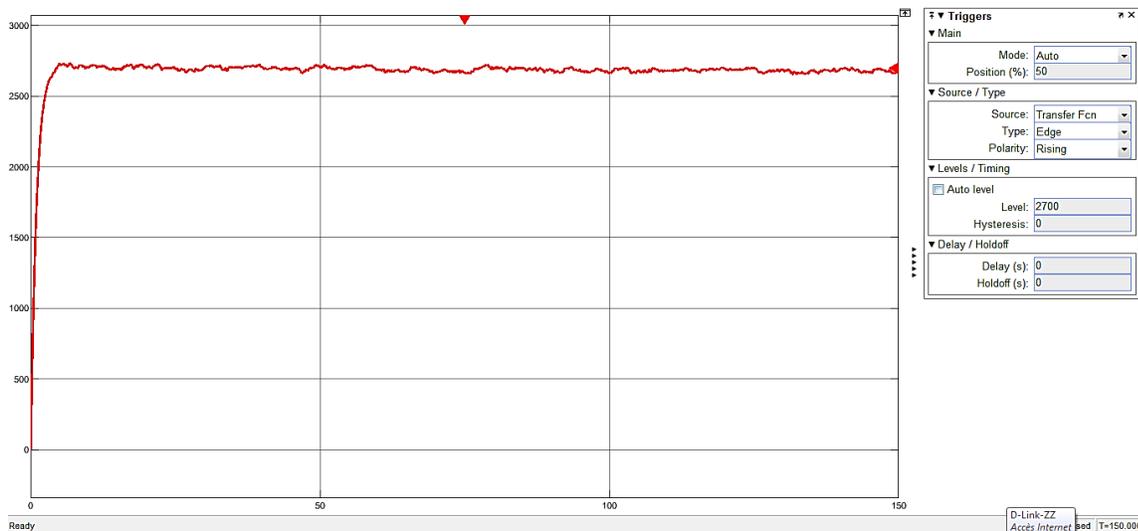


Fig 4.7 Résultats de simulation en boucle ouverte à vide

La figure 4.7 montre l'enregistrement de la réponse en boucle ouverte. On remarque que le moteur atteint sa valeur finale (2700tr/min) avec un temps de réponse de 9.45 seconde puis il presque se stabilise avec un bruit.

