

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEURET DE LA RECHERCHE

SCIENTIFIQUE

UNIVERSITE KASDI MERBAH OUARGLA

Faculté des Sciences appliquées

Département de Génie électrique

Mémoire fin d'études

MASTER ACADEMIQUE

Domaine: Sciences et technologie

Filière: Génie électrique

Spécialité : Electrotechnique industrielle

Présenté Par :

Djebrit mohammed said & Siradj abelmonime

Thème :

**Application de l'apprentissage par renforcement à  
la commande d'un MCC**

Soutenu publiquement le : 13/06/2023

Devant le jury composé de:

Dr. Naceur sonia

MCA (UKM Ouargla)

Président

Dr. Taïbi Djamel

MAA (UKM Ouargla)

Examinatrice

Dr Benmakhlouf abdesalem

MAA (UKM Ouargla)

Encadreur

Année Universitaire :2022/2023

## **Remerciement**

*Tout d'abord, je voudrais remercier **ALLAH** Tout-Puissant, le Très Miséricordieux, le Très Miséricordieux, qui m'a donné la force et la patience d'accomplir cet humble travail.*

*Je remercie sincèrement nos chers parents de tout mon cœur pour leurs encouragements tout au long de mon parcours.*

*Je remercie également notre promoteur, **Abdeslam Benmakhlouf***

*qui nous a donné l'opportunité d'explorer ce domaine, et qui nous a guidé tout au long de notre travail.*

*Je remercie également les membres du jury qui ont accepté d'évaluer notre travail.*

*Enfin, je remercie tous mes amis pour leurs encouragements et leur soutien.*

## Résumé:

Cette mémoire a pour but d'étudier et d'améliorer le contrôle de vitesse d'un moteur à courant continu en utilisant la technique d'apprentissage par renforcement. Un algorithme d'apprentissage par renforcement et une simulation dans un environnement dans MATLAB seront utilisés pour développer et évaluer le système de contrôle proposé. Ensuite, une comparaison sera faite entre les performances du moteur utilisant la technique d'apprentissage par renforcement et les performances du moteur utilisant le régulateur PID.

---

**Mots clés:** apprentissage par renforcement, courant continu, agent, algorithme, simulations, moteur à courant continu.

## الملخص:

تهدف هذه المذكرة إلى دراسة وتحسين التحكم في سرعة محرك التيار المستمر باستخدام تقنية التعلم المعزز. سيتم استخدام خوارزمية تعلم معزز والمحاكاة في البيئة مع الوكيل داخل MATLAB لتطوير وتقييم نظام التحكم المقترح. بعد ذلك، سيتم إجراء مقارنة بين أداء المحرك باستخدام تقنية التعلم المعزز وأداء المحرك باستخدام متحكم PID.

---

**الكلمات المفتاحية:** التعلم المعزز, التيار المستمر, الوكيل, خوارزمية, المحاكات, محرك التيار المستمر.

## Abridgement:

This note aims to study and improve the speed control of a DC motor using the reinforcement learning technique. A reinforcement learning algorithm and simulation in environment within MATLAB will be used to develop and evaluate the proposed control system. Next, a comparison will be made between the performance of the engine using the reinforcement learning technique and the performance of the engine using the conventional PID.

---

**Keywords:** reinforcement learning, continuous current, agent, algorithm, simulations, DC motor.

# SOMMAIRE

---

Introduction générale	2
Chapitre I : l'apprentissage par renforcement	
I.1-Introduction :	5
I.2-Apprentissage automatique (Machine learning):	5
I.2.1-Apprentissage supervise :	5
I.2.2-Apprentissage non supervise:	5
I.2.3-Apprentissage par renforcement:	6
I.3-Markov decision process:	7
I.4-Éléments de l'apprentissage par renforcement:	7
I.4.1-Environnement :	7
I.4.2-États :	7
I.4.3-Actions :	8
I.4.4-Modèle de transition:	8
I.4.5-Récompense :	8
I.4.6-Fonction d'utilité :	9
I.4.7-Politique :	9
I.5-algorithmes d'apprentissage	9
I.6-Programmation dynamique:	10
I.7-Méthodes monte carlo:	10
I.8-Les méthodes de différence temporelle :	10
I.8.1-L'algorithme Q-LEARNING :	11
I.8.2-Algorithme acteur-critique:	13
I.8.3-Algorithme Q profond (deep Q algorithm DQN) :	14
I.9-Application d'apprentissage par renforcement :	16
I.9.1-Robotique et contrôle autonome:	16
I.9.2-Communication et réseautage :	16
I.9.3-Système d'autorégulation :	17
I.10-Conclusion:	17
Chapitre II: modélisation de moteur à courant continu	
II.1-Introduction	19

---

<b>II.2-Importance des logiciels assistés par ordinateur pour modéliser les machines électriques :</b> .....	<b>19</b>
<b>II.3-La diversité des types de machines à courant continu et son impact sur la recherche et le contrôle :</b> .....	<b>19</b>
<b>II.3-équations fondamentales des moteurs à courant continu:</b> .....	<b>20</b>
<b>II.4-Modélisation de moteurs à courant continu séparément excités :</b> .....	<b>22</b>
<b>II.4.1-Modélisation à l'aide de schémas fonctionnels :</b> .....	<b>22</b>
<b>II.4.2-Modélisation à l'aide de blocs Simscape :</b> .....	<b>23</b>
<b>II.4.3-Modélisation à l'aide du bloc machine DC intégré :</b> .....	<b>23</b>
<b>II.5-Modélisation des moteurs à courant continu à aimants permanents:</b> .....	<b>24</b>
<b>II.5.1-Modélisation à l'aide de schémas fonctionnels :</b> .....	<b>25</b>
<b>II.5.2-Modélisation à l'aide des blocs Simscape :</b> .....	<b>25</b>
<b>II.5.3-Modélisation à l'aide du bloc machine DC intégré :</b> .....	<b>26</b>
<b>II.6-Modélisation des moteurs cc raccordés en shunt:</b> .....	<b>26</b>
<b>II.6.1-Modélisation à l'aide de diagrammes de blocs :</b> .....	<b>26</b>
<b>II.6.2-Modélisation à l'aide de blocs Simscape :</b> .....	<b>27</b>
<b>II.6.3-Modélisation à l'aide du bloc de machine à courant continu intégré :</b> .....	<b>28</b>
<b>II.7-Modélisation des moteurs à courant continu connectés en série:</b> .....	<b>29</b>
<b>II.7.1-Modélisation à l'aide d'un diagramme de blocs :</b> .....	<b>29</b>
<b>II.7.2-Modélisation à l'aide de blocs Simscape :</b> .....	<b>30</b>
<b>II.7.3-Modélisation à l'aide du bloc de machine à courant continu intégré :</b> .....	<b>30</b>
<b>II.8-Paramètres des machines à courant continu :</b> .....	<b>31</b>
<b>II.9- Conclusions:</b> .....	<b>32</b>
<b>Chapitre III : commande de la vitesse d'un mcc par RL</b>	
<b>III.1-Introduction:</b> .....	<b>33</b>
<b>III.2-Les étapes que vous pouvez suivre pour créer un environnement d'apprentissage par renforcement Simulink pour un moteur à courant continu:</b> .....	<b>33</b>
<b>III.2-Définition de l'environnement d'apprentissage :</b> .....	<b>34</b>
<b>III.2.1-Etat:</b> .....	<b>35</b>
<b>III.2.2-La fonction de récompense (REWARD) :</b> .....	<b>36</b>
<b>III.2.3-L'agent :</b> .....	<b>37</b>
<b>III.3-Créer l'environnements de RI dans Matlab:</b> .....	<b>38</b>
<b>III.3.1-Réseau critique :</b> .....	<b>38</b>

---

<b>III.3.2-Réseau d'acteur :</b> .....	<b>39</b>
<b>III.3.3-schéma final dans Simulink :</b> .....	<b>40</b>
<b>III.3.4-Sélection du modèle de moteur MCC et Paramètres :</b> .....	<b>40</b>
<b>III.4-Entrainement de l'agent (training agent) :</b> .....	<b>40</b>
<b>III.5-Moteur à courant continu à avec contrôleur PID :</b> .....	<b>41</b>
<b>III.5.1-Paramètres de PID contrôleur :</b> .....	<b>42</b>
<b>III.6-La simulation:</b> .....	<b>43</b>
<b>III.6.1-première test :</b> .....	<b>43</b>
<b>III.6.2-La deuxième test :</b> .....	<b>44</b>
<b>III.7-Comparaison de PID avec l'apprentissage par renforcement :</b> .....	<b>45</b>
<b>III.7.1-Avantages du contrôleur PID :</b> .....	<b>45</b>
<b>III.7.2-Avantages de l'apprentissage par renforcement :</b> .....	<b>45</b>
<b>III.8-Conclusion :</b> .....	<b>46</b>
<b>Conclusion générale:</b> .....	<b>48</b>
<b>Les références</b> .....	<b>49</b>

---

# Liste de Figure

---

Figure I.1-Schéma de l'apprentissage par renforcement.....	6
Figure I.2-Actor-critic algorithm .....	13
Figure (II.1)-MCC à excitation séparée.....	21
Figure (II.3)-MCC à excitation série.....	21
Figure (II.4)-MCC à excitation en dérivation .....	21
Figure II.5 -Un modèle d'un moteur à courant continu à excitation séparée utilisant des blocs de fonctions de transfert dans Simulink. ....	23
Figure II.6 -Un modèle de moteur à courant continu à aimant permanent utilisant des blocs Simscape. ....	23
Figure II.7- le bloc de machine DC intégré dans simulink. ....	24
Figure II.8-Un modèle d'un moteur à courant continu à aimant permanent utilisant des blocs de fonctions de transfert dans Simulink. ....	25
Figure II.10-le bloc de machine DC intégré dans Simulink.....	26
Figure II.11-Un modèle d'un moteur à courant continu connecté en dérivation utilisant des blocs de fonction de transfert dans Simulink.....	27
Figure II.12-Un modèle de moteur à courant continu connecté en dérivation utilisant des blocs Simscape .....	28
Figure II.14-Un modèle de moteur à courant continu connecté en série utilisant des blocs de fonction de transfert dans Simulink. ....	30
Figure II.14-Un modèle de moteur à courant continu connecté en série utilisant des blocs Simscape. ....	30
Figure II.15-le bloc de machine à courant continu intégré dans Simulink. ....	31
Figure III.1-L'environnement d'apprentissage .....	34
Figure III.2-Le block de vecteur d'état .....	35
Figure III.3-Le block de vecteur d'état (en détail) .....	36
Figure III.4-bloc de La fonction de récompense.....	37
Figure III.5-bloc de La fonction de récompense.....	37
Figure III.6-l'agent .....	38
Figure III.7-Réseau critique .....	39
Figure III.8-Réseau acteur.....	39
Figure III.9-L'environnement d'apprentissage .....	40
Figure III.10-Résultats de l'entraînement.....	41
Figure III.11-moteur mcc avec contrôleur PID. ....	42

<b>Figure III.12-Paramètres de PID contrôleur. ....</b>	<b>42</b>
<b>Figure III.13-Résultats du premier test et différence entre RL et PID. ....</b>	<b>43</b>
<b>Figure III.14-Résultats du deuxième test et différence entre RL et PID.....</b>	<b>44</b>
<b>Figure III.15-Zoom sur les résultats passés.....</b>	<b>44</b>



## Liste de tableaux

---

Tableau I.1 - L'algorithme <i>Q-Learning</i> .....	12
Tableau I.2 - Actor-Critic Algorithm.....	14
Tableau I.3 - Deep-Q Network Algorithm.....	15
Tableau II 2 - Paramètres de moteurs à courant continu utilisés dans la simulation.....	31

# **Introduction Générale**

## **Introduction générale**

L'apprentissage automatique est l'un des domaines les plus récents. Pour Il existe un système de commande intelligent avec plusieurs outils comme logique floue, réseaux de neurones artificiels et algorithmes génétiques, Il existe également trois grandes familles d'apprentissage, qui sont :

Apprentissage supervisé, non supervisé et apprentissage par renforcement

L'encadrement de ce dernier type est l'un des objectifs de notre recherche.

L'apprentissage par renforcement et l'entraînement des agents consistent à permettre à une machine d'apprendre à prendre des décisions et à agir de manière autonome dans un environnement complexe, en maximisant les récompenses obtenues. Cela se fait en utilisant des techniques d'apprentissage et des algorithmes spécifiques pour ajuster les actions de l'agent en fonction de son expérience et des récompenses reçues. et pour réaliser notre simulation nous avons utilisé du logiciel Matlab.

### **Problématique :**

à travers la technique de l'apprentissage par renforcement , comment peut-on contrôler les moteurs électriques pour obtenir les meilleures performances possibles?

### **Objective :**

Nous sommes toujours confrontés à des problèmes de contrôle des moteurs électriques, y compris ceux à courant continu, notamment dans leur vitesse, nous avons donc choisi l'apprentissage par renforcement, afin d'obtenir de meilleures performances dans la détermination de la vitesse de la séquence en fonction de l'environnement.

### **Organisation du mémoire :**

Le travail présenté dans ce mémoire est donc organisé en trois chapitres qui sont structurés comme suit :

- **Chapitre I :** Ce chapitre explique les concepts d'apprentissage par renforcement et d'apprentissage par renforcement profond et les différentes méthodes utilisées dans ces processus.
- **Chapitre II :** Ce deuxième chapitre est consacré à la modélisation de moteurs à courant continu
- **Chapitre III :** Ce dernier chapitre est consacré au contrôle moteur et à la simulation de ce système.

Et à la fin nous terminons notre travail par une conclusion générale.

**CHAPITRE I:  
APPRENTISSAGE PAR  
RENFORCEMENT**

## **I.1-Introduction :**

L'apprentissage par renforcement est un type d'apprentissage automatique où un agent apprend à prendre des décisions basées sur les commentaires de l'environnement, sous forme de récompenses ou de punitions.

L'objectif est de maximiser la récompense totale au fil du temps. Il est couramment utilisé dans la robotique, les jeux et les voitures autonomes, et présente des avantages dans les situations où il n'y a pas d'ensemble de données préexistant ou de données étiquetées. L'agent apprend de ses interactions avec l'environnement et peut s'adapter aux situations changeantes.

## **I.2-Apprentissage automatique (Machine learning):**

Les progrès de la technologie informatique et l'émergence de nouveaux algorithmes intelligents ont rapproché l'objectif de rapprocher l'intelligence artificielle (IA) de la réalité. L'IA fait référence à la capacité des machines programmables à présenter une intelligence simulée, dans le but d'imiter le cerveau humain. Dans le domaine de l'IA, l'apprentissage automatique (ML) est un sous-domaine qui se concentre sur le développement d'agents logiciels capables de s'améliorer automatiquement grâce à l'expérience [1].

Il existe principalement trois catégories de ML:

### **I.2.1-Apprentissage supervisé :**

L'apprentissage supervisé est un type d'apprentissage automatique dans lequel un algorithme apprend à faire des prédictions basées sur un ensemble de données étiqueté. L'algorithme reçoit un ensemble de paires entrée-sortie et apprend à mapper l'entrée à la sortie. Le processus d'apprentissage consiste à ajuster les paramètres de l'algorithme pour minimiser l'erreur entre ses prédictions et les résultats réels. L'apprentissage supervisé peut être utilisé pour diverses applications et est puissant dans l'apprentissage de modèles et de relations complexes dans les données. Cependant, il nécessite une grande quantité de données étiquetées et peut ne pas bien se généraliser aux nouvelles données invisibles si les données d'apprentissage ne représentent pas la distribution sous-jacente des données.

