

SPÉCIALITÉ GÉNIE ELECTRIQUE, QUATRIÈME ANNÉE - GE4
ANNÉE 2025-2026, SEMESTRE 7

STM-GE-08-PROJET GES7

Rapport final

Asservissement de machine à courant continu

ARCHIS - IDMONT - RAMDANI - SCHMITT STATUCKI

Table des matières

Introduction	4
1 Présentation des machines à courant continu utilisées	4
1.i Fonctionnement général d'une machine à courant continu	4
1.ii Caractéristiques des machines	5
1.iii Équations de fonctionnement	5
2 Structure du système complet	6
3 Cahier des charges	7
4 Temporalité du projet	7
5 Modélisation et simulation du système en boucle ouverte	8
5.i Modèle du moteur seul sans frottements ni charge	8
5.i.a Sous Matlab/Simulink	8
5.i.b Sous PSIM	9
5.i.c Comparaison des deux modèles	11
5.ii Prise en compte des frottements et de la charge	12
5.ii.a Notions théoriques	12
5.ii.b Implémentation sous PSIM	13
5.ii.c Implémentation sous Matlab/Simulink	13
5.ii.d Comparaison des deux modèles	14
5.iii Etude du régime permanent	16
5.iv Modélisation du hacheur	17
5.iv.a Notions théoriques	17
5.iv.b Implémentation sous PSIM	18
5.iv.c Implémentation sous Matlab/Simulink	18
5.v Prise en compte de la commande	18
5.v.a Notions théoriques	18
5.v.b Implémentation sous PSIM	19
5.v.c Implémentation sous Matlab/Simulink	19
5.vi Etude du régime permanent considérant la commande	20
5.vi.a Sous Matlab/Simulink	20
5.vi.b Sous PSIM	20
6 Dimensionnement et simulation des asservissements	22
6.i Préliminaire à l'asservissement	22
6.ii Asservissement en courant	23
6.ii.a Calcul du correcteur	23
6.ii.b Vérification du correcteur	26
6.iii Asservissement de vitesse	27
6.iii.a Démarche globale	27
6.iii.a.1 Calcul du correcteur	27
6.iii.a.2 Vérification du correcteur	31
6.iii.b Prise en compte des capteurs	32
6.iv Lien entre valeur de référence et valeur moyenne	34
7 Conception et dimensionnement des circuits électroniques	35
7.i Asservissements	35
7.i.a Correcteur de vitesse	35
7.i.b Correcteur de courant	35
7.i.c Comparateur	35
7.ii Circuit de mise en forme du signal tachymétrique	35
7.iii Circuit de mise en forme du signal du codeur incrémental	35
7.iv Limiteur de courant	35
7.v Limiteur de la tension de commande	35
7.vi Indicateur de sens de rotation	35

Liste des tableaux

1	Caractéristiques des MCC	5
---	------------------------------------	---

Table des figures

1	Schéma équivalent d'un moteur à courant continu	5
2	Structure globale du système	6
3	Diagramme de Gant prédictif	7
4	Fichier Matlab de déclaration des constantes : Moteur seul sans frottements ni charge *	8
5	Schéma bloc Simulink du moteur seul sans frottements ni charge *	9
6	Essai pour la détermination de la MCC équivalente sur PSIM *	9
7	Moteur seul sans frottements ni charge sur PSIM *	10
8	Comparaison Matlab/PSIM moteur seul : courant d'induit *	11
9	Comparaison Matlab/PSIM moteur seul : vitesse *	11
10	Schéma équivalent du système avec une charge	12
11	Montage du système muni des éléments de frottement et de la charge simulés sur PSIM *	13
12	Montage du système muni des éléments de frottement et de la charge simulé sur Matlab *	13
13	Comparaison des courants induit simulés sur Matlab en bleu et sur PSIM en rouge *	14
14	Zoom sur le pic de courant *	14
15	Comparaison des vitesses simulés sur Matlab en bleu et sur PSIM en rouge *	15
16	Zoom sur le régime permanent *	15
17	Schéma de principe d'un hacheur en pont	17
18	Signaux de commande MLI du hacheur	17
19	Implémentation du hacheur sur PSIM *	18
20	Signal d'entrée de la MCC	18
21	Modélisation de la commande sous PSIM *	19
22	Modélisation de la commande sous Matlab *	19
23	Reproduction de la formule $\langle U_{AB} \rangle$ sous PSIM *	20
24	Courant d'induit moteur avec hacheur (vert) et courant d'induit moteur avec modélisation du hacheur (rouge) *	21
25	Vitesse moteur avec hacheur (vert) et vitesse moteur avec modélisation du hacheur (rouge) *	21
26	Zoom sur l'ondulation de vitesse *	22
27	Schéma-bloc du système global	23
28	Schéma-bloc simplifié du système global	23
29	Schéma-bloc Simulink *	24
30	Comparaison du courant d'induit issu du schéma global et du modèle *	24
31	Boucle fermée de courant *	25
32	Réponse asservie du courant d'induit *	25
33	Schéma de vérification de l'asservissement de courant sur PSIM *	26
34	Courbes de vérification de l'asservissement de courant sur PSIM *	26
35	Fonction du huitième ordre *	27
36	Comparaison des 3 modèles *	28
37	Schéma-bloc de l'asservissement de vitesse final	29
38	Asservissement simplifié de la vitesse	30
39	Graphique de la vitesse asservie *	30
40	Vérification de l'asservissement de vitesse : Schéma PSIM *	31
41	Courbes de vérification de l'asservissement de vitesse sur PSIM *	31
42	Modifications prenant en compte les capteurs *	32
43	Schéma-bloc de l'asservissement avec capteurs *	33
44	Vérification de l'asservissement avec capteur (vitesse) *	33
45	Vérification de l'asservissement avec capteur (courant) *	34
46	Signaux liés au codeur incrémental	34

Introduction

Le projet GE-S7 mêle les domaines de l'automatique et de l'électrotechnique dans le contexte de l'asservissement de machines à courant continu. En particulier, nous réaliserons un asservissement de courant et de vitesse en accord avec un cahier des charges établi au préalable. Ce projet va se dérouler en plusieurs étapes majeures que nous pouvons résumer de la manière suivante :

- Étude du système à asservir
- Modélisation et simulation du système en boucle ouverte
- Dimensionnement et simulation des asservissements
- Conception et réalisation du circuit imprimé
- Tests réels et comparaison avec les simulations

Le projet a pour objectifs de réaliser l'asservissement d'une machine et d'en comprendre les enjeux à travers les différentes étapes décrites ci-dessus. Il nous permet de nous mettre dans un cadre concret pour développer des compétences qui seront importantes dans notre futur métier d'ingénieur et en particulier, la démarche de projet et le travail en équipe.

Nous serons amenés dans ce rapport à fournir de nombreuses captures d'écran. Pour éviter d'alourdir le document et faciliter la lecture, nous indiquerons l'origine de chaque capture d'écran dans la [section 7.vi](#) où le fichier d'origine de chaque figure sera indiquée. Il sera possible de s'y rendre en cliquant sur l'astérisque "" présente à la fin de chaque titre de figure, et de revenir à la figure en cliquant sur le numéro de cette dernière dans l'arborescence.*

1 Présentation des machines à courant continu utilisées

1.i Fonctionnement général d'une machine à courant continu

Une machine à courant continu est composée comme toute machine électrique tournante d'une partie fixe : le stator, et d'une partie rotative : le rotor. Le stator (*ou inducteur*) consiste soit en un aimant permanent, soit en un électroaimant. Le rotor (*ou induit*), lui, est bobiné et va être traversé par un courant continu.

L'interaction entre le courant au rotor et le champ magnétique du stator va générer une force sur les conducteurs rotoriques permettant la rotation de la machine. De plus, pour permettre à la machine de tourner et de ne pas se bloquer du fait de la polarité du champ magnétique, une pièce appelée "collecteur" permet d'inverser la polarité du courant traversant les enroulements lors de la rotation, permettant ainsi un mouvement continu de la machine.

De ce fait, si nous alimentons l'induit de notre MCC avec un courant continu (et éventuellement l'inducteur si elle n'est pas à aimant permanent), nous fonctionnerons en mode moteur, et à l'inverse, en appliquant un mouvement de rotation sur l'arbre de notre MCC, nous fonctionnerons en génératrice.

Dans ce rapport, nous abrègerons souvent "machine à courant continu" par MCC.

1.ii Caractéristiques des machines

Nous utiliserons donc dans ce projet deux machines **identiques**. Une fonctionnera en moteur, l'autre en génératrice. Le type de machine utilisé est à **aimant permanent**, il ne sera donc pas nécessaire d'alimenter l'inducteur. Les machines sont à deux paires de pôles et leurs caractéristiques sont listées dans le [tableau 1](#).

TABLE 1 – Caractéristiques des MCC

Caractéristique	Valeur	Unité	Symbole
Couple en rotation lente	0,54	$N.m$	M_O
Courant nominal	4,5	A	I
Tension nominale d'alimentation	49	V	U
Vitesse nominale	3000	tr/min	N
Tension maximal	65	V	U_{max}
Vitesse maximale	4800	tr/min	N_{max}
Courant impulsionnel	13	A	I_{max}
Fem par 1000 tr/min à 25°	13,3	$\frac{V}{10^3 \cdot tr/min}$	$K_{e(tr/min)}$
Coefficient de couple électromagnétique	0,127	$N.m/A$	K_ϕ
Couple de frottement sec	2,4	$N.cm$	T_f
Coefficient de viscosité par 1000 tr/min	0,53	$N.cm$	K_d
Résistance de bobinage à 25°	1,52	Ω	R
Inductance de bobinage	2,2	mH	L
Inertie du rotor	0,000083	$kg.m^2$	J
Constante de temps thermique	7	min	T_{th}
Masse	1,34	kg	M

1.iii Équations de fonctionnement

Pour les besoins du projet, nous aurons besoin des équations de fonctionnement de la MCC en mode moteur. À partir du schéma équivalent d'un moteur à courant continu illustré à la [figure 1](#) et des équations de la mécanique, nous obtenons les équations de fonctionnement.

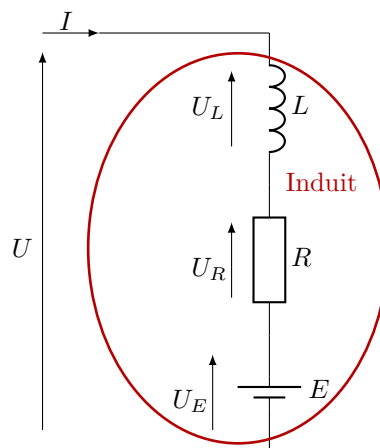


FIGURE 1 – Schéma équivalent d'un moteur à courant continu

Équations de fonctionnement :

$$\begin{aligned}
 & \bullet \quad u(t) = e(t) + Ri(t) + L \frac{di(t)}{dt} \\
 & \bullet \quad J \frac{d\omega}{dt} = C_m(t) - C_r(t) \\
 & \bullet \quad e(t) = K_e \omega(t) \\
 & \bullet \quad C_m = K_c i(t)
 \end{aligned} \tag{1}$$

Avec :

- J : l'inertie du rotor
- C_m : le couple magnétique
- C_r : le couple résistant ; comportant le couple de frottement et le couple de charge.
- K_e : la constante de force électromotrice
- K_c : la constante de couple
- ω : la vitesse de rotation du moteur
- R : la résistance d'induit (ou de bobinage)
- L : l'inductance d'induit (ou de bobinage)
- u : la tension aux bornes de l'induit
- e : la force électromotrice
- i : le courant d'induit

À noter qu'en fonctionnement moteur, nous sommes en convention récepteur. Pour un fonctionnement en génératrice, on passe en convention générateur et donc le sens du courant est inversé. Ainsi, on garde les mêmes équations à des signes près. De même, le signe des couples changera lui aussi. Il est ainsi aisé de déduire les équations en mode génératrice au besoin.

2 Structure du système complet

La structure globale du système est présentée en [figure 2](#) :

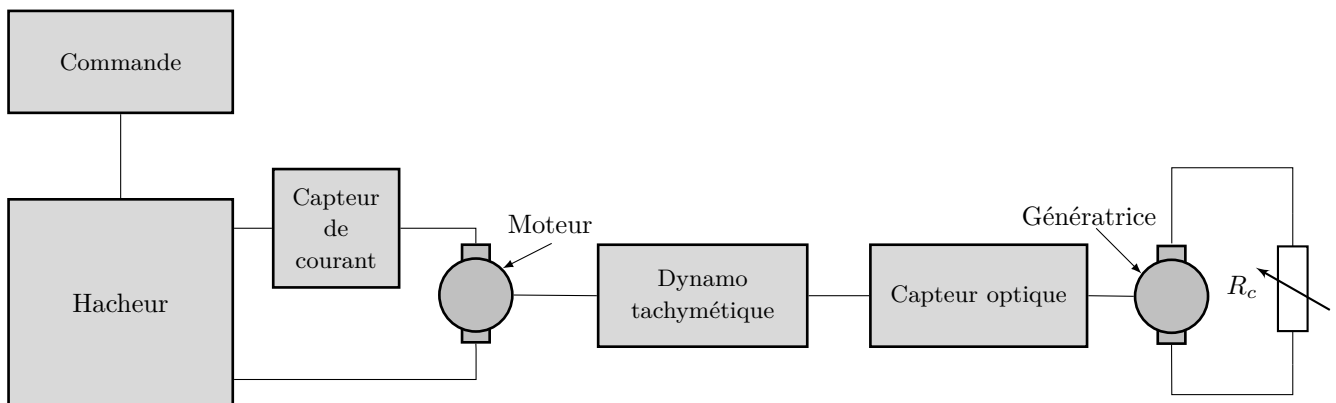


FIGURE 2 – Structure globale du système

Nous avons donc un hacheur en pont commandable alimenté en 48V-DC qui alimente notre moteur. À l'intérieur du hacheur se trouve un capteur de courant permettant de mesurer le courant en sortie de ce dernier. Le moteur et la génératrice se trouvent sur un même banc et sont mécaniquement couplés. On retrouve aussi sur ce banc une dynamo tachymétrique et un capteur optique permettant la mesure de la vitesse de rotation de l'arbre des deux machines. Le choix du capteur sera possible via un commutateur électrique. La génératrice alimentera un rhéostat R_c . Le moteur aura donc pour charge la génératrice alimentant le rhéostat.

Notre objectif sera d'ajouter à ce système une carte d'asservissement permettant de satisfaire les demandes du cahier des charges. Cette carte utilisera les données des différents capteurs à notre disposition et viendra agir sur les variables du système.

3 Cahier des charges

Nous avons pour ce projet un cahier des charges à respecter. Le cœur de ce dernier est comme nous l'avons dit l'asservissement en vitesse et en courant de notre MCC, c'est donc sur ces deux aspects que va se focaliser le cahier des charges.

Les contraintes que nous avons sont les suivantes :

- Le courant maximal ne devra pas être supérieur à 1,1 fois le courant nominal.
- La vitesse ne devra pas dépasser la vitesse maximale imposée par le constructeur.
- Les machines devront tourner à la vitesse de référence et le courant devra s'adapter en conséquence.

Ainsi, pour respecter les contraintes précédentes et avoir un système performant nous tâcherons de respecter le cahier des charges suivant :

- Les erreurs en régime permanent des deux grandeurs devront être nulles.
- Les dépassements en vitesse et en courant devront se situer entre 10% et 20% au maximum.
- Le temps de réponse de l'asservissement en courant devra être équivalent à dix fois la période du signal MLI du hacheur.
- Le temps de réponse de l'asservissement en vitesse doit être trois fois plus rapide que le temps de réponse du système en boucle ouverte.

Pour résumer, le cahier des charges impose une vitesse suivant une valeur de référence et un courant qui s'y adapte, cela avec des valeurs de dépassement comprises entre 10% et 20% au maximum et un temps de réponse fixé en fonction de la structure du système.

4 Temporalité du projet

Dans un soucis d'organisation, il nous a été suggéré de créer un diagramme de Gant. Cette représentation sert à organiser la temporalité de chaque activité en lien avec le projet. Pour chaque élément, des quantifications en semaines sont inscrites pour identifier le début et la durée totale de chacune d'entre elles. Voici donc notre diagramme de Gant prédictif :

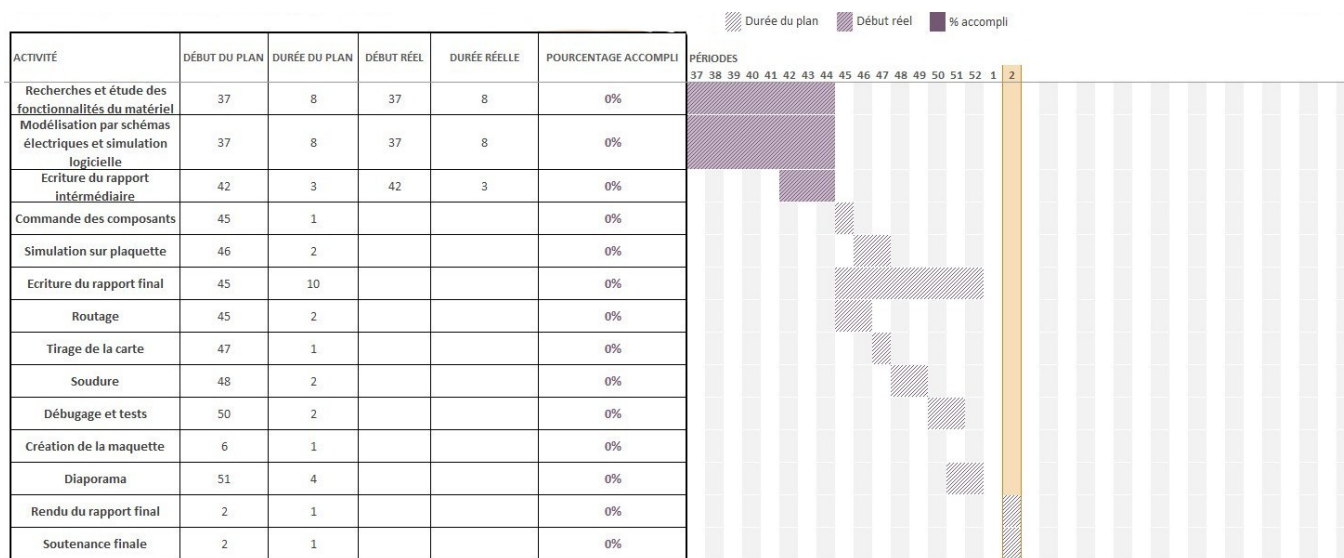


FIGURE 3 – Diagramme de Gant prédictif

5 Modélisation et simulation du système en boucle ouverte

Nous allons donc commencer par la modélisation et la simulation de notre système en boucle ouverte afin de pouvoir par la suite dimensionner nos correcteurs sur un modèle étant le plus fidèle possible par rapport à la réalité.

