

Commande de la machine synchrone à aimants permanents par retour d'état

1. Introduction.....	18
2. Stabilisation des modèles flous TS avec une loi de commande PDC	18
2.1 Stabilité des modèles flous de Takagi-Sugeno.....	18
2.2 Loi de commande PDC.....	20
2.3 Introduction aux outils LMIs.....	21
2.3.1 Changement de variable.....	22
2.3.2 Lemme de Schur	22
2.3.3 S-procédure	22
2.4 Synthèse de loi de commande PDC.....	23
3. Application à la machine synchrone à aimants permanents	27
4. Suivi de trajectoire des modèles flous TS.....	30
4.1 Synthèse de loi de commande.....	31
4.2 Application à la machine synchrone à aimants permanents	33
5. Interprétation des résultats.....	42
6. Conclusion	43

1. Introduction

Depuis plusieurs années de nombreux travaux se sont intéressés à la stabilité et la stabilisation des systèmes non linéaires décrits par des modèles flous de Takagi-Sugeno. L'analyse de stabilité des modèles flous est souvent réalisée par la technique de Lyapunov qui impose des exigences restrictives et donne des conditions de stabilité suffisantes. La stabilité dépend de l'existence d'une matrice commune, symétrique et définie positive, qui garantit la stabilité de tous les modèles locaux. Ces conditions de stabilité peuvent être exprimées en utilisant des inégalités linéaires matricielles (LMI).

L'une des premières idées de stabilisation de ces modèles flous a consisté à utiliser des retours d'état linéaires. Ces derniers ont vite été supplantés par une loi de commande qui permet de prendre en compte les non linéarités des modèles flous et connues sous le nom de PDC (Parallel Distributed Compensation).

Dans ce chapitre, nous présentons quelques théorèmes donnant les conditions suffisantes de stabilité et stabilisation des modèles flous de Takagi-Sugeno. L'approche proposée tout au long de ce mémoire repose sur les fonctions de Lyapunov. Il s'agit de chercher une matrice symétrique et définie positive et sa fonction de Lyapunov associée telles que certaines conditions garantissent les propriétés de stabilité, ensuite, nous présentons, le problème de suivi de trajectoires et la stabilisation de la machine synchrone à aimants permanents, par une loi de commande de type PDC.

2. Stabilisation des modèles flous TS avec une loi de commande PDC (Parallel Distributed Compensation)

2.1 Stabilité des modèles flous de Takagi-Sugeno

L'étude de la stabilité des modèles flous de Takagi-Sugeno s'effectue principalement en utilisant la méthode directe de Lyapunov [21]. Cette méthode s'appuie sur le choix d'une fonction candidate de Lyapunov.

Définition : Une fonction candidate de Lyapunov V est une fonction scalaire définie positive satisfaisant :

- $\|x\| \rightarrow \infty \Rightarrow V(x) \rightarrow \infty$
- $\alpha(\|x\|) \leq V(x) \leq \beta(\|x\|)$

où $\alpha, \beta \in \mathcal{C}^1$ sont des fonctions définies positives.

Théorème 3.1 : Un modèle $\dot{x} = f(x)$ est globalement asymptotiquement stable et converge vers $\mathbf{0}$ s'il existe une fonction candidate de Lyapunov V telle que :

$$\lim_{x \rightarrow \infty} V(x(t)) = \mathbf{0}, \forall x(0) \in \mathbb{R}^n$$

La fonction candidate de Lyapunov la plus couramment utilisée est la fonction quadratique donnée par :

$$V(x(t)) = x^T P x(t), P = P^T > 0 \quad (3.1)$$

La stabilité dans ce cas est dite stabilité quadratique.

Soit le modèle flou de Takagi-Sugeno continu en régime libre, suivant :

$$\dot{x}(t) = \sum_{i=1}^r h_i(z(t)) A_i x(t) \quad (3.2)$$

La stabilité quadratique s'étudie en calculant la dérivée de la fonction $V(x)$

$$\frac{d}{dt} V(x(t)) = \dot{x}(t)^T P x(t) + x(t)^T P \dot{x}(t) \quad (3.3)$$

En utilisant (3.2) :

$$\frac{d}{dt} V(x(t)) = \left(\sum_{i=1}^r h_i(z(t)) A_i x(t) \right)^T P x(t) + x(t)^T P \left(\sum_{i=1}^r h_i(z(t)) A_i x(t) \right) \quad (3.4)$$