### **I.2.2-Apprentissage non supervisé:**

L'apprentissage non supervisé est un type d'apprentissage automatique dans lequel l'algorithme identifie des modèles et des relations dans un ensemble de données non étiqueté en

regroupant des points de données similaires ou en réduisant la dimensionnalité des données. Ce type d'apprentissage est souvent utilisé dans l'analyse exploratoire des données, la détection d'anomalies et l'extraction de caractéristiques. L'algorithme est évalué en fonction de la façon dont il a identifié des modèles ou des structures dans les données. L'apprentissage non supervisé présente des avantages tels que la recherche de modèles inconnus et le traitement de divers types de données, mais présente également des limites telles que la difficulté d'évaluer les performances et d'interpréter les résultats. Les applications courantes incluent les systèmes de recommandation, la détection d'anomalies et la reconnaissance d'images et de paroles.

### I.2.3-. Apprentissage par renforcement:

L'apprentissage par renforcement implique qu'un agent apprend à se comporter dans un environnement par le biais d'interactions et de rétroactions sous forme de récompenses ou de punitions. L'environnement est modélisé comme un MDP, et l'objectif est de trouver une politique optimale qui maximise la récompense cumulée. Cela se fait à l'aide de méthodes basées sur la valeur ou sur les politiques. L'apprentissage par renforcement a été appliqué à divers problèmes, mais peut être difficile dans des scénarios réels. Les progrès de l'apprentissage par renforcement profond se sont révélés prometteurs pour relever ces défis

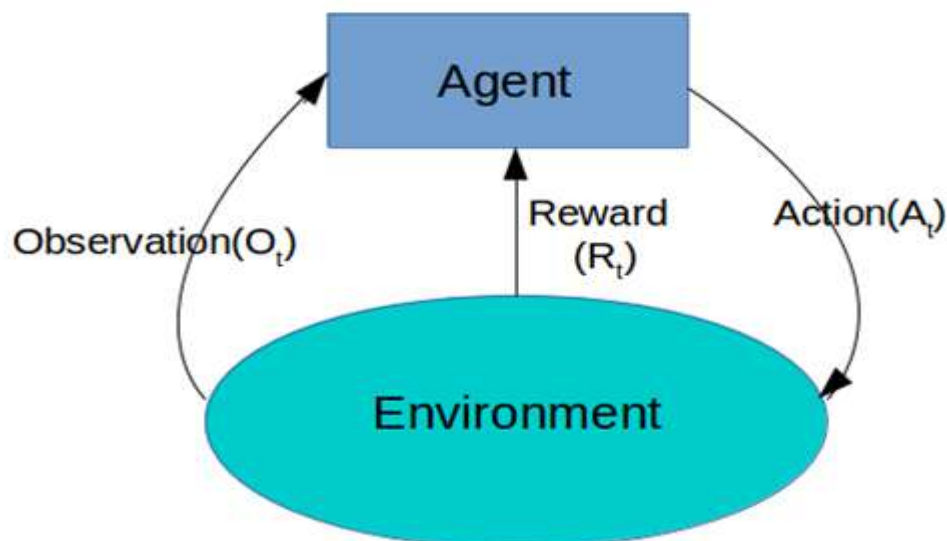


FIGURE I.1- Schéma de l'apprentissage par renforcement.

### **I.3-Markov decision process:**

Le processus de décision de Markov (MDP) est une méthode utilisée pour la prise de décision dans un environnement stochastique, qui est une réinterprétation des chaînes de Markov. L'objectif de MDP est de fournir une cartographie des actions optimales pour chaque état d'un environnement. Il est basé sur la propriété markovienne, où seul le présent compte, et les informations passées ne sont pas prises en compte. La prédiction de l'état suivant est indépendante des états passés, et la physique et les règles de l'environnement restent stationnaires. Les échecs sont un bon exemple de MDP où les règles restent constantes et les coups passés ne sont pas nécessaires pour jouer le coup suivant. Le MDP trouve des applications dans de nombreux domaines, tels que la recherche opérationnelle, la théorie du contrôle, les statistiques, les jeux, l'économétrie, l'IA, la robotique, le contrôle du dialogue, les investissements optimaux, les tests médicaux, la logistique et le contrôle des bioréacteurs. Pour acquérir une compréhension complète de MDP et RL, quelques termes doivent être définis [2].

### **I.4-Éléments de l'apprentissage par renforcement:**

#### **I.4.1 Environnement :**

L'environnement est un aspect vital du processus d'apprentissage par renforcement, car il définit les états, les actions et les récompenses possibles qu'un agent peut expérimenter au cours de son processus d'apprentissage. Définir l'environnement implique de spécifier les règles et les contraintes qui régissent les interactions entre l'agent et l'environnement. L'agent peut explorer l'environnement et apprendre par essais et erreurs pour maximiser les récompenses et atteindre son objectif. Un environnement bien conçu peut faciliter l'apprentissage, tandis qu'un environnement mal conçu peut entraver ou même empêcher l'apprentissage[1].

#### **I.4.2 États :**

Un état est un ensemble de positions, de lieux ou de conditions au sein d'un environnement qu'un agent peut atteindre ou visiter. Par exemple, dans un système de contrôle de la température, l'état pourrait être représenté par la valeur de température actuelle, tandis que dans un robot naviguant dans un bâtiment, un état pourrait être une pièce ou un emplacement particulier. Un état peut être représenté à l'aide de coordonnées ou de nombres/alphabets, et le nombre d'états peut être fini ou infini. Certains états d'un environnement sont considérés comme des états terminaux, ce qui signifie que les atteindre met fin au processus. Lorsqu'un agent passe d'un état à un autre, cela



s'appelle une visite, et plusieurs visites de l'état de départ à n'importe quel état terminal sont appelées un épisode[1].

### **I.4.3 ACTIONS:**

Les actions font référence à l'ensemble de choix ou de comportements qu'un agent ou un robot peut effectuer dans un environnement particulier. Par exemple, dans un monde en grille, un agent peut choisir de se déplacer dans l'une des quatre directions cardinales, ou il peut choisir de rester dans son état actuel. De même, dans un environnement à une seule ligne horizontale, l'agent peut se déplacer à gauche ou à droite. Généralement, un agent peut opter pour n'importe quelle action autorisée dans un scénario donné[1].

### **I.4.4 MODÈLE DE TRANSITION:**

Les règles, la dynamique ou la physique de l'environnement sont définies par la fonction de transition, notée  $T(s, a, s_0)$ , qui comprend l'état actuel, l'action choisie et le nouvel état résultant. La fonction de transition spécifie la probabilité d'une action dans un état donné, c'est-à-dire la probabilité de passer de l'état actuel  $s$  au nouvel état  $s_0$  tout en effectuant une action  $a$  et en recevant une récompense  $R$ . Par conséquent, le modèle de transition est basé uniquement sur l'état actuel, l'action choisie et l'état résultant, suivant la propriété de Markov, qui est indépendante des états et actions passés[1].

### **I.4.5 RÉCOMPENSE:**

La fonction  $R(s)$  est appelée la fonction de récompense qui attribue une valeur numérique à l'agent pour être dans un état après avoir effectué une action. Il sert de rétroaction à l'agent pour déterminer si un état est bénéfique ou non. Un agent est plus susceptible d'évoluer vers des états utiles avec des récompenses plus élevées et d'éviter les indésirables avec des récompenses plus faibles. La nature et le montant de la récompense peuvent affecter les actions et la politique de l'agent, et même des changements mineurs peuvent avoir un impact significatif sur son comportement[1].

la récompense future cumulative également appelée rendement est donnée comme suit :

$$R_t = r_{t+1} + r_{t+2} + r_{t+3} + \dots + r_{\infty} \dots \dots \dots (1)$$

Dans ce contexte, 't' en indice représente une instance particulière dans le temps. Notre objectif est de conclure une série de récompenses au lieu de rendements infinis. Nous nous

concentrons sur les récompenses cumulatives à prix réduit et les tâches épisodiques. Ces tâches sont celles qui se terminent après un certain pas de temps ou lorsque l'agent atteint un état terminal. L'utilité d'un état peut être représentée par la récompense actualisée à long terme.

#### I.4.6 FONCTION D'UTILITÉ :

L'agent reçoit une récompense à chaque état et nous pouvons l'utiliser

pour calculer la fonction d'utilité de l'histoire des états telle que définie dans

équation (2).

$$U_h = R(s_0) + \gamma R(s_1) + \gamma^2 R(s_2) + \gamma^3 R(s_3) + \dots + \gamma^n R(s_n) \dots (2)$$

$\gamma$  est un facteur d'actualisation compris entre zéro et un. L'utilité d'un état unique est définie comme suit:

$$U(s) = \sum_{t=0}^{\infty} \gamma^t R(st) \dots (3)$$

#### I.4.7 POLITIQUE :

Une politique agit comme un guide pour orienter un agent vers une action appropriée dans un état donné. Il sert de carte, reliant un groupe d'états à un ensemble d'actions correspondant. Une politique optimale maximise la récompense totale qu'un agent pourrait accumuler au cours de sa vie. Bien qu'un plan puisse être déduit d'une politique, ils diffèrent les uns des autres. Un plan dicte la séquence d'actions de l'état de départ à l'état de destination, tandis qu'une politique conseille le meilleur plan d'action qu'un agent doit adopter dans un état ou une situation spécifique. Cela découle de la propriété markovienne qui repose uniquement sur l'état actuel. Par conséquent, une politique guide un agent sur la façon d'agir. Afin de déterminer la politique optimale, il est nécessaire de connaître l'équation de Bellman, qui sera présentée dans la section suivante.

$$U(s) = R(s) + \gamma \max_a \sum_{s'} T(s, a, s') U(s') \dots (4)$$

### I.5- algorithmes d'apprentissage

Les algorithmes d'apprentissage par renforcement (RL) peuvent être classés comme basés sur des politiques, basés sur des valeurs ou une combinaison des deux. Ils peuvent également être

classés comme sans modèle, qui s'appuie sur des essais et des erreurs pour mettre à jour les connaissances, ou basés sur des modèles, qui utilisent des modèles déjà appris pour faire des prédictions. Le RL basé sur un modèle permet à un agent de prédire les états et les récompenses correspondantes. Des exemples d'algorithmes sans modèle sont Monte Carlo et TD.

### **I.6- Programmation dynamique:**

La programmation dynamique est un ensemble de méthodes permettant de calculer une politique optimale dans un MDP connu, en utilisant les propriétés des équations de Bellman. Nous supposons donc dans cette section que le modèle de l'environnement est connu. La programmation dynamique va chercher à estimer la fonction de valeur optimale  $V^*$  afin d'en déduire une politique optimale. [2]

### **I.7-Méthodes monte carlo:**

la technique Monte Carlo (MC) utilise le hasard pour résoudre un problème. MC est une méthode sans modèle qui apprend de l'épisode complet (pas d'amorçage) utilisant l'idée de moyenner.

Nous pouvons utiliser la méthode MC pour la prédiction et le contrôle et nous distinguons le RL passif et actif respectivement. Dans le RL passif, le but de l'agent est d'apprendre la valeur de la fonction de valeur à l'aide d'une stratégie en RL actif, l'agent essaie de trouver une politique optimale lors de l'interaction avec l'environnement. Les avantages des méthodes MC par rapport à celles du RL sont les suivants :

- Il peut être appliqué avec des modèles d'échantillons (simulations).
- Les méthodes MC sont faciles à mettre en œuvre et pour les environnements avec de petits sous-ensembles d'états.
- MC trouve des solutions optimales, via une interaction directe avec le monde.

### **I.8- Les méthodes de différence temporelle :**

Le terme apprentissage par différence de temps (TD) fait référence à une classe de méthodes d'apprentissage par renforcement sans modèle qui apprennent en s'amorçant à partir de l'estimation actuelle de la fonction de valeur.

Ces méthodes échantillonnent l'environnement, comme les méthodes de Monte Carlo, et se mettent à jour en fonction des estimations existantes, comme les méthodes de programmation dynamique. Cependant, alors que les méthodes de Monte Carlo n'ajustent leurs estimations que lorsque le résultat final est connu, les méthodes TD ajustent leurs prédictions pour s'aligner ultérieurement sur des prédictions plus précises concernant l'avenir avant que le résultat final ne soit connu [1].

### I.8.1-L'algorithme Q-LEARNING :

Q-LEARNING est un modèle et d'apprentissage prospectif hors politique de la TD

Algorithme pour le contrôle. Sa règle de mise à jour est donnée comme suit :

$$Q(st, at) \leftarrow Q(st, at) + \sigma [rt+1 + \gamma \max_a Q(st, at)] \dots\dots\dots (5)$$

Dans l'algorithme Q-learning, l'action future  $a$  est prise en utilisant une politique gourmande, c'est-à-dire choisir une action qui a un maximum Valeur Q de l'état suivant. La convergence dans le Q-learning peut être accéléré en utilisant des traces d'éligibilité qui sont mises à jour dans Q-learning( $\lambda$ ) comme suit [1].

$$et(s, a) = I_{sst} . I_{aat} + \gamma \lambda et - 1(s, a) \dots\dots\dots (6)$$

$$\text{Si } Q_{t-1} = \max_a Q_{t-1}(st, a) \dots\dots\dots(7)$$

$$\text{Si non } et(s, a) = 0 \dots\dots\dots(8)$$

$I_{sst}$  est utilisé ici pour l'indication d'identité. Pour  $s = st$

, les  $I_{sst}$  et  $I_{aat}$  sont égaux à 1. Le  $\delta$  pour le Q-learning est défini comme :

$$\delta s = rt+1 + r \max_a Q_t(st+1, a) - Q_t(st, at) \dots\dots\dots (9)$$

La règle de mise à jour résultante pour l'algorithme Q-learning par

utiliser trace est :

$$Q_{t+1}(s, a) = Q_t(s, a) + \alpha \delta t et(s, a) \text{ for all } s \in S \dots\dots\dots (10)$$

**Algorithme 1.1** : Le Q-Learning

---

```

/*  $\alpha_n$  est un taux
d'apprentissage */
Initialiser( $Q_0$ )

Pour  $n \leftarrow 0$  jusqu'à  $N_{tot}-1$  faire

     $s_n \leftarrow$  Choix-Etat

     $a_n \leftarrow$  Choix-Action

     $(s'_n, r_n) \leftarrow$  Simuler( $s_n, a_n$ )

    {mise à jour
    de  $Q_n$  :}
    Début

         $Q_{n+1} \leftarrow Q_n$ 

         $\delta_n \leftarrow r_n + \gamma \max_b (s'_n, b) - Q_n(s_n, a_n)$ 

         $Q_{n+1}(s_n, a_n) \leftarrow Q_n(s_n, a_n) + \alpha_n(s_n, a_n) \delta_n$ 

    fin

retourner  $Q_{tot}$ 

```

---

Tableau I.1 : L'algorithme *Q-Learning*

**I.8.2-Algorithme acteur-critique:**

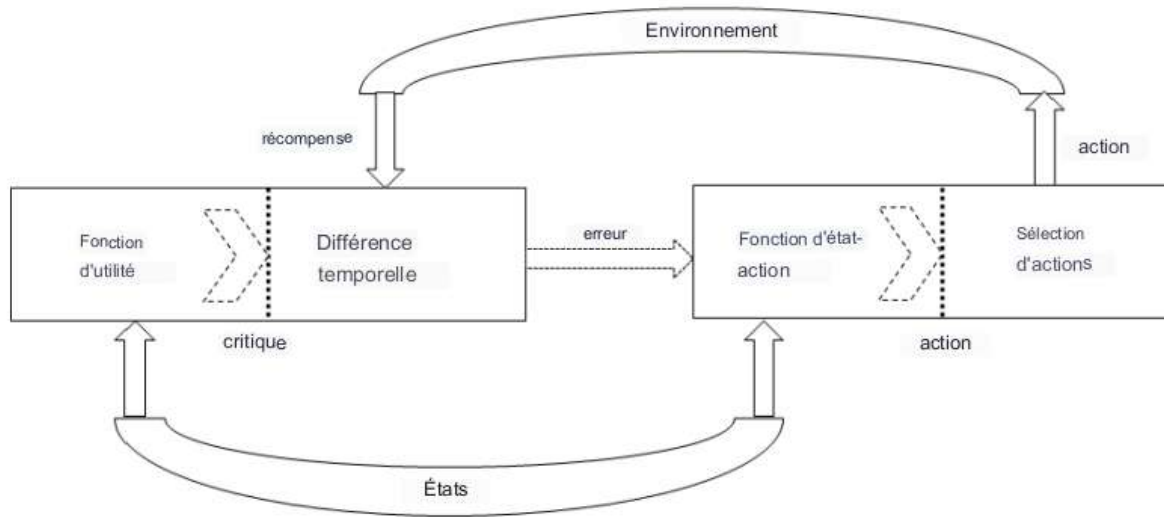
Acteur-Critique (AC) est une approche hybride basée à la fois sur politique et fonction de valeur. La fonction de valeur est estimée par le critique comme décrit par l'équation (28) tandis que la politique est mise à jour par l'acteur sur la base des commentaires du critique.