Dans tout ce rapport on considère que la puissance électromagnétique est la même que cela soit d'un point de vue électrique ou mécanique, ainsi on a dans l'équation 1 : $K_c = K_e = K_\phi$.

5.i Modèle du moteur seul sans frottements ni charge

Dans un premier temps, nous ne prendrons pas en compte les frottements et le moteur fonctionnera seul, sans charge. Ainsi, on aura $C_r = 0$ (cf. équation 1).

5.i.a Sous Matlab/Simulink

Nous voulons étudier le courant et la vitesse de notre système, ainsi, nous allons faire en sorte que dans notre modèle nous puissions accéder à ces grandeurs pour les analyser.

A partir des équations (1), l'induit de la MCC utilisé en moteur peut être modélisé par l'équation suivante :

$$u(t) = R \cdot i_M(t) + L \cdot \frac{di_M(t)}{dt} + K_\phi \omega(t)$$

En utilisant la transformée de Laplace, on a :

$$U - K_\phi \Omega = I_M(R + Lp)$$

On peut en déduire la fonction de transfert donnant accès au courant suivante :

$$H_1(p) = \frac{I}{U - K_\phi \Omega} = \frac{1}{R + L \cdot p}$$

De plus, toujours grâce aux équations (1) et sachant que $C_r = 0$, on a :

$$J \cdot \frac{d\Omega}{dt} = K_\phi i_M$$

De la même façon, on obtient la fonction de transfert donnant accès à la vitesse suivante :

$$H_2(s) = \frac{\Omega}{I_M} = \frac{K_\phi}{J \cdot p}$$

Les deux fonctions de transfert nous permettent bien d'obtenir le courant de l'induit moteur ainsi que la vitesse du moteur dans une même simulation.

Dans Matlab, on utilise l'outil Simulink pour modéliser notre système puis on implémente les deux fonctions de transfert. On identifie le courant moteur et la vitesse afin d'utiliser les valeurs de leurs courbes pour s'assurer de l'équivalence avec la simulation PSIM que nous effectuerons plus tard. Le schéma est montré en figure 5.

Tout au long du projet, nous utiliserons un fichier Matlab qui nous permettra de définir et stocker toutes nos constantes. Voici ce dernier en figure 4 incluant les constantes nécessaires à la simulation de cette partie. Nous le compléterons au fil des simulations avec les grandeurs nécessaires mais sa structure restera la même :

```
% données pour 1 moteur
R=1.52;           % résistance de l'induit
Rch=10;           % Résistance de charge
C0=2.2e-2;        % Couple de frottement sec pour une machine
f=0.53e-2/1000*(30/pi); % coefficient de frottement visqueux pour une machine
L=2.2e-3;         % inductance propre de l'induit
J=8.3e-5;         % moment d'inertie pour 1 machine
U=48;            % tension d'alimentation
KPhi=0.127;       % constante de couple, de fem
UM=15;
u_comande=11.25;
```

FIGURE 4 – Fichier Matlab de déclaration des constantes : Moteur seul sans frottements ni charge *

On réalise ainsi le schéma bloc suivant :

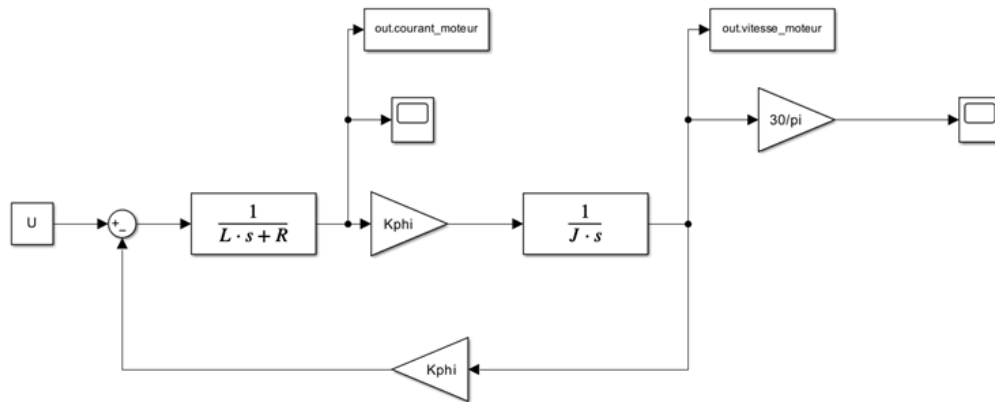


FIGURE 5 – Schéma bloc Simulink du moteur seul sans frottements ni charge *

Les blocs "out." permettent d'acquérir et de rendre accessible les valeurs du courant d'induit et de la vitesse de rotation du moteur dans les scripts Matlab. Le facteur $\frac{30}{\pi}$ permet de fournir une réponse en tr/min et non en rad/s dans l'oscilloscope.

5.i.b Sous PSIM

Parallèlement à cela, nous réalisons une simulation PSIM de notre moteur. Il y a un point important à aborder avant de continuer. Notre moteur réel est une MCC à aimants permanents, mais sur le logiciel PSIM, nous ne disposons pas du modèle de ce moteur. Nous avons néanmoins accès à un modèle de machine à excitation bobinée, qui possède les mêmes caractéristiques générales (résistance d'induit, moment d'inertie, etc). Nous devons donc dans un premier temps faire des essais avec ce modèle afin de trouver un coefficient à ajouter à la tension d'inducteur permettant de réaliser une équivalence. En effet, une MCC à excitation bobinée est définie par ces caractéristiques internes mais aussi par sa tension d'excitation. C'est sur cette dernière qu'il faudra jouer pour obtenir l'équivalence.

Pour ce faire, nous commençons par nous placer en charge avec une seconde machine en génératrice, dont les caractéristiques seront les mêmes que pour notre MCC réelle et avec une tension d'excitation arbitraire mais fixée (on choisira 10 V). La disposition de cette essai est présentée en figure 6.

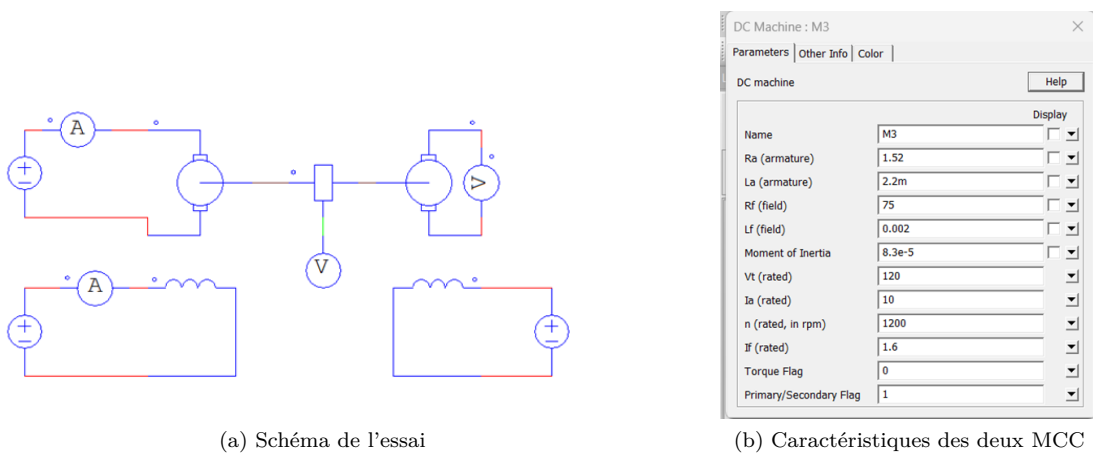


FIGURE 6 – Essai pour la détermination de la MCC équivalente sur PSIM *

L'idée est ici la suivante. Nos deux types de moteurs, à aimant permanent et à excitation bobinée, ont un coefficient de force électromotrice qui leur est propre. Nous allons donc chercher ce coefficient pour le moteur sur PSIM afin de pouvoir calculer le rapport des coefficients des deux machines, la réelle et celle sur PSIM avec la tension d'excitation précédemment mentionnée. Rappelons la formule du coefficient de force électromotrice :

$$K_{\phi} = \frac{E_{gen}}{\Omega}$$

Où E_{gen} est la tension induite au générateur et Ω la vitesse de rotation de l'ensemble.

On mesure donc la vitesse qui est donnée ici en tr/min , puis on la convertit en rad/s . Il suffit pour cela de la multiplier par $\frac{2\pi}{60}$. On obtient ainsi une vitesse angulaire de $690,67 \text{ rad/s}$. La force électromotrice E_{gen} est en régime permanent, la tension que l'on a imposé à l'induit, soit $48V$, on calcule alors K_{ϕ} :

$$K_{\phi} = \frac{48}{690,67} = 0,0694976577$$

On peut à présent faire le rapport des coefficients de nos deux modèles, on calcule :

$$\frac{0,127}{0,0694976577} = 1,827480936166667$$

Ainsi, comme la relation entre la tension d'excitation et le coefficient de force électromotrice est proportionnelle (c'est à dire qu'affecter à la tension d'excitation un certain coefficient revient à affecter ce même coefficient au coefficient force électromotrice tout en gardant la tension constante), nous devons multiplier la tension d'inducteur par le coefficient précédent. Dès lors, dans toutes nos simulations, nous devons ajouter ce coefficient à nos tensions d'inducteurs pour chaque machine. C'est bien à l'inducteur et non à l'induit que ce coefficient apparaît, car c'est l'inducteur qui diffère entre les deux machines.

Nous pouvons maintenant enfin passer à la simulation en tant que telle. Rappelons que nous cherchons tout d'abord à simuler le moteur à vide, plaçons alors un moteur sans aucune charge. Le schéma est présenté en figure 7.

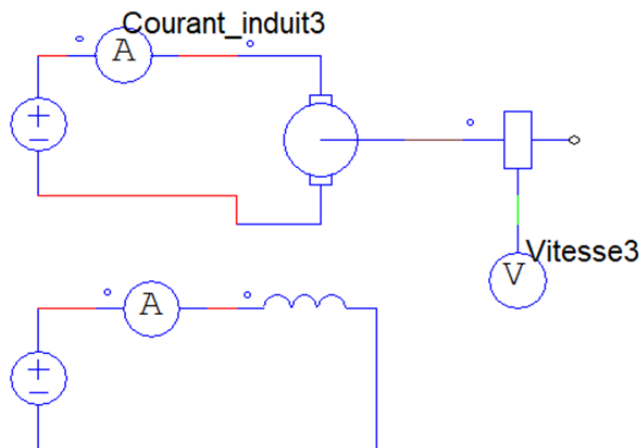


FIGURE 7 – Moteur seul sans frottements ni charge sur PSIM *

5.i.c Comparaison des deux modèles

Afin de nous assurer que notre modèle est le bon, superposons la courbe du courant d'induit du modèle Matlab avec celui du modèle PSIM. Pour ce faire, nous pouvons copier tous les points de la courbe PSIM et les reporter dans un fichier Matlab. Nous allons ensuite la tracer à l'aide de la fonction plot, et la superposer à celle que nous avons obtenu via Simulink. Nous obtenons le graphe suivant en [figure 8](#) :

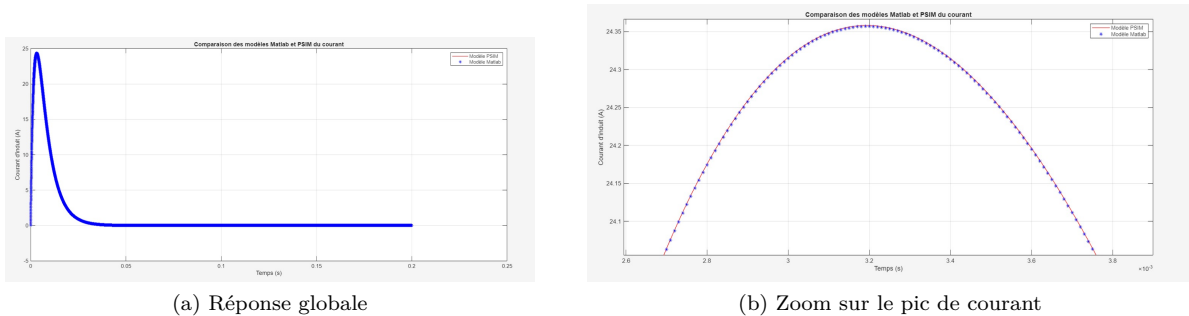


FIGURE 8 – Comparaison Matlab/PSIM moteur seul : courant d'induit *

On a en abscisse le temps en seconde et en ordonnée le courant en ampère.

Faisons de même pour la vitesse :

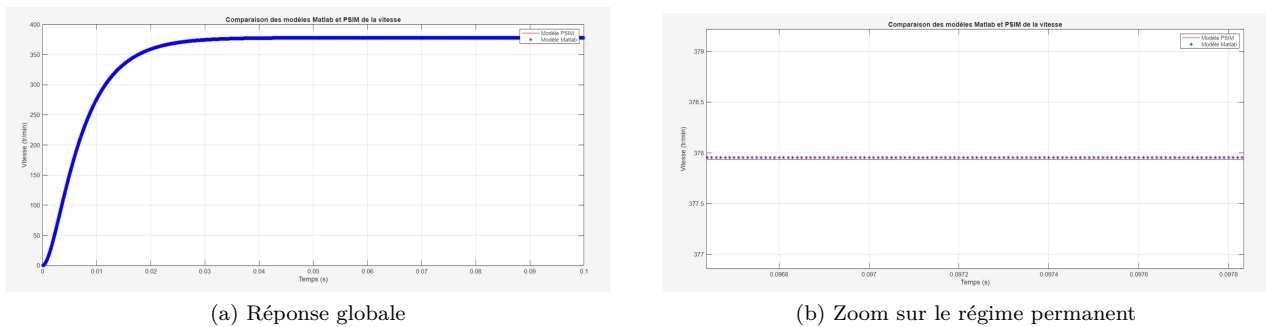


FIGURE 9 – Comparaison Matlab/PSIM moteur seul : vitesse *

On a en abscisse le temps en seconde et en ordonnée la vitesse en $10^1 \cdot \text{tr}/\text{min}$.

Nous pouvons observer une très bonne corrélation entre les deux simulations, la différence observée étant vraiment minime par rapport aux grandeurs en jeux. Ceci confirme donc que notre modèle est bon, car la simulation sur deux logiciels distincts nous donne le même résultat.

5.ii Prise en compte des frottements et de la charge

5.ii.a Notions théoriques

Prenons maintenant en compte la charge et les frottements . On établit les relations en fonction des frottements et de la charge :

- **Frottements :**

A partir des données de la machine, on connaît les valeurs de frottement sec et de coefficient de viscosité qui nous permettront de simuler les frottements. Les valeurs de frottement doivent être doublées car il faut compter les frottements des deux machines accouplées. On a donc l'équation suivante :

$$C_f = 2 \cdot C_0 + 2 \cdot f\omega$$

Avec C_f , le couple de frottement, C_0 , le couple de frottement sec (cf. " T_f ", [tableau 1](#)) et $f = \frac{K_d \cdot 30}{1000\pi}$ car le coefficient de viscosité est donné pour 1000 tr/min (cf. [tableau 1](#)).

- **Charges :**

En prenant en compte la génératrice qu'entraîne le moteur et la charge sur laquelle cette génératrice débite, on obtient le schéma suivant :

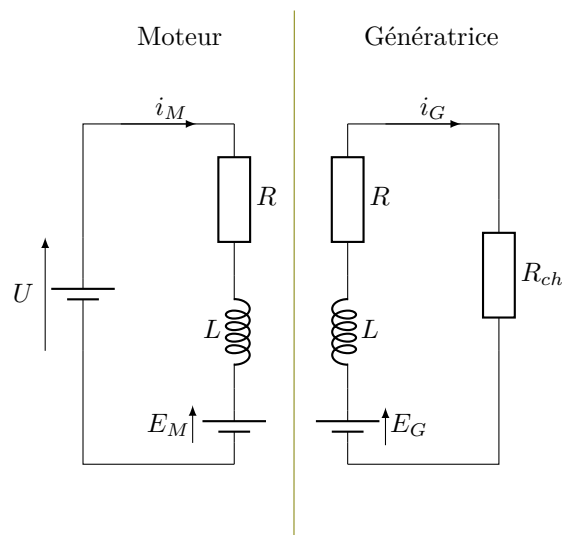


FIGURE 10 – Schéma équivalent du système avec une charge

À partir de la [figure 10](#), de ce qui a été montré pour les frottements, et des équations (1) adaptées à une MCC génératrice, on obtient les relations suivantes :

$$K_\phi \Omega = (R_{ch} + R)i_G + L \frac{di_G}{dt} \quad (2)$$

$$2J \cdot \frac{d\Omega}{dt} = K_\phi(i_M - i_G) - 2 \cdot C_0 - 2 \cdot \Omega f \quad (3)$$

L'équation 2 nous permet d'établir la fonction de transfert reliant le courant du générateur à la vitesse. L'équation 3 nous permet de réaliser la "boucle de rétroaction" des frottements et de la charge. Ces dernières nous seront utiles pour implémenter ces éléments à la simulation sur Matlab.