En posant :

$$A_z = \sum_{i=1}^r h_i(z(t)) A_i$$

On obtient :

$$\frac{d}{dt} V(x(t)) = \left(x(t)^T (A_z^T P + P A_z) x(t) \right) \quad (3.5)$$

Prouver la stabilité quadratique du modèle (3.2) revient à résoudre le problème suivant :

$$\text{Trouver une matrice } \mathbf{P} > 0 \text{ telle que } \mathbf{A}_z^T \mathbf{P} + \mathbf{P} \mathbf{A}_z < 0 \forall x(\mathbf{t}) \in \mathbb{R}^n$$

Théorème 3.2 [23] : S'il existe $\mathbf{P} > 0$ telle que :

$$\mathbf{A}_i^T \mathbf{P} + \mathbf{P} \mathbf{A}_i < 0, i = \{1..r\} \quad (3.6)$$

alors le système (3.2) est quadratiquement stable.

2.2 Loi de commande PDC (Parallel Distributed Compensation)

La loi de commande PDC est donnée par [24] :

$$\mathbf{u}(\mathbf{t}) = -\sum_{i=1}^r h_i(\mathbf{z}(\mathbf{t})) \mathbf{F}_i \mathbf{x}(\mathbf{t}) \quad (3.7)$$

Où \mathbf{F}_i est le gain de retour local relatif au $i^{\text{ème}}$ modèle. Le principe de cette méthode illustrée dans la figure 3.1, est d'élaborer une commande par retour d'état pour chaque modèle local, puis d'effectuer une interconnexion floue d'une manière similaire à la technique utilisée pour les modèles locaux. L'avantage majeur de cette loi de commande, est de respecter la même structure de découpage des non linéarités que celle utilisée pour l'obtention du modèle flou TS. Dans le cas où le modèle flou TS est obtenu par découpage exact, cette loi de commande est donc valable quelque soit le point de sous espace compact de l'espace d'états. La commande PDC, notamment la stabilisation quadratique, se base sur la fonction de Lyapunov pour montrer la convergence du modèle flou en boucle fermée, en l'occurrence déterminer les retours d'états correspondants à chaque modèle local composant le modèle flou TS. Ainsi, à partir d'une telle fonction, on peut trouver une matrice définie positive \mathbf{P} commune entre tous les modèles locaux en boucle fermée [25].

Pour obtenir le modèle flou en boucle fermée, on applique la commande PDC au modèle flou TS. Ainsi le modèle flou obtenu est le suivant :

$$\dot{x}(t) = \sum_{i=1}^r \sum_{j=1}^r h_i(z(t)) h_j(z(t)) G_{ij} x(t) \quad (3.8)$$

avec

$$G_{ij} = A_i - B_i K_j$$

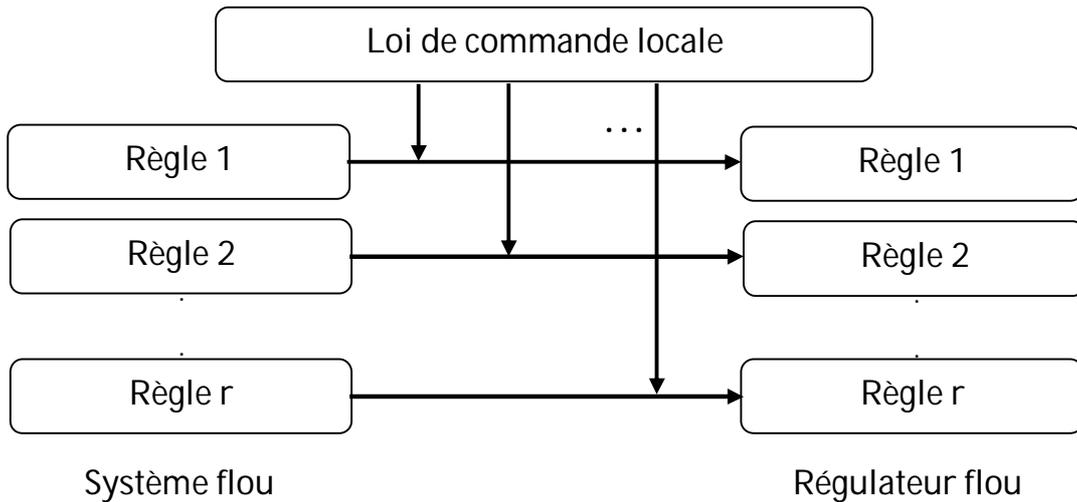


Figure 3.1 : Principe de la commande PDC

Les conditions de synthèse s'obtiennent en utilisant formellement les conditions d'analyse. Le problème ne se formule pas directement comme un problème des LMIs, mais s'appuie sur des contraintes de type BMIs (Bilinear Matrix Inequalities). C'est le cas de la stabilisation par retour d'état et par retour de sortie. Dans ce qui suit nous présentons quelques notions sur les LMIs.

