

Projet GE S7

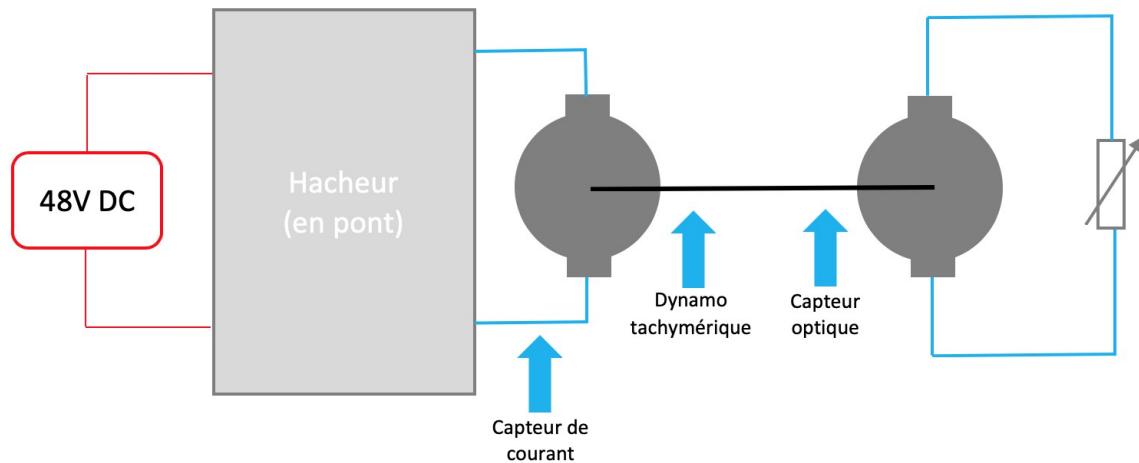
Asservissement d'une machine à courant continu

TRENCHANT Evan

TROULLIER Laël

VIRQUIN Rudy

GE4, groupe 2



Professeur : Damien Flieller

7 novembre 2025

Introduction

Dans le domaine de l'électronique de puissance et de l'automatique, la commande des machines électriques constitue un enjeu majeur pour de nombreuses applications industrielles. Parmi ces machines, la machine à courant continu (MCC) occupe une place particulière de par sa simplicité de commande et sa capacité à fournir des couples élevés à basse vitesse.

Ce projet, réalisé dans le cadre du semestre 7 de la formation en Génie Électrique à l'INSA Strasbourg, porte sur l'étude et la réalisation de l'asservissement d'une machine à courant continu avec charge. L'objectif principal est de développer un système de contrôle permettant d'asservir précisément la vitesse et le courant de la machine tout en respectant les contraintes de performance imposées.

Contexte et problématique

L'asservissement des machines électriques nécessite une approche méthodique combinant la modélisation théorique, la simulation numérique et la validation expérimentale. Dans le cas particulier de la machine à courant continu, plusieurs défis doivent être relevés :

- La modélisation précise du comportement dynamique de la machine et de sa charge
- La conception de correcteurs adaptés pour les boucles de courant et de vitesse
- La limitation des dépassements lors des régimes transitoires
- L'optimisation des performances en régime permanent et dynamique

Objectifs du projet

Ce projet vise à concevoir et valider un système d'asservissement complet pour une machine à courant continu. Les objectifs spécifiques sont les suivants :

1. **Modélisation et simulation** : Développer un modèle mathématique précis de la machine à courant continu et de sa charge, puis l'implémenter dans les environnements MATLAB/Simulink et PSIM
2. **Asservissement en courant** : Concevoir et régler une boucle de régulation de courant permettant de contrôler précisément le couple de la machine
3. **Asservissement en vitesse** : Implémenter une boucle de régulation de vitesse en cascade avec la boucle de courant, utilisant soit un capteur tachymétrique soit un codeur incrémental
4. **Respect des spécifications** : Garantir que les dépassements en vitesse et en courant restent dans la plage de 10 à 20% lors des transitoires
5. **Validation comparative** : Comparer les résultats obtenus entre les simulations MATLAB/Simulink et PSIM pour valider la cohérence des modèles

Approche méthodologique

La démarche adoptée suit une approche progressive et structurée :

- **Phase 1** : Étude et modélisation de la machine seule
- **Phase 2** : Intégration de la charge mécanique et validation du modèle complet
- **Phase 3** : Conception et réglage de la boucle de courant

- **Phase 4 :** Conception et réglage de la boucle de vitesse
- **Phase 5 :** Optimisation globale et validation finale du système d'asservissement

Chaque phase fait l'objet d'une validation croisée entre les outils MATLAB/Simulink et PSIM, permettant de garantir la fiabilité des résultats et d'identifier d'éventuelles divergences de modélisation.

Structure du rapport

Ce rapport présente de manière détaillée l'ensemble des travaux réalisés. Il s'articule autour des axes suivants :

- La modélisation théorique et numérique de la machine à courant continu
- L'analyse comparative des outils de simulation
- La conception des correcteurs et leur réglage
- La validation expérimentale des performances obtenues
- L'analyse critique des résultats et les perspectives d'amélioration

Les résultats obtenus démontrent la faisabilité d'un asservissement efficace de la machine à courant continu tout en respectant les spécifications imposées, ouvrant ainsi la voie à des applications industrielles concrètes.

Table des matières

Introduction	2
1 Cahier des charges	5
1. Notations des variables pour le projet	5
2. Spécifications du système	5
2 Planification du projet	7
3 Étude de la MCC et de son hacheur	8
1. Modélisation de la MCC à vide	8
2. Modélisation de la MCC avec charge	12
3. Modélisation de la MCC avec charge et hacheur	15
4. Simplification du schéma bloc	18
4 Étude de l'asservissement de la MCC	20
1. Asservissement en courant	20
2. Asservissement en vitesse	22
5 Dimensionnement des asservissements	27
1. Dimensionnement des correcteurs réels pour l'asservissement en courant	27
Liste des figures et chemins d'accès	29
Conclusion	30

1 - Cahier des charges

1. Notations des variables pour le projet

Le tableau suivant récapitule l'ensemble des notations utilisées dans ce projet, ainsi que les valeurs et unités correspondantes pour notre moteur à courant continu.

Tableau 1 : Notations des variables pour le projet

Notations	Nom	Valeurs	Unités
Paramètres nominaux			
M_0	couple en rotation lente	0,54	N.m
I_0	courant permanent rotation lente	4,5	A
U	tension d'alimentation de définition	49	V
N	vitesse de définition	3 000	tr/min
Paramètres maximaux			
U_{max}	tension maximale	65	V
N_{max}	vitesse maximale	4 800	tr/min
I_{max}	courant impulsional	13	A
Constantes électromécaniques			
K_e	constante de fem (à 25°C)	13,3	V/(1000 tr/min)
K_Φ	constante de couple électromagnétique	0,127	N.m/A
Paramètres de frottement			
T_f	couple de frottement sec	2,4	N.cm
K_d	coefficient de viscosité	0,53	N.cm/(1000 tr/min)
Paramètres électriques			
R	résistance du bobinage (à 25°C)	1,52	Ω
L	inductance du bobinage	2,2	mH
R_{ch}	résistance de charge	10	Ω
Paramètres mécaniques			
J	inertie du rotor	$8,3 \cdot 10^{-5}$	kg.m ²
M	masse du moteur	1,34	kg
T_{th}	constante de temps thermique	7	min
Variables dynamiques			
Ω	vitesse de rotation		rad/s
C_m	couple électromagnétique		N.m
C_r	couple résistant total		N.m
i	courant d'induit		A
u	tension d'alimentation		V
e	force contre-électromotrice		V

2. Spécifications du système

Le cahier des charges du projet est défini comme suit :

Asservissement en courant

- Temps de réponse maximal de 10 fois la période de la MLI soit 0,45 ms
- Dépassemement maximal de 20%

Asservissement en vitesse

- Dépassemement maximal de 20%

2 - Planification du projet

La réalisation de ce projet d'asservissement d'une machine à courant continu nécessite une planification rigoureuse pour garantir l'atteinte des objectifs dans les délais impartis. Cette section présente l'organisation temporelle du projet et le diagramme de Gantt détaillant les différentes phases de développement.

Diagramme de Gantt

Le diagramme de Gantt ci-dessous illustre la planification détaillée du projet sur 9 semaines, avec les dépendances entre les tâches et les jalons importants.