5.ii.b Implémentation sous PSIM

On ajoute d'abord les frottements sur PSIM. Le bloc "Mecanical load" (charge mécanique) nous permet de simuler les frottements entre le moteur et le générateur. On y rentre les valeurs de frottement sec et de coefficient de viscosité.

Pour la charge, on ajoute simplement une résistance choisie de manière arbitraire à 10Ω dans le circuit induit du générateur comme suit en [figure 11](#) :

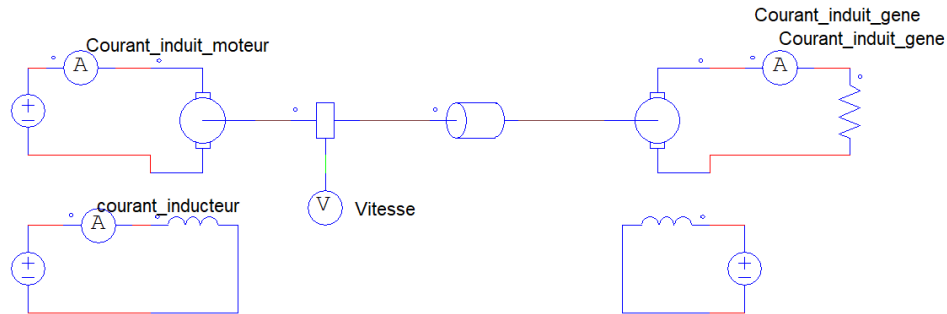


FIGURE 11 – Montage du système muni des éléments de frottement et de la charge simulés sur PSIM *

5.ii.c Implémentation sous Matlab/Simulink

On peut ensuite faire l'équivalent sur Matlab. A l'aide des équations précédemment établies ([2](#) et [3](#)), on ajoute les différents éléments au montage Simulink et leurs valeurs au fichier de données.

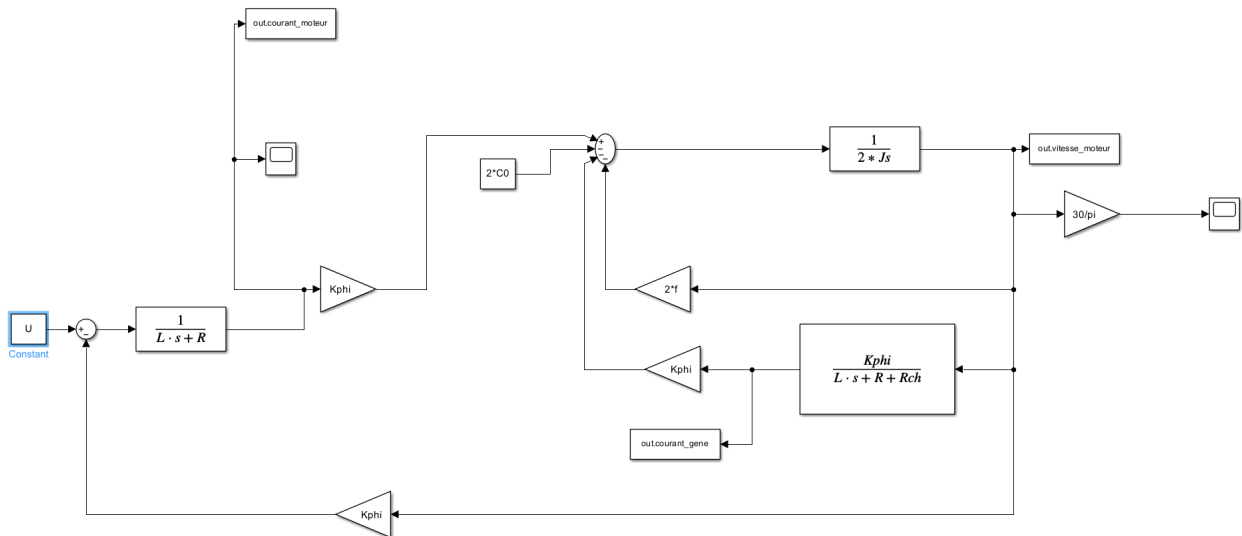


FIGURE 12 – Montage du système muni des éléments de frottement et de la charge simulé sur Matlab *

5.ii.d Comparaison des deux modèles

On relève les valeurs de simulation de PSIM pour les comparer à celles obtenues avec Matlab pour comparer les montages et s'assurer de la validité de nos simulations.

On applique la même méthode que dans la partie précédente.

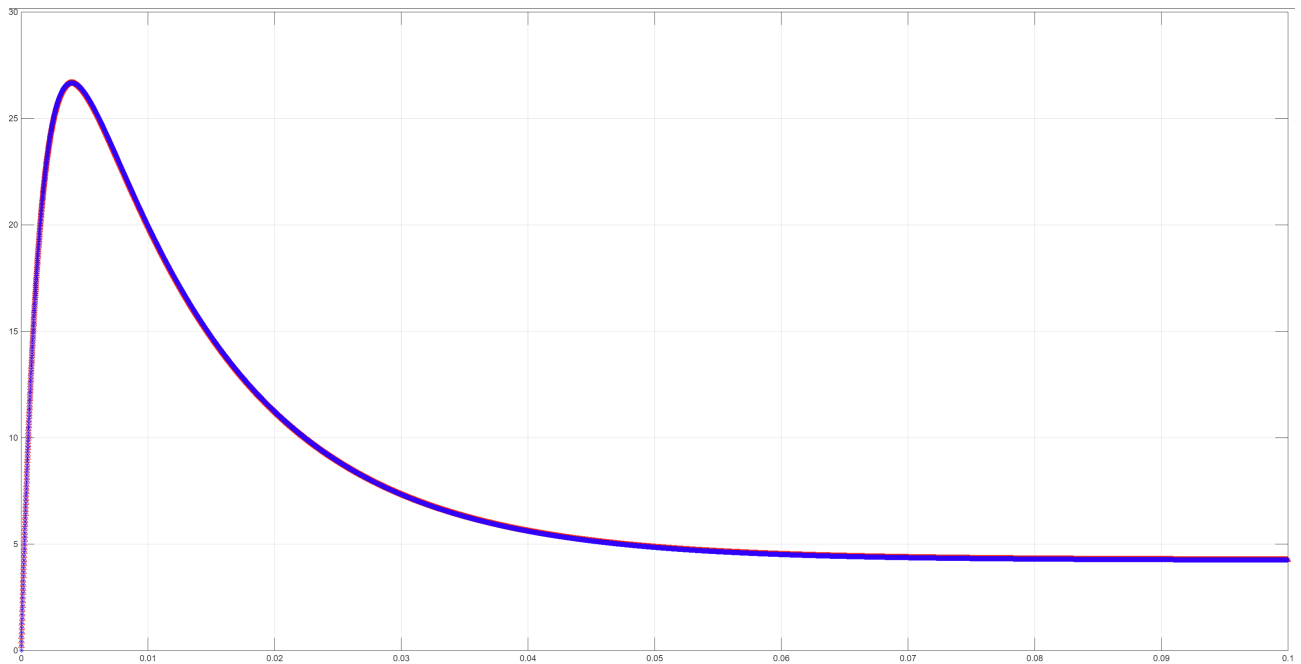


FIGURE 13 – Comparaison des courants induit simulés sur Matlab en bleu et sur PSIM en rouge *

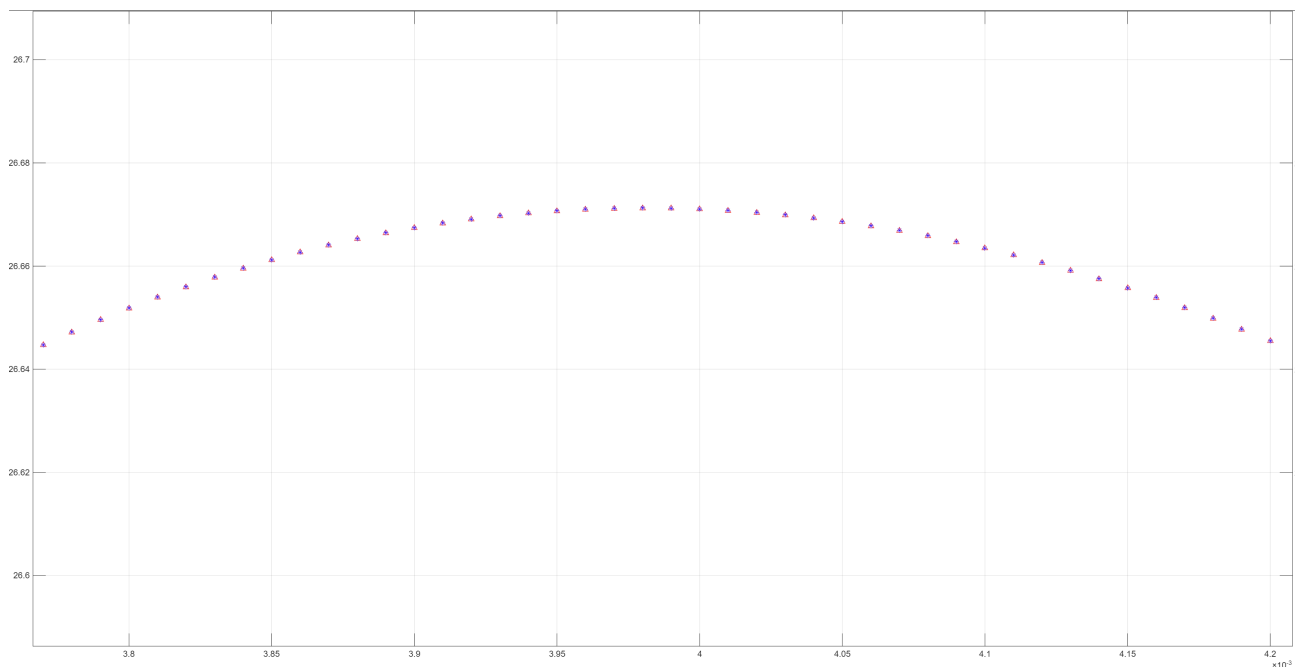


FIGURE 14 – Zoom sur le pic de courant *

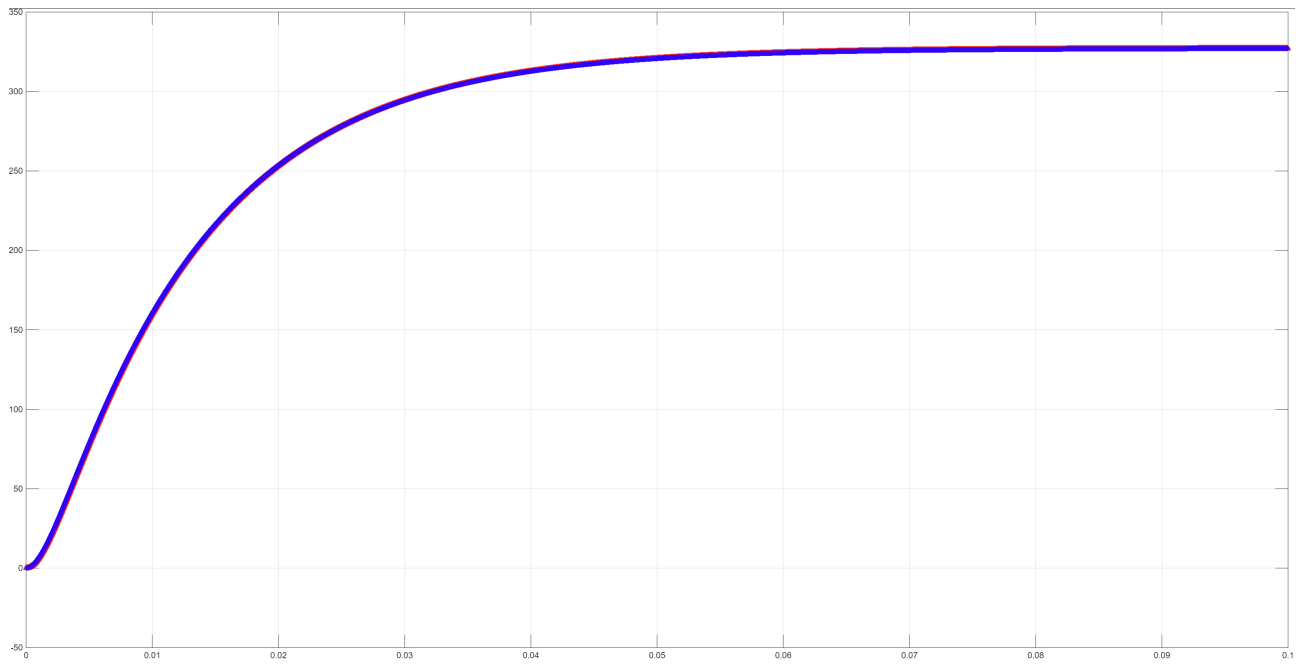


FIGURE 15 – Comparaison des vitesses simulés sur Matlab en bleu et sur PSIM en rouge *

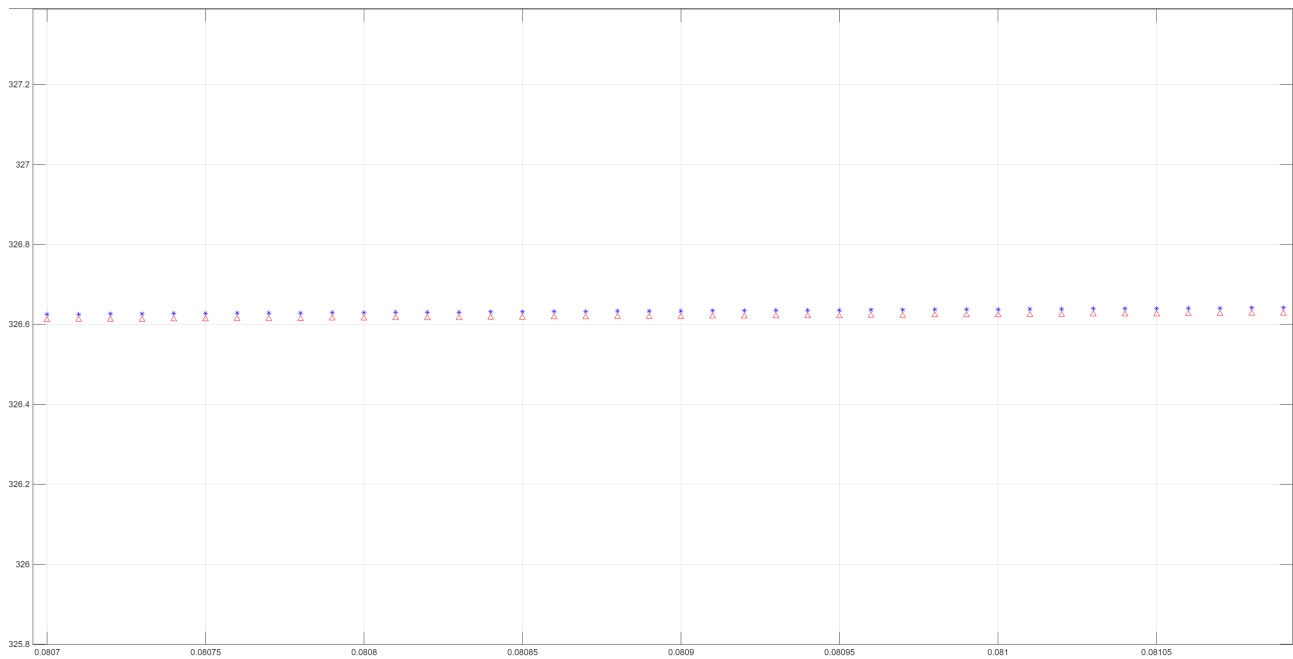


FIGURE 16 – Zoom sur le régime permanent *

Nous pouvons observer encore une fois une très bonne corrélation entre les deux simulations, la différence observée étant vraiment minime par rapport aux grandeurs en jeux. Ceci confirme donc que notre modèle est correct avec l'ajout de ces nouveaux éléments.

5.iii Etude du régime permanent

En amont de la réalisation de l'asservissement de notre système, nous allons réaliser l'étude du régime permanent de celui-ci. Pour cela, nous allons utiliser un outil de modélisation sous forme de systèmes d'états. On définit le vecteur d'état X composé de 3 variables d'état : le courant d'induit moteur i_m , le courant d'induit génératrice i_g et la vitesse du moteur Ω . Le vecteur peut ainsi se noter ainsi :

$$X = \begin{pmatrix} i_m \\ i_g \\ \Omega \end{pmatrix}$$

L'étude du régime permanent est donc équivalente à l'étude de l'évolution de ce vecteur lorsque $t \rightarrow \infty$. En pratique, le régime permanent est atteint lorsque t est largement plus élevé que les constantes de temps du système. A partir des équations de moteur chargé par la génératrice, on peut écrire les relations entre les différentes variables d'états :

$$\begin{cases} U = Ri_m + L \frac{di_m}{dt} + K_\phi \Omega \\ K_\phi \Omega = (R_{ch} + R)i_g + L \frac{di_g}{dt} \\ 2J \frac{d\Omega}{dt} = K_\phi(i_m - i_g) - 2C_0 - 2f\Omega \end{cases} \Leftrightarrow \begin{cases} \frac{di_m}{dt} = \frac{U}{L} - \frac{R}{L}i_m - \frac{K_\phi}{L}\Omega \\ \frac{di_g}{dt} = \frac{K_\phi}{L}\Omega - \frac{(R+R_{ch})}{L}i_g \\ \frac{d\Omega}{dt} = \frac{K_\phi}{2J}i_m - \frac{K_\phi}{2J}i_g - \frac{C_0}{J} - \frac{f}{J} \end{cases}$$

Soit sous forme matricielle :

$$\frac{dX}{dt} = \begin{pmatrix} -\frac{R}{L} & 0 & -\frac{K_\phi}{L} \\ 0 & -\frac{(R+R_{ch})}{L} & \frac{K_\phi}{L} \\ \frac{K_\phi}{2J} & -\frac{K_\phi}{2J} & -\frac{f}{J} \end{pmatrix} \begin{pmatrix} i_m \\ i_g \\ \Omega \end{pmatrix} + \begin{pmatrix} \frac{U}{L} \\ 0 \\ -\frac{C_0}{J} \end{pmatrix} = AX + BU$$

Les constantes de temps du système τ_i sont liées aux valeurs propres λ_i de la matrice A d'après la formule : $\tau_i = \frac{-1}{\lambda_i}$. Pour calculer ces dernières, on utilisera Matlab. On entre la matrice A sur le logiciel et on calcule les constantes de temps avec la commande : `cstesdetemps=-1./eig(A)`. On obtient alors les valeurs suivantes :

$$\begin{cases} \tau_1 = 0,0016 \text{ s} \\ \tau_2 = 0,0122 \text{ s} \\ \tau_3 = 0,0002 \text{ s} \end{cases}$$

Ainsi, dans nos simulations, si nous voulons observer avec certitude le régime permanent nous tacherons de prendre ce qui vient d'être énoncé en compte.