2.3 Introduction aux outils LMIs

Un grand nombre de problèmes concernant la stabilité des systèmes peuvent se résoudre par l'intermédiaire de problèmes convexes d'un type particulier, les programmes semi définis (SDP). Ils sont aussi connus sous le nom de LMIs (Linear Matrix Inequalities) en automatique, du fait des conditions engendrées par la formulation du problème SDP.

L'intérêt principal des SDPs est le calcul d'un minimum global en temps polynomial (par opposition aux techniques d'optimisation stochastiques qui n'ont pas de limite de temps de résolution), en utilisant la méthode des points intérieurs [26].

Une contrainte LMI (Linear Matrix Inequality) s'écrit sous la forme [20] :

$$F \triangleq F_0 + \sum_{i=1}^m x_i F_i < 0 \text{ avec } F_i = F_i^T \in \mathbb{R}^m \quad (3.9)$$

$F(x)$ est une fonction affine en les variables de décision $x_i, i = \{1 \dots m\}$. L'ensemble des solutions $S = \{x \in \mathbb{R}^m, F(x) < 0\}$ est convexe. Un problème de faisabilité LMI s'écrit :

$$\text{Trouver } x_i, i = \{1 \dots m\} \text{ telle que } F(x) < 0$$

La résolution de ce problème peut se faire par des algorithmes performants comme l'algorithme du point intérieur. Pour pouvoir passer d'une inégalité non-linéaire ou une BMI (Bilinear Matrix Inequalities) à une LMI, on utilise les techniques suivantes :

2.3.1 Changement de variable : Cette transformation permet de rendre les équations linéaires en de nouvelles variables.

2.3.2 Lemme de Schur : Ce lemme permet de transformer des non linéarités convexes en des LMIs.

Lemme : Soient $Q = Q^T, R = R^T$ et S des matrices de tailles appropriées. La condition

$$\begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \geq 0$$

est équivalente à

$$R \geq 0, Q - SR^*S^T \geq 0, S(I - RR^*) = 0$$

où R^* dénote l'inverse de Moore-PenRose de R .

2.3.3 S-procédure : Ce lemme est une méthode permettant d'approximer un ensemble de contraintes quadratiques non convexes par une contrainte convexe.

Lemme : Soient $F_0, \dots, F_n, \forall i, F_i \geq 0$, des fonctions quadratiques de la variable $\zeta \in \mathbb{R}^n$

$$F_i(\zeta) \triangleq \zeta^T T_i \zeta + 2u_i^T \zeta + v_0, i = 0, \dots, p \quad (3.10)$$

où $T_i^T = T_i$. On considère la condition suivante sur F_0, \dots, F_n :

$$F_0(\zeta) \geq \mathbf{0} \text{ Pour tout } \zeta \text{ telle que } F_i(\zeta) \geq \mathbf{0}, i = 0, \dots, p$$

De manière évidente, S'il existe

$$\tau_1 \geq \mathbf{0}, \dots, \tau_p \geq \mathbf{0} \text{ telle que } \begin{bmatrix} T_0 & \mathbf{u}_0 \\ \mathbf{u}_0^T & \mathbf{v}_0 \end{bmatrix} - \sum_{i=1}^p \begin{bmatrix} T_i & \mathbf{u}_i \\ \mathbf{u}_i^T & \mathbf{v}_i \end{bmatrix} \geq \mathbf{0}$$

alors (3.10) est vérifiée. Quand $p = 1$, la réciproque est vraie, à condition qu'il existe $\zeta(\mathbf{0})$ vérifiant $F_1(\zeta_0) > \mathbf{0}$.