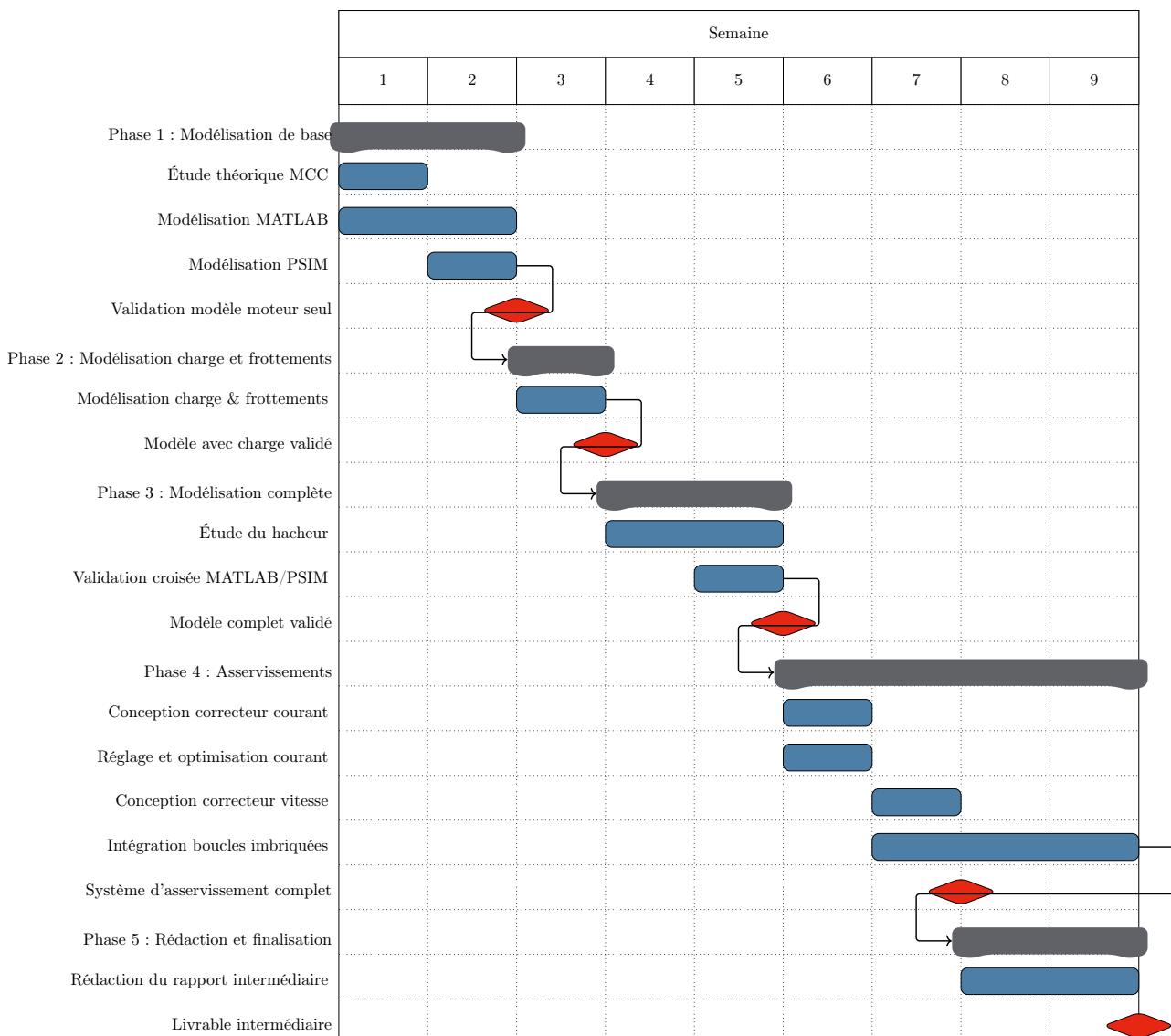


FIGURE 2.1 – Diagramme de Gantt — récapitulatif jusqu'à 9 semaines

3 - Étude de la MCC et de son hacheur

1. Modélisation de la MCC à vide

1.1. Équations de la MCC

La machine à courant continu (MCC) est un dispositif électromécanique capable de convertir l'énergie électrique en énergie mécanique, et inversement. Cette conversion bidirectionnelle s'effectue grâce au passage d'un courant continu dans son induit, générant ainsi un couple moteur qui entraîne l'arbre en rotation. En mode générateur, le principe est inversé : la rotation de l'arbre produit une force électromotrice aux bornes de la machine. La MCC constitue ainsi un élément clé dans de nombreux systèmes d'entraînement industriels et de conversion d'énergie.

Formules d'une MCC sans pertes et sans charge

Les équations régissant le comportement d'une machine à courant continu sans pertes et sans charge sont les suivantes :

Équations temporelles de la MCC

$$u(t) = R \cdot i(t) + L \frac{di(t)}{dt} + e(t)$$

$$C_m(t) = K_c \cdot i(t)$$

$$e(t) = K_e \cdot \omega(t)$$

$$J \frac{d\omega}{dt} = C_m(t)$$

On note ici : $K_c = K_e = K_\Phi$

Transformée de Laplace

Les équations précédentes peuvent être exprimées dans le domaine de Laplace, ce qui facilite l'analyse et la modélisation du système :

Équations dans le domaine de Laplace

$$I(s) = \frac{U(s) - K_\Phi \Omega}{R + Ls}$$

$$C_m(s) = K_\Phi \cdot I(s)$$

$$E(s) = K_\Phi \cdot \Omega(s)$$

$$\Omega(s) = \frac{K_\Phi \cdot I(s)}{Js}$$

Conversion vitesse-tension

Le moteur possède une force contre-électromotrice E proportionnelle à sa vitesse de rotation. Les données constructeur indiquent : $E = 13,3$ V pour 1000 tr/min.

En notant N la vitesse en tr/min et Ω la vitesse en rad/s, on obtient les relations suivantes :

$$E = \frac{13,3}{1000} \cdot N$$

$$\Omega = \frac{\pi N}{30} \Rightarrow E = \underbrace{\frac{13,3}{1000} \cdot \frac{30}{\pi}}_{0,127} \cdot \Omega$$

La constante de fcem est donc : $K_\Phi = 0,127$ V/(rad/s).

Une fois le modèle mathématique de la MCC défini à partir des équations présentées ci-dessus, nous pouvons procéder à sa simulation numérique. Cette simulation sera réalisée successivement sur deux environnements logiciels complémentaires : Simulink, qui permet une approche par schéma-blocs, et PSIM, orienté vers la simulation de circuits de puissance. La confrontation des résultats obtenus avec les deux outils permettra de valider notre modélisation théorique.

1.2. Simulation avec Simulink

Dans l'environnement Simulink, notre démarche consiste à traduire le système d'équations différentielles sous forme d'un schéma-blocs interconnectés, où chaque bloc représente une opération mathématique élémentaire.

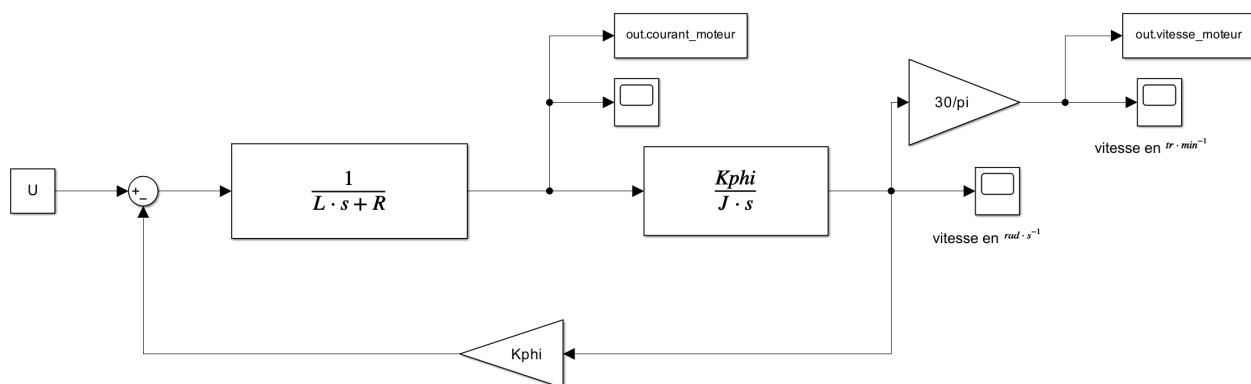


FIGURE 3.1 – Schéma Simulink du moteur à courant continu à vide

On remarque que le schéma Simulink présenté en Figure 3.1 intègre les équations de la MCC sans charge. Les simulations effectuées avec ce modèle permettent d'obtenir les réponses temporelles du courant et de la vitesse du moteur à vide, illustrées dans la Figure 3.2.

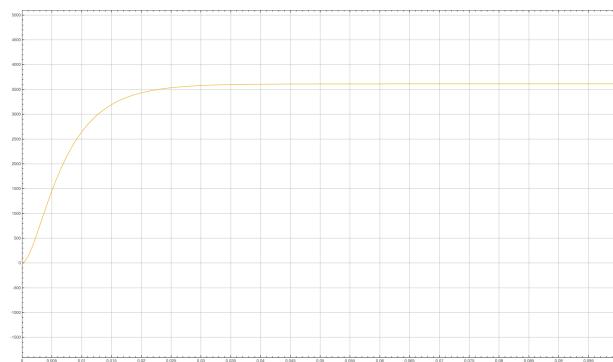
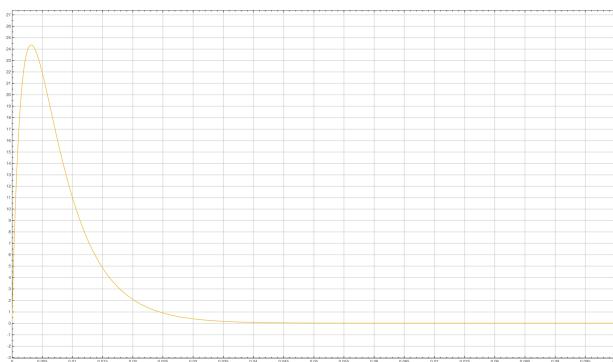


FIGURE 3.2 – Courbes de réponse du moteur à courant continu

1.3. Simulation sur PSIM

Pour la simulation sur PSIM, nous adoptons une approche méthodologique complémentaire. L'utilisation de deux méthodes distinctes (Simulink et PSIM) constitue une validation croisée : si les résultats concordent, nous pourrons confirmer la validité de nos modélisations.

Contrairement à Simulink qui nécessite la construction du modèle à partir des équations différentielles, PSIM propose une approche plus directe basée sur l'utilisation de composants pré-définis. Le logiciel met à disposition une bibliothèque de machines à courant continu standards qu'il suffit de paramétrier avec les caractéristiques de notre moteur réel. Cette méthodologie se rapproche davantage du fonctionnement physique du système et réduit significativement le nombre de calculs intermédiaires.

Le principe de modélisation sous PSIM repose sur l'assemblage de composants dont on définit ensuite les propriétés. Par exemple, pour simuler notre moteur, nous plaçons simplement un bloc « moteur DC » dans l'environnement de travail, puis nous renseignons ses paramètres électriques et mécaniques.

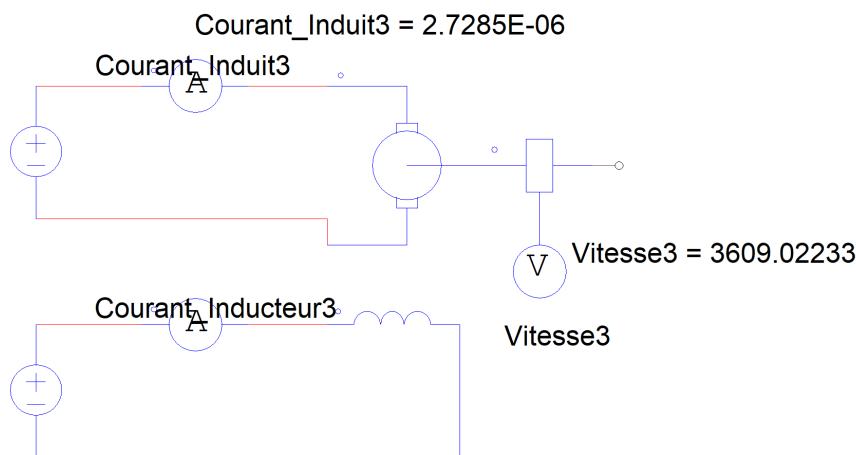


FIGURE 3.3 – Modélisation du moteur à courant continu sur PSIM

1.4. Comparaison des résultats

Nous comparons les résultats avec un programme *MATLAB*. Pour cela, nous avons besoin de récupérer les valeurs de notre modélisation sous forme de tableau : $X = [t \ i(t) \ \omega(t)]$.

On retrouve bien les valeurs attendues, notamment en vitesse où on a la relation suivante à

l'état stable :

$$\omega = \frac{U}{K_\Phi} = \frac{48}{0,127} = 377,95 \text{ rad/s} \quad \text{ou} \quad \omega = \frac{U}{K_e} = \frac{48}{13,3} = 3609 \text{ tr/min}$$

Les courbes de la Figure 3.4 montrent une excellente concordance entre les résultats obtenus avec Simulink et PSIM, validant ainsi notre modélisation du moteur à courant continu à vide. Nous pouvons donc passer aux simulations suivantes : le moteur avec des pertes et sa charge.