On cherche finalement à trouver les valeurs finales des 3 variables d'état, c'est-à-dire leur valeur en régime permanent. On note X_∞ le vecteur contenant ces valeurs finales. Cela revient à résoudre :

$$\frac{dX}{dt} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \Leftrightarrow MX_\infty + N = 0 \Leftrightarrow X_\infty = -(M)^{-1}N$$

En utilisant Matlab à nouveau, on obtient :

$$X_\infty = \begin{pmatrix} 4,2452 \text{ A} \\ 3,6065 \text{ A} \\ 327,144515 \text{ rad/s} \approx 3124 \text{ tr/min} \end{pmatrix}$$

5.iv Modélisation du hacheur

5.iv.a Notions théoriques

Jusqu'à présent, nous avons utilisé dans nos modèles une source de tension continue pour alimenter notre MCC. Hors, dans les faits, c'est bien un hacheur qui alimentera notre moteur. Ainsi, nous allons modéliser plus finement notre système en modélisant le comportement du hacheur. Nous effectuerons cela sur PSIM.

Nous savons que le hacheur utilisé est un hacheur en pont fonctionnant en mode bipolaire dont le schéma de principe est donné à la [figure 17](#).

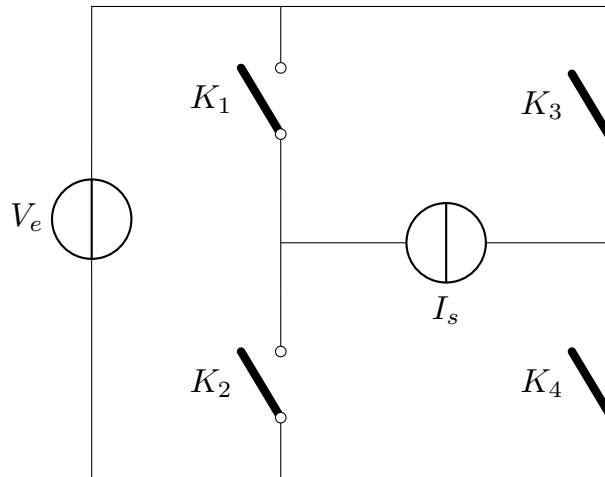


FIGURE 17 – Schéma de principe d'un hacheur en pont

V_e sera la tension en entrée de notre hacheur, c'est-à-dire 48V, la source de courant sera notre moteur et les interrupteurs seront des transistors MOFSET. Pour obtenir le comportement voulu, les transistors sont commandés par un signal MLI (modulation largeur d'impulsion) ayant la forme présentée en [figure 18](#) :

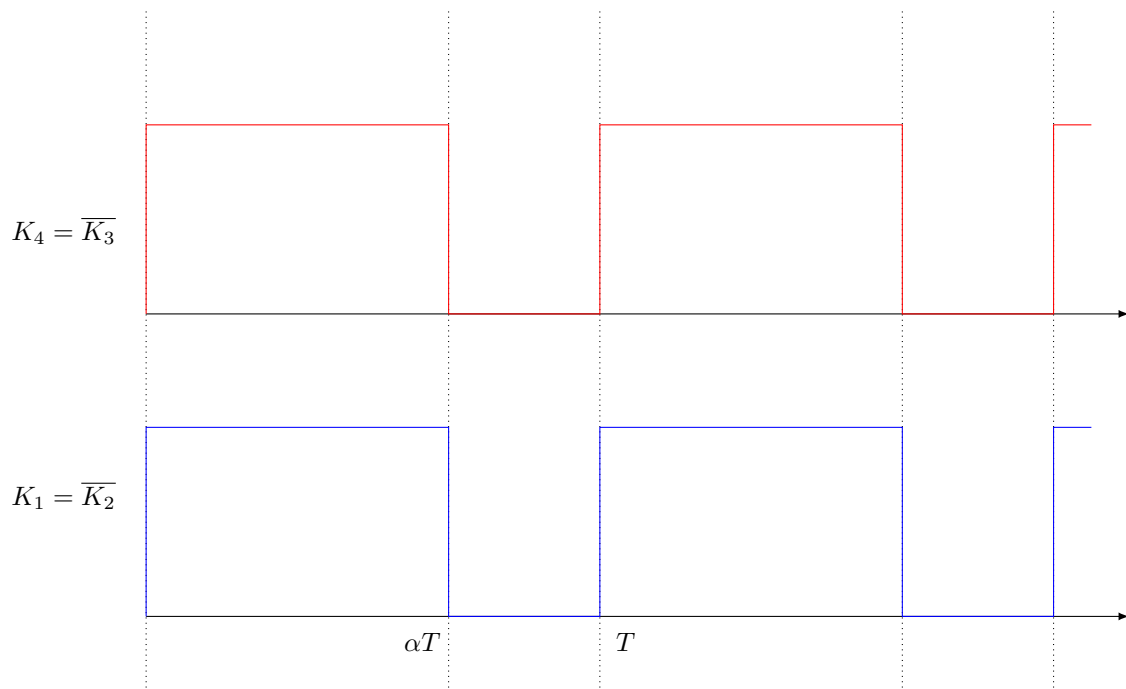


FIGURE 18 – Signaux de commande MLI du hacheur

5.iv.b Implémentation sous PSIM

On réalise donc le schéma du hacheur ayant la structure précédemment mentionnée et permettant d'obtenir les signaux de commande souhaités sur PSIM. Ce dernier est présenté en [figure 19](#) :

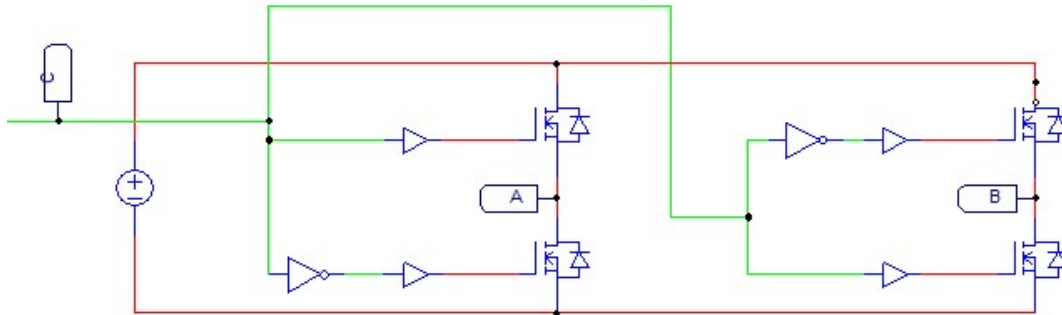


FIGURE 19 – Implémentation du hacheur sur PSIM *

Pour l'instant, le signal de commande des MOSFETs 'C' est un signal carré alternant entre 0 et 10 V avec un rapport cyclique de 75%. Nous prendrons en compte la commande du système dans la partie suivante.

5.iv.c Implémentation sous Matlab/Simulink

Pour la modélisation sur Matlab, nous allons simplement utiliser la valeur moyenne de la tension en sortie du hacheur, ce que nous avons déjà fait depuis le début des simulations.

5.v Prise en compte de la commande

5.v.a Notions théoriques

Comme évoqué précédemment, on a cherché à prendre en compte l'alimentation U_{AB} du moteur issue du hacheur. Pour sa modélisation sur Matlab, nous avons utilisé sa valeur moyenne. En pratique, comme nous l'avons vu, la tension issue du hacheur est un signal carré compris entre -48V et 48V et dont la valeur moyenne va ajuster la vitesse de rotation du moteur. Le hacheur est piloté par un signal *PMW* (Pulse Width Modulation) ou en français, *MLI* (Modulation de Largeur d'Impulsion) dont on va contrôler le rapport cyclique qui permettra donc d'ajuster la valeur moyenne en sortie. Pour créer un signal MLI, on utilisera un comparateur : celui-ci va comparer une tension de commande DC $u_{commande}$ et un signal triangulaire. Ainsi, lorsque la tension de commande est supérieure au signal triangle, la tension de sortie du comparateur prend une valeur haute sinon elle vaut 0. Visuellement, on a la forme d'ondes suivantes :

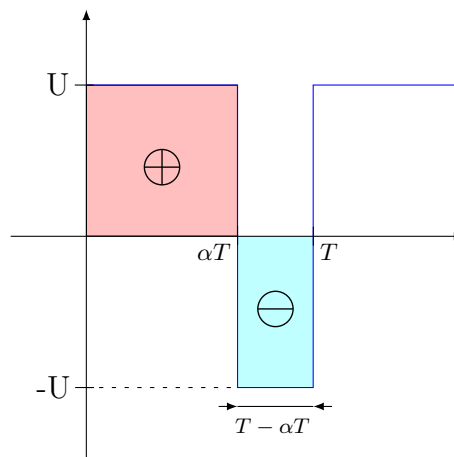


FIGURE 20 – Signal d'entrée de la MCC

Grâce à la méthode des aires, on peut établir la relation entre la tension d'alimentation du hacheur (ici 48 V) et sa tension moyenne de sortie, on a donc :

$$\langle U_{AB} \rangle = \frac{\text{aire} \oplus - \text{aire} \ominus}{T} = \frac{U\alpha T - U(T - \alpha T)}{T} = (2\alpha - 1)U = \left(2\frac{u_{\text{commande}}}{U_M} - 1\right)U \quad (4)$$

Nous avons ainsi modélisé le signal fourni par le hacheur par le calcul de sa valeur moyenne. Cela sera particulièrement utile pour les simulations sur Simulink qui ne contient pas de modèle de hacheur intégré.

5.v.b Implémentation sous PSIM

A partir de ce qui a été dit précédemment, on implémente la commande de la manière suivante sur PSIM :

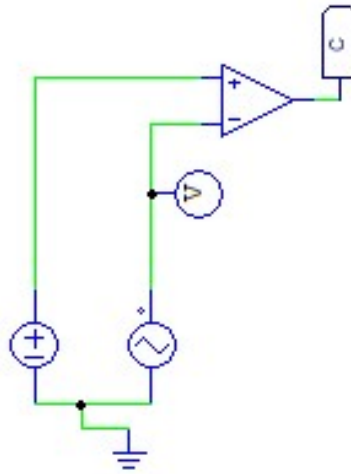


FIGURE 21 – Modélisation de la commande sous PSIM *

5.v.c Implémentation sous Matlab/Simulink

De même, à partir de l'équation 4, on implémente la commande sur Simulink en changeant la constante U précédente par ce qui est montré en figure 22 :

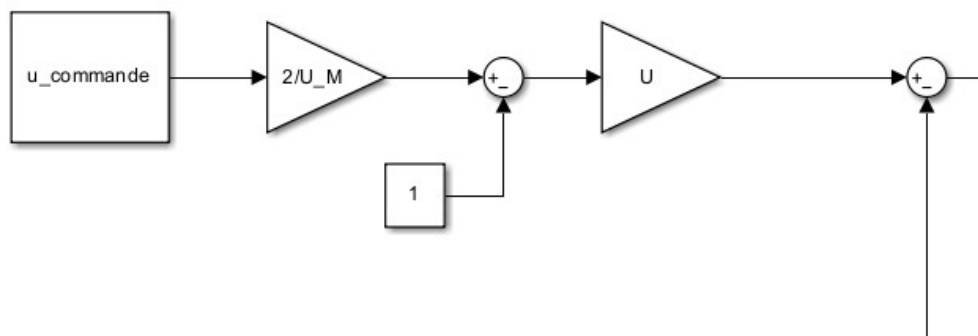


FIGURE 22 – Modélisation de la commande sous Matlab *

5.vi Etude du régime permanent considérant la commande

Nous pouvons à présent vérifier la conformité de la modélisation de la commande sur Simulink puis sur PSIM. Pour cela, nous prendrons pour l'exemple un rapport cyclique $\alpha = 0,75$ avec $u_{commande} = 11,25V$ et $U_M = 15V$.

5.vi.a Sous Matlab/Simulink

On peut recalculer sur Matlab, les valeurs finales de nos variables d'états en remplacement U par la valeur de $\langle U_{AB} \rangle$ établie en (4) et grâce aux valeurs de l'exemple évoquées précédemment. On obtient alors :

$$X'_\infty = \begin{pmatrix} 2,2882A \\ 1,78142A \\ 1543tr/min \end{pmatrix}$$

5.vi.b Sous PSIM

Testons à présent notre modélisation sur PSIM. Pour cela, nous allons reproduire la formule de $\langle U_{AB} \rangle$ à l'aide de blocs de la manière suivante :

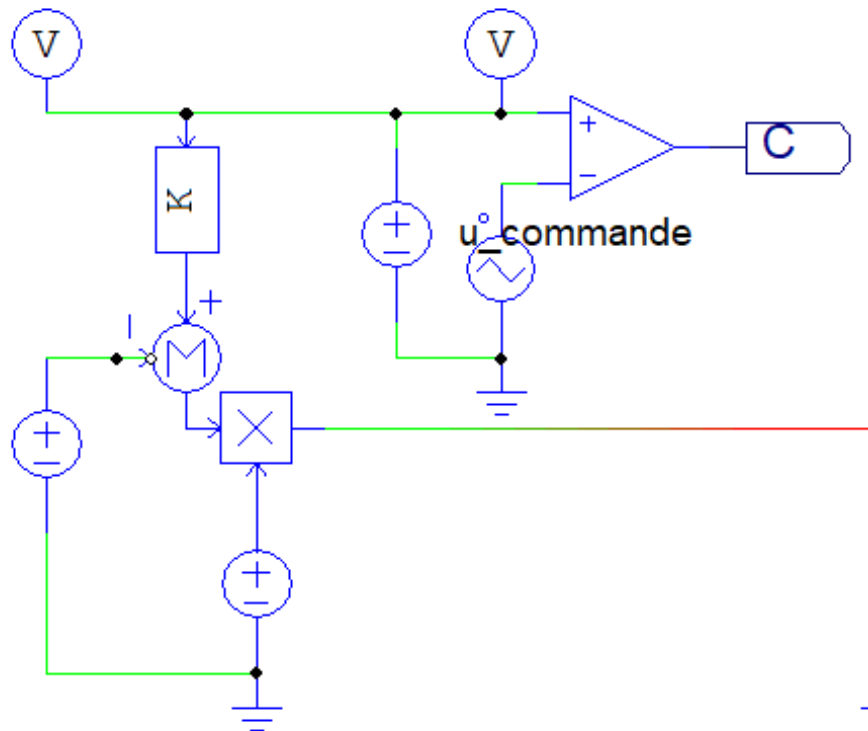


FIGURE 23 – Reproduction de la formule $\langle U_{AB} \rangle$ sous PSIM *

On observe ici la création de 2 signaux : le premier (label C) est le signal MLI évoqué plus tôt qui va piloter le hacheur et en second, on crée le signal $\langle U_{AB} \rangle = (2 \frac{u_{commande}}{U_M} - 1)U$ à partir de $u_{commande}$, ici une source de tension continue. Nous pouvons ainsi comparer l'évolution de nos variables d'états avec 2 entrées différentes : l'une est le signal directement fourni par le hacheur et l'autre est le signal de sortie modélisé.