2.4. Synthèse de loi de commande PDC(Parallel Distributed Compensation)

Le premier résultat concernant la stabilisation des modèles TS par la commande PDC a été donné par Wang et al [24], [27], énoncé dans le théorème suivant :

Théorème 3.3 : L'équilibre du modèle flou continu donné par (3.8) est asymptotiquement stable, s'il existe une matrice $P = P^T$ telle que :

$$\begin{aligned} G_{ii}^T P + P G_{ii} &< \mathbf{0} \quad i = 1 \dots, r \quad (3.11) \\ \frac{(G_{ij} + G_{ji})^T}{2} P + P \frac{(G_{ij} + G_{ji})}{2} &\leq \mathbf{0} \quad i, j < j \end{aligned}$$

avec

$$h_i(z(t))h_j(z(t)) \neq \mathbf{0}, t > 0$$

$$G_{ij} = A_i - B_i F_j$$

Pour mettre au point le régulateur PDC, on se ramène donc à un problème de faisabilité des LMIs qui peut être résolu à l'aide des outils issus de l'optimisation convexe, c'est pour cela qu'on effectue les changements de variables bijectifs usuels suivants : $P = X^{-1}$ et $F_i = M_i X^{-1}$, d'où après congruence, on peut réécrire le théorème 3.3 sous forme des LMIs comme suivant :

L'équilibre du modèle flou continu donné par (3.8) est asymptotiquement stabilisé via la loi de commande PDC, s'il existe une matrice X et F_i telles que les LMIs suivantes soient vérifiées :

$$\begin{aligned}
& -XA_i^T + A_iX - M_i^T B_i^T + B_i M_i > 0, i = 1, 2 \dots r \quad (3.12) \\
& -XA_i^T - A_iX - XA_j^T - A_jX + M_j^T B_i^T + B_i M_j + M_j^T B_i^T + B_j M_i \geq 0, i < j
\end{aligned}$$

Exemple

Pour illustrer cette approche, considérons le problème de stabilisation d'un pendule inversé (figure 3.2). Les équations du mouvement du pendule sont [22] :

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{g \sin(x_1(t)) - amlx_2^2 \sin(2x_1)/2 - a \cos(x_1)u}{\frac{4}{3} - aml \cos^2(x_1)} \end{cases}$$

où $x_1(t)$ désigne l'angle du pendule et $x_2(t)$ sa vitesse angulaire, $g = 9,8 \text{ m/s}^2$ est la gravité, m est la masse du pendule, M est la masse du chariot, $2l$ est la longueur du pendule, et u est la force exercée sur le chariot et $a = 1 / (m + M)$.

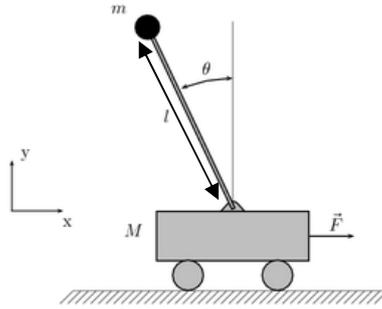


Figure 3.2 : Représentation du pendule inversé

Lorsque $x_1(t)$ est proche de zéro, les équations non-linéaires peuvent être simplifiées comme suit :

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{gx_1 - au(t)}{\frac{4l}{3} - aml} \end{cases}$$

Lorsque $x_1(t)$ est proche de $\mp\pi/2$, les équations non-linéaires peut être simplifiée comme suit :

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{gx_1/\pi - a\beta u(t)}{\frac{4l}{3} - aml\beta^2} \end{cases}$$

où $\beta = \cos(88^\circ)$

La dynamique du pendule peut être décrite par le modèle flou TS suivant :

régle 1 : Si x_1 est proche de 0 alors $\dot{x}(t) = A_1 x(t) + B_1 u$

régle 2 : Si x_1 est proche de $\mp\pi/2$ alors $\dot{x}(t) = A_2 x(t) + B_2 u$

avec

$$A_1 = \begin{bmatrix} 0 & 1 \\ \frac{4l}{3} - aml & 0 \end{bmatrix}, A_2 = \begin{bmatrix} 0 & 1 \\ \frac{2g}{\pi(\frac{4l}{3} - aml\beta^2)} & 0 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0 \\ a\beta \\ -\frac{4l}{3} - aml \end{bmatrix}, B_2 = \begin{bmatrix} 0 \\ a\beta \\ -\frac{4l}{3} - aml\beta^2 \end{bmatrix}$$

Pour stabiliser le pendule dans son point d'équilibre, on utilise une loi de commande PDC donnée par :

$$u(t) = - \sum_{i=1}^2 h_i(z(t)) F_i x(t)$$

La résolution des LMIs (3.12), nous a donné les résultats suivants :

$$F_1 = [-120.6667 \quad -22.6667]$$

$$F_2 = [-2551.6 \quad -764.0]$$

$$P = \begin{bmatrix} 3.6250 & 0.6250 \\ 0.6250 & 0.2812 \end{bmatrix}$$

Les figures 3.3, 3.4, et 3.5 montrent respectivement l'angle du pendule inversé, sa vitesse angulaire et la commande pour les conditions initiales $[15deg \ 0deg/s]^T$.