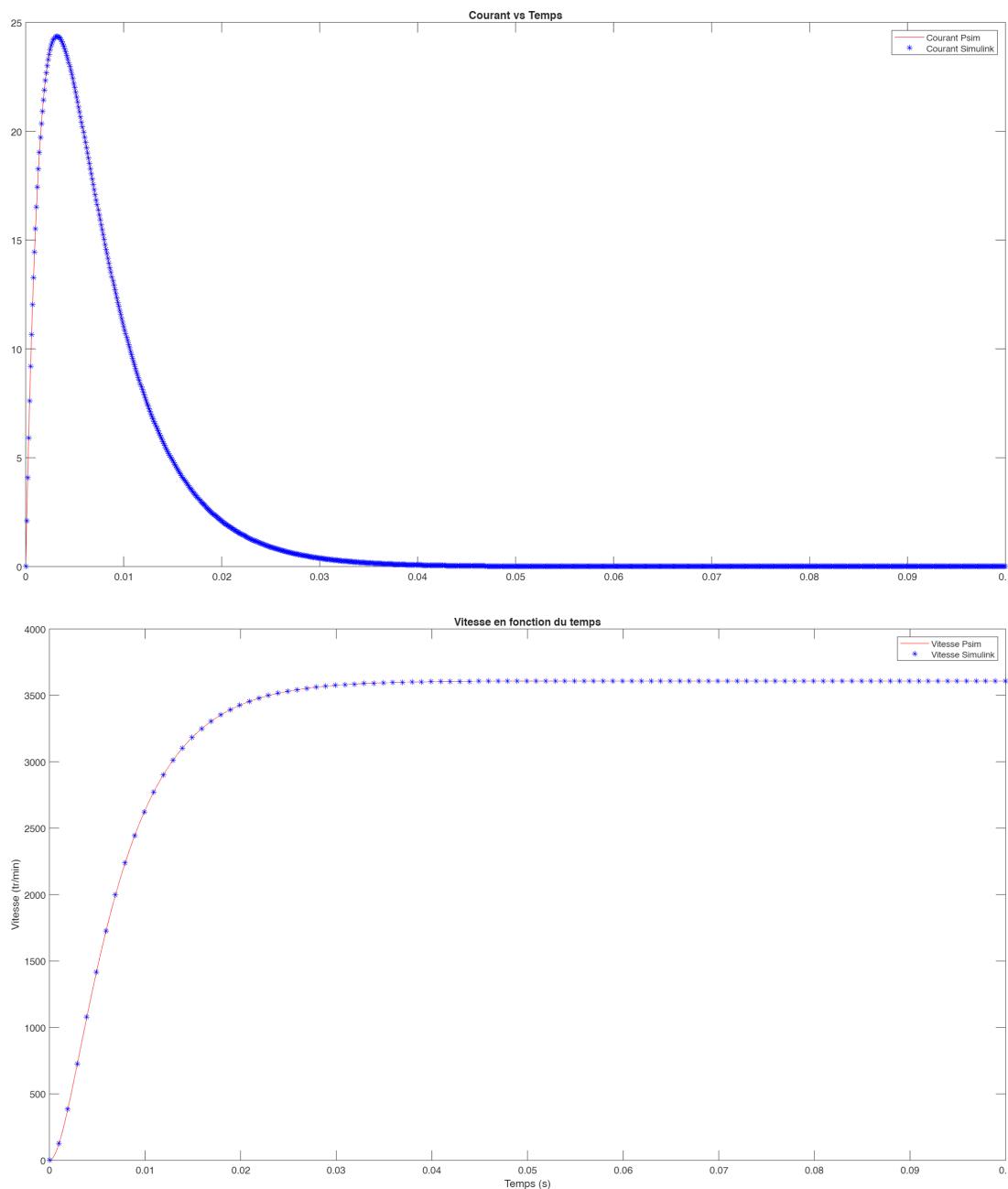


FIGURE 3.4 – Comparaison des résultats Simulink et PSIM pour le moteur à vide

2. Modélisation de la MCC avec charge

Dans cette partie, nous allons effectuer la même logique qu'avec le moteur à vide. Cette fois-ci le moteur a des pertes, appelées frottements, et le couple résistant du moteur est pris en compte. On note :

$$f = C_f \cdot \Omega = \frac{30}{\pi} \cdot \frac{1}{1000} \cdot 0,53 \cdot 10^{-2} = 5,06 \cdot 10^{-5} \text{ N.m/rad.s}$$

$$C_r = 2 \cdot C_0 = 0,024 \text{ N.m}$$

2.1. Équations de la MCC avec pertes

Les équations de notre système vont donc changer. En effet en ajoutant les frottements, le système devient :

Formules d'une MCC avec pertes

$$u(t) = R \cdot i(t) + L \frac{di(t)}{dt} + e(t)$$

$$C_m(t) = K_c \cdot i(t)$$

$$e(t) = K_e \cdot \omega(t)$$

$$J \frac{d\omega}{dt} = C_m(t) - f \cdot \omega(t) - C_r(t)$$

Ce qui donne dans le domaine de Laplace :

Équations dans le domaine de Laplace

$$I(s) = \frac{U(s) - K_\Phi \Omega}{R + Ls}$$

$$C_m(s) = K_\Phi \cdot I(s)$$

$$E(s) = K_\Phi \cdot \Omega(s)$$

$$\Omega(s) = \frac{C_m + C_r}{Js + f}$$

Une fois les équations trouvées, nous allons pouvoir déterminer le schéma bloc et effectuer les mesures.



2.2. Simulation Simulink

Le schéma bloc n'est plus le même. En effet, celui-ci devient plus complexe, puisque qu'on y ajoute le couple résistant C_r et les frottements f .

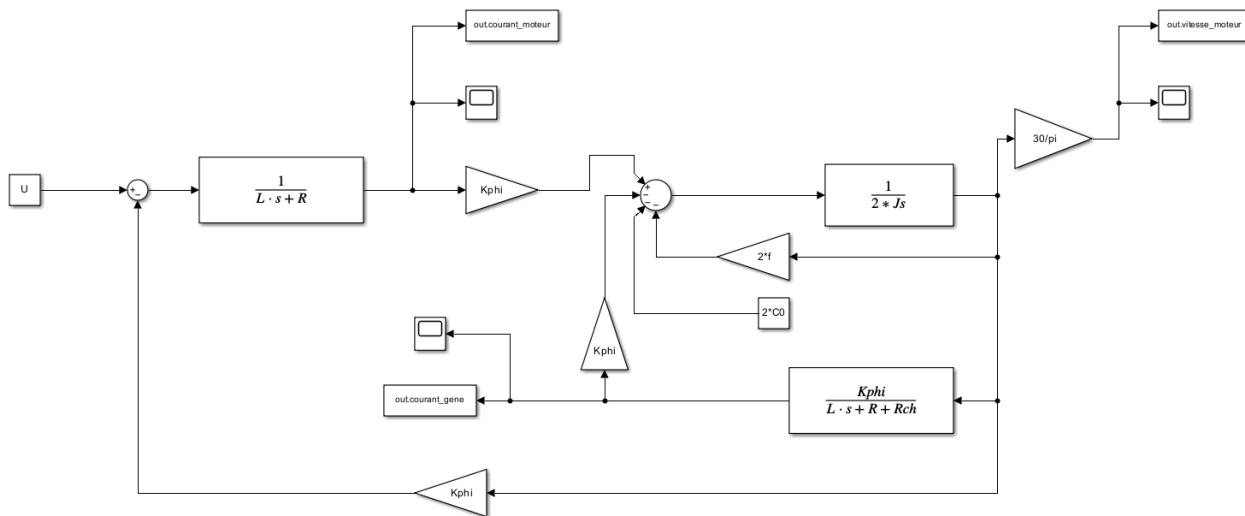


FIGURE 3.5 – Schéma Simulink du moteur à courant continu avec charge et frottements

2.3. Simulation sur PSIM

De manière analogue à la simulation du moteur à vide, nous utilisons PSIM pour valider le modèle Simulink incluant cette fois-ci la charge et les frottements. La démarche reste similaire : nous utilisons le bloc moteur à courant continu de PSIM, mais en y ajoutant les paramètres de couple résistant (C_r) et de frottement visqueux (f). Cette approche directe, basée sur des composants physiques, permet de vérifier la cohérence des résultats obtenus par la modélisation par équations dans Simulink.