En figure 24, figure 25 et figure 26 sont présentées, l'allure du courant d'induit et de la vitesse avec ces deux signaux :

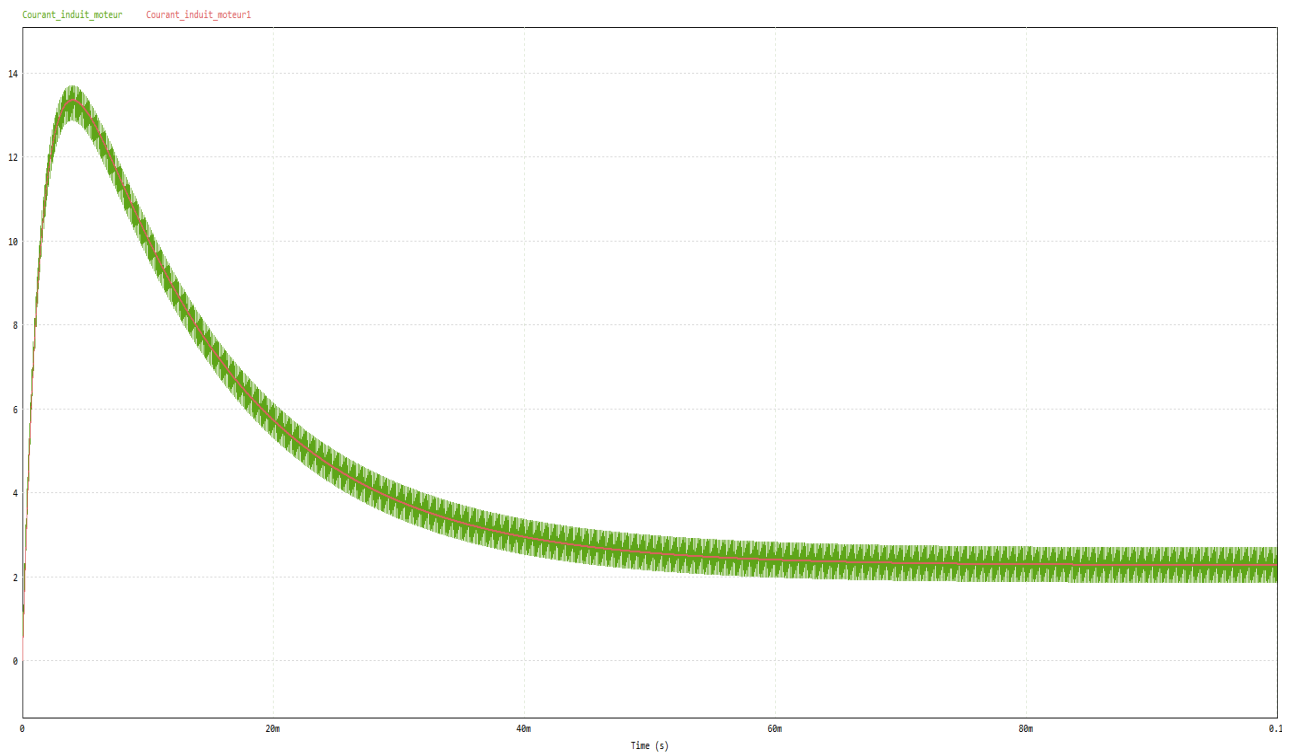


FIGURE 24 – Courant d'induit moteur avec hacheur (vert) et courant d'induit moteur avec modélisation du hacheur (rouge) *

On a en abscisse le temps en seconde et en ordonnée le courant en ampère.

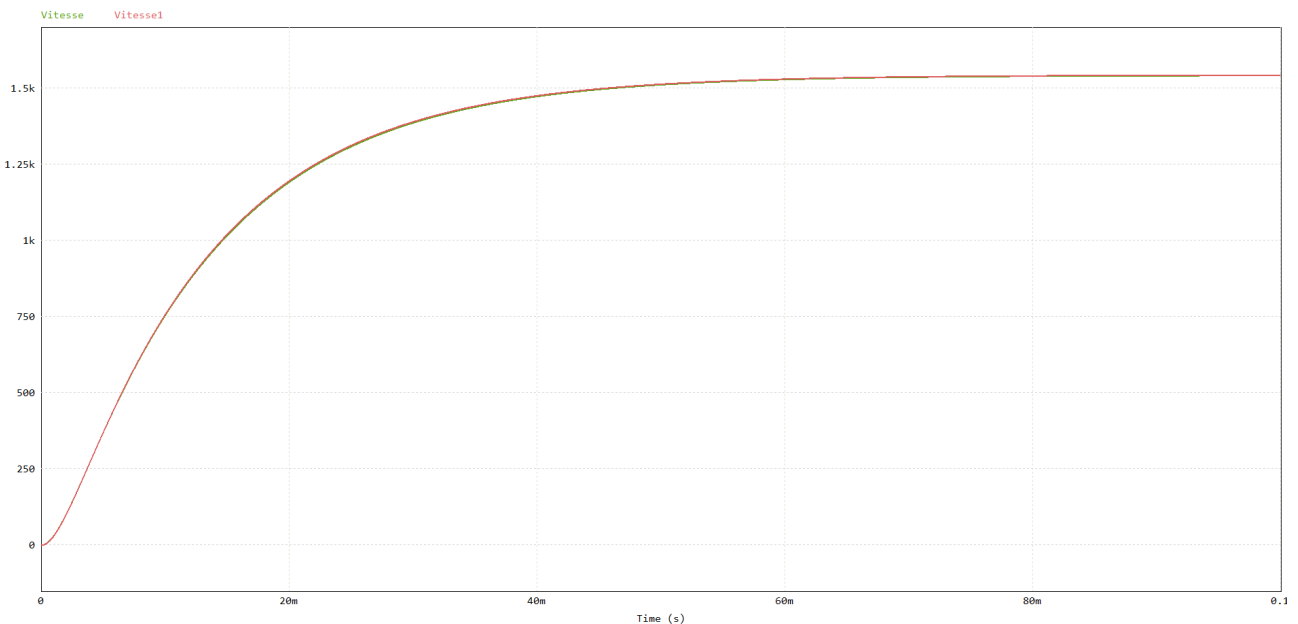


FIGURE 25 – Vitesse moteur avec hacheur (vert) et vitesse moteur avec modélisation du hacheur (rouge) *

On a en abscisse le temps en seconde et en ordonnée la vitesse en tr/min.

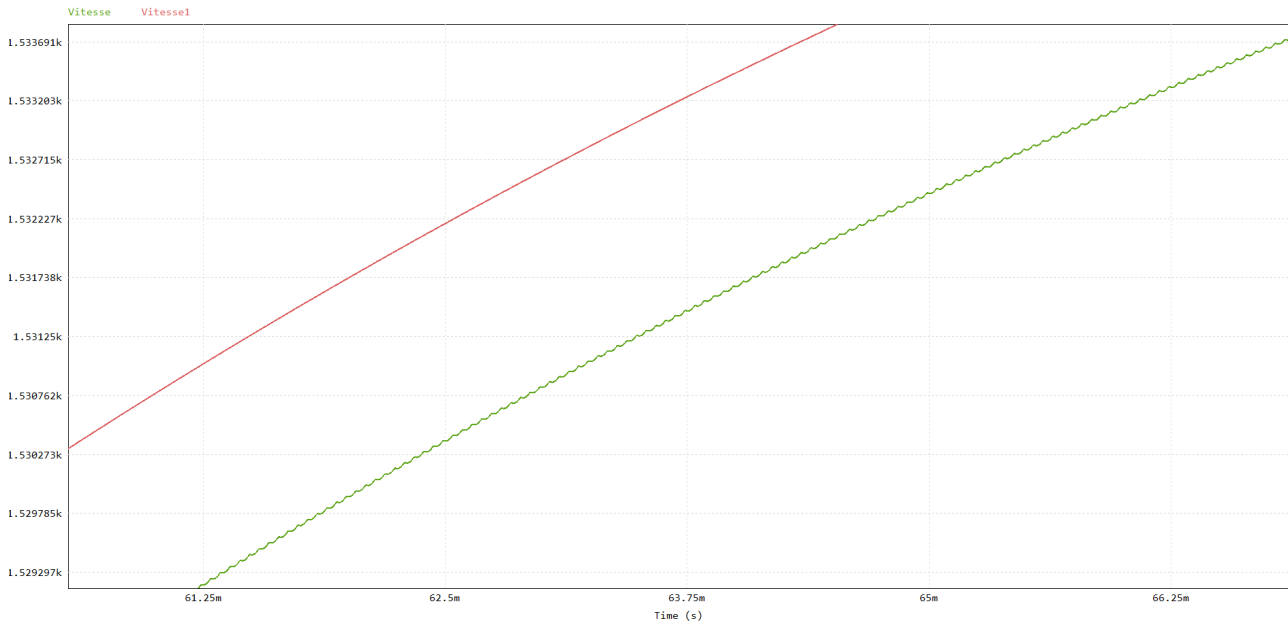


FIGURE 26 – Zoom sur l'ondulation de vitesse *

On a en abscisse le temps en seconde et en ordonnée la vitesse en tr/min .

On constate plusieurs choses :

- Tout d'abord avec le signal directement fourni par le hacheur, on constate une ondulation sur les deux courbes. Celle-ci est dû au signal du hacheur qui n'est pas continu mais carré.
- Malgré cette ondulation, on constate que l'évolution des grandeurs est la même que celle avec la modélisation du hacheur, ce qui confirme la bonne conception de notre hacheur sur PSIM. De plus, la courbe avec modélisation du hacheur est pratiquement la même que celle que nous obtenons sur Matlab avec la même modélisation de la commande. Cela conforte encore notre modèle.
- Enfin, les valeurs des grandeurs en régime permanent obtenus avec ces courbes sont cohérentes avec celles calculées via Matlab précédemment.

On voit que PSIM nous permet donc de simuler plus finement le système en prenant en compte les ondulations réelles du système.

6 Dimensionnement et simulation des asservissements

6.1 Préliminaire à l'asservissement

Avant de commencer le dimensionnement de nos asservissements, nous devons légèrement modifier notre modèle. En effet, pour réaliser nos asservissements, on va considérer que l'on se situe autour d'un point d'équilibre et que nous allons linéariser le système autour de ce point d'équilibre. Ainsi, nous ne prendrons pas en compte les éléments invariants à une perturbation δ sur notre système. De ce fait, nous négligerons dans notre asservissement les frottements secs $2C_0$ et prendrons en compte uniquement le gain du hacheur à une perturbation, soit $\frac{2U}{U_M} = 6,4$. On ajustera donc ces éléments dans nos modèles PSIM et Matlab. On utilisera aussi un limiteur de tension pour limiter la tension de commande afin qu'elle ne sorte pas de sa plage de fonctionnement normale.

6.ii Asservissement en courant

6.ii.a Calcul du correcteur

Pour simplifier la réalisation de l'asservissement, nous allons regrouper les différentes parties qui constituent notre schéma bloc global et établir la fonction de transfert en boucle ouverte du courant. On a le schéma bloc suivant :

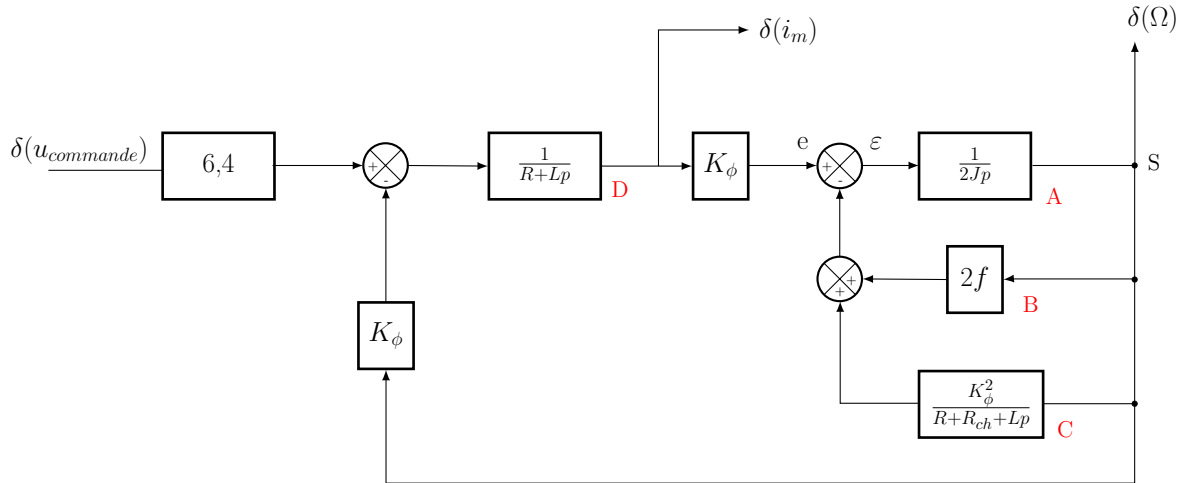


FIGURE 27 – Schéma-bloc du système global

A partir de cette structure, nous pouvons établir la fonction de transfert souhaitée. Nous allons dans un premier temps établir une fonction de transfert H qui sera le rapport entre la vitesse Ω et le courant i_m . Nous pouvons écrire les équations suivantes :

$$\begin{cases} \varepsilon = e - B \cdot S - C \cdot S \\ S = A \cdot \varepsilon = A(e - (B + C)S) \end{cases} \quad (5)$$

On en tire l'expression suivante :

$$S = \frac{A \cdot e}{1 + A(B + C)}$$

En nommant $H = K_\phi \cdot \frac{A}{1 + A(B + C)}$, on a finalement :

$$H = \frac{\Omega(p)}{i_m(p)} = K_\phi \cdot \frac{A}{1 + A(B + C)}$$

Nous pouvons alors simplifier le schéma bloc de la manière suivante :

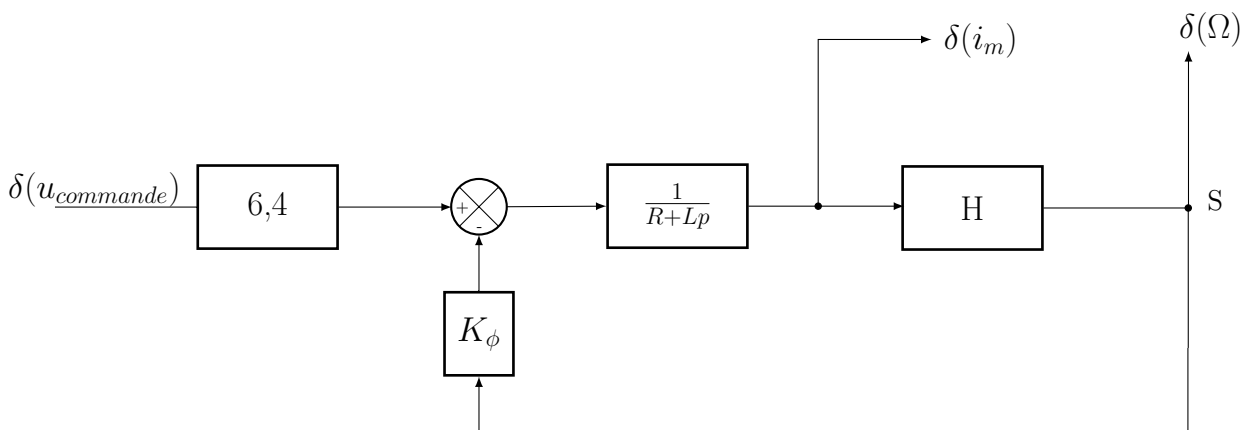


FIGURE 28 – Schéma-bloc simplifié du système global

Nous pouvons finalement établir l'expression du courant d'induit moteur en fonction de la tension de commande :

$$\delta(i_m) = 6,4 \cdot \frac{1}{\frac{R + Lp}{K_\phi^2 \cdot H} \cdot \delta(u_{commande})} \quad (6)$$

A présent, il convient de comparer l'exactitude de notre modélisation du courant sur Simulink. Nous avons le schéma-bloc suivant sur Simulink :

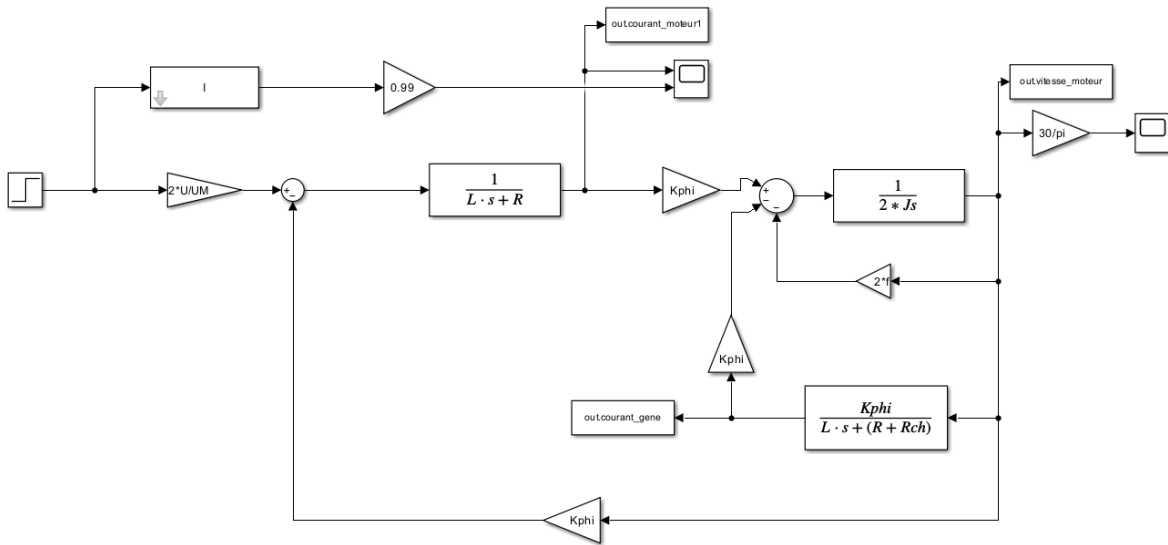


FIGURE 29 – Schéma-bloc Simulink *

La partie basse correspond au schéma-bloc complet du système et au-dessus, le bloc I correspond à la fonction de transfert établie plus haut sur l'équation 6. Le facteur 0,99 permettra de mieux visualiser les deux courbes sur le scope. On réalise un essai indiciel :

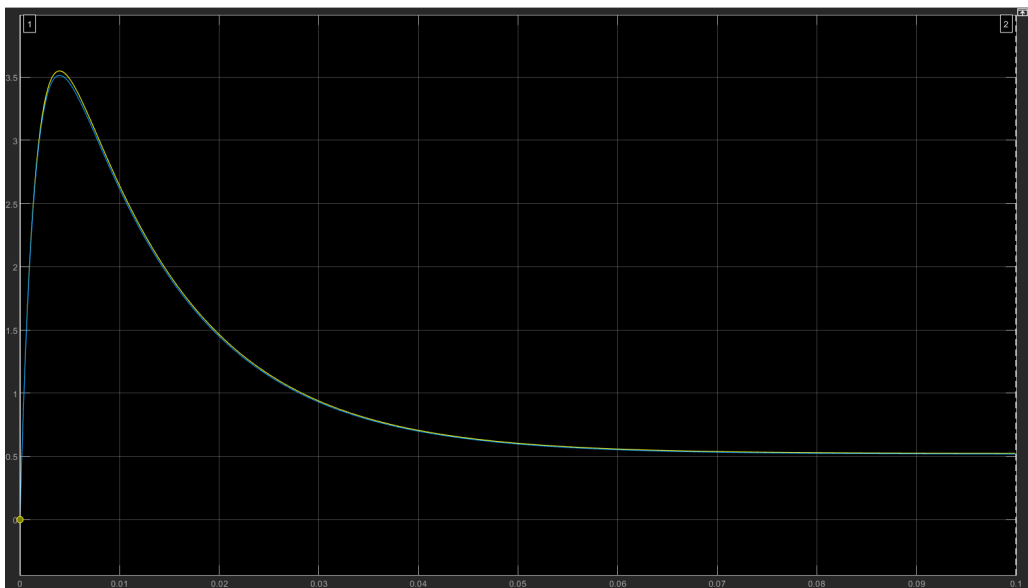


FIGURE 30 – Comparaison du courant d'induit issu du schéma global et du modèle *

On observe une correspondance exacte entre les deux courbes ce qui témoigne de la bonne modélisation du courant d'induit à partir de la fonction de transfert en boucle ouverte.