4.4.2 Essais en boucle ouverte (avec génératrice et changement de vitesse)

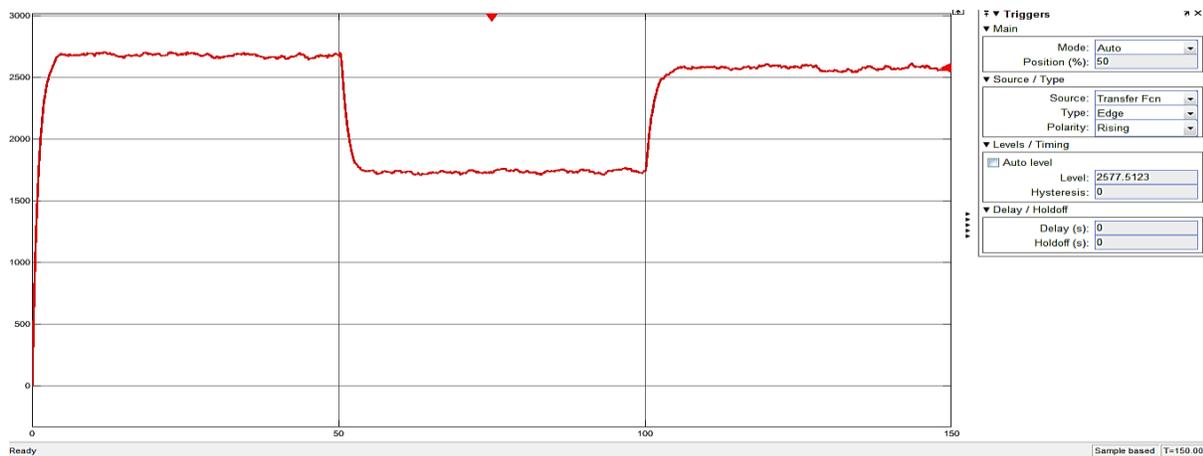


Fig 4.8 Démarrage et changement de vitesse à vide

On a variée la vitesse de **2700 à 1750 tr/min** à l'instant ($t = 50$ s) et de **1750 à 2500 tr/min** à l'instant ($t = 100$ s), le temps nécessaire quand on a diminué la vitesse pour atteindre la valeur désirée est de 5.65s et presque le même quand on augmente.(voir la figure 4.8).

4.4.3 Essais en boucle fermée (régulateur PI avec génératrice à vide)

Quand on utilise un régulateur PI, il est notable que le temps de réponse de la vitesse atteindre son but tôt en 4,98 s, mais, on remarque qu'il y a un dépassement qui est référé au régulation afin de la stabilisation en régime permanent avec moins de bruit, comme le montre la figure 4.9.

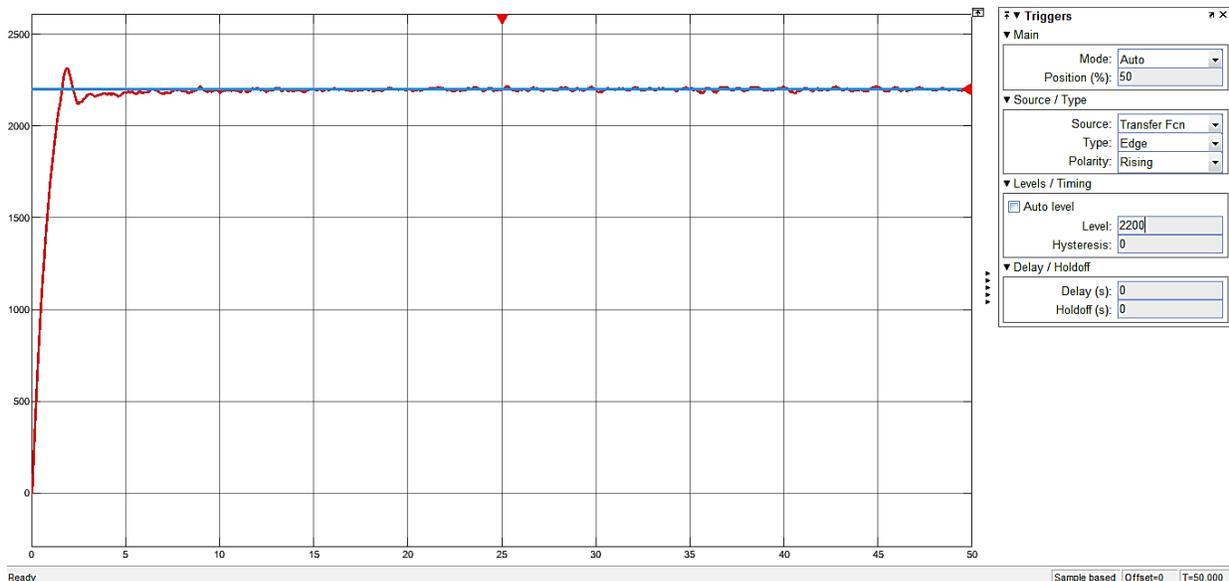


Fig 4.9 Réglage de vitesse avec un PI (à vide)

4.4.4 Essais en boucle fermée (régulateur PI avec génératrice et charge)

La figure 4.10 illustre l'enregistrement de cet essai. En appliquant un couple résistant C_r à l'instant ($t = 30.39$ s), on remarque que la vitesse diminue à cause du couple résistant appliqué (branchement une **diode LED** comme charge au niveau sortie de génératrice), mais le régulateur va régler la vitesse pour suivre la consigne.