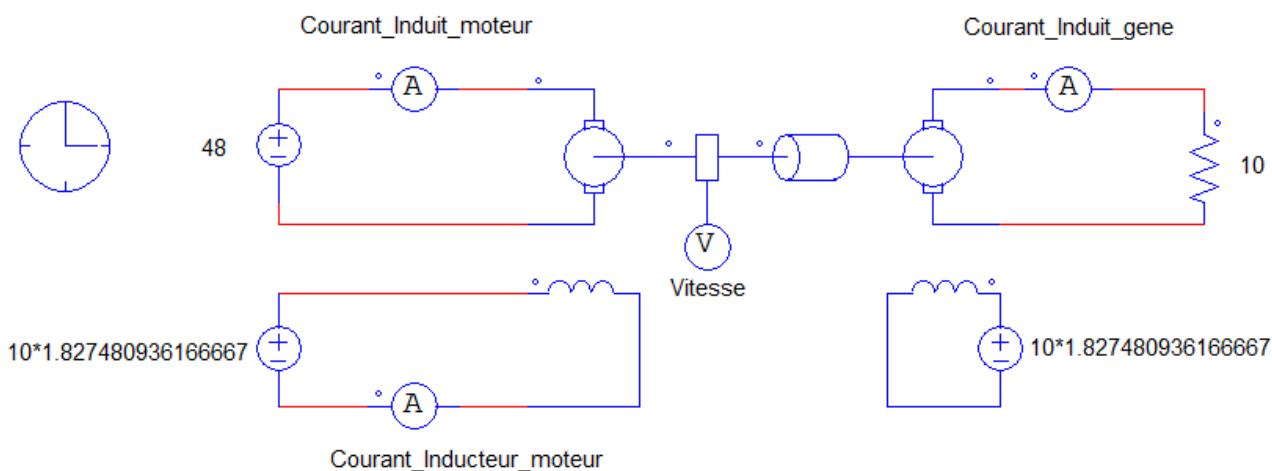


FIGURE 3.6 – Modélisation du moteur avec charge et frottements sur PSIM

2.4. Comparaison des résultats

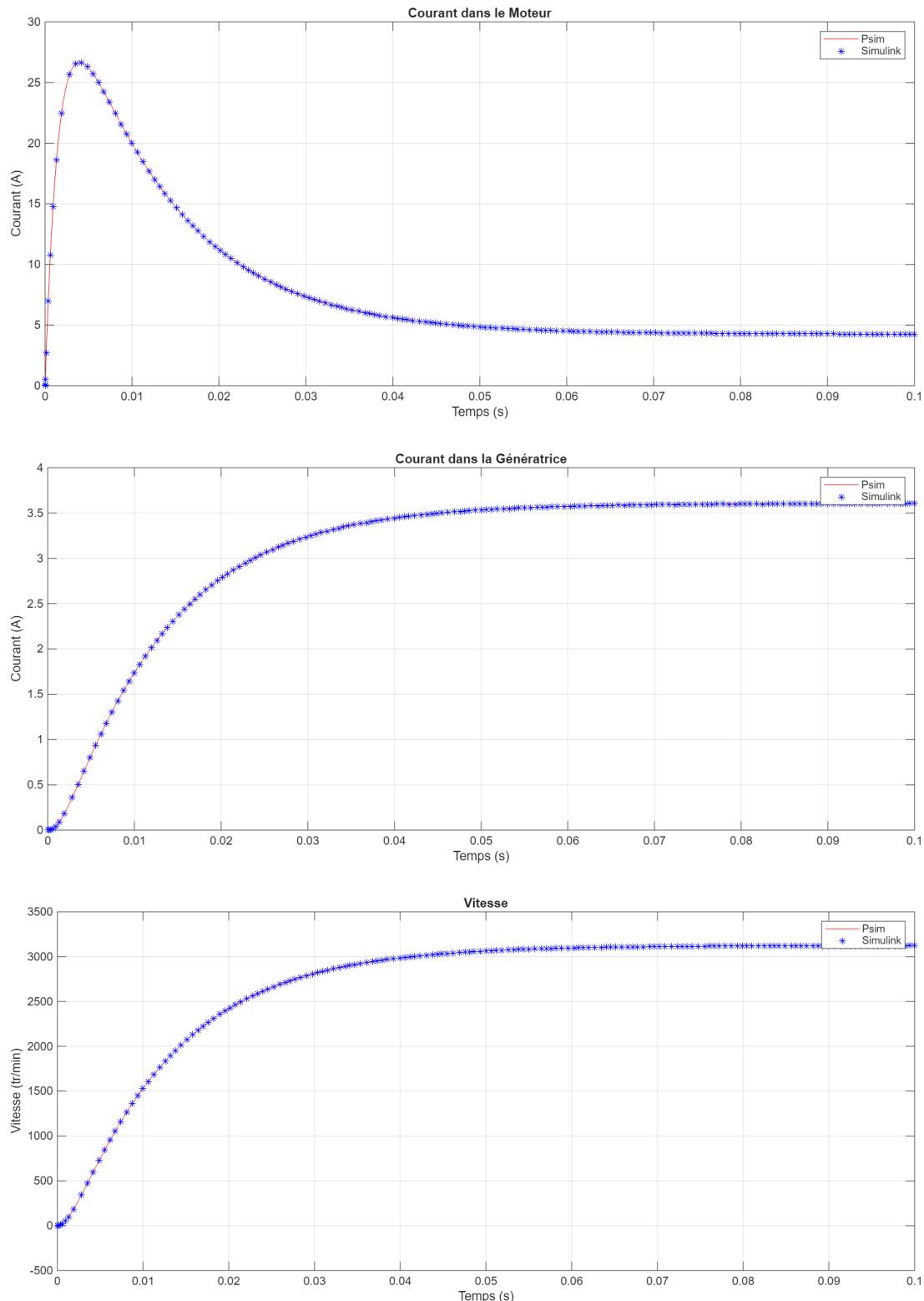


FIGURE 3.7 – Comparaison des résultats Simulink et PSIM pour le moteur avec charge

Les courbes de la Figure 3.7 confirment la validité de notre modélisation avec charge et frottements. On observe une superposition quasi-parfaite des résultats Simulink (marqueurs) et PSIM (ligne continue) sur l'ensemble des grandeurs mesurées. La vitesse atteint un régime permanent à environ 3100 tr/min, valeur inférieure à celle du moteur à vide (3609 tr/min) en raison de la présence de la charge de 10Ω , du couple résistant C_r et des frottements visqueux. Le cou-

rant d'induit présente un pic initial caractéristique du démarrage, puis se stabilise à une valeur permanente correspondant à l'équilibre entre le couple moteur et les couples résistifs. Cette concordance valide l'implémentation des pertes mécaniques dans les deux environnements de simulation.

3. Modélisation de la MCC avec charge et hacheur

3.1. Modélisation du hacheur

Le hacheur, ou convertisseur continu–continu, est un dispositif d'électronique de puissance qui permet de modifier la valeur moyenne d'une tension continue à l'aide d'interrupteurs commandés. Il est couramment utilisé pour piloter des charges électriques (ici une machine à courant continu) avec un bon rendement.

Sur PSIM, le hacheur est modélisé par un pont en H composé de quatre transistors MOSFET pilotés en paires. Le schéma fonctionnel comporte le pont en H alimenté par une source continue (ici 48 V), un générateur de signal triangulaire et un comparateur qui produit le signal PWM de commande.

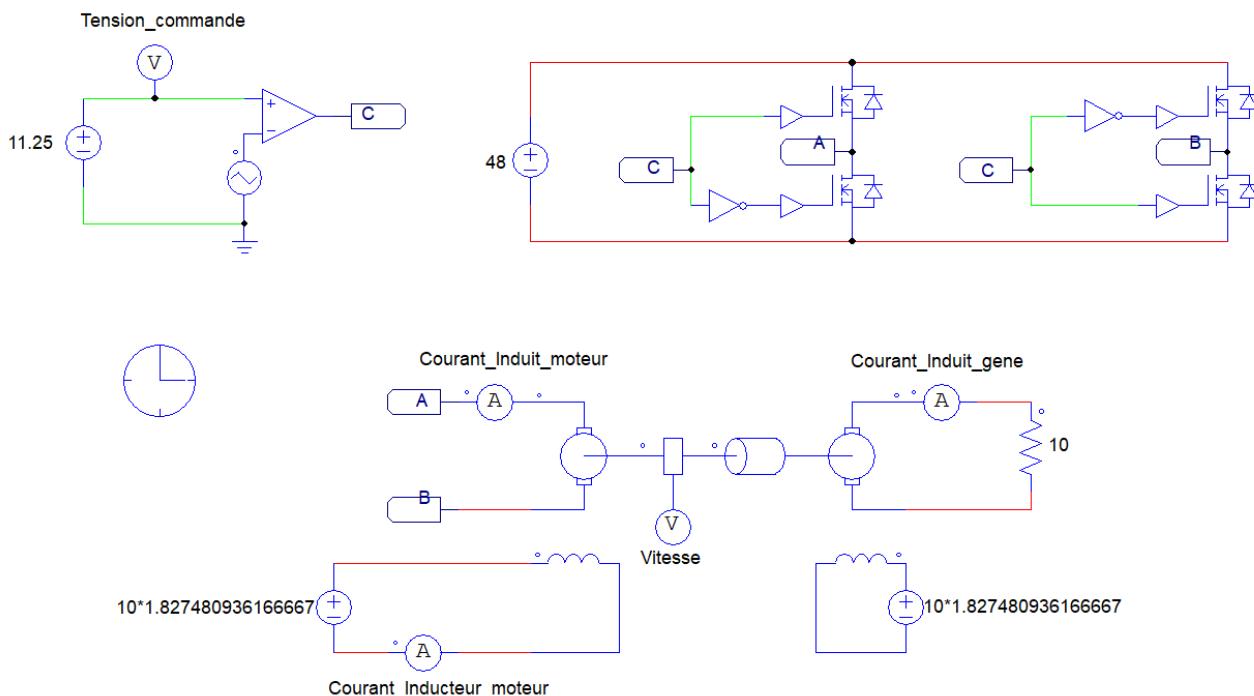


FIGURE 3.8 – Modélisation du hacheur sur PSIM

Le hacheur fournit une tension de sortie moyenne qui peut être positive ou négative (mode bipolaire). La valeur moyenne de la tension de sortie V_s dépend de trois grandeurs : la tension d'alimentation (ici 48 V), la tension de commande continue (notée V_{commande}) et le rapport cyclique α du signal de commande (duty cycle) appliqué aux interrupteurs.

La commutation des transistors s'effectue par paires (par exemple 1 et 4 d'une part, 2 et 3 d'autre part) de façon complémentaire afin de piloter le pont en H. Le rapport cyclique α correspond à la proportion de la période pendant laquelle l'interrupteur est fermé. Pour $\alpha = 0$ la tension de sortie vaut $V_s = -48 \text{ V}$, pour $\alpha = 1$ on obtient $V_s = +48 \text{ V}$, et pour des valeurs intermédiaires de α la tension de sortie varie de manière (approximativement) linéaire entre ces extrêmes. Ainsi, une augmentation de α entraîne une augmentation de V_s .

La commande PWM est générée en comparant, au moyen d'un comparateur (ou d'un AOP configuré), une tension continue de référence $V_e = V_{\text{commande}}$ avec un signal triangulaire V_{triangle}

qui varie typiquement de 0 à 15 V. Le comparateur produit alors le signal V_{pwm} dont le rapport cyclique est proportionnel à V_{commande} . En sortie du comparateur, des suiveurs (buffers) sont placés pour assurer une adaptation d'impédance et piloter correctement les étages de commande des MOSFET.

3.2. Caractéristique du hacheur

Le hacheur utilisé possède une caractéristique de transfert non-linéaire avec une zone morte. La tension de sortie moyenne $\langle u_{AB} \rangle$ dépend de la tension de commande u_{commande} selon la relation suivante :

Relation tension de sortie :

$$\begin{aligned}\langle u_{AB} \rangle &= \frac{96}{15} (u_{\text{commande}} - 7,5) \\ &= \frac{96}{15} u_{\text{commande}} - 48 \\ &= 6,4 u_{\text{commande}} - 48\end{aligned}$$

Caractéristiques principales :

- Zone morte : $0 \leq u_{\text{commande}} < 7,5$ V
- Pente : 6,4 V/V
- Tension maximale : 48 V à 15 V de commande
- Point de fonctionnement intermédiaire (en rouge) : (11,25 V; 24 V)

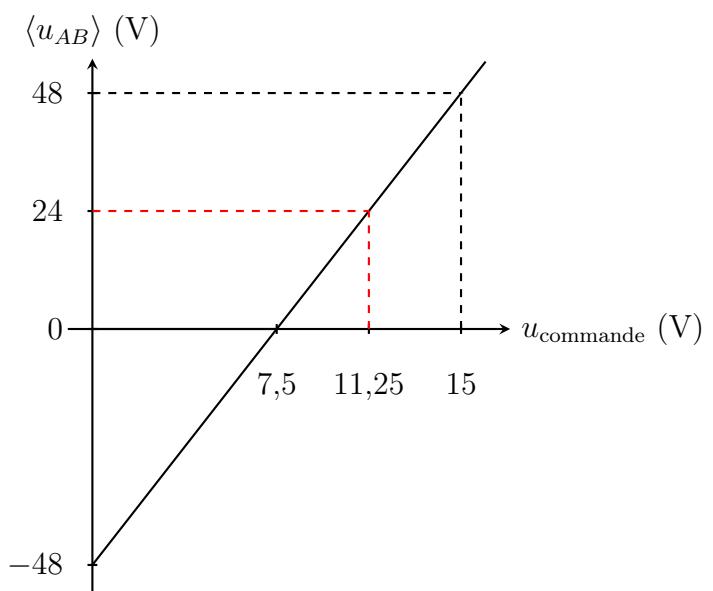


FIGURE 3.9 – Caractéristique de transfert du hacheur

Cette caractéristique montre que le hacheur présente une zone morte pour les faibles tensions de commande (inférieures à 7,5 V), puis un comportement linéaire avec un gain de 6,4. La tension de sortie est limitée à ± 48 V.