Le modèle maintenant établi et validé, nous pouvons procéder à son asservissement. Nous avons le schéma bloc suivant :

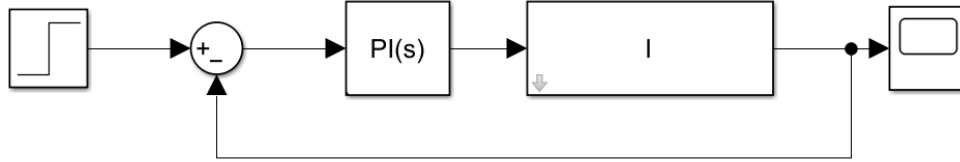


FIGURE 31 – Boucle fermée de courant *

Grâce à la modélisation effectuée précédemment, ce schéma est très simplifié : le correcteur va nous permettre d'ajuster la réponse du courant selon notre cahier des charges. Notre fonction de transfert est en pratique trop complexe pour être étudiée à la main, nous allons donc utiliser l'outil *PID Tuner* de Simulink qui va calculer automatiquement un correcteur selon nos préférences. On rappelle que l'on souhaite un temps de réponse à 5 % égal à au moins 10 fois la période du signal MLI qui pilote le hacheur. La fréquence de ce signal est fixée à $f_{MLI} = 20\text{kHz}$ d'où $T_{MLI} = \frac{1}{f_{MLI}} = 5 \times 10^{-5}\text{s}$. Nous devons donc fixer $t_{r5\%} = 10 \times T_{MLI} = 5 \times 10^{-4}\text{s}$. On rappelle que l'on devra limiter le dépassement entre 10 % et 20 % de la valeur finale.

Ainsi grâce au *PID Tuner*, on règle la réponse en boucle fermée pour qu'elle soit robuste, c'est-à-dire peu sensible aux perturbations et assez rapide pour correspondre à notre cahier des charges. Nous obtenons les valeurs suivantes :

$$PI(p) = P + I \frac{1}{p}$$

$$P = 6,2186$$

$$I = 41446 \text{ s}^{-1}$$

La réponse à un essai indiciel est la suivante :

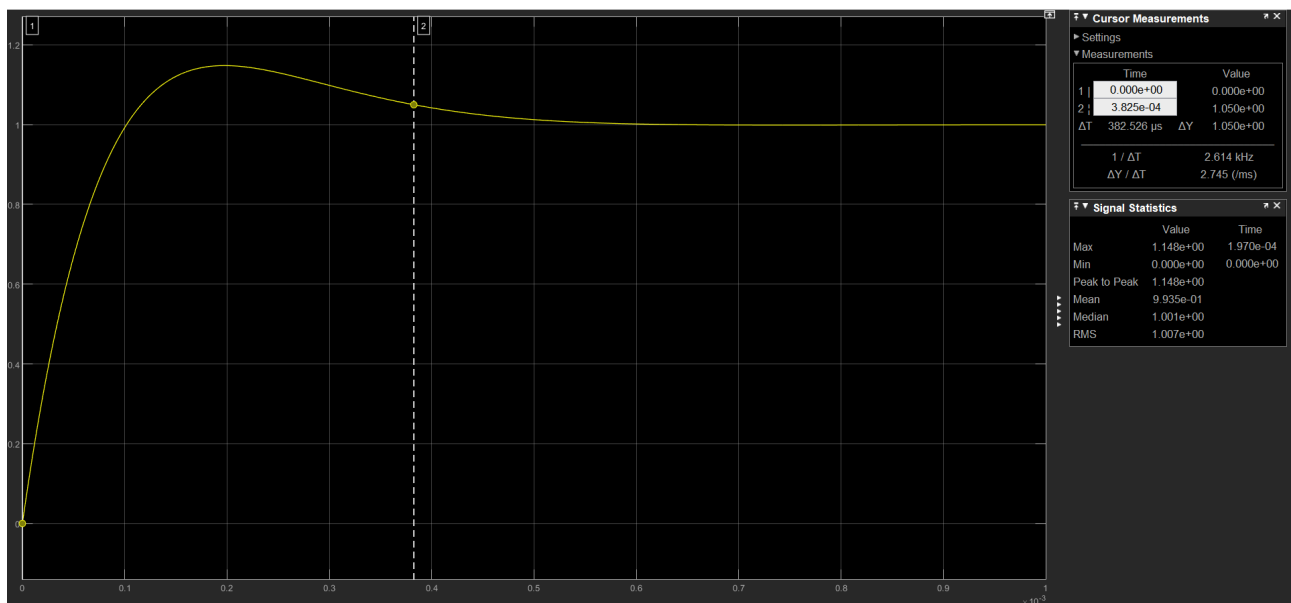


FIGURE 32 – Réponse asservie du courant d'induit *

La réponse correspond bien à nos attentes avec un dépassement maximum de 14,8 % et un temps de réponse effectif de 383 μs , inférieur au temps de réponse maximum imposé de 500 μs .

6.ii.b Vérification du correcteur

Nous allons maintenant vérifier sur PSIM que notre correcteur a bien été dimensionné. On réalise donc l'asservissement sur PSIM avec les mêmes valeurs de correcteur que trouvées précédemment à l'aide du schéma suivant :

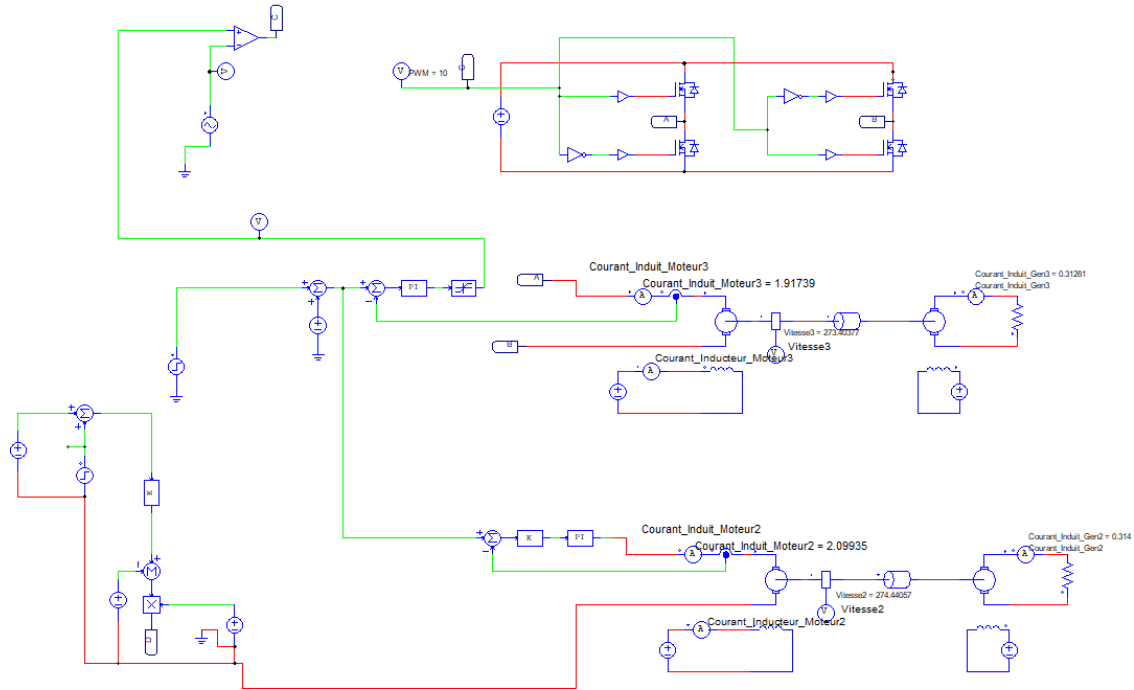


FIGURE 33 – Schéma de vérification de l'asservissement de courant sur PSIM *

Nous avons la partie du bas qui simule le correcteur avec le hacheur modélisé et la partie juste au dessus qui utilise le hacheur directement. L'objectif est de voir ici si la réponse à une perturbation avec le hacheur et le correcteur suit la même évolution que la modélisation avec le même correcteur.

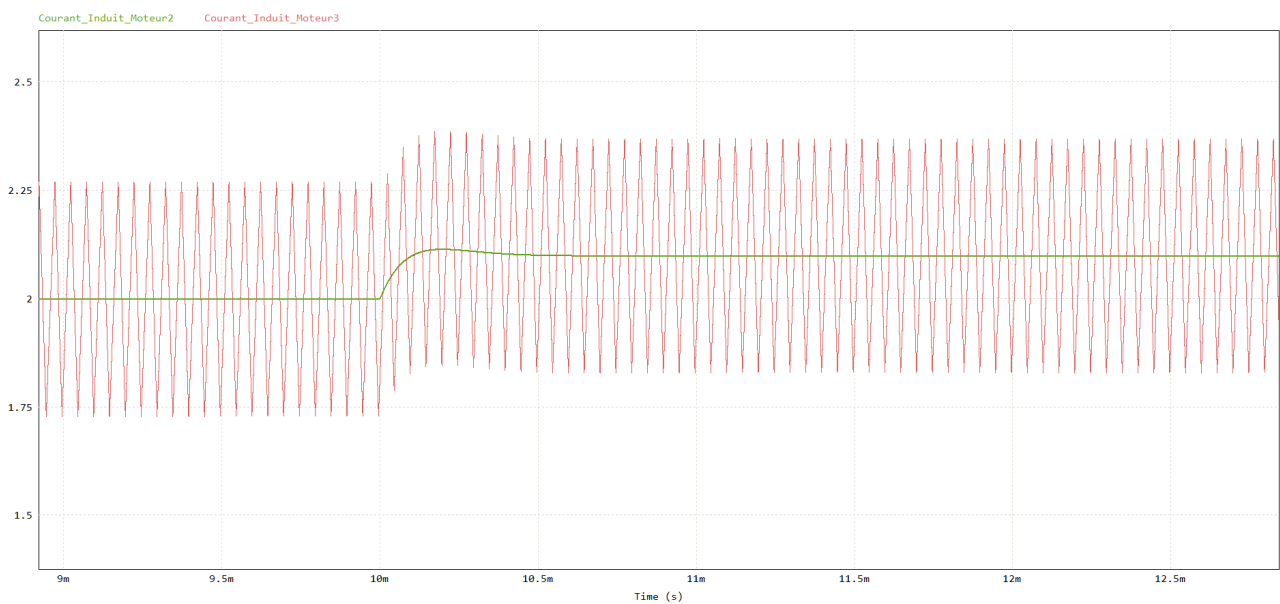


FIGURE 34 – Courbes de vérification de l'asservissement de courant sur PSIM *

En vert, on a la courbe avec le hacheur modélisé et en rouge, la courbe avec le hacheur. Comme nous avons considéré être autour d'un point d'équilibre, il est impossible de comparer les réponses à la mise en route du système, il faut donc comparer les réponses à une perturbation en régime établi comme sur la [figure 34](#). On constate que les réponses suivent la même évolution, ce qui valide le calcul de notre correcteur.

On a bien pris soin de s'assurer que l'asservissement avec la modélisation du hacheur sur PSIM était équivalente à celle sur Matlab (temps de réponse, gain statique, dépassement) avant d'effectuer les comparaisons précédentes.

6.iii Asservissement de vitesse

Pour l'asservissement de vitesse, étant donné que nous utiliserons deux types de capteur qui seront susceptibles de ne pas avoir les mêmes gains, nous expliciterons dans un premier temps la démarche et la réflexion en faisant l'hypothèse que nous avons un gain de capteur unitaire. Par la suite, il nous suffira simplement de prendre en compte les gains (ou le gain si les capteurs possèdent le même gain) des capteurs et d'effectuer la même démarche. Ainsi, lors de la prise en compte des capteurs, nous expliciterons uniquement la validité des résultats et non plus la démarche qui sera donc la même que celle présentée en amont.

6.iii.a Démarche globale

6.iii.a.1 Calcul du correcteur

De manière similaire à l'asservissement de courant, nous pouvons déterminer la fonction de transfert en boucle fermée équivalente globale pour la vitesse. Celle-ci se calcule grâce à la formule de Black, non plus au niveau de la sortie de courant mais bien celle de vitesse.

Si nous reprenons les écritures précédentes, elle peut s'écrire :

$$\frac{K_{\phi} \cdot H \cdot D}{1 + K_{\phi} \cdot H \cdot D}$$

A l'aide de Matlab, nous pouvons rapidement calculer cette fonction. Il s'agit d'une fonction de transfert du huitième ordre. Lorsque nous simulons notre système en remplaçant notre schéma bloc par cette seule fonction, voici ce que nous obtenons.

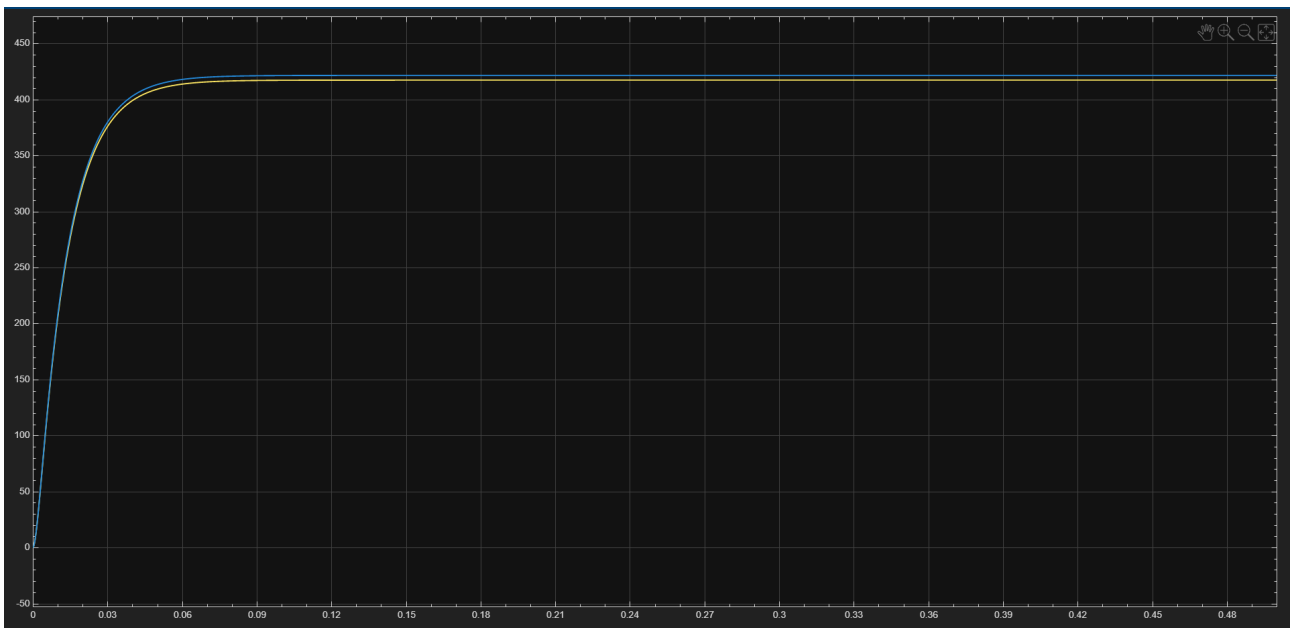


FIGURE 35 – Fonction du huitième ordre *

Nous avons rajouté un facteur 0,99 à la fonction de transfert nouvellement calculée afin de pouvoir discerner nos deux courbes. On remarque effectivement que cette fonction est la bonne modélisation de notre système.

Seulement, cette fonction a été calculée à partir de H, une fonction de transfert du troisième ordre : cela peut rendre sa manipulation fastidieuse, particulièrement lors du calcul du correcteur. Pour résoudre ce problème, nous pouvons réaliser une simplification d'ordre physique. La constante de temps de l'inductance de la génératrice étant très rapide, nous pouvons simplement la négliger devant le système complet. Nous pouvons donc simplifier les expressions de certaines fonctions en supposant L nulle.

On peut donc dans la modélisation de la génératrice modifier la fonction que nous avons appelée "C", celle-ci devient :

$$C = \frac{K_\phi^2}{R + R_{ch} + Lp} = \frac{K_\phi^2}{R + R_{ch}}$$

On remarque que C est à présent une constante, nous pouvons alors l'ajouter à notre autre constante $B = 2f$. Nous obtenons ainsi une nouvelle constante que nous noterons $f_{eq} = \frac{K_\phi^2}{R + R_{ch}} + 2f$.