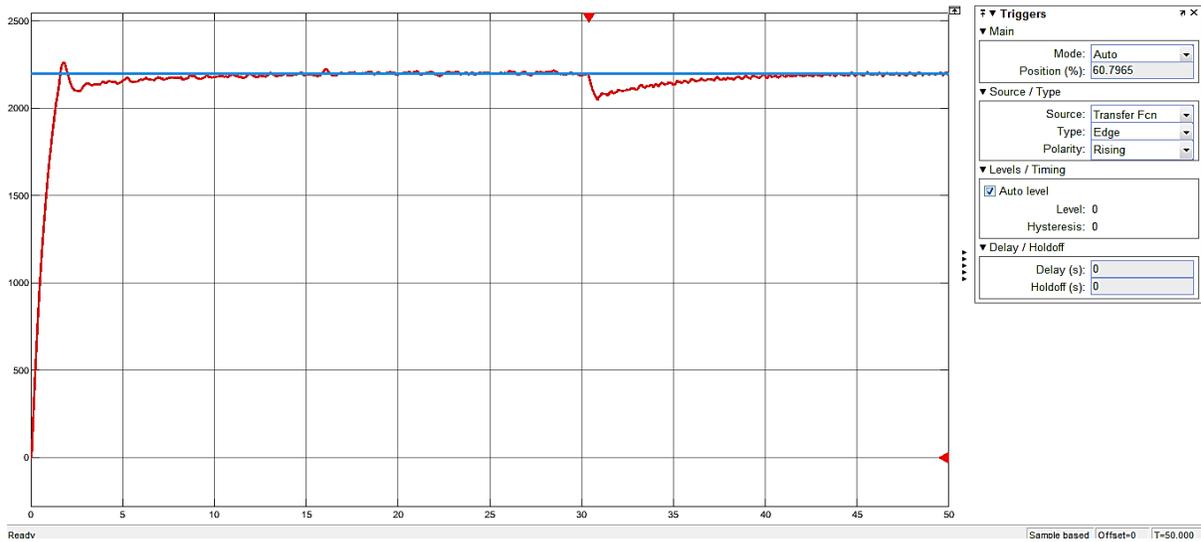


Fig 4.10 Réglage de vitesse avec un PI (en charge)

4.4.5 Essais en boucle fermée (régulateur PI avec génératrice et variation de vitesse)

En faisant une variation du vitesse de **2400 à 1350** à l’instant (**t = 50s**) et de **1350 à 1800** à l’instant (**t = 100s**) , on peut noter que la vitesse diminue ou augmente presque instantanément à la valeur désirée mais avec un dépassement avant la stabilisation, comme le montre la figure 4.11.