3.3. Simulation sur Simulink

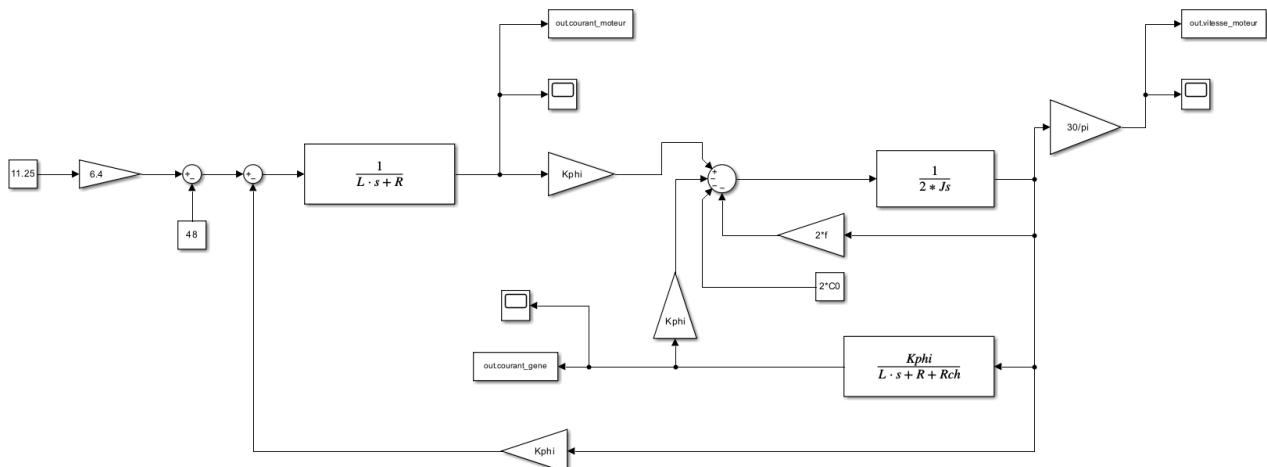


FIGURE 3.10 – Modélisation du moteur avec hacheur sur Simulink

3.4. Comparaison des résultats

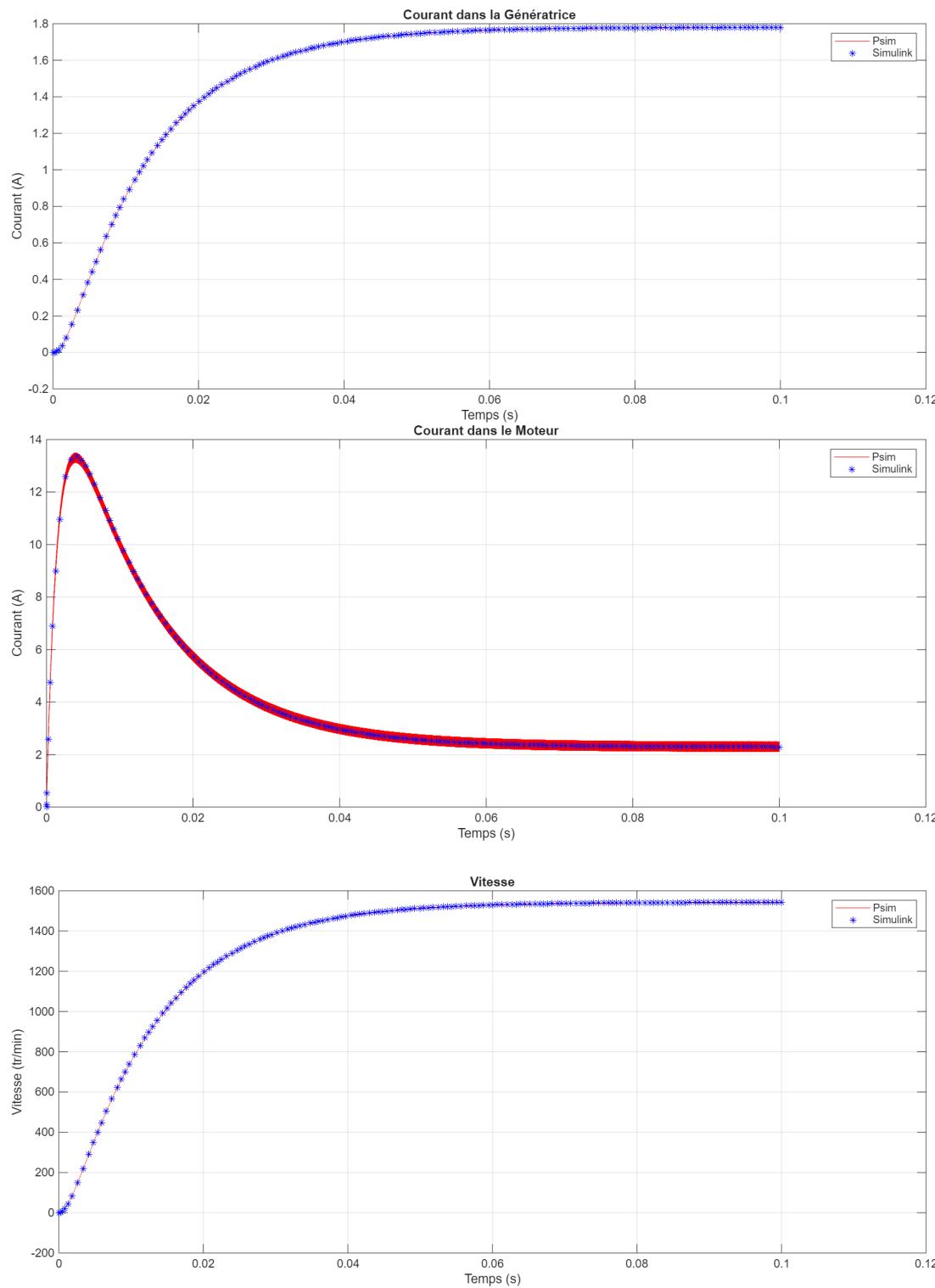


FIGURE 3.11 – Comparaison des résultats Simulink et PSIM pour le moteur avec hacheur

Les résultats présentés dans la Figure 3.11 démontrent la cohérence entre les modélisations Simulink et PSIM du système complet moteur-hacheur. Les trois grandeurs observées (courant génératrice, courant moteur et vitesse) montrent une excellente superposition des courbes issues des deux simulateurs. On constate notamment que l'introduction du hacheur impose une dynamique légèrement différente par rapport à l'alimentation directe, avec des ondulations

caractéristiques de la modulation PWM. La vitesse converge vers une valeur de régime permanent proche de 3100 tr/min, cohérente avec le point de fonctionnement imposé par la tension de commande du hacheur. Cette validation croisée confirme la justesse de notre modélisation du convertisseur statique et de son interaction avec la charge mécanique.

4. Simplification du schéma bloc

A ce stade de l'analyse, nous disposons du schéma bloc complet de notre système, avant asservissement en vitesse. Comme illustré dans la Figure 3.12, ce schéma est assez complexe en raison de la présence de plusieurs boucles de rétroaction imbriquées. Pour faciliter l'analyse et la conception des contrôleurs, il est nécessaire de simplifier ce schéma bloc en combinant les différents éléments.

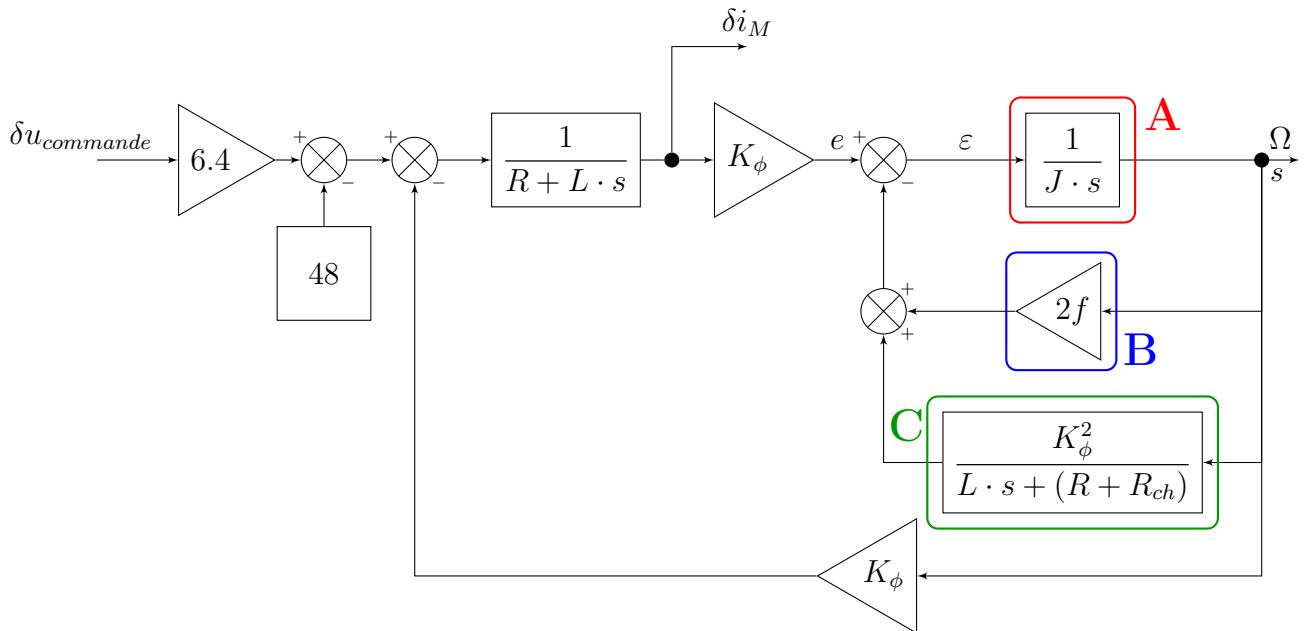


FIGURE 3.12 – Schéma bloc de l'asservissement en courant avec boucle de courant imbriquée

Cette simplification permet d'obtenir une représentation plus concise du système, facilitant ainsi l'analyse et la conception des contrôleurs.

$$\begin{cases} \varepsilon = e - \textcolor{blue}{B} \cdot \Omega - \textcolor{green}{C} \cdot \Omega \\ s = \textcolor{red}{A} \cdot \varepsilon = \textcolor{red}{A} \cdot (e - (\textcolor{blue}{B} + \textcolor{green}{C}) \cdot s) \end{cases} \implies s = \frac{\textcolor{red}{A} \cdot e}{1 + \textcolor{red}{A} \cdot (\textcolor{blue}{B} + \textcolor{green}{C})} = H \cdot e$$

Le diagramme bloc simplifié de l'asservissement en vitesse est présenté dans la Figure 3.13.