La méthode AC peut être utilisée aussi bien pour les petites grands espaces d'action étatique. Pour les problèmes ayant un petit état et l'espace d'action, la critique est estimée par la fonction Q.

tandis que la politique de Boltzmann ou – cupide est utilisée pour acteur l'estimation des politiques. [3]

$$V^\pi (s') = \sum_a \pi (s, a) \sum_{s'} p( s' | s, a ) [R (s | s', a ) + \gamma V^\pi (s' ) ] \dots\dots\dots (11)$$

$$Q^\pi (s, a) = \sum_{s'} p( s' | s, a ) [R (s | s', a ) + \gamma Q^\pi (s', a' ) ] \dots\dots (12)$$



**FIGURE I.2-Actor-critic algorithm**

**Algorithm 2: Actor-Critic Algorithm [2]**

Initialisation

Récompenses pour les paires état-action  $R_{s,a}$ 

$$\gamma = 0.9$$

initialiser  $s$ 1. sélectionnez  $a$ à  $st$ 2. Obtenez  $st+1$ 3. Obtenez le premier,  $a$ 4. Mettre à jour la fonction utilitaire de l'état  $st$  (critique)

$$U(st) \leftarrow U(st) + \alpha [rt+1 + \gamma U(st+1) - U(st)]$$

5. Mettre à jour la probabilité  $a_t$  en utilisant l'erreur (acteur)

$$\delta = rt+1 + \gamma U(st+1) - U(st)$$

Tableau I.2 Actor-Critic Algorithm

**I.8.3-Algorithm Q profond (deep Q algorithm DQN) :**

Deep Q-Network (DQN) est considéré comme le principal utilisateur de réseau neuronal (NN). Objectif principal de cet algorithme était de traiter les différentes variables environnementales ayant un grand espace d'état. L'estimation des variables est l'objectif principal à traiter. L'équation (30) se concentre sur le déploiement Méthode Q-learning pour mettre à jour les variables us :

$$g \text{ NN. } r_j + \gamma \max_a Q(\phi_{j+1}, a) \dots \dots \dots (13)$$

La méthode Deep Q-Network (DQN) est présentée dans l'algorithme 6. L'expérience Replay conserve les expériences qui contiennent l'action, les transitions d'état correspondantes et les récompenses et des mini-lots sont faits pour la mise à jour de NN. L'estimation de la valeur Q nécessite un modèle basé sur l'instabilité des paramètres. Basé sur NN, la production de modèles aboutit à l'estimation de la fonction de valeur Q. A cet effet, deux NN sont utilisés normalement et nommé d'après la valeur Q et l'autre est nommé d'après la cible Q Network [4].

Au cours des dernières années, l'apprentissage par renforcement profond a attiré l'attention de nombreux chercheurs en raison de sa capacité à apprendre des actions et à être capable de travailler dans des environnements dynamiques en constante évolution et contrôler un manipulateur robotique comme un hélicoptère ou jouer à un jeu informatique sont quelques applications de RL profond.

**Algorithm 3: Algorithme Q profond**

Configurer RM

Initialiser  $Q(s,a)$

Initialiser la cible

**Répéter**

Initialiser la séquence  $s_1 = x_1$  et précédée

Séquence  $\theta_1 = \theta(S_1)$

Choisissez au hasard  $a_i$  avec probabilité ou sélectionnez à

Choisissez  $a_i = \operatorname{argmax}_a (Q(\varphi(st), a; \theta))$

Obtenir  $r_{t+1}$  et  $x_{t+1}$

$$st+1 = (st, at, xt+1)$$

$$\varphi_{t+1} = \varphi(st+1)$$

Transition de magasin  $(\varphi_t, at, r_{t+1}, \varphi_{t+1})$  dans RM

Exemple de minilot aléatoire de transitions  $(\varphi_t, at, r_{t+1}, \varphi_{t+1})$  de RM

**Si** l'épisode se termine à l'étape  $k+1$  alors

$$y_k = r_k$$

**Autre**

$$y_k = r_k + \gamma \max_a Q(\varphi_{k+1}, a, \theta)$$

**Fin**

Avoir une étape de descente de pente sur

$$(y_k - Q(\varphi_k, a_k, \theta))$$

Chaque jeu d'étapes  $C \quad Q^* = Q$

**jusqu'à**  $t = 1, T$  ;

**jusqu'à** l'épisode = 1, M;

Tableau I.3 Deep-Q Network Algorithm



## **I.9- Application d'apprentissage par renforcement :**

Cette section présente quelques applications importantes de l'apprentissage par renforcement dans les soins de santé, la robotique et l'autonomie, contrôle, communication et mise en réseau, langage naturel, traitement, jeux et système auto-organisé, planification, gestion et configuration des ressources, IoT et vision par ordinateur[5].

### **I.9.1-Robotique et contrôle autonome:**

La robotique et le contrôle autonome sont des domaines d'études concernés par la conception, le développement et le contrôle de systèmes robotiques. La robotique implique la création et le fonctionnement d'appareils mécaniques qui peuvent être programmés pour effectuer des tâches de manière autonome ou avec une télécommande.

Le contrôle autonome fait référence à la capacité d'un système robotique à fonctionner de manière indépendante, sans nécessiter d'intervention humaine constante. Cela nécessite l'utilisation de capteurs, d'algorithmes et d'intelligence artificielle pour permettre au robot de percevoir et d'interagir avec son environnement de manière significative.

Les applications de la robotique et du contrôle autonome sont vastes et diverses, allant de l'automatisation industrielle à l'exploration spatiale, de la chirurgie médicale à l'assistance à domicile. Les systèmes robotiques ont le potentiel d'améliorer l'efficacité, la sécurité et la qualité de vie dans une variété de domaines, ce qui en fait un domaine crucial de recherche et de développement à l'ère moderne.

### **I.9.2-Communication et réseautage :**

L'apprentissage par renforcement, en particulier le RL profond, peut potentiellement être appliqué pour résoudre de nombreux défis et problèmes liés à la communication et au réseautage. Réseaux modernes tels que les véhicules aériens sans pilote (UAV), hétérogènes Les réseaux (HeNets) et l'Internet des objets (IoT) peuvent être Entités réseau IF autonomes, ad hoc et décentralisées c'est-à-dire que les UAV, les utilisateurs mobiles et les appareils IoT sont capables de prendre décisions autonomes, p. ex. choix du débit de données, spectre Accès, association de station de base et contrôle de la puissance d'émission.

Ces dernières années, la RL profonde a été appliquée dans les communications

et les réseaux réussissent à atteindre l'objectif souhaité.

### **I.9.3 Système d'autorégulation :**

dans un système de chauffage autorégulateur, l'agent d'apprentissage par renforcement peut apprendre à ajuster la puissance de chauffage en fonction de la rétroaction du capteur de température pour maintenir une plage de température définie. L'agent peut également apprendre à s'adapter aux changements de l'environnement, tels que les changements de température extérieure, pour maintenir la plage de température souhaitée.

L'utilisation de l'apprentissage par renforcement dans les systèmes autorégulateurs permet un contrôle adaptatif et efficace de systèmes complexes qui seraient difficiles ou impossibles à contrôler manuellement. En apprenant de l'expérience, l'agent peut améliorer son comportement au fil du temps et répondre à des situations nouvelles et changeantes.

### **I.10-Conclusion:**

En conclusion, l'apprentissage par renforcement est une méthode puissante pour enseigner à une machine comment prendre des décisions en fonction des récompenses ou des punitions qu'elle reçoit en fonction de ses actions. Cette approche est particulièrement utile pour résoudre des problèmes complexes où les solutions ne sont pas connues à l'avance et où la machine doit explorer différentes options pour trouver la meilleure solution possible.

L'apprentissage par renforcement est utilisé dans de nombreux domaines, notamment les jeux, la robotique, la finance et la santé. Cette méthode a permis des avancées significatives dans des domaines tels que la reconnaissance vocale, la traduction automatique, la vision par ordinateur et la reconnaissance de formes.

Cependant, l'apprentissage par renforcement présente également des défis, notamment la nécessité de concevoir des systèmes de récompenses appropriés pour guider le processus d'apprentissage et la difficulté de garantir que la machine ne développe pas de comportements indésirables. Néanmoins, avec des progrès constants dans les technologies de l'IA et des algorithmes d'apprentissage par renforcement plus sophistiqués, l'avenir de cette méthode promet de nombreuses applications passionnantes.

# **Chapitre II**

## **modélisation de**

### **moteur à courant**

## II.1- Introduction

La modélisation des moteurs à courant continu est une méthode utilisée pour décrire le comportement de ces dispositifs électromécaniques. Les moteurs à courant continu sont largement utilisés dans de nombreuses applications

La modélisation d'un moteur à courant continu implique de représenter ses différents éléments constitutifs et d'établir des équations mathématiques qui décrivent les relations entre ces éléments.

Le modèle résultant permet de prédire le fonctionnement du moteur dans différentes conditions de charge, de tension d'alimentation et de vitesse de rotation.

## II.2 - Importance des logiciels assistés par ordinateur pour modéliser les machines électriques :

Les logiciels assistés par ordinateur, tels que MATLAB/Simulink, sont importants pour un apprentissage efficace des cours sur les machines électriques, en particulier pour les modèles dynamiques de ces machines. La capture du comportement dynamique des machines électriques est importante pour de nombreux aspects, tels que la commande de la vitesse et de l'angle des moteurs. Les moteurs à courant continu ont toujours été utilisés dans des applications où une bonne contrôlabilité est requise comme les systèmes robotiques Simulink offre un environnement graphique pour modéliser les machines dynamiques [6].

Les systèmes peuvent être facilement modélisés à l'aide de diagrammes de blocs ou de blocs Simscape qui imitent les caractéristiques des composants électriques et mécaniques. Plusieurs modèles de moteurs à courant continu Simulink ont été rapportés dans la littérature. Cependant, ces études manquent généralement de détails suffisants qui permettraient au lecteur de reconstruire adéquatement le modèle, principalement les équations et les sous-systèmes dans le modèle. De plus, aucun modèle basé sur la bibliothèque Simscape n'a été présenté qui sont visuellement plus liés aux composants de circuits électriques auxquels les étudiants sont familiers [7].

## II.3- La diversité des types de machines à courant continu et son impact sur la recherche et le contrôle :

Une machine à courant continu a un stator et un rotor. Le stator héberge généralement l'enroulement de champ, tandis que le rotor est constitué d'un enroulement d'induit. Les enroulements de champ et d'induit dans les machines à courant continu peuvent être alimentés par des sources de tension distinctes (c'est-à-dire excitation séparée) ou à partir de la même source

(c'est-à-dire auto-excité). De plus, l'aimant permanent peut remplacer l'enroulement de champ dans le stator, d'où la construction de machines à courant continu à aimant permanent. Dans les moteurs auto-excités, la technique de connexion des enroulements de champ et d'induit forme les types de base de machines à courant continu, tels que les machines à courant continu connectées en dérivation, connectées en série et connectées en composé. La majorité des modèles présentés dans la littérature sont pour les machines à courant continu à aimant permanent, malgré les caractéristiques souhaitées d'autres types de machines à courant continu. Par exemple, les machines à courant continu connectées en série, qui présentent un couple de démarrage élevé, sont utilisées dans les systèmes de traction, et les machines à courant continu connectées en dérivation, qui ont une vitesse relativement constante à différents niveaux de charge, sont utilisées pour les entraînements industriels tels que les perceuses. Par conséquent, la diversité de la recherche liée à la modélisation et au contrôle des moteurs à courant continu peut être affectée [8].

La contribution de cet article est de présenter des packages simples et modifiables pour la modélisation et la simulation de différents types de moteurs à courant continu. Cet article présente [11], pour la première fois, un laboratoire virtuel de modèles dynamiques de (I) excitation séparée, (II) aimant permanent, (III) connecté en dérivation et (IV) connecté en série de moteurs à courant continu développés à l'aide de MATLAB/Simulink, où le comportement transitoire de ces moteurs peut être étudié. Trois méthodes différentes pour modéliser les quatre types de moteurs à courant continu, dans l'environnement MATLAB/Simulink, sont présentées. Premièrement, le diagramme de blocs du moteur est développé en utilisant des blocs de fonction de transfert [6].