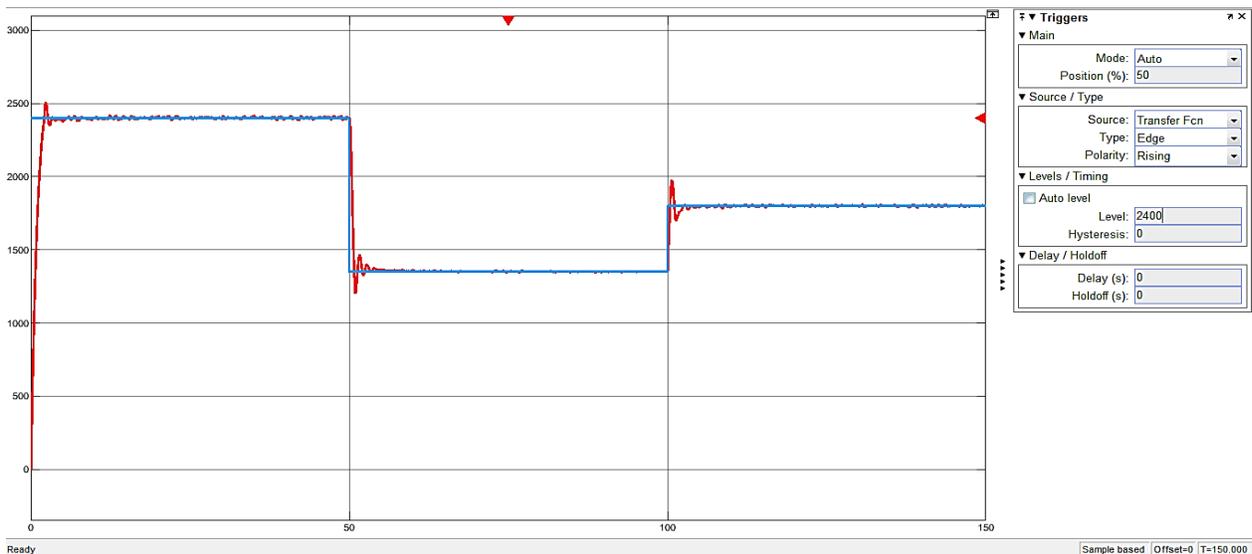


Fig 4.11 Variation de vitesse avec un PI

4.4.6 Essais en boucle fermée (régulateur RST avec génératrice à vide)

Dans ce cas, on a varié la vitesse de référence de **2000 à 1500** à l'instant ($t = 50s$) et de **1500 à 2300** à l'instant ($t = 100s$). Le dépassement et le temps de réponse sont moins importantes par rapport au résultat du PI, mais lors de l'augmentation, la valeur de la vitesse ne montre aucun dépassement on que le temps de réponse augment, comme le montre la figure 4.12.

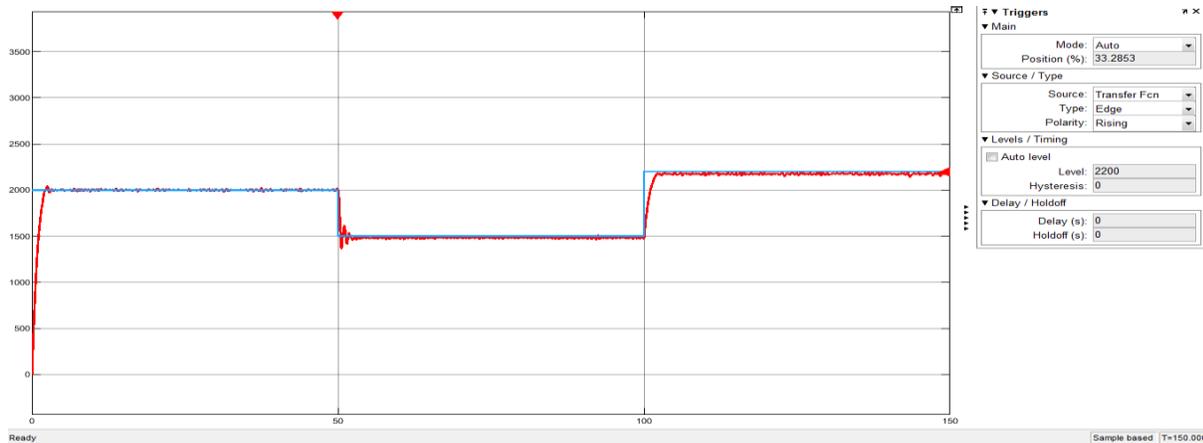


Fig 4.12 Réglage de vitesse avec un RST (à vide)

4.4.7 Essais en boucle fermée (régulateur RST avec perturbation)

En introduisant un couple résistant C_r à l'instant ($t = 77.51s$), on observe que la vitesse diminue en raison du couple résistant mis en œuvre. Cependant, le régulateur intervient pour ajuster la vitesse afin qu'elle corresponde à la consigne.