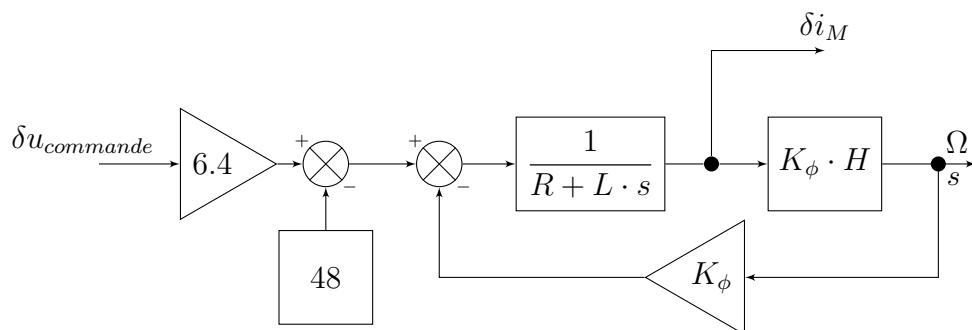


FIGURE 3.13 – Schéma bloc de l'asservissement en courant simplifié

On peut alors écrire la fonction de transfert en boucle ouverte du système simplifié :

$$\delta_{i_M}(s) = \frac{1}{1 + \frac{(K_\phi)^2 \cdot H}{R + L \cdot s}} \cdot \delta u_{commande}(s) = FTBO(s) \cdot \delta u_{commande}(s)$$

Pour s'assurer que notre fonction de transfert est correcte, il faudrait la valider par des simulations numériques en comparant les résultats obtenus. Il serait alors possible de poursuivre l'analyse en utilisant cette fonction de transfert simplifiée pour concevoir et optimiser le contrôleur de l'asservissement en vitesse du moteur.



4 - Étude de l'asservissement de la MCC

1. Asservissement en courant

L'asservissement en courant constitue la boucle interne de notre système de régulation en cascade. Son rôle principal est d'assurer que le courant absorbé par le moteur suive fidèlement la consigne fournie par l'asservissement en vitesse. Cette régulation permet de contrôler précisément le couple moteur et d'améliorer la dynamique globale du système.

1.1. Modélisation sur Simulink

L'asservissement en courant doit appliquer la consigne déterminée par l'asservissement en vitesse. Pour cela, nous mettons en œuvre un correcteur Proportionnel-Intégral (PI). Le cahier des charges impose un temps de réponse égal à 10 fois la période de la modulation de largeur d'impulsion (MLI), soit 0.45 ms, avec un dépassement maximal de 20 %.

À l'aide de l'outil PID Tuner de Simulink, nous obtenons les paramètres optimaux suivants pour le correcteur PI :

- Gain proportionnel : $K_p = 5,053$
- Gain intégral : $K_i = 36349,55$

Il est important de souligner que la notation du correcteur PI diffère entre Simulink et PSIM. Sur PSIM, la valeur du paramètre τ représente le rapport $\frac{K_p}{K_i}$, tandis que sur Simulink, les deux gains sont exprimés directement.

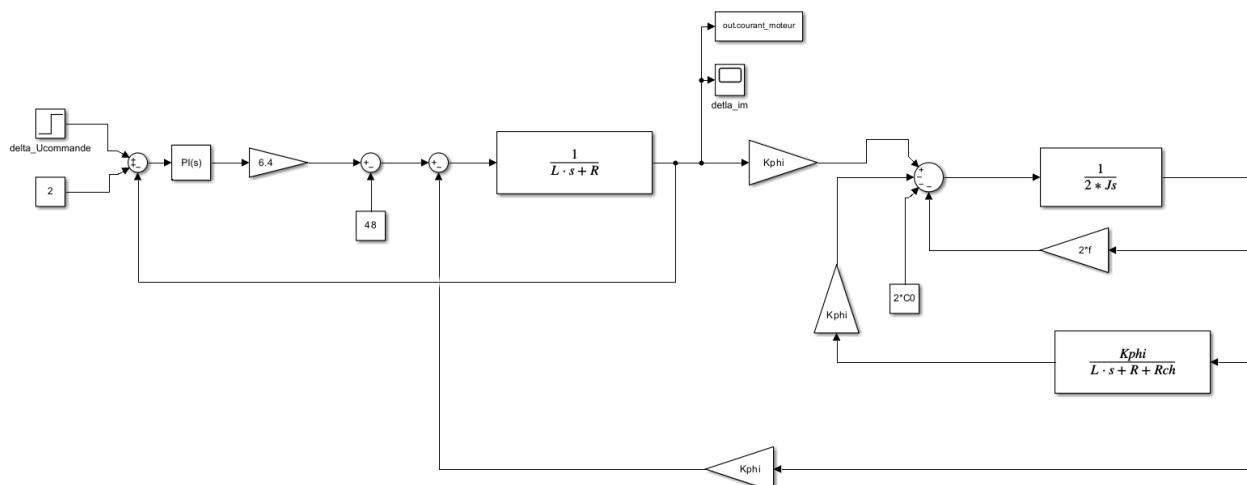


FIGURE 4.1 – Schéma de l'asservissement en courant du moteur dans Simulink

1.2. Modélisation sur PSIM

Pour adapter le correcteur PI de Simulink à PSIM, il faut convertir les paramètres en utilisant les relations suivantes :

Dans Simulink :

$$G(s) = K_p + \frac{K_i}{s} = K_p \frac{s + \frac{K_i}{K_p}}{s}$$

Dans PSIM :

$$G(s) = k \frac{1 + sT}{sT} \Rightarrow k = K_p \text{ et } T = \frac{K_p}{K_i}$$

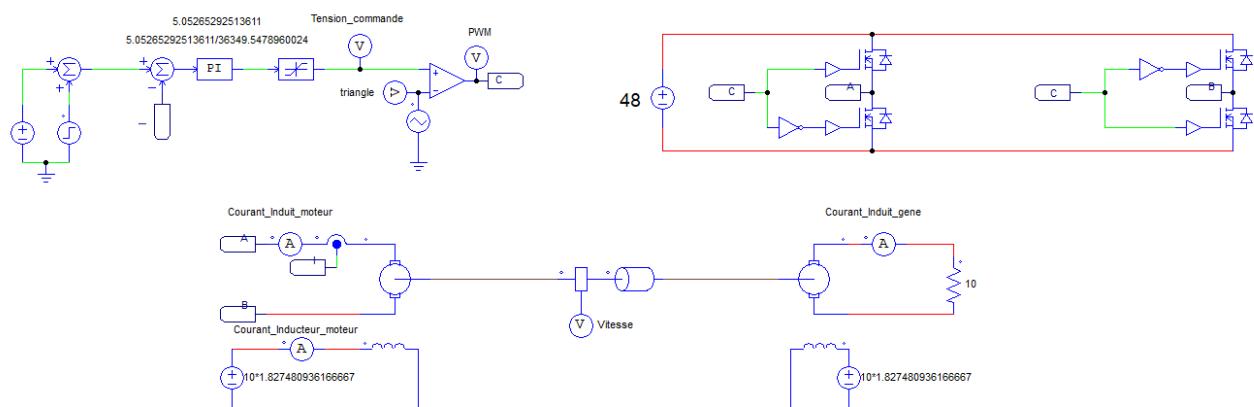


FIGURE 4.2 – Schéma de l’asservissement en courant du moteur dans PSIM

1.3. Comparaison des résultats

Dans cette section, nous allons comparer les résultats obtenus avec les deux outils de simulation, Simulink et PSIM.

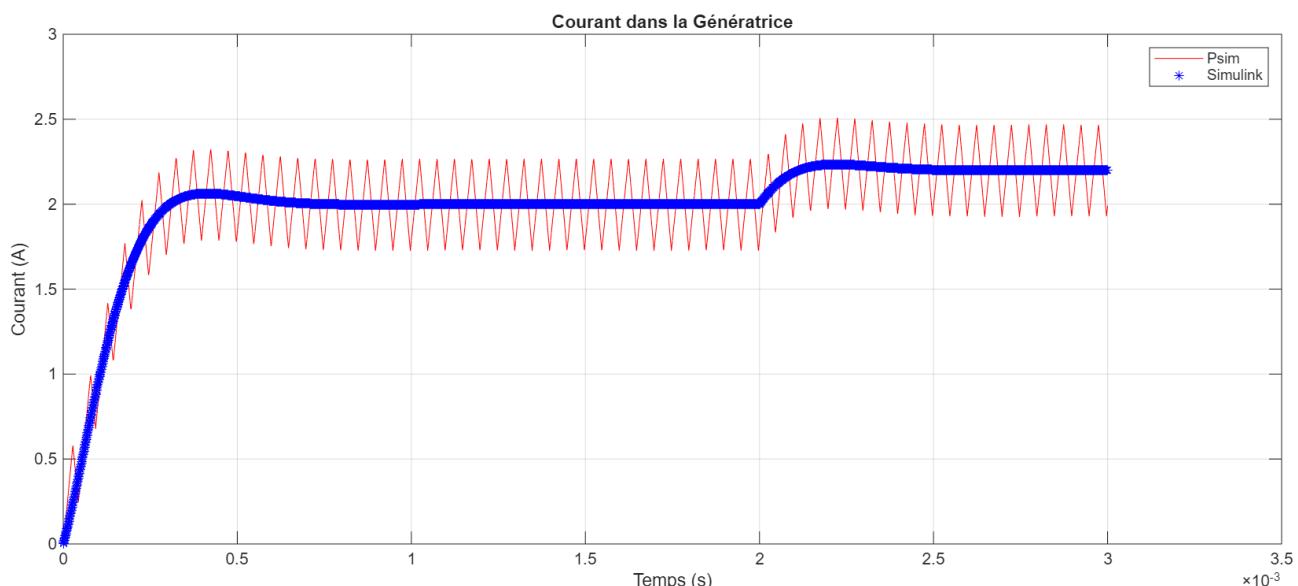


FIGURE 4.3 – Comparaison des réponses indicielles de l’asservissement en courant du moteur entre Simulink et PSIM

La figure 4.3 illustre la comparaison des réponses indicielles de l’asservissement en courant du moteur entre Simulink et PSIM. Les deux courbes sont quasiment superposées, ce qui confirme la cohérence des modélisations réalisées dans les deux environnements.

On constate que la réponse indicelle de l’asservissement en courant du moteur présente un dépassement d’environ 19% et un temps de réponse de 0,35 ms. Ce qui respecte les spécifications du cahier des charges qui exige un dépassement maximal de 20% et un temps de réponse maximal de 10 fois la périodicité de la MLI soit 0,45 ms.

2. Asservissement en vitesse

L'asservissement en vitesse constitue la boucle externe et la régulation principale de notre système de contrôle en cascade. Contrairement à l'asservissement en courant qui opère sur une grandeur électrique interne, l'asservissement en vitesse agit directement sur la grandeur mécanique d'intérêt pour l'application finale. C'est donc cet asservissement qui détermine les performances réelles du système vis-à-vis des exigences du cahier des charges.