### II.3- équations fondamentales des moteurs à courant continu:

Le comportement dynamique des moteurs à courant continu peut être dérivé des équations électromagnétiques du système. Les équations de tension d'induit et de champ sont les suivantes [9]:

$$v_a = (r_a + \frac{dL_{AA}}{dt})i_a + \omega_r L_{AF}i_f \quad (1)$$

$$v_f = (r_f + \frac{dL_{FF}}{dt})i_f \quad (2)$$

Où

**ra, ra:** auto-résistance des enroulements de champ et d'induit

**LFF, LAA:** auto-inductances des enroulements de champ et d'induit

**if , ia:** courants circulant à travers les enroulements de champ et d'induit

**Or:** vitesse du rotor (induit).

**LAF:** inductance mutuelle entre les bobines de champ et d'induit.

Le couple électromagnétique ( $T_e$ ) peut être exprimé par :

$$T_e = L_{AF} i_f i_a \tag{3}$$

Le couple électromagnétique et le couple de charge ( $T_l$ ) sont liés par :

$$T_e = J \frac{d\omega_L}{dt} + B_m \omega + T_l \tag{4}$$

Où

**J:** est l'inertie du rotor (plus toute charge mécanique connectée).

**Bm:** est le coefficient d'amortissement associé au système de rotation mécanique de la machine.

Le circuit équivalent de différents types de moteurs à courant continu est présenté dans les figures ;

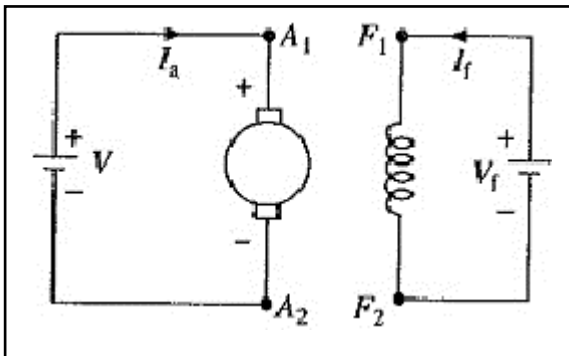


Figure (II.1) MCC à excitation séparée

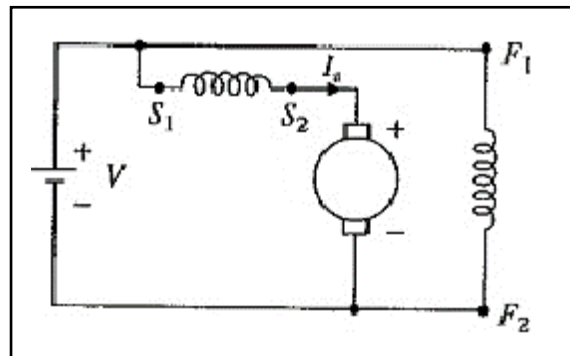


Figure (II.2) MCC à aimant permanent

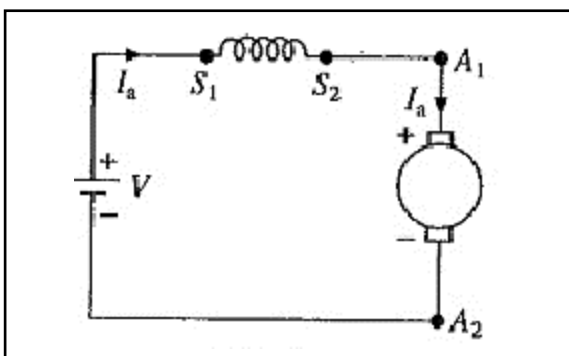


Figure (II.3) MCC à excitation série

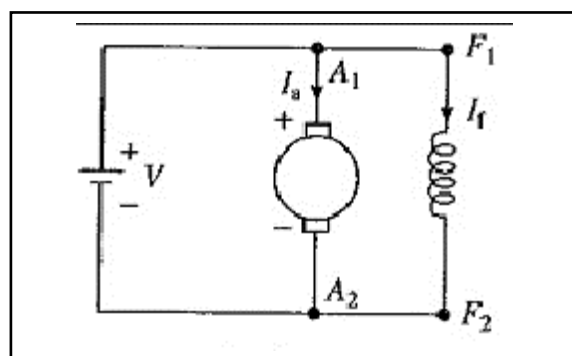


Figure (II.4) MCC à excitation en dérivation

## II.4-Modélisation de moteurs à courant continu séparément excités :

Le circuit équivalent des moteurs à courant continu à excitation séparée est représenté dans la figure II.1 .

Les circuits de champ et d'induit sont alimentés par des sources de tension différentes (séparées).

### II.4-1. Modélisation à l'aide de schémas fonctionnels :

La fonction de transfert des systèmes d'équations différentielles linéaires et invariantes dans le temps peut être trouvée en suivant les quatre étapes suivantes [10] :

- Trouver les équations qui décrivent le système.
- Identifier les variables d'entrée et de sortie du système. Étant donné que les moteurs à courant continu à commande de vitesse ont des applications courantes, la vitesse de rotation est choisie comme variable de sortie. En revanche, les tensions d'induit et de champ ainsi que le couple de charge sont choisis comme variables d'entrée.
- Prendre la transformée de Laplace de ces équations, en supposant des conditions initiales nulles.
- La fonction de transfert du système est le rapport entre les variables de sortie et d'entrée.

Une représentation du moteur à l'aide de blocs de fonctions de transfert peut être développée à partir des équations différentielles données dans les équations 1, 2, 3 et 4. En prenant la transformée de Laplace de ces équations et en considérant que les conditions initiales sont nulles, on obtient les équations suivantes qui peuvent être simplement présentées sous forme de diagramme de blocs [7] :

$$V_a(s) = r_a I_a(s) + sL_{AA}I_a + \omega_r L_{AF} I_f(s); \therefore I_a(s) = \frac{1}{(r_a + sL_{AA})} (V_a(s) - \omega_r L_{AF} i_f) \quad (5.a)$$

$$V_a(s) = (r_f + sL_{FF}) I_f(s); \therefore I_f(s) = \frac{1}{(r_f + sL_{FF})} V_f(s) \quad (5.b)$$

$$T_e = (sJ + Bm)\omega_r(s) + T_l(s), T_e(s) = L_{AF} I_f(s) I_a(s) \quad (5.c)$$

$$\omega_r(s) = \frac{1}{(sJ + Bm)} (L_{AF} I_f(s) I_a(s) - T_l(s))$$

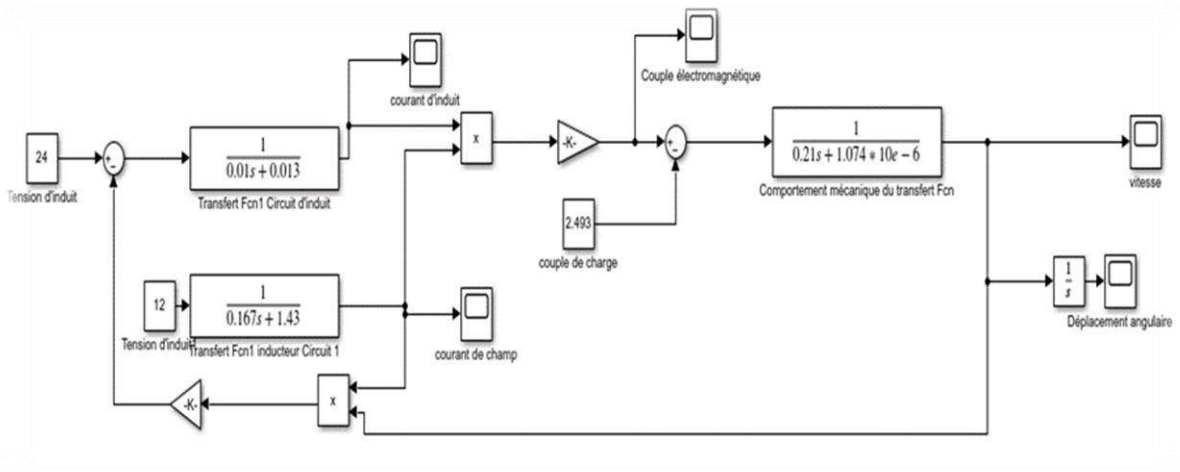


Figure II.5 ; Un modèle d'un moteur à courant continu à excitation séparée utilisant des blocs de fonctions de transfert dans Simulink.

**II.4.2- Modélisation à l'aide de blocs Simscape :**

Les moteurs à courant continu à excitation séparée peuvent être représentés par des composants électriques et mécaniques fournis par Simscape, qui imitent les caractéristiques électriques et mécaniques de la machine. La Figure II.6 montre un modèle de moteur à courant continu à excitation séparée dans Simscape/Simulink.

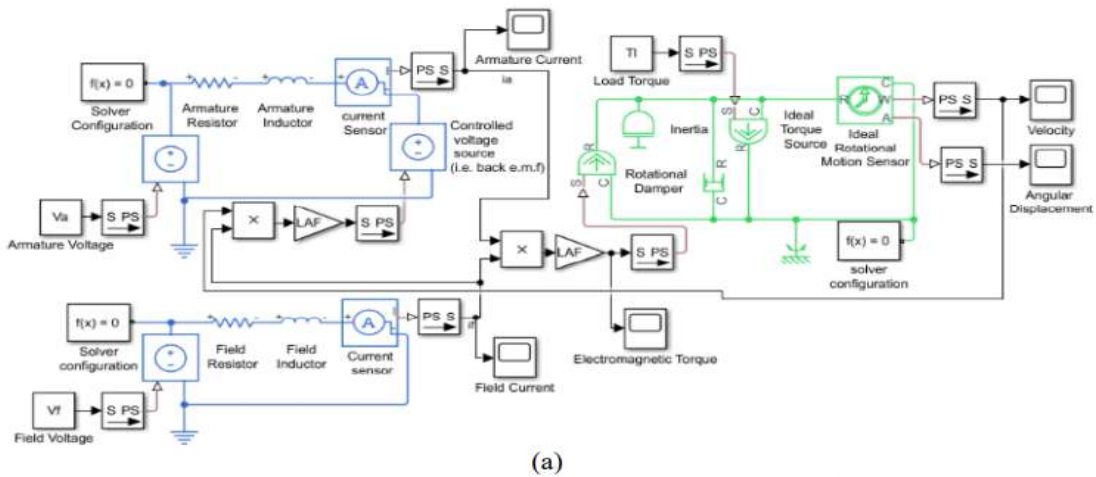


Figure II.6 . Un modèle de moteur à courant continu à aimant permanent utilisant des blocs Simscape.

**II.4.3- Modélisation à l'aide du bloc machine DC intégré :**

Simscape/la bibliothèque de technologie spécialisée fournit un bloc prêt à l'emploi, qui représente les machines à courant continu. Les entrées du bloc sont les bornes des enroulements d'induit et de champ et le couple de charge. Par conséquent, il peut modéliser différents types de



machines à courant continu, lorsque des sources de tension d'induit et de champ sont connectées différemment à ses bornes. Le bloc intégré a été utilisé pour simuler un moteur à courant continu à excitation séparée, comme le montre la figure :

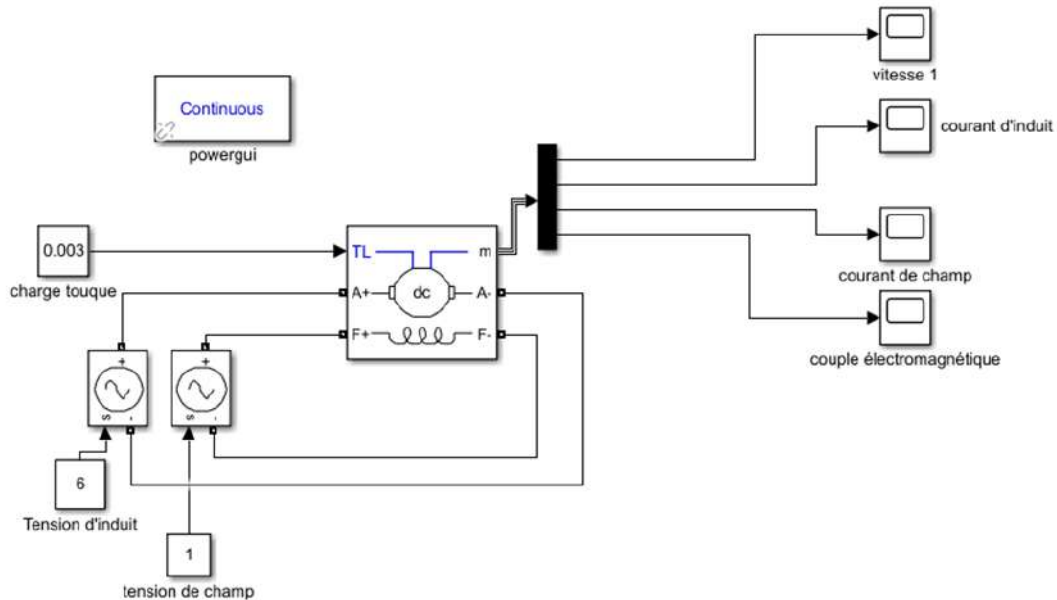


Figure II.7 ; le bloc de machine DC intégré dans simulink.

### II.5- Modélisation des moteurs à courant continu à aimants permanents:

Un moteur à courant continu à aimant permanent (PM) n'a pas de circuit de champ ; à la place, l'aimant permanent produit un flux de champ constant, ce qui est analogue à un moteur SEDC avec un  $K_v$  constant. Les équations de tension d'induit et de couple électromagnétique sont réécrites comme suit [10] :

$$v_a = \left( r_a + \frac{dL_{AA}}{dt} \right) i_a + \omega_r k_v \quad (6)$$

$$T_e = K_v i_a \quad (7)$$

Le circuit équivalent pour ce type de machine est représenté sur la figure suivant :

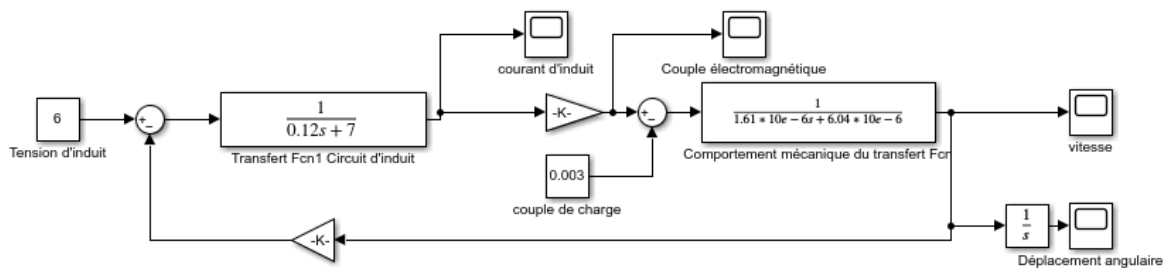
**II.5.1-Modélisation à l'aide de schémas fonctionnels :**

Un modèle de moteur PMDC peut être dérivé de la transformée de Laplace des équations (4), (6) et (7), en supposant des conditions initiales nulles. Les équations suivantes peuvent être représentées par des blocs de fonction de transfert, [10] comme indiqué dans la Figure II.8 :

$$V_a(s) = r_a I_a(s) + sL_{AA} I_a + \omega_r K_v; \therefore I_a(s) = \frac{1}{(r_a + sL_{AA})} (V_a(s) - \omega_r k_v) \tag{8.a}$$

$$T_e = (sJ + Bm)\omega_r(s) + T_l(s), T_e(s) = K_v I_a(s); \therefore$$

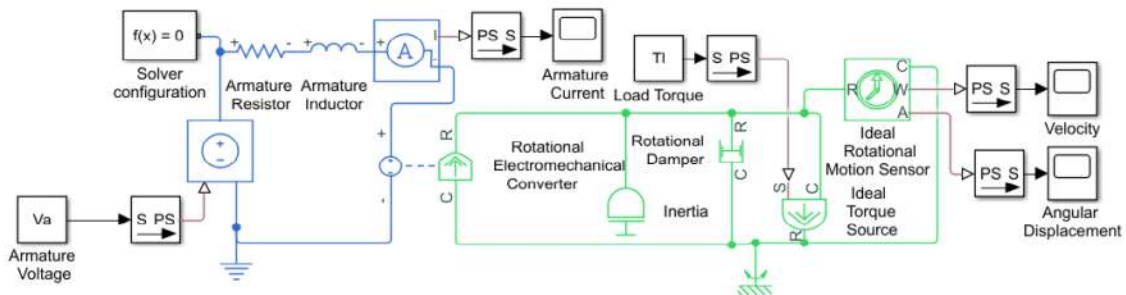
$$\omega_r(s) = \frac{1}{(sJ + Bm)} (K_v I_a(s) - T_l(s)) \tag{8.b}$$



**Figure II.8 :Un modèle d'un moteur à courant continu à aimant permanent utilisant des blocs de fonctions de transfert dans Simulink.**

**II.5.2- Modélisation à l'aide des blocs Simscape :**

Les moteurs à courant continu à aimant permanent peuvent être représentés par des circuits électriques (c'est-à-dire des inductances et des résistances), en plus de composants qui imitent la caractéristique mécanique de la machine (c'est-à-dire l'inertie, l'amortisseur rotatif), comme le montre la figure suivant :



**Figure II.9 : Un modèle de moteur à courant continu à aimant permanent utilisant des blocs Simscape.**

### II.5.3- Modélisation à l'aide du bloc machine DC intégré :

Le bloc de machine CC intégré de la bibliothèque de technologies Simscape / spécialisée a été utilisé pour simuler un moteur à courant continu à aimant.