Ensuite, nous pouvons calculer la nouvelle fonction de transfert du point de vue de la vitesse. Celle-ci se calcule aisément par la formule de Black suivante :

$$G = \frac{A}{1 + A \cdot f_{eq}} = \frac{\frac{1}{2Jp}}{1 + \frac{1}{2Jp} f_{eq}} = \frac{1}{2Jp + f_{eq}}$$

On se retrouve maintenant avec une fonction de transfert du premier ordre, bien plus simple à manipuler que H. Néanmoins, cette approximation pourrait être trop "simplificatrice", comparons donc nos 3 courbes jusqu'ici obtenues afin de valider notre modèle.

Pour ce faire, calculons la nouvelle fonction de transfert en boucle fermée. En effet, G est la nouvelle expression de la fonction que nous avons appelée H. On reprend alors l'expression que nous avons en remplaçant H par G :

$$\frac{K_\phi \cdot G \cdot D}{1 + K_\phi \cdot G \cdot D}$$

Nous obtenons une fonction de transfert en boucle fermée du quatrième ordre.

Comparons maintenant nos 3 courbes, celle sans avoir simplifié le schéma bloc, celle en ayant simplifié mais en ayant gardé L non nul, et celle avec le L de la génératrice nul. Notons qu'un facteur 0.99 et 1.01 ont été ajoutés aux courbes simplifiées pour une meilleure visibilité.

On obtient :

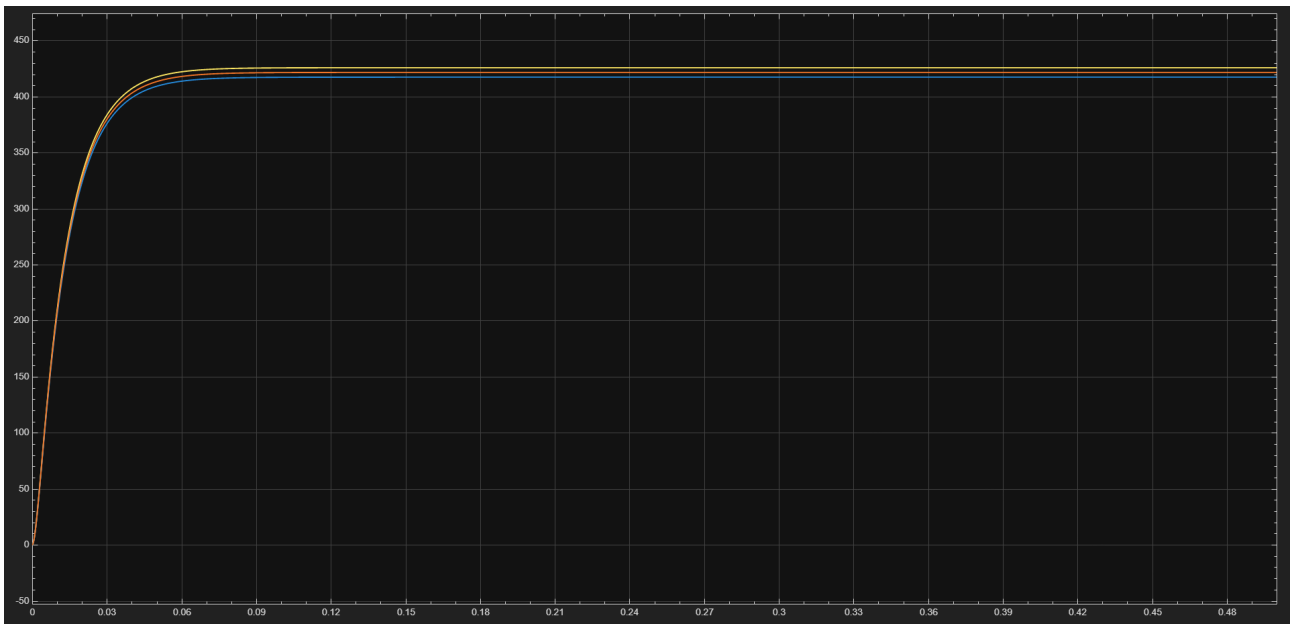


FIGURE 36 – Comparaison des 3 modèles *

Au vue de ces courbes, il va sans dire que notre modèle avec une fonction du premier ordre plutôt que du troisième ordre est parfaitement justifié.

Avant de calculer l'asservissement, il est possible d'effectuer une dernière simplification dans le schéma bloc qui va alléger nos calculs. Nous savons que l'asservissement de vitesse ne peut se faire sans celui du courant, il faudrait alors prendre toute la partie précédente en compte ici pour calculer notre correcteur et nos fonctions de transfert équivalentes.

Cependant, rappelons nous que le temps de réponse de la boucle de courant est de l'ordre de la centaine de μs tandis que celui la vitesse est de l'ordre du dixième de seconde. Ainsi, comme $\tau_{courant} \ll \tau_{vitesse}$, nous pouvons assimiler l'entièreté de la boucle de courant à un gain unitaire, cette dernière n'ayant quasiment aucun impact sur le réponse en vitesse de notre système.

Maintenant que ce modèle a été validé, nous pouvons passer à l'asservissement. L'avantage que nous avons à présent, est que nous pouvons calculer facilement à la main les coefficients de notre correcteur proportionnel-intégral. On a donc le schéma bloc suivant :

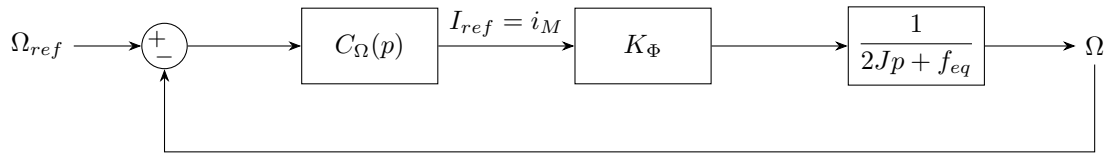


FIGURE 37 – Schéma-bloc de l'asservissement de vitesse final

Rappelons les calculs :

$$F(p) = \frac{C(p)G(p)}{1 + C(p)G(p)} = \frac{K_i \frac{1 + \tau p}{\tau p} \frac{A}{1 + \tau p}}{1 + K_i \frac{1 + \tau p}{\tau p} \frac{A}{1 + \tau p}} = \frac{\frac{K_i A}{\tau p}}{1 + \frac{K_i A}{\tau p}} = \frac{1}{1 + \frac{\tau}{K_i A} p} = \frac{K_{BF}}{1 + \tau_{BF} p}$$

Où $C(p)$ est le correcteur, $T(p)$ est la fonction de transfert de notre système en boucle ouverte et $H(p)$ est la fonction de transfert de notre système asservi en boucle fermée.

On peut réécrire l'expression de G :

$$\frac{1}{2Jp + f_{eq}} = \frac{\frac{1}{f_{eq}}}{1 + \frac{2J}{f_{eq}} p}$$

On en déduit

$$\tau_{BO} = \frac{2J}{f_{eq}}$$

De même on trouve $A = \frac{K_{Phi}}{f_{eq}}$. Il ne faut pas oublier le facteur K_{Phi} dans la formule de Black.

De ça, on calcule ensuite $\tau_{BF} = \frac{\tau_{BO}}{K_i A} = \frac{\frac{2J}{f_{eq}}}{K_i \cdot \frac{1}{f_{eq}} \cdot K_{Phi}}$

Or, le cahier des charges stipule un temps de réponse en boucle fermée 3 fois inférieur à celui en boucle ouverte, on en déduit alors que $K_i \cdot \frac{1}{f_{eq}} \cdot K_{Phi} = 3 \rightarrow K_i = \frac{3f_{eq}}{K_{Phi}} = 0,0354$.

Précisons que lors des calculs, nous avons volontairement choisi $T_i = \tau_{BO}$ afin de simplifier l'expression. Nous savons que cette méthode n'est pas la plus optimale, mais elle fournit un résultat satisfaisant pour ce que nous cherchons. Nous comparerons nos calculs aux résultats trouvés avec le PID Tuner pour s'en assurer.

Nos coefficients de correcteur maintenant calculés, testons notre asservissement sur Matlab.

Pour ce faire, de façon analogue à l'asservissement de courant, nous venons créer une simple boucle de rétroaction munie d'un correcteur PI en essai indiciel.



FIGURE 38 – Asservissement simplifié de la vitesse

Ici, dans le PI, nous remplissons $P = K_i = 0,0354$ et $I = \frac{1}{\tau_{BO}} = 27,1 \text{ s}^{-1}$.

Lorsque nous lançons la simulation avec un échelon de 1, nous obtenons la courbe suivante :

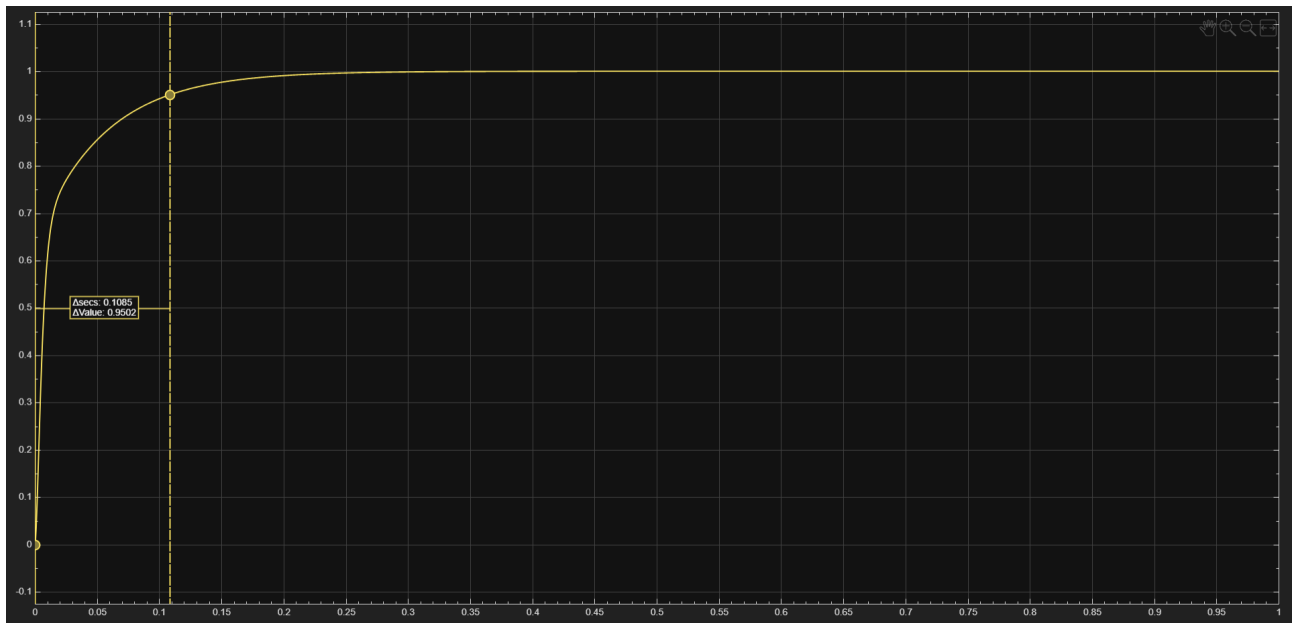


FIGURE 39 – Graphe de la vitesse asservie *

A l'aide des curseurs, nous pouvons déterminer le temps de réponse à 5%, ici nous le mesurons à 0.1085s.

$$\text{Rappelons } tr_{5\%BF} = 3\tau_{BF} = 3 \times \frac{\frac{2J}{f_{eq}}}{K_i \times \frac{1}{f_{eq}} \times K_{Phi}} = \frac{2J}{f_{eq}} = 0.1107s$$

On trouve ce que l'on cherchait, le correcteur PI est donc bien dimensionné.

Avec le PID Tuner, nous trouvons :

$$K_i = 0,0110$$

$$T_i = 0,0142s$$

6.iii.a.2 Vérification du correcteur

Comme précédemment pour le courant, nous allons vérifier notre asservissement sur PSIM. Étant donné que pour calculer notre correcteur de vitesse, nous avons fait l'hypothèse que le courant s'établissait instantanément, il ne faut donc pas oublier de tester l'asservissement de vitesse **avec** l'asservissement de courant. Nous aurons donc ici notre système complet asservi. Voici le schéma de ce dernier sur la figure suivante :

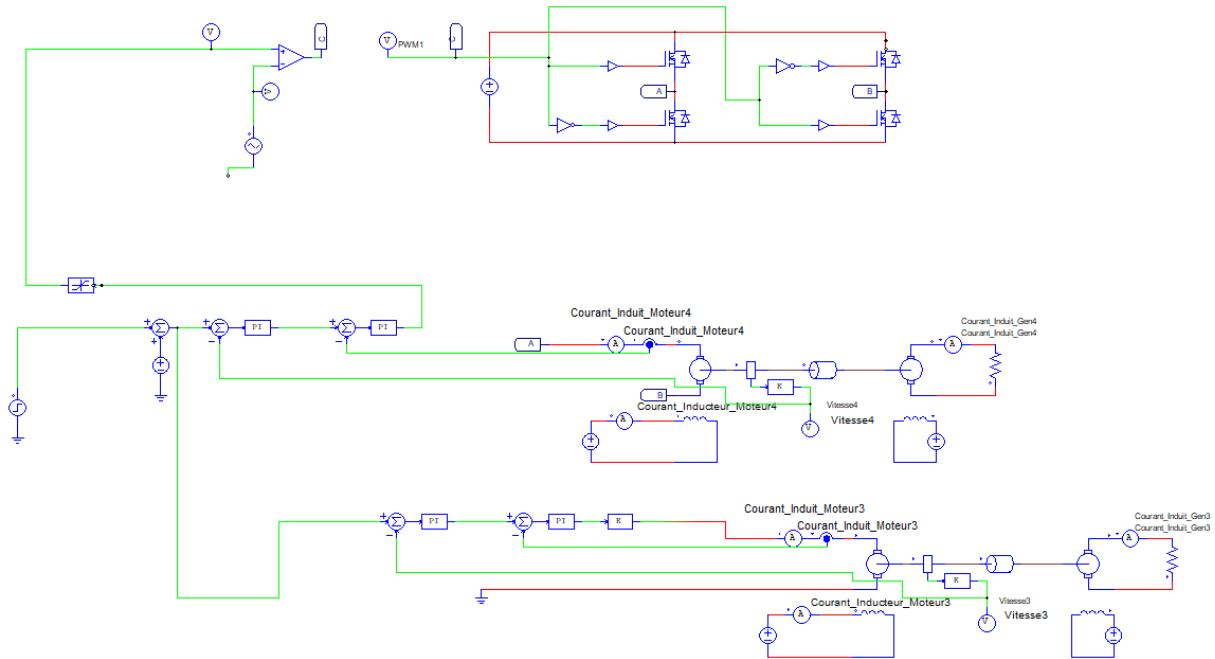


FIGURE 40 – Vérification de l'asservissement de vitesse : Schéma PSIM *

Nous avons la partie du bas qui simule le correcteur avec le hacheur modélisé et la partie juste au dessus qui utilise le hacheur directement. Le premier correcteur à gauche est celui de vitesse et le deuxième à droite, celui de courant. Comme précédemment, l'objectif est de voir ici si la réponse à une perturbation avec le hacheur et le correcteur suit la même évolution que la modélisation avec le même correcteur.

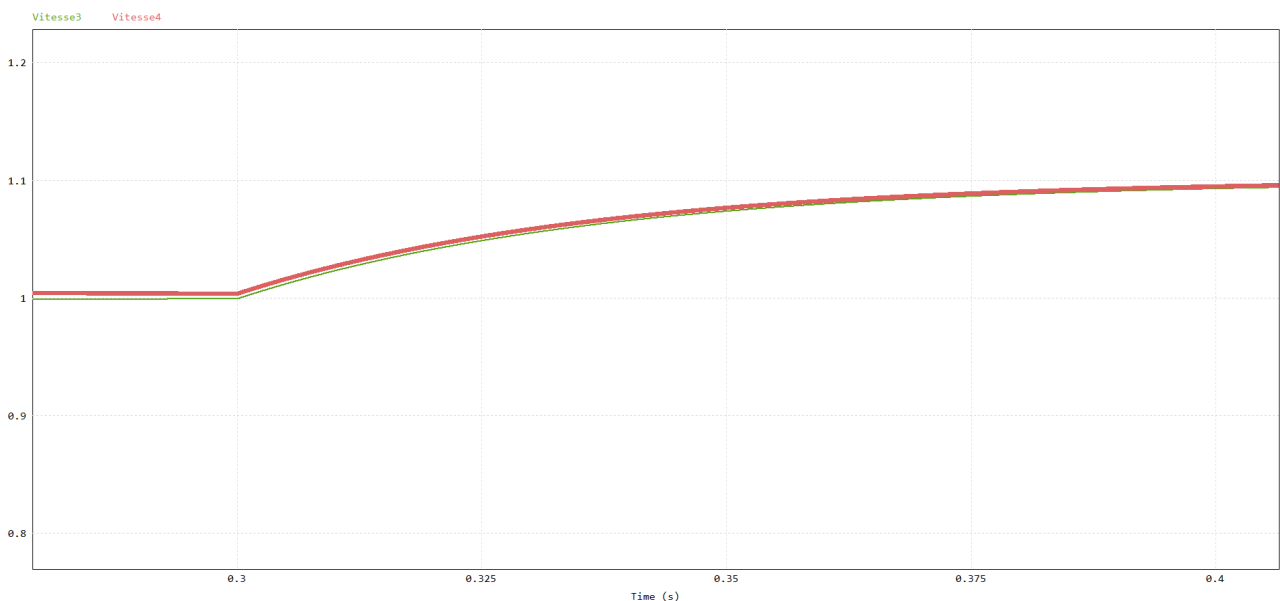


FIGURE 41 – Courbes de vérification de l'asservissement de vitesse sur PSIM *

En vert, on a la courbe avec le hacheur modélisé et en rouge, la courbe avec le hacheur. Comme nous avons considéré être autour d'un point d'équilibre, il est impossible de comparer les réponses à la mise en route du système, il faut donc comparer les réponses à une perturbation en régime établi comme sur la [figure 41](#). On constate que les réponses suivent la même évolution, ce qui valide le calcul de notre correcteur.