La régulation en vitesse permet de garantir que le moteur atteint et maintient la vitesse de rotation souhaitée, indépendamment des perturbations externes telles que les variations de charge ou les frottements. Elle assure également la rapidité de la réponse du système face aux changements de consigne, tout en limitant les dépassements pour éviter les sollicitations mécaniques excessives.

Dans cette section, nous allons d'abord caractériser le comportement du système en boucle ouverte pour identifier ses limitations, puis nous concevrons un correcteur PI permettant d'améliorer significativement les performances dynamiques du moteur.

2.1. Caractérisation en boucle ouverte

Nous commençons par la simulation de la vitesse du moteur en boucle ouverte sur Simulink afin d'établir une référence de performance. Nous réalisons le schéma suivant :

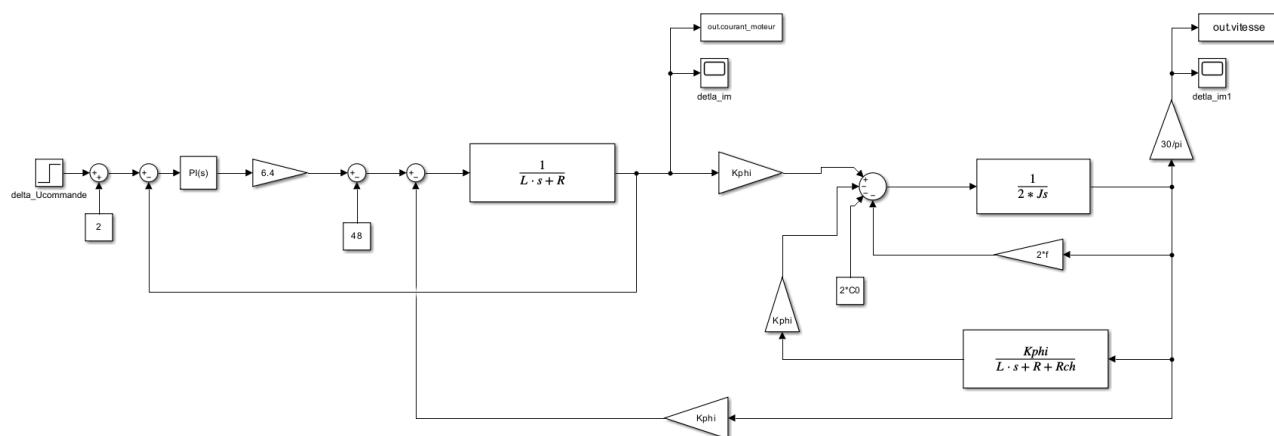


FIGURE 4.4 – Schéma de la simulation en boucle ouverte de la vitesse du moteur dans Simulink

La simulation en boucle ouverte nous permet d'observer la réponse naturelle du système sans rétroaction. Cette étape est essentielle pour comprendre les limites intrinsèques du système et établir les objectifs d'amélioration pour l'asservissement. Pour effectuer cette caractérisation, nous attendons d'abord que le système atteigne son régime permanent, puis nous appliquons une consigne en échelon pour observer la dynamique de réponse.

L'analyse de la réponse indicielle révèle un temps de réponse à 5 % en boucle ouverte de 124.1 ms. Cette valeur relativement élevée indique une dynamique lente qui ne serait pas acceptable pour de nombreuses applications industrielles nécessitant des temps de réaction rapides.

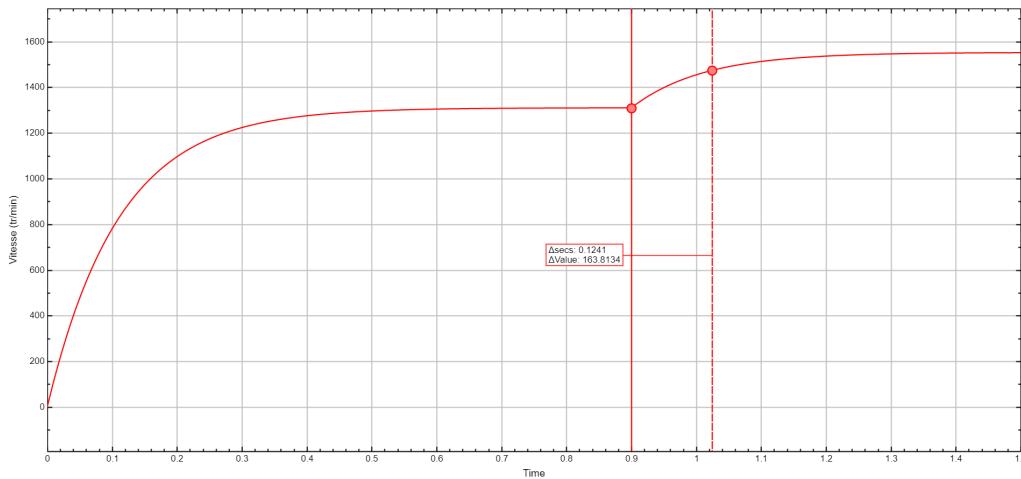


FIGURE 4.5 – Réponse de la vitesse du moteur en boucle ouverte dans Simulink

2.2. Conception du correcteur PI

Face aux limitations observées en boucle ouverte, nous mettons en œuvre un asservissement en boucle fermée avec un correcteur PI. L'objectif de cet asservissement est d'améliorer drastiquement les performances dynamiques du système. Nous visons un temps de réponse à 5 % trois fois plus rapide qu'en boucle ouverte, soit 41.37 ms, tout en maintenant un dépassement maximal de 20 % pour limiter les contraintes mécaniques.

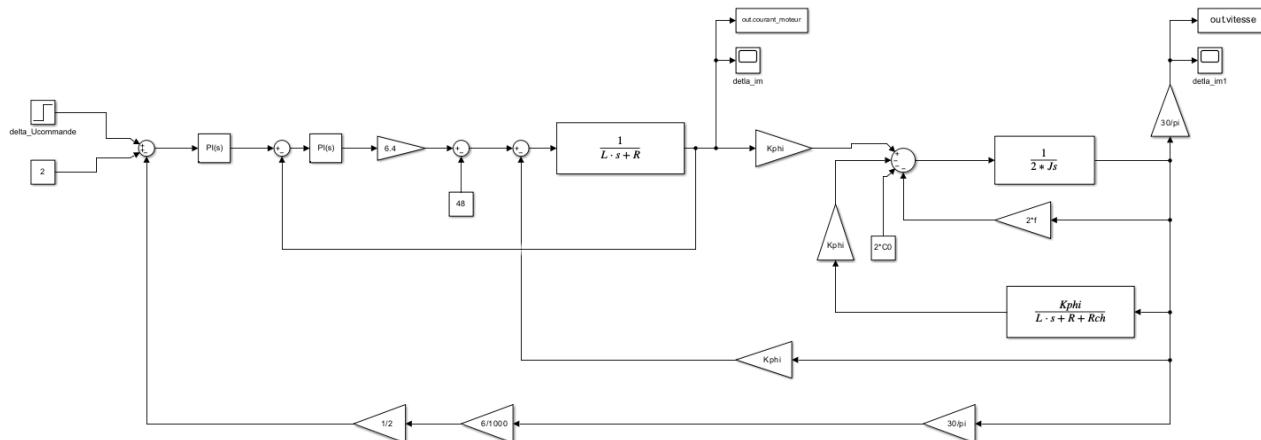


FIGURE 4.6 – Schéma de l'asservissement en vitesse du moteur dans Simulink

Le schéma présente l'architecture complète du système de régulation en cascade : la boucle externe d'asservissement en vitesse génère une consigne de courant qui est ensuite régulée par la boucle interne d'asservissement en courant. Cette structure en cascade permet de bénéficier d'une meilleure robustesse et d'une dynamique optimisée.

À l'aide de l'outil PID Tuner de Simulink, nous déterminons les paramètres optimaux du correcteur PI permettant de satisfaire les spécifications temporelles imposées :

- Gain proportionnel : $K_p = 6,68$
- Gain intégral : $K_i = 496,43$

Ces valeurs résultent d'un compromis entre rapidité et stabilité, garantissant une réponse dynamique tout en évitant les oscillations excessives qui pourraient endommager le système mécanique.

2.3. Validation sur PSIM

Après avoir conçu et validé le correcteur PI sur Simulink, nous procédons à l'implémentation du système complet sur PSIM. Cette étape est cruciale car PSIM offre une modélisation plus proche de la réalité physique, notamment en ce qui concerne les composants électroniques de puissance et leurs caractéristiques non-idiées. Comme pour l'asservissement en courant, il est nécessaire d'adapter les paramètres du correcteur PI de Simulink vers la notation utilisée par PSIM, en utilisant les relations de conversion présentées précédemment :

$$k = K_p \quad \text{et} \quad T = \frac{K_p}{K_i}$$

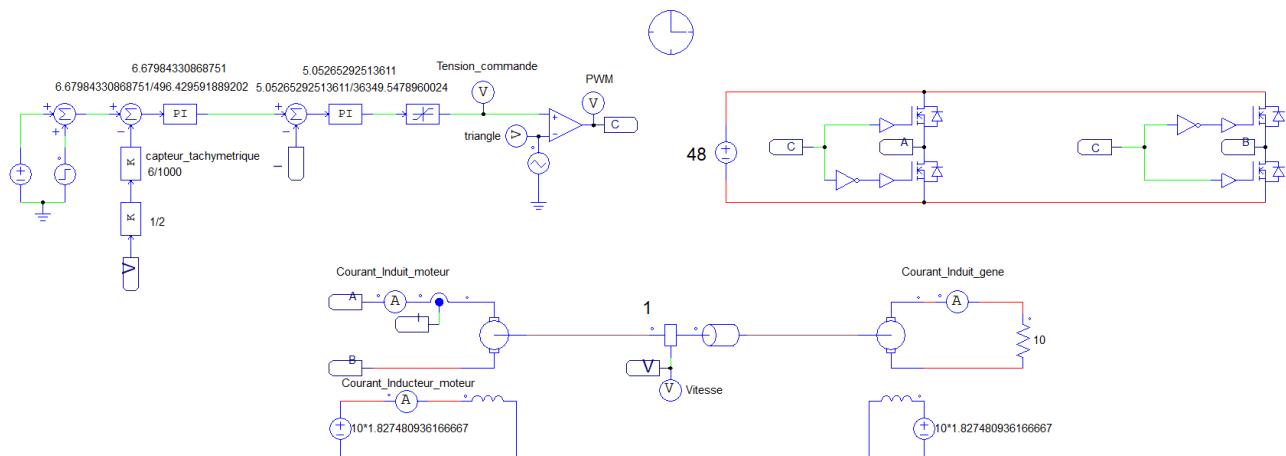


FIGURE 4.7 – Schéma de l'asservissement en vitesse du moteur dans PSIM

Le schéma PSIM intègre l'ensemble de la chaîne de régulation en cascade : l'asservissement en vitesse commande l'asservissement en courant, qui lui-même pilote le hacheur quatre quadrants alimentant le moteur à courant continu. Cette modélisation complète permet de simuler le comportement réel du système dans des conditions proches de l'implémentation finale.