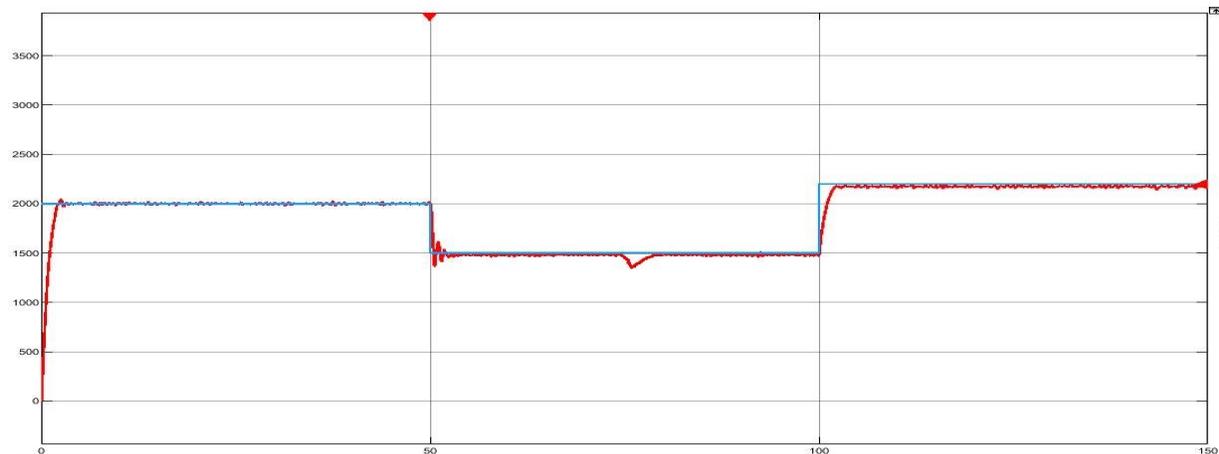


Fig 4.13 Réglage de vitesse avec un RST (avec perturbation)

CONCLUSION GENERALE

Notre travail concerné la modélisation détaillée, le réglage et la mise en pratique d'une chaîne à vitesse variable de la machine à courant continu à aimants permanents. L'identification des paramètres de la machine a été priorisée, facilitant son utilisation dans la partie expérimentale. Nous avons profité de l'extension du logiciel MatLab pour utiliser son application "Parameter Estimator " par définir les paramètres de notre moteur. Pour le contrôle de la vitesse, la méthode de compensation des pôles a été utilisée pour déterminer les valeurs des paramètres du régulateur, ainsi que l'application dédiée au régulateur PID par l'icone "Tunning". Celles-ci, nous a fournissant un contexte précieux pour discuter du comportement du moteur, ou élément central de ce travail a été l'accent mis sur l'amélioration des performances de réglage de vitesse. Il a été proposé que le régulateur PI puis remplacé par un régulateur RST.

Ce travail nous a donné l'occasion d'approfondir nos connaissances dans la compréhension de concepts tels que la discrétisation, le régulateur PID discret et la commande RST a été approfondie, tous ces éléments étant cruciaux pour le projet de contrôle du moteur à courant continu. La transition d'un modèle en temps continu à un modèle en temps discret, le régulateur PID discret offrant un équilibre entre une bonne réponse transitoire et une faible erreur en régime permanent, et l'approche avancée de contrôle RST ont été étudiés en détail.

La dernière partie de notre travail a été consacré sur l'application pratique (réalisation d'une maquette expérimentalement) de ces théories, avec l'implémentation des techniques étudiées et la réalisation de tests expérimentaux pour évaluer les performances du système de contrôle. Ce travail expérimental permettra non seulement de valider les théories abordées, mais aussi de souligner les forces et les faiblesses de chaque méthode, et, finalement, de perfectionner le fonctionnement du moteur à courant continu à aimants permanents.