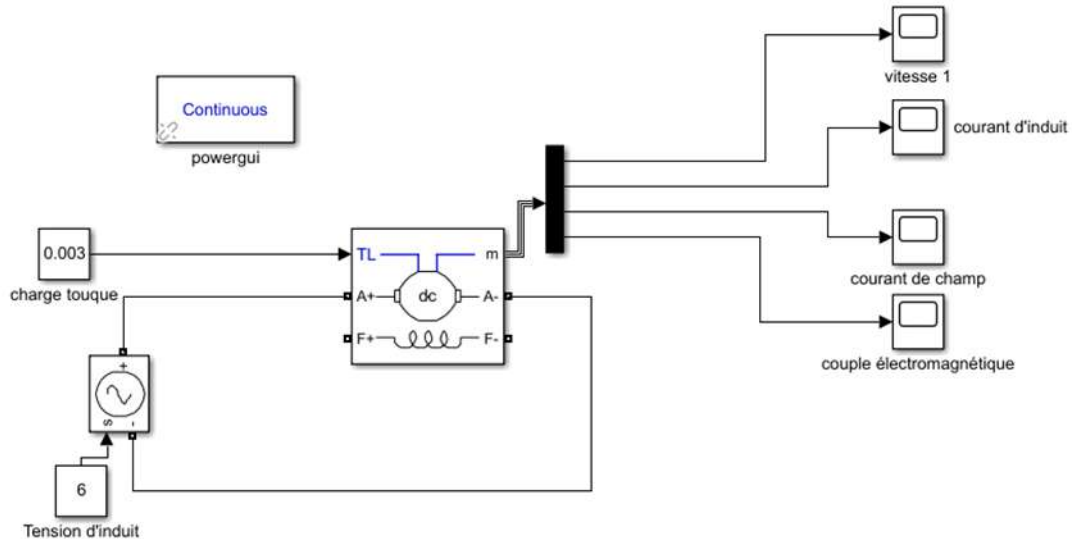


Figure II.10 : le bloc de machine DC intégré dans Simulink.

### II.6- Modélisation des moteurs cc raccordés en shunt:

Lorsque les enroulements de champ et d'induit peuvent être connectés en parallèle, comme le montre schématiquement la Figure 1©, on parle alors d'une machine à courant continu connectée en dérivation. Comme l'enroulement de champ est connecté entre les bornes de l'induit, [10]

$$V_f = V_a \quad (9)$$

Et le courant total ( $I_t$ ) est

$$I_f = I_a + I_f \quad (10)$$

#### II.6.1- Modélisation à l'aide de diagrammes de blocs :

Pour construire un modèle de moteurs à courant continu connectés en dérivation, la transformée de Laplace des équations (1)-(4) et (9) est considérée, avec des conditions initiales nulles. En conséquence, les équations suivantes sont représentées par un diagramme de blocs,

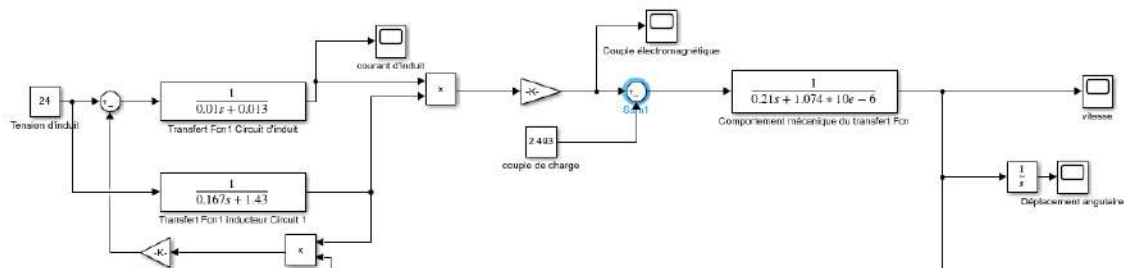
$$V_f(s) = V_a(s); V_a(s) = r_a I_a(s) + sL_{AA} I_a + \omega_r L_{AF} I_f; \quad (11.a)$$

$$\therefore I_a(s) = \frac{1}{(r_a + sL_{AA})} (V_a(s) - \omega_r L_{AF} I_f)$$

$$V_f(s) = (r_f + sL_{FF}) I_f(s); \quad \therefore I_f = \frac{1}{(r_f + sL_{FF})} V_f(s) \quad (11.b)$$

$$T_e = (sJ + Bm)\omega_r(s) + T_l(s), T_e(s) = L_{AF} I_f(s) I_a(s)$$

$$\omega_r(s) = \frac{1}{(sJ + Bm)} (L_{AF} I_f(s) I_a(s) - T_l(s)) \quad (11.c)$$



**Figure II.11: Un modèle d'un moteur à courant continu connecté en dérivation utilisant des blocs de fonction de transfert dans Simulink.**

### II.6.2- Modélisation à l'aide de blocs Simscape :

Un modèle de moteur à courant continu connecté en dérivation peut être construit à l'aide de blocs Simscape, comme le montre suivant :

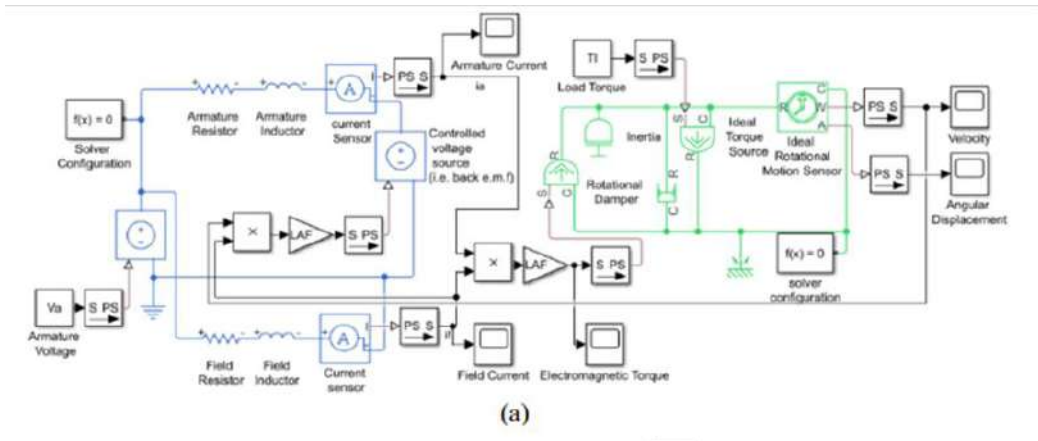


Figure II.12: Un modèle de moteur à courant continu connecté en dérivation utilisant des blocs Simscape

### II.6.3-Modélisation à l'aide du bloc de machine à courant continu intégré :

Pour valider les modèles présentés précédemment, un modèle de machine à courant continu prêt à l'emploi dans la bibliothèque de technologies spécialisées de Simscape a été utilisé. Les enroulements d'armature et de champ ont été connectés à la même source de tension, comme indiqué dans la Figure II.13 :

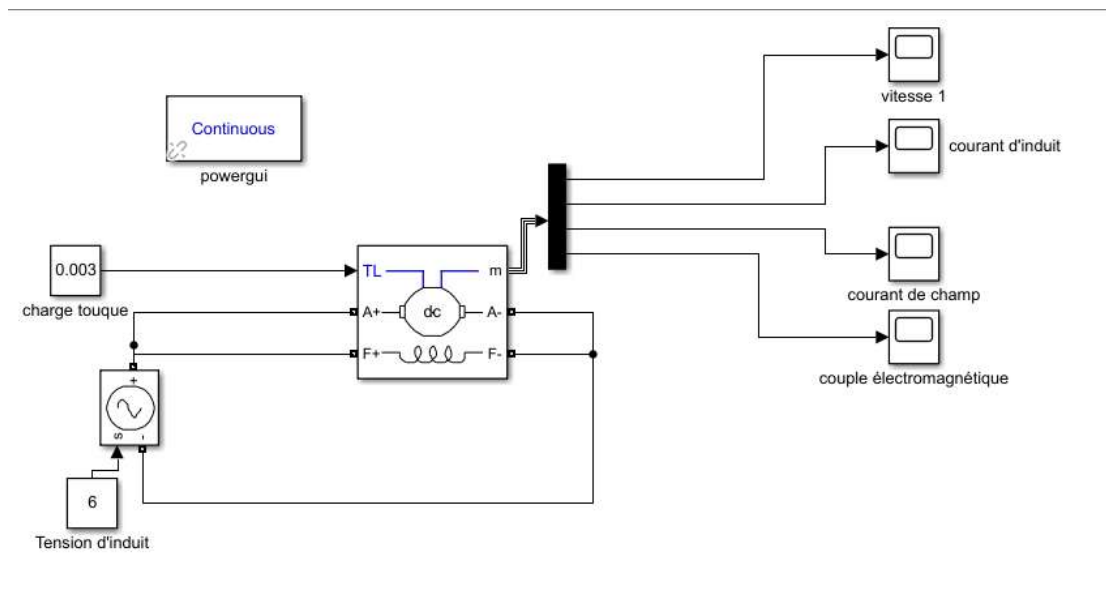


Figure II.13: le bloc de machine à courant continu intégré dans Simulink.

## II.7- Modélisation des moteurs à courant continu connectés en série:

Lorsque le circuit de champ est connecté en série avec le circuit d'induit, comme illustré dans la Figure 1(d), la machine est appelée machine à courant continu connectée en série ou machine à courant continu série. Ainsi, les deux circuits sont alimentés à partir de la même source de tension continue ( $v_t$ ) et le courant qui circule dans les deux circuits est le même [10]:

$$V_t = V_a + V_f \quad (12);$$

$$I_a = I_f \quad (13).$$

### II.7.1- Modélisation à l'aide d'un diagramme de blocs :

Un moteur à courant continu série peut être représenté à l'aide d'un diagramme de blocs à travers les équations différentielles données dans (1) à (6) et (2) à (5). En prenant la transformée de Laplace de ces équations, en considérant que les conditions initiales sont nulles, on obtient les équations suivantes, qui ont été mises sous forme de diagramme de blocs présenté dans la Figure :

$$V_a = r_a I_a + sL_{AA} I_a + \omega_r L_{AF} I_a$$

$$V_f = (r_f + sL_{FF}) I_f$$

$$I_f = I_a$$

$$V_t = V_a + V_f = (r_a + r_f + sL_{AA} + sL_{FF}) I_a + \omega_r L_{AF} I_a \quad (14.a)$$

$$I_a = \frac{1}{(r_a + r_f + sL_{AA} + sL_{FF})} (V_t - \omega_r L_{AF} I_a)$$

$$T_e = (sJ + Bm)\omega_r + T_l, T_e = L_{AF} I_a^2;$$

$$\omega_r(s) = \frac{1}{(sJ + Bm)} (L_{AF} I_a^2 - T_l); \quad (14.b)$$

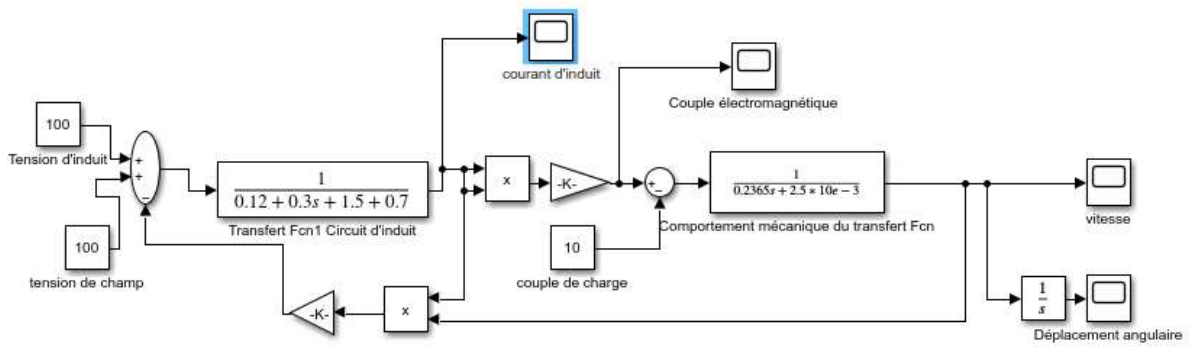


Figure II.14: Un modèle de moteur à courant continu connecté en série utilisant des blocs de fonction de transfert dans Simulink.

**II.7.2- Modélisation à l'aide de blocs Simscape :**

Les moteurs à courant continu série peuvent être représentés par des composants électriques et mécaniques pour imiter les caractéristiques de la machine. La Figure II.14 montre une représentation d'un moteur à courant continu connecté en série dans Simscape.

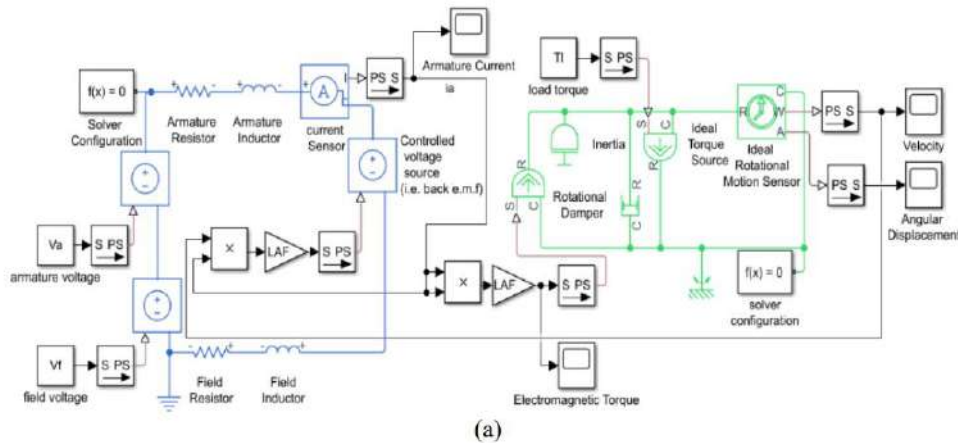


Figure II.14: Un modèle de moteur à courant continu connecté en série utilisant des blocs Simscape.