2.4. Comparaison et validation des performances

La validation de notre approche de modélisation nécessite une comparaison rigoureuse entre les résultats obtenus sur Simulink et PSIM. Cette confrontation permet non seulement de vérifier la cohérence de nos modèles, mais également de confirmer que les performances visées sont effectivement atteintes dans les deux environnements de simulation.

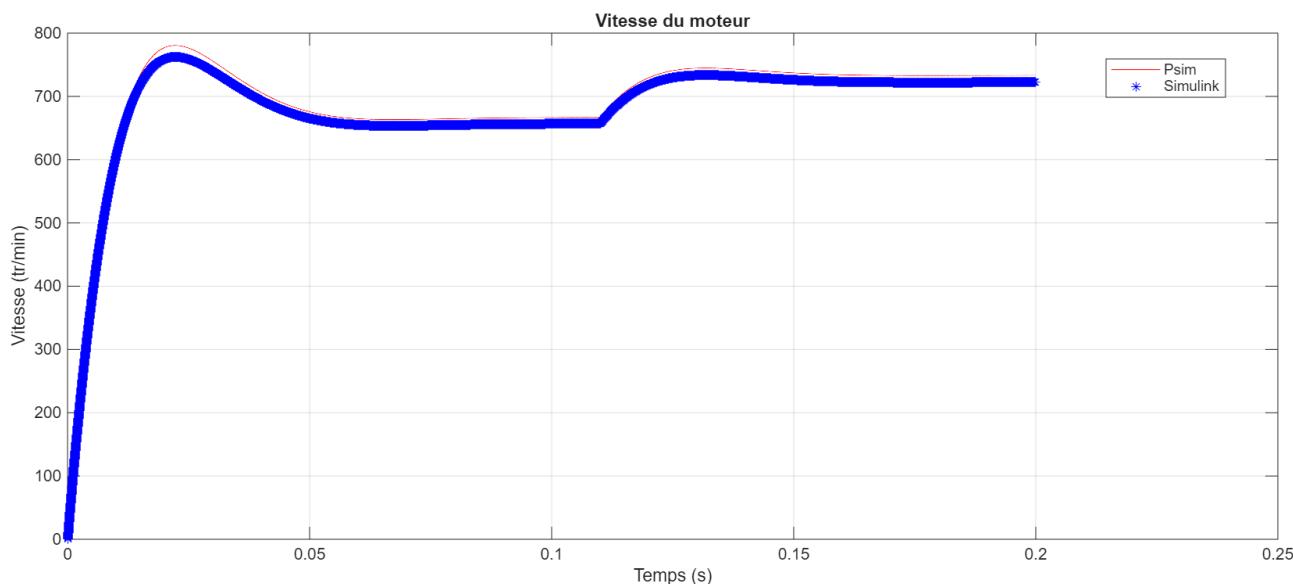


FIGURE 4.8 – Comparaison des réponses en vitesse entre Simulink et PSIM

La figure 4.8 illustre une superposition quasi-parfaite des réponses temporelles obtenues dans les deux environnements. Cette concordance remarquable valide la cohérence de nos modélisations et confirme que les paramètres du correcteur PI ont été correctement adaptés entre Simulink et PSIM. Pour faciliter la visualisation des deux courbes, un facteur de 0.99 a été appliqué aux valeurs de vitesse issues de Simulink.

L'analyse des performances révèle que l'asservissement en vitesse respecte pleinement les spécifications du cahier des charges :

- Le temps de réponse à 5 % est d'environ 41.37 ms, soit exactement trois fois plus rapide que la réponse en boucle ouverte (124.1 ms)
- Le dépassement observé reste inférieur à 20 %, ce qui limite les contraintes mécaniques sur le système d' entraînement
- Le régime permanent est atteint sans erreur statique grâce à l'action intégrale du correcteur PI

Ces résultats démontrent l'efficacité de la stratégie de régulation en cascade mise en œuvre : la boucle rapide d'asservissement en courant assure une dynamique élevée et un contrôle précis du couple, tandis que la boucle externe d'asservissement en vitesse garantit le suivi de consigne et le rejet des perturbations mécaniques.

2.5. Asservissement en vitesse avec le codeur incrémental

Le cahier des charges impose la possibilité d'asservir la vitesse du moteur à l'aide d'un codeur incrémental.

Principe de fonctionnement du codeur incrémental

Le fonctionnement du codeur incrémental repose sur un disque rotatif équipé de fentes. Celui-ci est interposé entre une diode électroluminescente et un capteur photodiode. Lorsque le disque tourne, les fentes permettent à la lumière de passer à travers, générant ainsi des impulsions électriques dans le capteur. Le nombre d'impulsions générées par tour dépend de la résolution du codeur. Le codeur incrémental fournit est composé de deux pistes de fentes décalées d'un quart de période, ce qui permet de déterminer le sens de rotation du moteur en analysant la séquence des impulsions générées par les deux capteurs.

Calcul de la vitesse et du sens de rotation

Pour mesurer la vitesse de rotation du moteur, il suffit de compter le nombre d'impulsions générées par le codeur sur une période de temps donnée. La vitesse angulaire peut être calculée en utilisant la formule suivante :

$$\omega = \frac{N \cdot 2\pi}{P \cdot T} \quad (4.1)$$

où :

- ω est la vitesse angulaire en radians par seconde (rad/s)
- N est le nombre d'impulsions comptées
- P est le nombre d'impulsions par tour du codeur
- T est la période de temps pendant laquelle les impulsions sont comptées en secondes (s)
- 2π est une constante pour convertir les tours en radians

Pour connaître le sens de rotation, on analyse la séquence des impulsions des deux pistes. Si la piste A précède la piste B, le moteur tourne dans un sens. Si la piste B précède la piste A, le moteur tourne dans le sens inverse.

Rotation positive (A précède B)



Rotation négative (B précède A)



FIGURE 4.9 – Chronogrammes selon le sens de rotation avec le codeur incrémental

5 - Dimensionnement des asservissemens

1. Dimensionnement des correcteurs réels pour l'asservissement en courant

Après avoir validé la structure de contrôle avec des régulateurs PI idéaux en simulation, il est nécessaire de dimensionner les circuits électroniques réels permettant d'implémenter ces correcteurs. Cette étape consiste à traduire les fonctions de transfert théoriques en circuits à amplificateurs opérationnels (AOP).

1.1. Structure du correcteur PI

La structure du correcteur PI à implémenter est représentée à la figure 5.1.

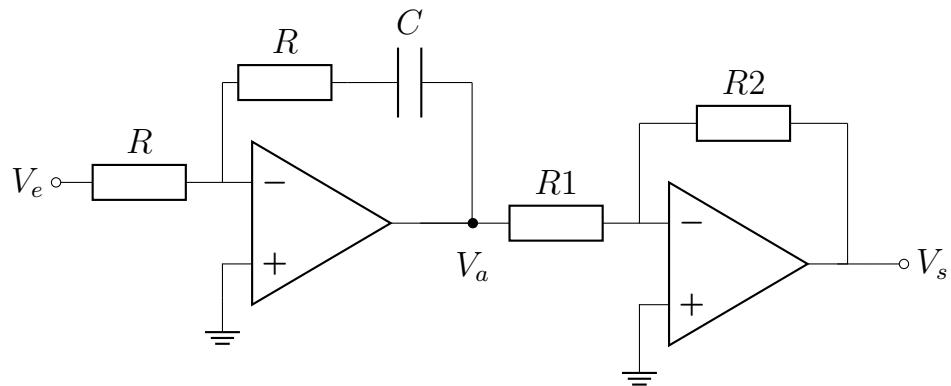


FIGURE 5.1 – Schéma d'un correcteur PI avec des amplificateurs opérationnels

1.2. Calcul de la fonction de transfert globale

La fonction de transfert globale du système s'exprime comme suit :

$$\frac{V_e(s)}{V_r(s)} = \frac{V_e(s)}{V_a(s)} \cdot \frac{V_a(s)}{V_r(s)} = H_1(s) \cdot H_2(s) \quad (5.1)$$

Dans cette configuration, on suppose que les amplificateurs opérationnels sont idéaux, ce qui implique $\varepsilon = 0 \implies V^+ = V^-$. Le dimensionnement des correcteurs s'effectue en séparant les deux fonctions de transfert.

1.3. Dimensionnement du correcteur $H_1(s)$

Pour le premier étage du correcteur, on introduit une résistance équivalente :

$$R_{eq} = R + \frac{1}{sC} \quad (5.2)$$

En appliquant le théorème de Millman au point A, on obtient :

$$V^- = \frac{V_e \cdot R1 + V_a \cdot Z_{eq}}{R1 \cdot Z_{eq}} \quad (5.3)$$

Puisque $V_a \cdot R = -V_e \cdot (R + \frac{1}{sC}) = -V_e \cdot \frac{s[RC]+1}{sC}$, la fonction de transfert du premier étage s'écrit :

$$\frac{V_a(s)}{V_e(s)} = \frac{RC + 1}{sC} \quad (5.4)$$



1.4. Dimensionnement du correcteur $H_2(s)$

Pour le second étage, on applique le théorème de Millman au point B :

$$V^- = \frac{\frac{1}{R1} + \frac{1}{R2}}{\frac{1}{R1} + \frac{1}{R2}} \Rightarrow \frac{V_r \cdot R1 + V_a \cdot R2}{R1 + R2} \quad (5.5)$$

La fonction de transfert du second étage s'exprime alors par :

$$H_2(s) = \frac{V_r}{V_a} = -\frac{R2}{R1} \quad (5.6)$$

1.5. Identification des paramètres

Par identification avec la structure PI souhaitée, on obtient les relations suivantes :

$$K_{posm} = -\frac{R2}{R1} \quad (5.7)$$

$$\tau = RC \quad (5.8)$$

1.6. Choix des composants

Les valeurs des composants retenues pour l'implémentation physique sont les suivantes :

Composant	Valeur
$R = R_a = R1$	10 k Ω
$R2$	82 k Ω
C	58 nF

Ces valeurs permettent d'obtenir les caractéristiques dynamiques du correcteur PI avec les performances souhaitées en termes de temps de réponse et de stabilité du système asservi.



Liste des figures et chemins d'accès

Cette section recense l'ensemble des figures présentes dans ce rapport ainsi que le chemin d'accès aux fichiers de simulation correspondants.

Figure 3.1 : Modelisation/01_12_09_2025/moteur_seul_12_09_2025.slx

Figure 3.2 : Modelisation/01_12_09_2025/moteur_seul_12_09_2025.slx

Figure 3.3 : Modelisation/01_12_09_2025/moteur_seul_12_09_2025.psimsch

Figure 3.4 : Modelisation/01_12_09_2025/comparasion_moteur_seul_12_09_2025.m

Note : Cette liste permet de retrouver rapidement les fichiers sources utilisés pour générer les figures du rapport.



Conclusion