RÉFÉRENCES

- [1] Chaouch Souad, Machine à courant continu, Cours. Université de Batna, 2020.
- [2] Abdallah Ismail , Ahmed Belkacem Ilyes, Contrôle du sens de la vitesse d'un moteur à courant continu sous interface graphique (GUI MatLab), Mémoire de fin d'études de Master en électrotechnique, Centre Universitaire Belhadj Bouchaib d'Ain-Temouchent, 2020
- [3] Bensaoula Mohamed Amine, Conception du Contrôleur PID pour le Moteur à Courant Continu (MCC) à Excitation indépendante Simulation sous MatLab/Simulink, Mémoire de fin d'études de Master en Électromécanique, Université Badji Mokhtar Annaba,
- [4] Bouazza Mohamed Amine , Mouhous Samir, Etude et simulation de la commande en vitesse d'un moteur à courant contenu sous Proteus, Mémoire de fin d'études de Master en automatique, Université ibn-Khaldoun de Tiaret, 2019.
- [5] Rezgui Maroua , Zelghi Aya, Commande adaptative d'un moteur à courant continu, Mémoire de fin d'études de Master en électrotechnique, Université Mohamed Khider de Biskra, 2022
- [6] Ayeb Fakhreddine, Conception et implémentation d'une commande PID numérique d'un moteur à courant continu, Mémoire de fin d'études de Master en génie électrique, Université Larbi Ben M'hidi D'oum El Boguai, 2014.
- [7] Mohamed Lamine Hamida, Farid Kadi, Identification et contrôle de la machine à courant continu par la Dspace 1103, Mémoire de fin d'études de Master en génie électrique, Université Mouloud Mammeri De Tizi-Ouzou, 2015.
- [8] <https://webetab.ac-bordeaux.fr/Pedagogie/Physique/Physico/Electro/e07mesur.htm>
- [9] <https://www.mathworks.com/discovery/parameter-estimation.html>

REFERENCES

- [10] <https://www.mathworks.com/videos/estimating-dc-motor-parameters-97057.html>
- [11] Slimani Sofiane, Hachour Ahcen, Contribution à l'Implémentation d'un Régulateur Flou Simplifié sous un API S7-300, Mémoire de Fin d'Etude de MASTER ACADEMIQUE en Automatique et Informatique Industriel, Université Mouloud Mammeri De Tizi-Ouzou, 2018.
- [12] <https://www.mathworks.com/help/control/ref/pidtuner.html>
- [13] Frédéric Gouaisbaut, La commande RST, Cours , Université Toulouse Iii-Paul Sabatier, 2007
- [14] Hugues Garnier, Systèmes à temps discret, Cours, Université de Lorraine.
- [15] Annemarie Kökösy, Conception de régulateurs numériques, Cours, 2006.
- [16] Youness El Abbari, Taibi Zidouz, Synthèse des correcteurs numériques par la méthode polynomiale R-S-T, Mémoire de fin d'études de Master en génie des procédés, Université Sidi Mohamed Ben Abdellah - Ecole Supérieure de Technologie Fès, 2009.
- [17] Milat Aimen, Application de la commande RST Robuste pour l'asservissement de vitesse d'une Machine à Courant Continu, Mémoire de fin d'études de Master en électrotechnique, Université Mohamed Seddik – Jijel, 2020.
- [18] <https://www.arduino.cc/en/about>
- [19] <https://store.arduino.cc/products/arduino-mega-2560-rev3>
- [20] M. Ben Ahmed Bouzid, M. Chelbi Farid, Implémentation sur DSP F28335 des Stratégies de Commande d'un Convertisseur DC/AC, Mémoire de fin d'études Master en électromécanique, Université A. MIRA-BEJAIA, 2018.

REFERENCES

[21] Bensayah Amel, Contrôle de vitesse d'un moteur à courant continu, Mémoire de fin d'études de Master en électrotechnique, Université Larbi Ben M'hidi – Oum El Bouaghi, 2016

[22] <https://fr.mathworks.com/products/matlab.html>

[23] <https://fr.mathworks.com/products/simulink.html>

[24] <https://www.mathworks.com/hardware-support/arduino-matlab.html>