**II.7.3. Modélisation à l'aide du bloc de machine à courant continu intégré :**

La bibliothèque de technologies spécialisées de Simscape fournit un modèle prêt à l'emploi de la machine à courant continu série. La Figure II.15 montre une représentation d'un moteur à courant continu connecté en série dans Simulink.

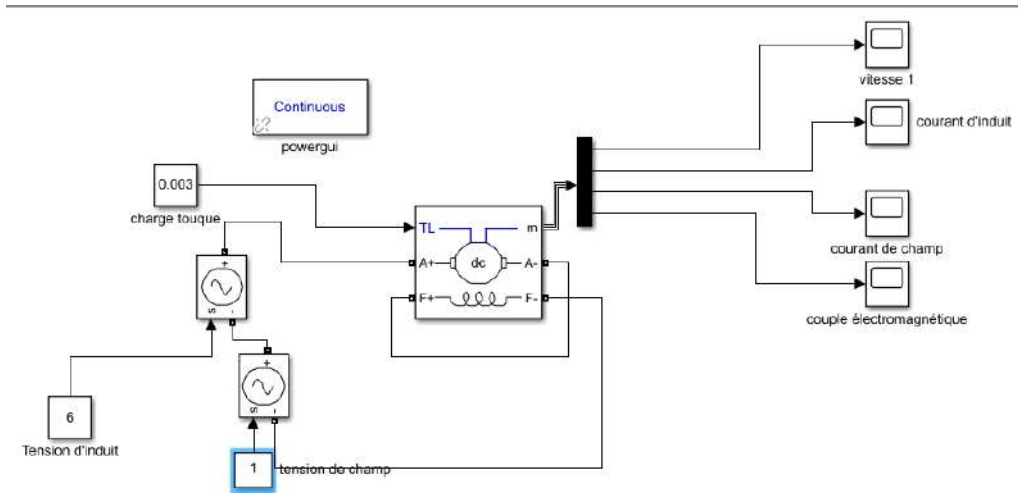


Figure II.15: le bloc de machine à courant continu intégré dans Simulink.

### II.8-: Paramètres des machines à courant continu :

Les modèles présentés dans les sections précédentes sont simulés et leurs caractéristiques de démarrage sont représentées et discutées dans cette section. Les paramètres de la machine utilisés dans la simulation sont présentés dans le Tableau 1. Tous les modèles ont été construits et simulés à l'aide de MATLAB 2022b.

Paramètres	Excité séparément	Aimant permanent	Connecté en dérivation	Connecté en série
$r_a$ ( $\Omega$ )	0.013	7	0.013	1.5
$L_{AA}$ (H)	0.01	0.12	0.01	0.12
$V_f$	12	--	24	100
$V_a$	24	6	24	100
$T_l$ (N.m)	2.493	0.003	2.493	10
J (Kg.m <sup>2</sup> )	0.21	$1.61 \cdot 10^{-6}$	0.21	0.2365
$B_m$ (N.m/rad/sec)	$1.074 \cdot 10^{-6}$	$6.04 \cdot 10^{-6}$	$1.074 \cdot 10^{-6}$	$2.5 \cdot 10^{-3}$
$r_f$ ( $\Omega$ )	1.43	--	1.43	0.7
$L_{FF}$ (H)	0.167	--	0.167	0.03
$L_{AF}$ (H)	0.004	--	0.004	0.0675
$K_v$ (V.sec/rad)	--	0.0141	--	--

Tableau II 2. Paramètres de moteurs à courant continu utilisés dans la simulation.



**II.9- Conclusions:**

Une approche de modélisation de diagramme de blocs a été utilisée, ce qui permet une modification des modèles pour diverses applications, telles que la commande de vitesse des moteurs à courant continu. La deuxième approche de modélisation utilise les composants de la bibliothèque Simscape, qui imitent les parties électriques et mécaniques des machines à courant continu, pour construire des modèles. Tous les modèles développés permettent de modifier tous les paramètres de la machine pour des fins de surveillance et d'évaluation, créant ainsi un environnement d'apprentissage amélioré. Dans la troisième approche de modélisation, le bloc machine à courant continu intégré de Simulink a été adopté, ce qui fournit une référence pour valider les premières et deuxièmes approches de modélisation.

# **Chapitre III**

## **commande de la vitesse d'un mcc par RL**

### **III.1-Introduction:**

L'apprentissage par renforcement offre une approche prometteuse pour la commande de la vitesse des moteurs à courant continu, permettant d'automatiser le processus de réglage des paramètres du contrôleur. En exploitant les capacités d'apprentissage autonome de l'agent RL, il est possible de développer des systèmes de contrôle plus adaptatifs et performants, capables de s'adapter à des environnements variables et de répondre aux exigences spécifiques de chaque application, à la fin on fait une comparaison avec le régulateur pid.

### **III.2-Les étapes que vous pouvez suivre pour créer un environnement d'apprentissage par renforcement Simulink pour un moteur à courant continu:**

1. Définissez l'espace d'état
2. Définissez l'espace d'action.
3. pour créer un modèle du système de moteur à courant continu. Le modèle composé de la tension appliquée au moteur.
4. Définissez la fonction de récompense.
5. Mettez en œuvre l'algorithme d'apprentissage par renforcement : Utilisez l'algorithme d'apprentissage par renforcement de votre choix, tel que l'apprentissage par renforcement Q ou les réseaux Q profonds (DQN), pour entraîner un agent à contrôler le moteur à courant continu.
6. Testez et évaluez l'agent : Une fois que l'agent a été formé, testez-le sur l'environnement du moteur à courant continu pour voir à quel point il se comporte bien. Évaluez ses performances en fonction de la vitesse requise, du temps nécessaire pour atteindre la vitesse et de la stabilité du contrôle.

### III.2-Définition de l'environnement d'apprentissage :

L'environnement d'apprentissage par renforcement est un cadre dans lequel un agent intelligent apprend à prendre des décisions en interagissant avec un environnement dynamique. L'agent prend des actions et observe les réponses de l'environnement, puis ajuste son comportement en fonction des récompenses ou des punitions reçues.

L'environnement d'apprentissage par renforcement peut être utilisé pour entraîner des algorithmes d'apprentissage automatique à accomplir des tâches spécifiques, telles que jouer à un jeu ou contrôler un robot. L'agent apprend à optimiser une fonction de récompense en prenant des actions qui maximisent la probabilité de recevoir une récompense positive.

Dans cet environnement, l'agent est en mesure d'apprendre de manière autonome en explorant différentes stratégies et en ajustant son comportement en fonction des résultats observés. En résumé, l'environnement d'apprentissage par renforcement est un cadre d'apprentissage qui permet à un agent intelligent de prendre des décisions autonomes dans un environnement dynamique en se basant sur des récompenses ou des punitions.

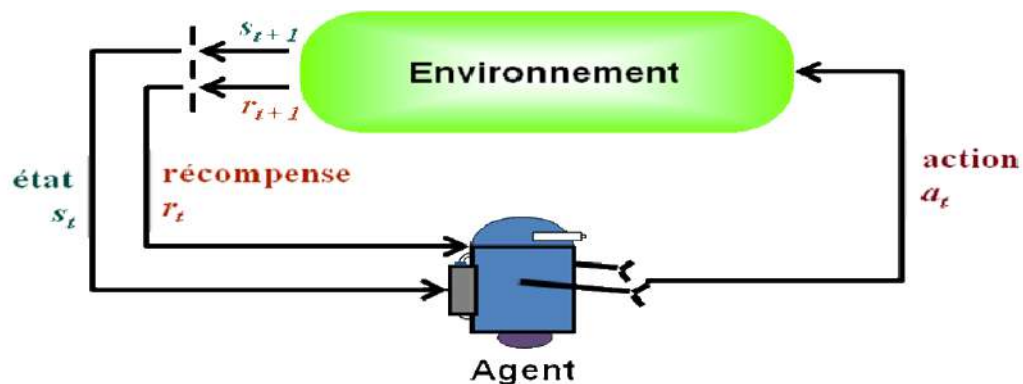


Figure III.1. L'environnement d'apprentissage

### III.2.1-Etat:

L'état d'apprentissage par renforcement peut être défini de deux manières différentes selon le contexte :

1. L'état d'apprentissage par renforcement peut faire référence à l'état de l'agent au cours de l'apprentissage. Cela inclut les connaissances et les compétences que l'agent a acquises jusqu'à présent, ainsi que les politiques qu'il a apprises pour prendre des décisions en fonction de l'état actuel de l'environnement.
2. L'état d'apprentissage par renforcement peut également faire référence à l'état de l'environnement tel que perçu par l'agent. Cela inclut les informations que l'agent perçoit de l'environnement telles que la position, la vitesse, l'orientation, la couleur, etc. Ces informations sont généralement traitées par l'agent pour prendre une décision quant à l'action à prendre.

Dans les deux cas, l'état est une information importante qui permet à l'agent de prendre des décisions en fonction de l'environnement. En utilisant l'état actuel de l'environnement, l'agent peut choisir l'action la plus appropriée à prendre pour maximiser les récompenses attendues. Ensuite, l'agent met à jour ses politiques d'apprentissage pour s'adapter à l'environnement et améliorer ses performances futures.

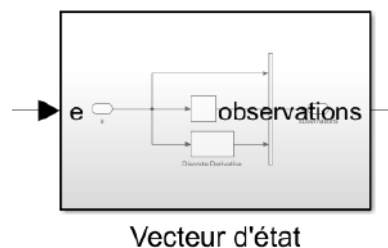


Figure III.2. Le block de vecteur d'état

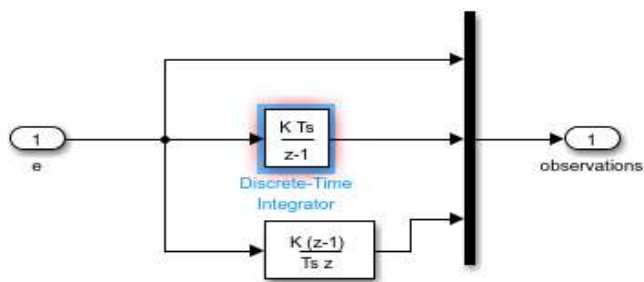


Figure III.3. Le block de vecteur d'état (en détail)

**III.2.2-La fonction de récompense (REWARD) :**

Dans le contrôle de la vitesse du moteur par apprentissage par renforcement, la fonction de récompense est une fonction qui mesure la performance de l'agent de contrôle (le contrôleur) en fonction de ses actions et de l'état de l'environnement. Cette fonction définit la récompense (ou pénalité) que l'agent reçoit pour chaque action qu'il effectue. L'objectif de cette fonction est de fournir une indication à l'agent sur la qualité de ses actions et de ses décisions. En général, la fonction de récompense est conçue de manière à encourager l'agent à atteindre un objectif spécifique, tel que maintenir une vitesse stable et précise du moteur ou minimiser la consommation d'énergie. En conséquence, les actions qui aident à atteindre cet objectif seront récompensées, tandis que les actions qui nuisent à cet objectif seront pénalisées. La conception de la fonction de récompense est une étape clé dans l'apprentissage par renforcement car elle a une influence directe sur la stratégie de l'agent et sur la qualité des résultats obtenus. Une fonction de récompense bien conçue doit être adaptée à la tâche spécifique de contrôle de la vitesse du moteur et doit encourager des comportements bénéfiques à long terme.

La fonction de récompense :

$$r_t = -10^{-5}(e_t)^2 \dots\dots\dots (1)$$

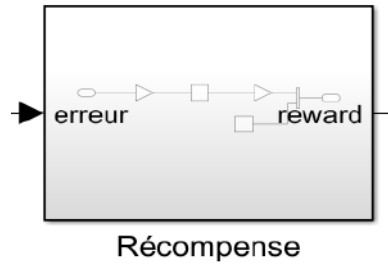


Figure III.4 bloc de La fonction de récompense

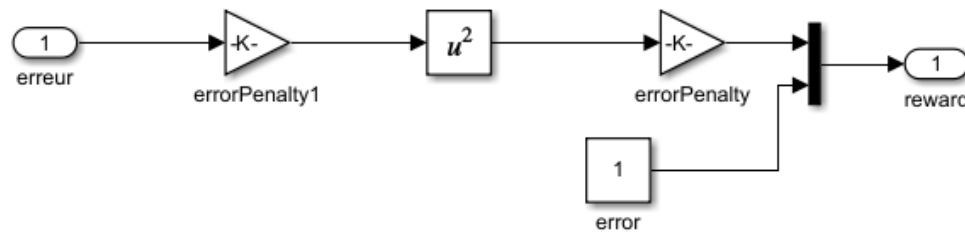


Figure III.5 bloc de La fonction de récompense.

### III.2.3-L'agent :

Dans le contexte de la commande de vitesse par apprentissage par renforcement, l'agent est le système de contrôle autonome qui interagit avec l'environnement pour apprendre comment contrôler la vitesse du moteur. Il s'agit généralement d'un algorithme d'apprentissage automatique qui utilise la technique de l'apprentissage par renforcement pour ajuster ses actions en fonction de la récompense (ou pénalité) qu'il reçoit de l'environnement. L'agent apprend progressivement à partir de son expérience et à travers des itérations successives d'interaction avec l'environnement. Pendant chaque itération, l'agent effectue une action en réponse à l'état actuel de l'environnement, et reçoit une récompense (ou pénalité) correspondante en fonction de la qualité de cette action. L'objectif de l'agent est d'apprendre une politique de contrôle optimale, c'est-à-dire une stratégie qui maximise les récompenses sur le long terme. L'apprentissage par renforcement permet à l'agent de s'adapter à des situations changeantes et de trouver une solution à des problèmes complexes sans nécessiter une programmation explicite. Cependant, pour que l'agent puisse apprendre efficacement, il est important de définir une fonction de récompense appropriée et de bien paramétrer les hyper paramètres de l'algorithme d'apprentissage. DDPG algorithme est une

méthode "hors politique" utilisée pour les actions continues et le "déterminisme" dans DDPG fait référence à l'acteur calculant directement l'action plutôt que de distribuer les probabilités aux actions.

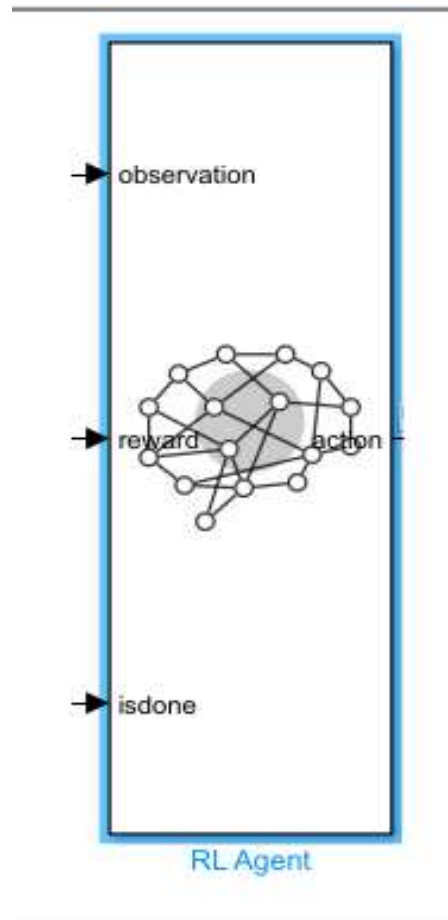


Figure III.6. l'agent

### III.3- Créer l'environnements de RL dans Matlab:

#### III.3.1-Réseau critique :

Le réseau critique est une technique utilisée par les chefs de projet pour créer un calendrier de projet et estimer la durée totale d'un projet.