On a bien pris soin de s'assurer que l'asservissement avec la modélisation du hacheur sur PSIM était équivalente à celle sur Matlab (temps de réponse, gain statique, dépassement) avant d'effectuer les comparaisons précédentes.

6.iii.b Prise en compte des capteurs

Comme mentionné précédemment, notre système réel comporte des capteurs pour mesurer la vitesse de rotation de la MCC. En particulier, nous disposons de deux types de capteurs de vitesse différents : une dynamo-tachymétrique et un codeur incrémental. Nous considérons que leurs constantes de temps sont largement plus rapides que celles de notre système et les modéliserons donc seulement par des gains. Pour la dynamo-tachymétrique, son gain est spécifié comme étant de 6 V pour 1000 tr/min. Nous veillerons à diviser ce gain par deux pour faire en sorte que la tension de commande reste inférieur à 15 V.

Finalement, dans notre modèle Matlab, la vitesse de sortie est en rad/s , on multiplie donc cette vitesse par $\frac{30}{\pi}$ pour la convertir en tr/min . Le gain final à appliquer sur Matlab pour la dynamo-tachymétrie est donc :

$$k = \frac{1}{2} \cdot \frac{6}{1000} \cdot \frac{30}{\pi} \approx 28 \cdot 10^{-3}$$

Sur PSIM, le capteur nous retourne déjà une vitesse en tr/min , il suffira donc d'appliquer le gain de $\frac{3}{1000}$. Les modifications apportées sont présentées ci-dessous :

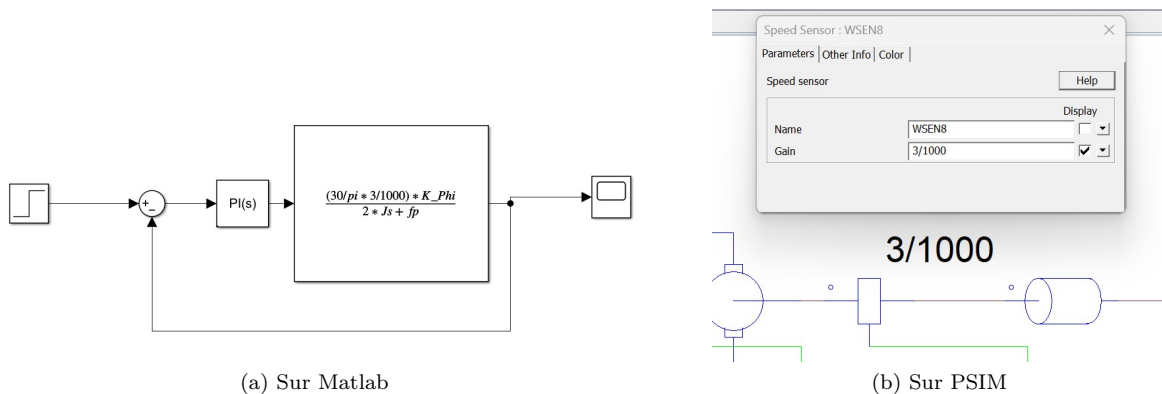


FIGURE 42 – Modifications prenant en compte les capteurs *

Pour le codeur incrémental, nous ferons en sorte qu'il ait un gain identique à la dynamo-tachymétrie pour n'avoir qu'à dimensionner qu'un seul correcteur.

Après calcul des nouveaux correcteurs grâce au *PID Tuner*, nous obtenons les valeurs suivantes :

$$C(p) = P \left(1 + I \frac{1}{p} \right) \text{ avec } P = 2.27 \text{ et } I = 22.8 \text{ s}^{-1}$$

Comme précédemment, nous avons vérifié ce correcteur et obtenu des résultats concluants. Cependant, nous allons cette fois-ci mettre en avant une comparaison complète de notre système. Pour cela nous simulerons le système complet et corrigé (courant et vitesse) sur PSIM, ainsi que le système complet et corrigé **sans les hypothèses simplificatrices** sur Matlab. C'est-à-dire que nous garderons tous les éléments de notre schéma-bloc hormis les éléments évoqués dans la partie préliminaire à l'asservissement.

Nous avons ainsi le schéma Simulink suivant :

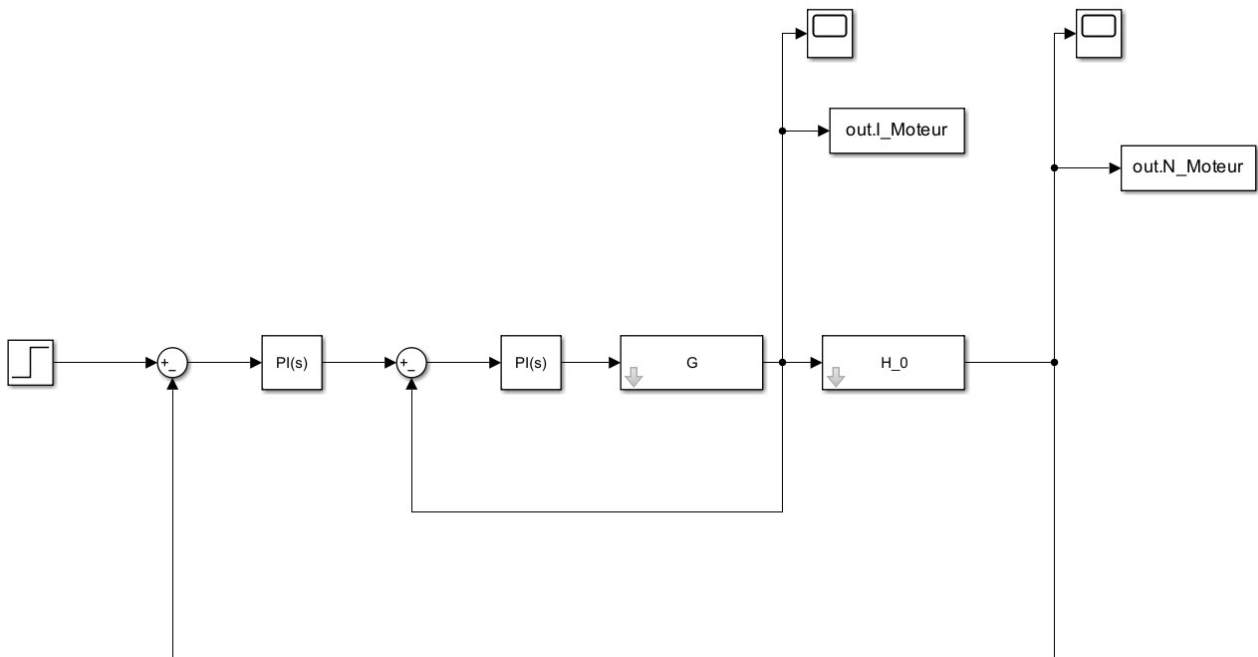


FIGURE 43 – Schéma-bloc de l'asservissement avec capteurs *

Où G et H_0 sont les fonctions de transfert permettant d'obtenir respectivement le courant (ou sa tension image considérant un gain unitaire) à partir de la commande issue des correcteurs et la tension image de la vitesse à partir du courant moteur. Le schéma utilisé pour PSIM est le même que celui utilisé pour la vérification de l'asservissement de vitesse, ce dernier faisant déjà intervenir tous les éléments.

Nous retrouverons sur les figures suivantes le modèle avec le hacheur de PSIM en rouge et le modèle Matlab en bleu.

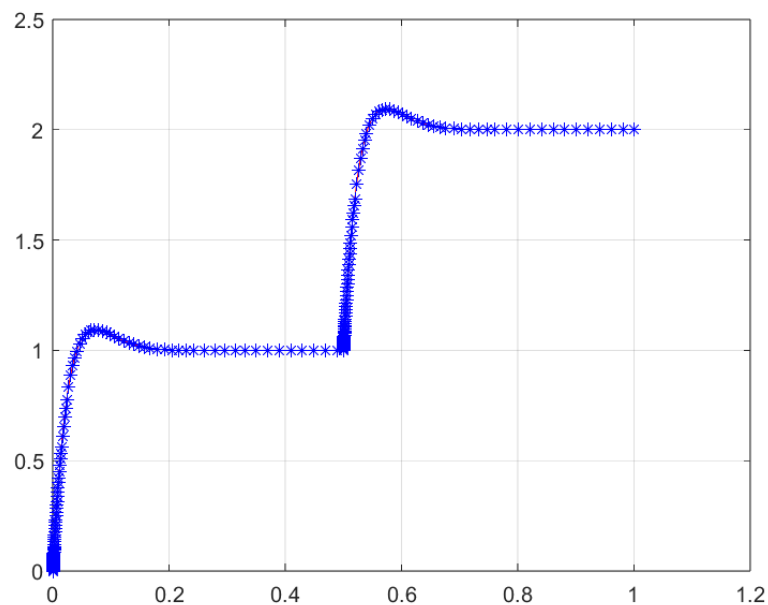


FIGURE 44 – Vérification de l'asservissement avec capteur (vitesse) *

On a en abscisse le temps en seconde et en ordonnée l'image de la vitesse en volt.

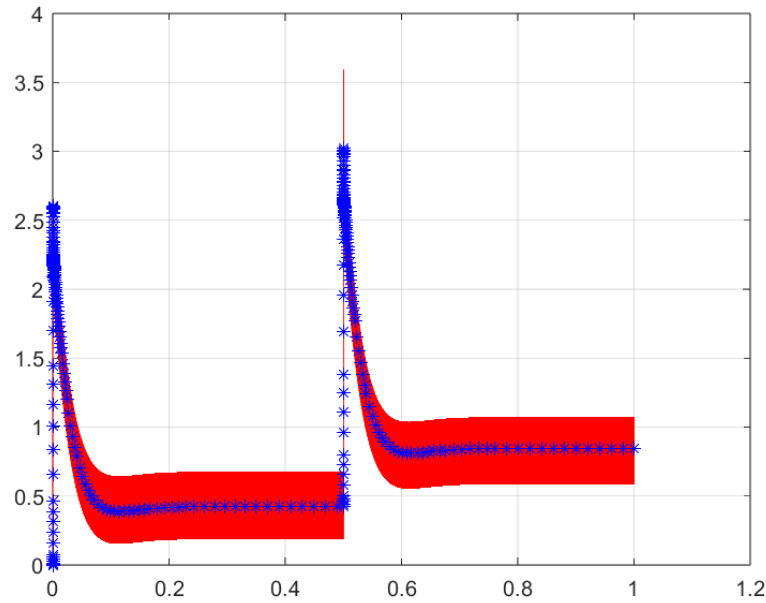


FIGURE 45 – Vérification de l’asservissement avec capteur (courant) *

On a en abscisse le temps en seconde et en ordonnée le courant en ampère (ou son image en volt).

On observe une concordance parfaite qui valide complètement notre cos correcteurs finaux.

6.iv Lien entre valeur de référence et valeur moyenne

Le but d’un asservissement est de faire tendre une grandeur caractéristique d’un système vers une valeur de référence (une consigne). Cependant, nous avons vu dans les études précédentes que les grandeurs que nous avons à asservir présentent des ondulations plus ou moins importantes. Il est donc légitime de se questionner sur le sens physique de la valeur de référence, étant donné que, la sortie n’étant pas constante, il est théoriquement impossible pour le système d’atteindre cette valeur. Dans notre cas, en considérant le fait que nous avons un intégrateur dans nos correcteurs, nous allons montrer que les valeurs de références Ω_{ref} et I_{Mref} sont égales aux valeurs moyennes $\langle \Omega \rangle$ et $\langle i_M \rangle$. Cela implique que c’est la valeur moyenne en sortie de notre système que nous allons asservir. La valeur de référence fixera la valeur moyenne de sortie du système. Nous allons démontrer cela pour le courant, et le cas de la vitesse sera purement analogue.

Soit y la variable d’état supplémentaire due à la présence d’un intégrateur dans notre système. On sait que :

$$y(t) = y(t_0) + \int_{t_0}^t (I_{Mref} - i_M) dt$$

Où t_0 est l’instant initial, c’est-à-dire le moment auquel nous avons mis en route le système.

Si l’on se place en régime permanent, on peut représenter la variation de courant comme suit :

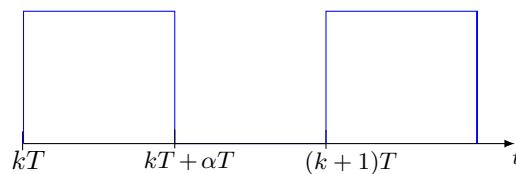


FIGURE 46 – Signaux liés au codeur incrémental

Où T représente une période de MLI du hacheur et α son rapport cyclique.

On peut en déduire que $i_M((k+1)T) = i_M(kT)$. De plus, on a :

$$\begin{cases} y((k+1)T) = y(t_0) + \int_{t_0}^{(k+1)T} (I_{Mref} - i_M) dt \\ y(kT) = y(t_0) + \int_{t_0}^{kT} (I_{Mref} - i_M) dt \end{cases}$$

Ainsi :

$$y((k+1)T) - y(kT) = \int_{kT}^{(k+1)T} (I_{Mref} - i_M) dt = T \cdot I_{Mref} - \int_{kT}^{(k+1)T} i_M dt = 0$$

Il est donc enfin possible d'obtenir le résultat final :

$$T \cdot I_{Mref} - \int_{kT}^{(k+1)T} i_M dt = 0 \Leftrightarrow I_{Mref} = \frac{1}{T} \int_{kT}^{(k+1)T} i_M dt = \langle i_M \rangle$$

7 Conception et dimensionnement des circuits électroniques

7.i Asservissements

7.i.a Correcteur de vitesse

7.i.b Correcteur de courant

7.i.c Comparateur

7.ii Circuit de mise en forme du signal tachymétrique

7.iii Circuit de mise en forme du signal du codeur incrémental

7.iv Limiteur de courant

7.v Limiteur de la tension de commande

7.vi Indicateur de sens de rotation

Arborescence des fichiers par figure :

Si le visualisateur de PDF utilisé est Adobe Acrobat, alors il est possible d'accéder au fichier directement en cliquant sur le lien. Autrement, il faudra suivre l'arborescence.

- Figure (4) : [donnees_moteur.m](#)
- Figure (5) : [/Modelisation et simulation du systeme en boucle ouverte/MATLAB/simu_moteur_a_vide_SIMULINK.slx](#)
- Figure (6) : [/Modelisation et simulation du systeme en boucle ouverte/PSIM/Moteur_a_vide_PSIM.psimsch](#)
- Figure (7) : [/Modelisation et simulation du systeme en boucle ouverte/PSIM/Moteur_a_vide_PSIM.psimsch](#)
- Figure (8) : [/Modelisation et simulation du systeme en boucle ouverte/MATLAB/Comparaison_courant_psim_MATLAB.m](#)
- Figure (9) : [/Modelisation et simulation du systeme en boucle ouverte/MATLAB/Comparaison_vitesse_psim_MATLAB.m](#)
- Figure (11) : [/Frottements et charges/PSIM/Simulation_frottement_et_charges.psimsch](#)
- Figure (12) : [/Frottements et charges/MATLAB/Simulation_frottement_et_charges.slx](#)
- Figure (13) : [/Frottements et charges/MATLAB/Comparaison_courant_et_vitesse_frottements_et_charges.m](#)
- Figure (14) : [/Frottements et charges/MATLAB/Comparaison_courant_et_vitesse_frottements_et_charges.m](#)
- Figure (15) : [/Frottements et charges/MATLAB/Comparaison_courant_et_vitesse_frottements_et_charges.m](#)
- Figure (16) : [/Frottements et charges/MATLAB/Comparaison_courant_et_vitesse_frottements_et_charges.m](#)
- Figure (19) : [/Systeme complet avec hacheur/PSIM/Systeme_complet_BO.psimsch](#)
- Figure (21) : [/Prise en compte de la commande/PSIM/Systeme_complet_BO.psimsch](#)
- Figure (22) : [/Prise en compte de la commande/MATLAB/Systeme_complet_avec_commande.slx](#)
- Figure (23) : [/Systeme complet avec hacheur/PSIM/Systeme_complet_BO.psimsch](#)
- Figure (24) : [/Systeme complet avec hacheur/PSIM/Systeme_complet_BO.psimsch](#)
- Figure (25) : [/Systeme complet avec hacheur/PSIM/Systeme_complet_BO.psimsch](#)
- Figure (26) : [/Systeme complet avec hacheur/PSIM/Systeme_complet_BO.psimsch](#)
- Figure (29) : [/Asservissement en courant/MATLAB/comparaison_FT_courant.slx](#)
- Figure (30) : [/Asservissement en courant/MATLAB/comparaison_FT_courant.slx](#)
- Figure (31) : [/Asservissement en courant/MATLAB/asserv_courant.slx](#)
- Figure (32) : [/Asservissement en courant/MATLAB/asserv_courant.slx](#)
- Figure (33) : [/Asservissement en courant/PSIM/Boucle_courant_asservie_hacheur.psimsch](#)
- Figure (34) : [/Asservissement en courant/PSIM/Boucle_courant_asservie_hacheur.psimsch](#)
- Figure (35) : [/Asservissement de vitesse/MATLAB/Fonction_de_transfert_courant_et_vitesse_MATLAB.m](#)
- Figure (36) : [/Asservissement de vitesse/MATLAB/Comparaisons_asservissement_de_vitesse_SIMULINK.slx](#)
- Figure (39) : [/Asservissement de vitesse/MATLAB/Comparaisons_asservissement_de_vitesse_SIMULINK.slx](#)
- Figure (40) : [/Asservissement de vitesse/PSIM/Systeme_complet_asservie_hacheur.psimsch](#)
- Figure (41) : [/Asservissement de vitesse/PSIM/Systeme_complet_asservie_hacheur.psimsch](#)