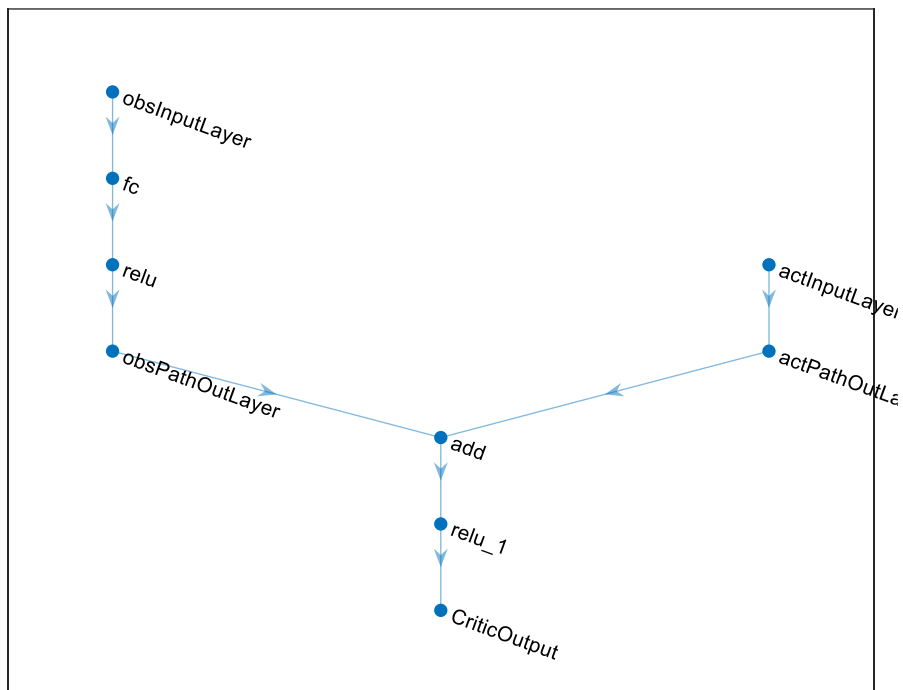


Figure III.7 Réseau critique

### III.3.2 Réseau d'acteur :

Les réseaux d'acteurs peuvent être utilisés pour étudier les interactions entre les différents éléments du système d'apprentissage par renforcement.

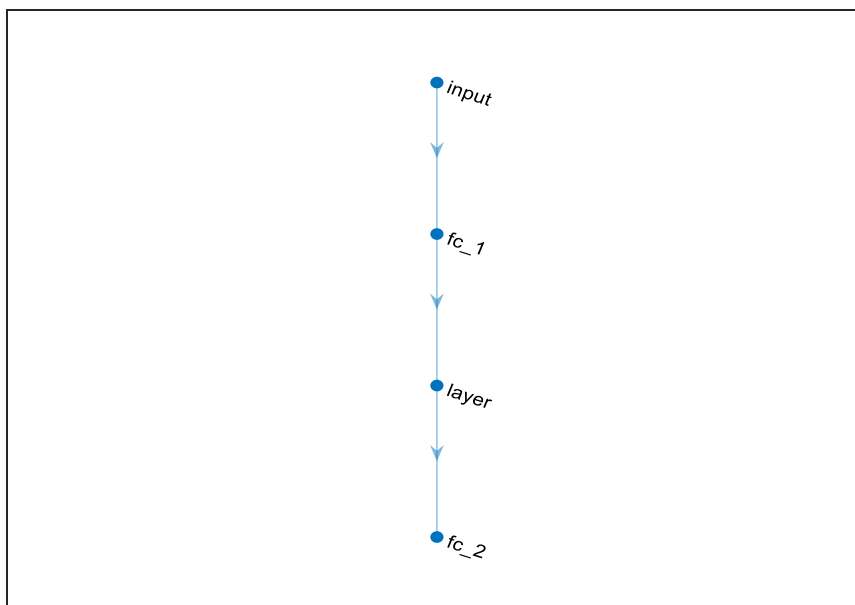


Figure III.8 Réseau acteur

### III.3.3 schéma final dans Simulink :

Nous créons le schéma suivant en utilisant Simulink et installons notre propre module moteur dans le système (Mcc).

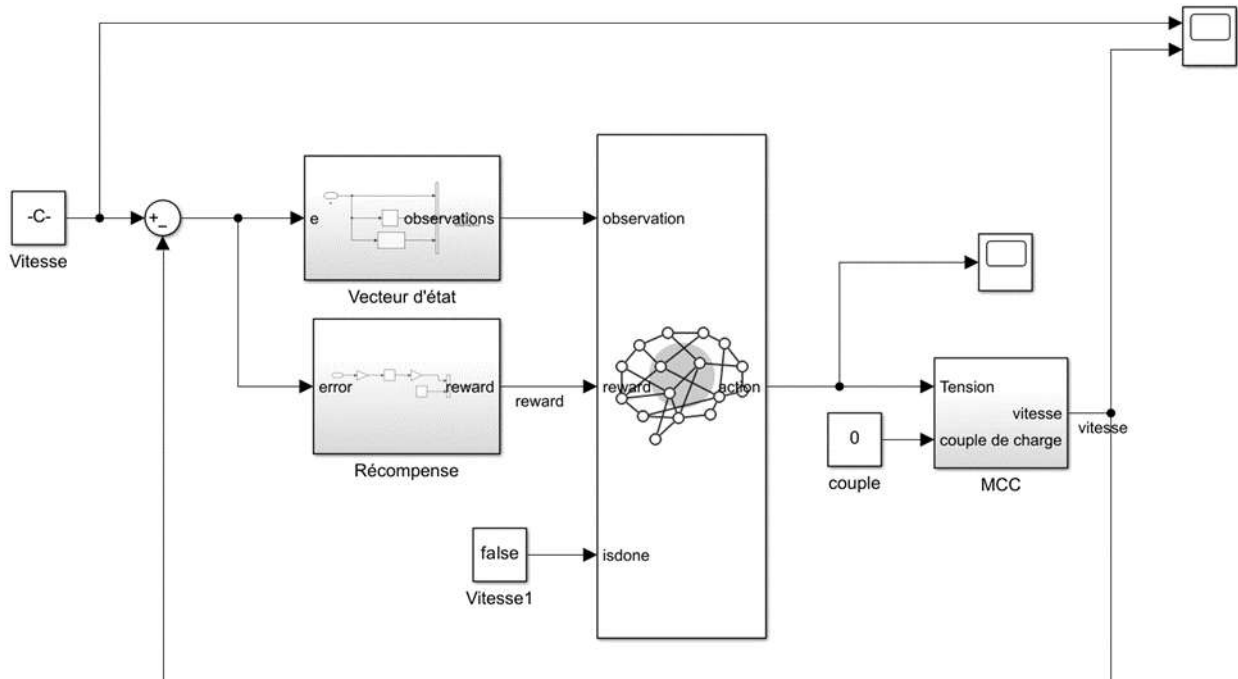


Figure III.9. L'environnement d'apprentissage

### III.3.4- Sélection du modèle de moteur MCC et Paramètres :

Nous avons choisi le moteur du type à moteur à aimant permanent et nous en avons traité dans le deuxième chapitre

Les paramètres de la machine utilisés dans la simulation sont présentés dans le deuxième chapitre  
Temps d'échantillonnage fondamental ( $T_s$ )=0.01s, temps d'arrêt ( $T_f$ )=20

### III.4-Entraînement de l'agent (training agent) :

L'Entraînement est basé sur l'apprentissage de l'erreur, et lorsqu'elle atteint la valeur requise, elle s'arrête et enregistre la solution.

L'entraînement nécessite un ordinateur doté d'un matériel puissant (RAM, processeur et carte graphique), ce qui contribue à la rapidité de l'entraînement.

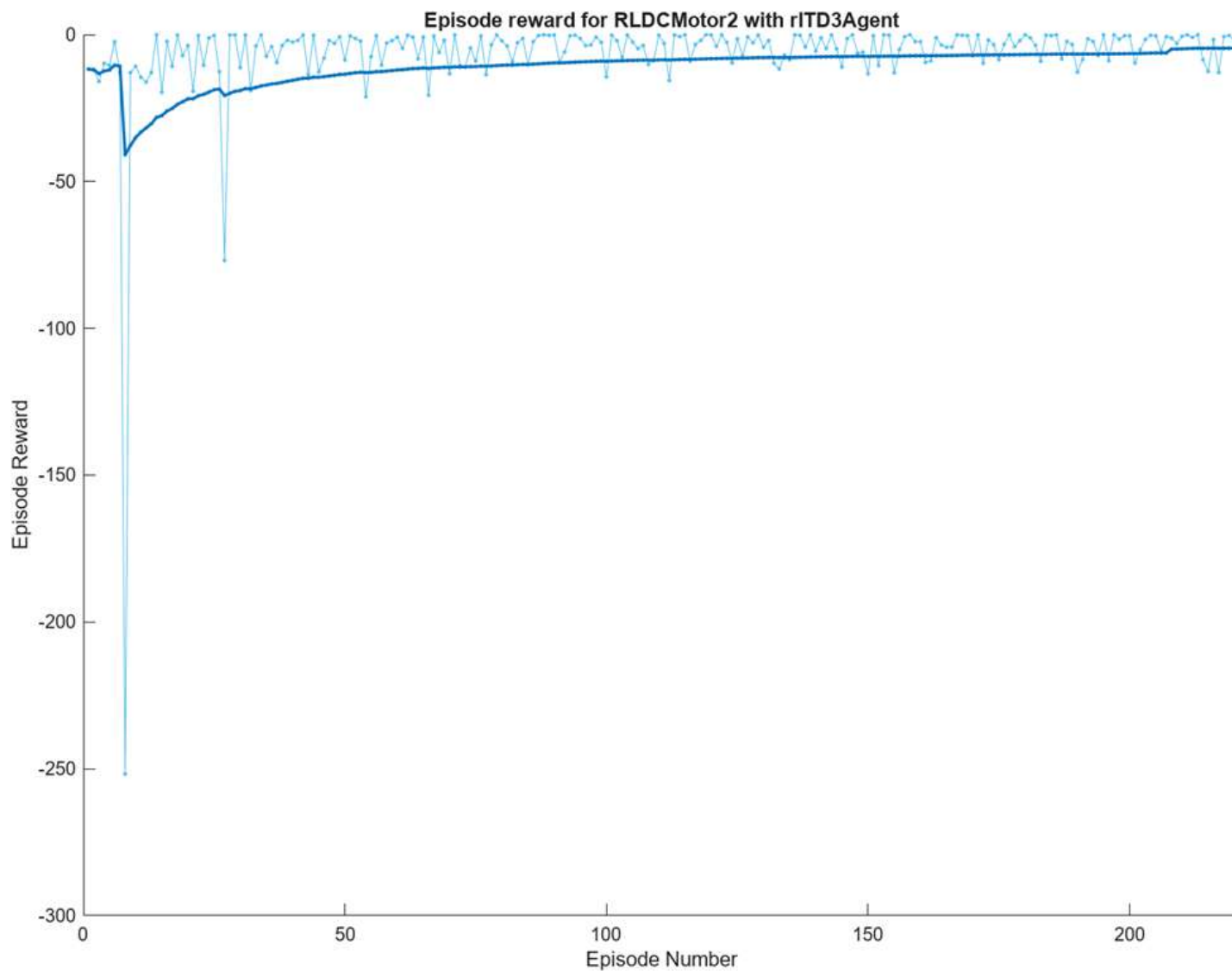


Figure III.10. Résultats de l'entraînement.

### III.5-Moteur à courant continu à avec contrôleur PID :

Un contrôleur PID peut être utilisé pour contrôler la vitesse. Il utilise un algorithme de contrôle pour ajuster la vitesse du moteur en fonction de l'entrée de commande.

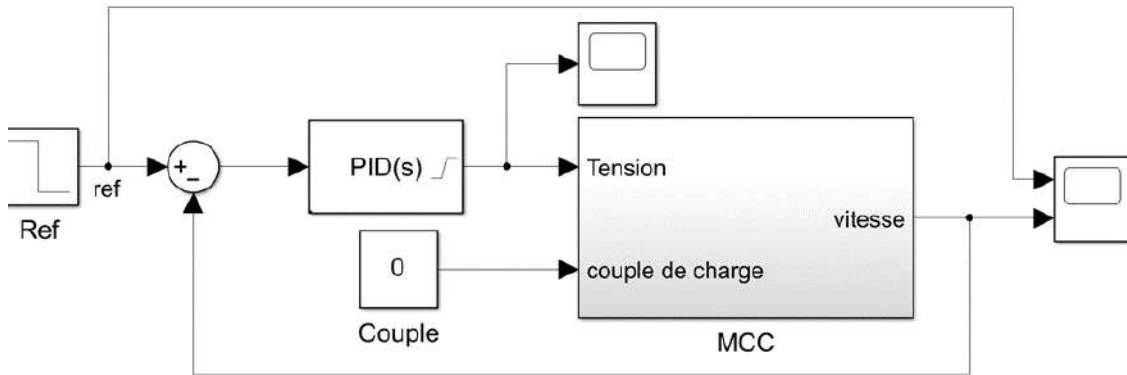


Figure III.11. moteur mcc avec contrôleur PID.

### III.5.1-Paramètres de PID contrôleur :

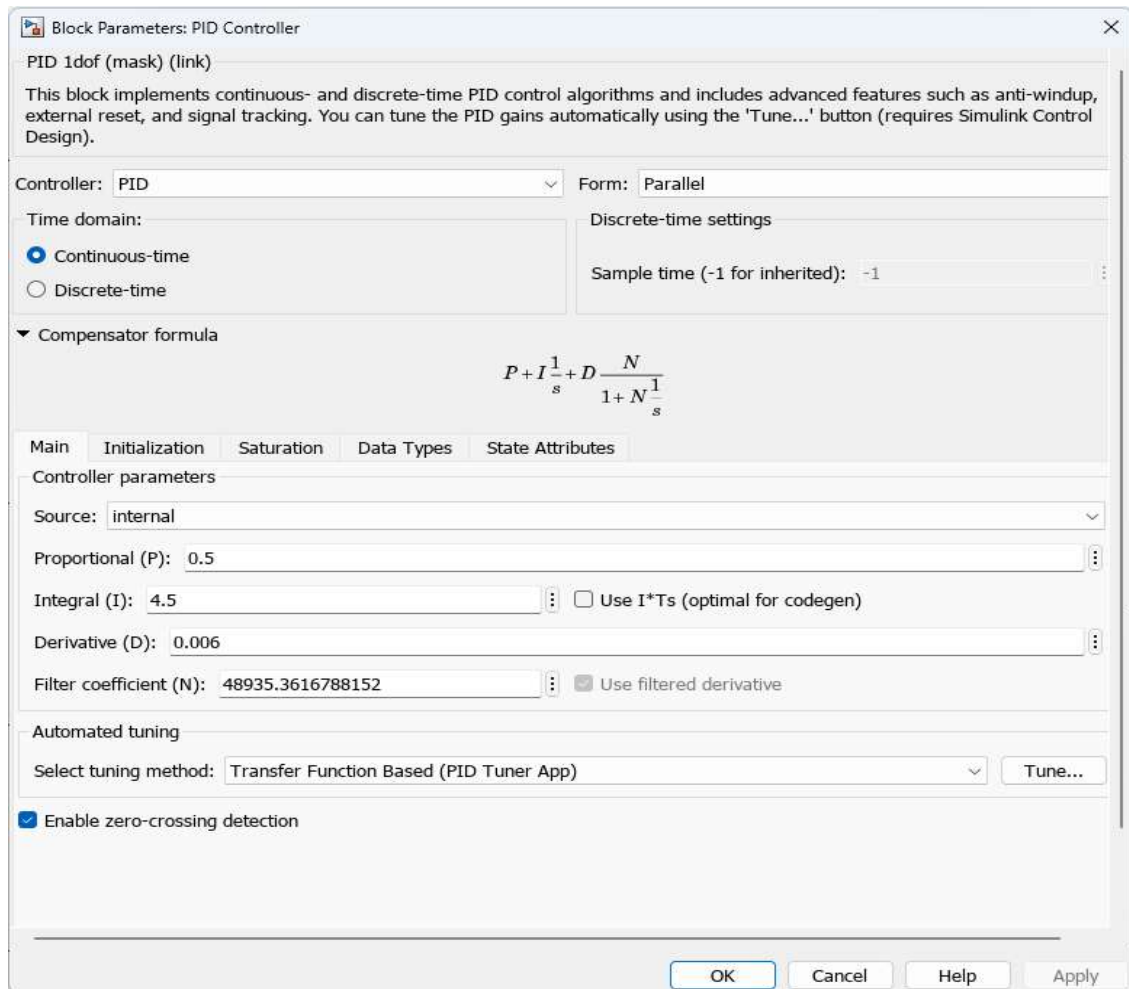


Figure III.12. Paramètres de PID contrôleur.

### III.6-La simulation:

#### III.6.1-première test :

Application d'un couple résistant d'amplitude  $T_r = -0.015$  N.m à l'instant  $t = 1.5$  s.

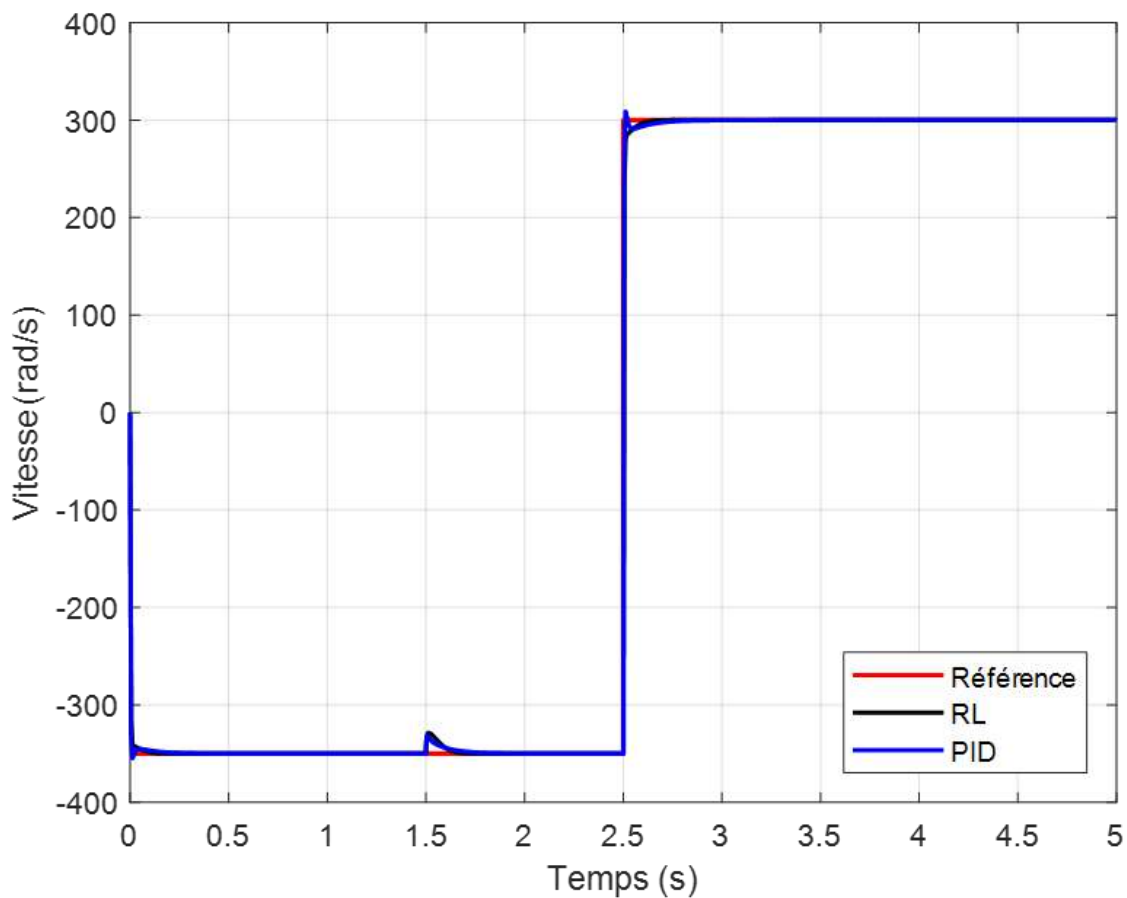


Figure III.13 Résultats du premier test et différence entre RL et PID.

### III.6.2-La deuxième test :

Application d'un couple résistant d'amplitude  $T_r = -0.015$  N.m à l'instant  $t = 1.5$  s.

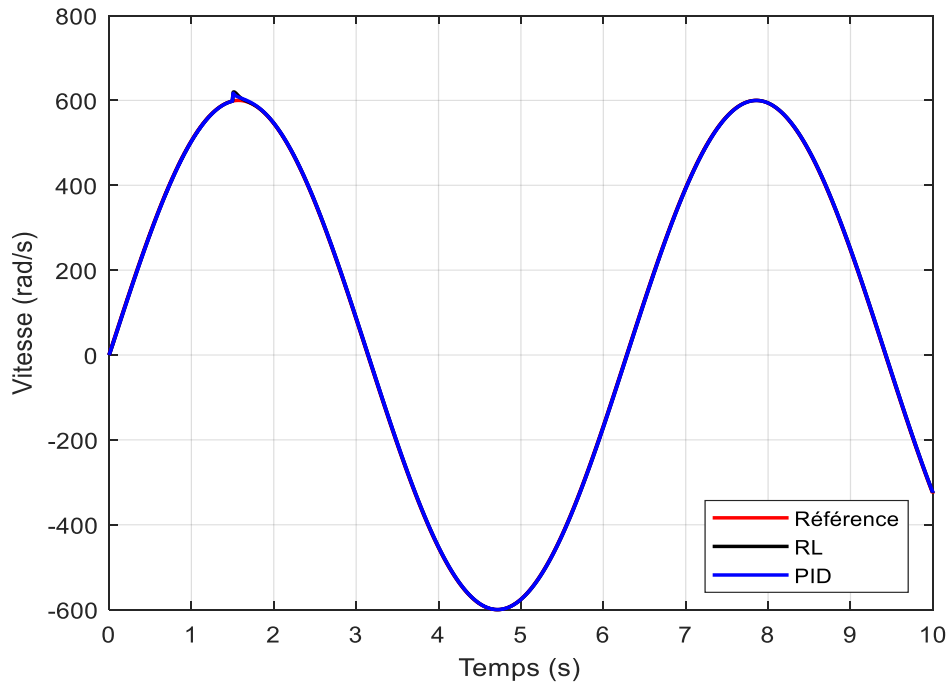


Figure III.14 Résultats du deuxième test et différence entre RL et PID

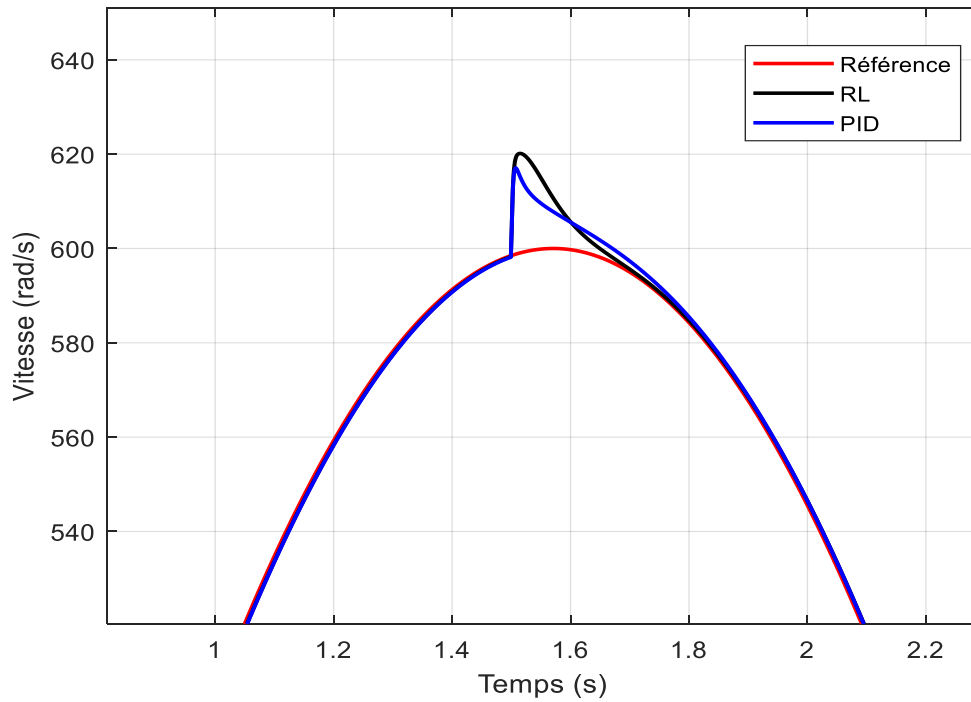


Figure III.15 Zoom sur les résultats passés

### **III.7-Comparaison de PID avec l'apprentissage par renforcement :**

Le contrôle de la vitesse du moteur peut être réalisé à l'aide de différentes techniques, telles que le contrôleur PID (proportionnel-intégral-dérivé) et l'apprentissage par renforcement. Voici une comparaison entre ces deux approches :

#### **III.7.1-Avantages du contrôleur PID :**

**Simplicité :** Le contrôleur PID est relativement simple à mettre en œuvre et ne nécessite pas une quantité importante de données pour fonctionner.

**Prédictibilité :** Le comportement du contrôleur PID est généralement prévisible et stable, ce qui peut être souhaitable dans de nombreuses applications industrielles.

#### **III.7.2-Avantages de l'apprentissage par renforcement :**

**Adaptabilité :** L'apprentissage par renforcement peut s'adapter à des situations variables et à des conditions de fonctionnement changeantes.

**Performances potentiellement supérieures :** L'apprentissage par renforcement peut aboutir à des performances supérieures par rapport à des méthodes traditionnelles dans des scénarios complexes ou lorsque la dynamique du moteur est difficile à modéliser.

**Limitations de l'apprentissage par renforcement :**

**Besoin de données d'apprentissage :** L'apprentissage par renforcement nécessite généralement une grande quantité de données d'apprentissage, ce qui peut être coûteux ou peu pratique à obtenir dans certaines situations.

**Apprentissage non déterministe :** Étant donné que l'apprentissage par renforcement repose sur des récompenses et des pénalités, il peut être difficile de garantir un comportement précis ou reproductible dans toutes les situations.

En résumé, le contrôleur PID est une méthode éprouvée et simple pour le contrôle de la vitesse du moteur, tandis que l'apprentissage par renforcement offre une plus grande adaptabilité et peut potentiellement atteindre de meilleures performances dans des situations.

**III.8-Conclusion :**

La commande de vitesse par apprentissage par renforcement est une méthode prometteuse pour améliorer les performances des systèmes de contrôle de moteurs. Elle permet à un agent autonome d'apprendre à contrôler la vitesse du moteur en recevant des récompenses ou des pénalités en fonction de la qualité de ses actions. L'avantage de cette méthode est sa capacité à s'adapter à des situations changeantes sans programmation explicite, mais il est important de bien définir la fonction de récompense et de paramétrer correctement les paramètres de l'algorithme d'apprentissage pour une efficacité optimale.



# **Conclusion Générale**

## **Conclusion générale:**

Dans cette thèse, nous avons abordé l'apprentissage automatique, où nous avons choisi la méthode de l'apprentissage par renforcement.

Comment obtenons-nous la vitesse requise du moteur MCC par apprentissage par renforcements.

Cette méthode repose sur l'interaction et les rétroactions entre l'agent et l'environnement de travail (le moteur avec l'agent).

Pour résoudre ce type d'apprentissage, il existe plusieurs méthodes. Dans cette thèse, nous avons choisi la méthode des différences temporelles, utilisant l'algorithme Actor-Critic. Cette algorithme repose sur l'interaction et les rétroactions (récompenses et sanctions). Nous devons entrainent l'agent pour collecter des données.

Dans le dernier, nous faisons une simulation et comparons (le contrôleur PID et l'apprentissage par renforcement).

La méthode d'apprentissage par renforcement reste la meilleure dans la bonne utilisation de cette technologie.

Nous avons fait ce travail avec l'application MATLAB 2022B.

## LES RÉFÉRENCES

- [1] Tom M. Mitchel. Machine Learning : sous-titre de l'ouvrage.1<sup>ère</sup>. McGraw-Hill Science; 1997.
- [2] C. Watkins, Learning from delayed rewards, Tech. Rep., 1989; vol. 3: 25.
- [3] V. R. Konda and J. N. Tsitsiklis, Actor-critic algorithms, in Proc. Adv. Neural Inf. Process, 2000;1008–1014.
- [4] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, Et M. Riedmiller, Playing atari with deep reinforcement learning, arXiv: 2013.1312.5602.
- [5] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y.-C. Liang, and D. I. Kim, “Applications of deep reinforcement learning in communications and networking: A survey,” IEEE Commun. Surveys Tuts.2019 vol. 21, no. 4, ;3133–3174
- [6] F. A. Salem, “Modeling, simulation, controller selection and design of electric motor for mechatronics motion applications, using different control strategies and verification using MATLAB/simulink,” Eur. Sci. J.;2013, vol. 9, no. 27,
- [7] P. C. Sen, Thyristor DC drives. Wiley, 1981.
- [8] K.-K. Nguyen and T.-T. Nguyen, “The sensorless control system for controlling the speed of direct current motor,” Indonesian Journal of Electrical Engineering and Computer Science;2019 vol. 16, no. 3, : 1171-1178.
- [9] N. Farah et al., “Analysis and investigation of different advanced control strategies for high-performance induction motor drives,” TELKOMNIKA Telecommunication, Computing, Electronics and Contro ; 2020 vol. 18, no. 6, pp. 3303-3314.
- [10] W. C. Koech, S. Rotich, T. Rotich, and F. Nyamwala, “Dynamic model of a DC motor-gear-alternator (MGA) system,” Asian Res. J. Math.; 2016 vol. 1, no. 4, pp. 1–16,
- [11] R. Krishnan, Electric motor Drives modeling, Analysis, and control. upper saddle rive Prentice Hall, 2001