On remarque que le pendule converge vers son point d'équilibre.

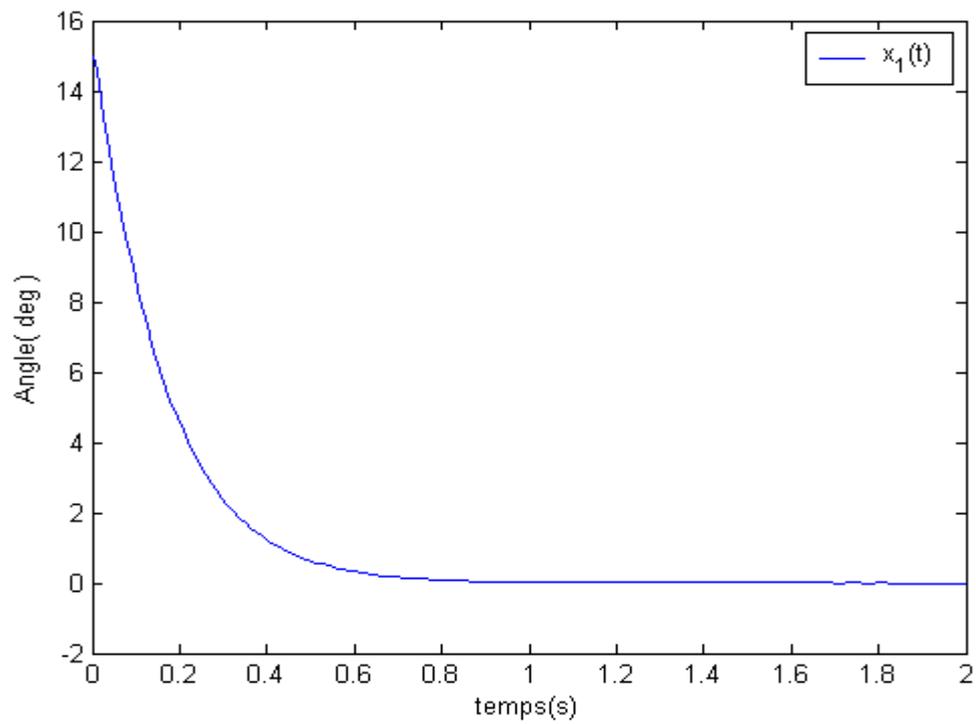


Figure 3.3 : Variation de l'angle du pendule

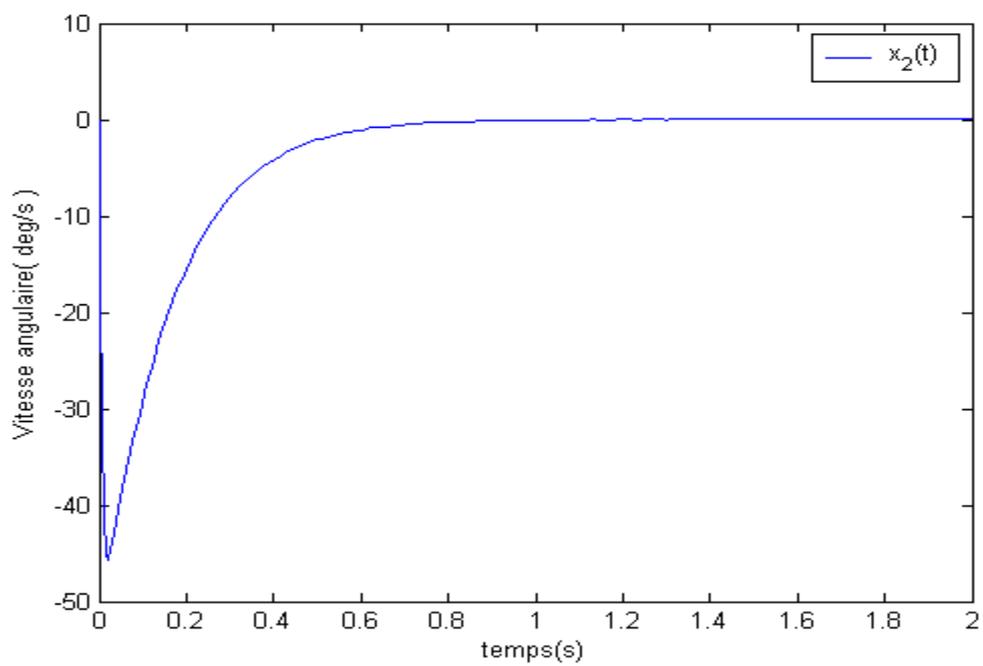


Figure 3.4 : Variation de la vitesse angulaire du pendule

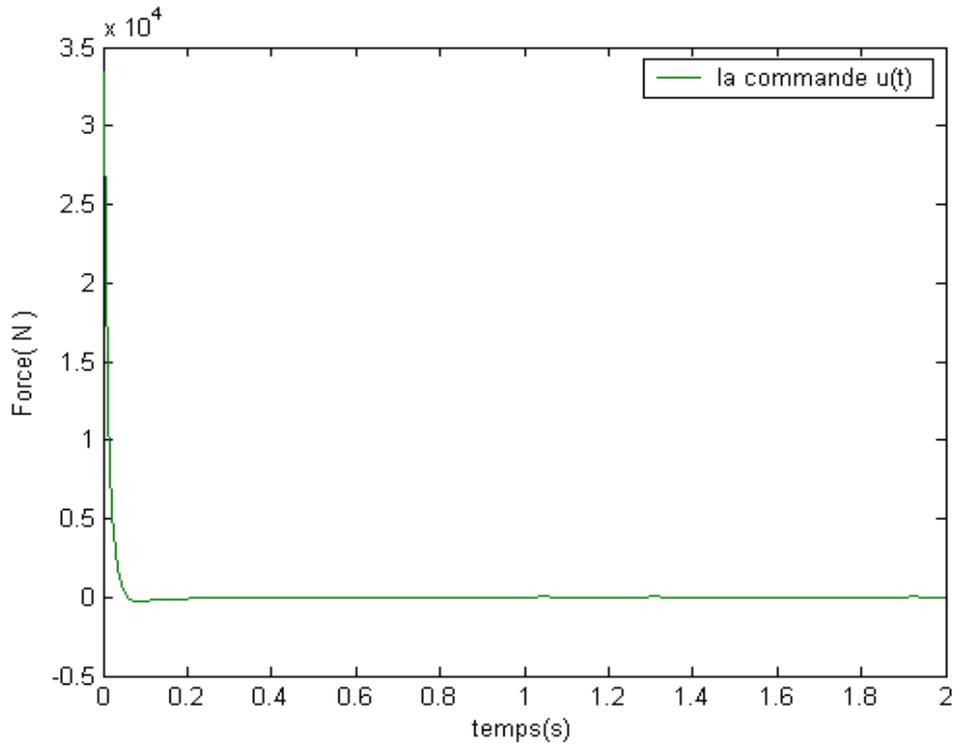


Figure 3.5 : Variation de la commande $u(t)$

3. Application à la machine synchrone à aimants permanents

Pour stabiliser la machine synchrone à aimants permanents décrite par le modèle flou à deux règles données au paragraphe (2.3), on utilise une loi de commande PDC donnée par :

$$u(t) = - \sum_{i=1}^2 h_i(\omega) F_i x(t)$$

où

$$h_1(\omega) = F_{11} \text{ et } h_2(\omega) = F_{12}$$

$$\omega \in [d, D]$$

$$F_{11} = \frac{\omega - d}{D - d} \text{ et } F_{12} = \frac{D - \omega}{D - d}$$

La résolution des LMIs (2.13), nous a donné les résultats suivants :

$$P = \begin{bmatrix} 3.1 & -10.8 & 0.4 \\ -10.8 & 1755.5 & -66.7 \\ 0.4 & -66.7 & 204.1 \end{bmatrix}$$

$$F_1 = \begin{bmatrix} -0.6160 & -2.5490 & 0.1030 \\ 0.0826 & -10.6374 & -3.8083 \end{bmatrix}$$

$$F_2 = \begin{bmatrix} -0.6182 & -2.1635 & 0.0884 \\ 0.0204 & -1.6938 & -4.1936 \end{bmatrix}$$

Les figures 3.6, 3.7, 3.8 et 3.9 montrent respectivement, la variation de la vitesse angulaire ω , la variation du courant i_q , la variation du courant i_d et la variation des tensions de commande u_q et u_d , pour les conditions initiales $[40\text{rd/s } 0.5\text{A } 0.5\text{A}]^T$.

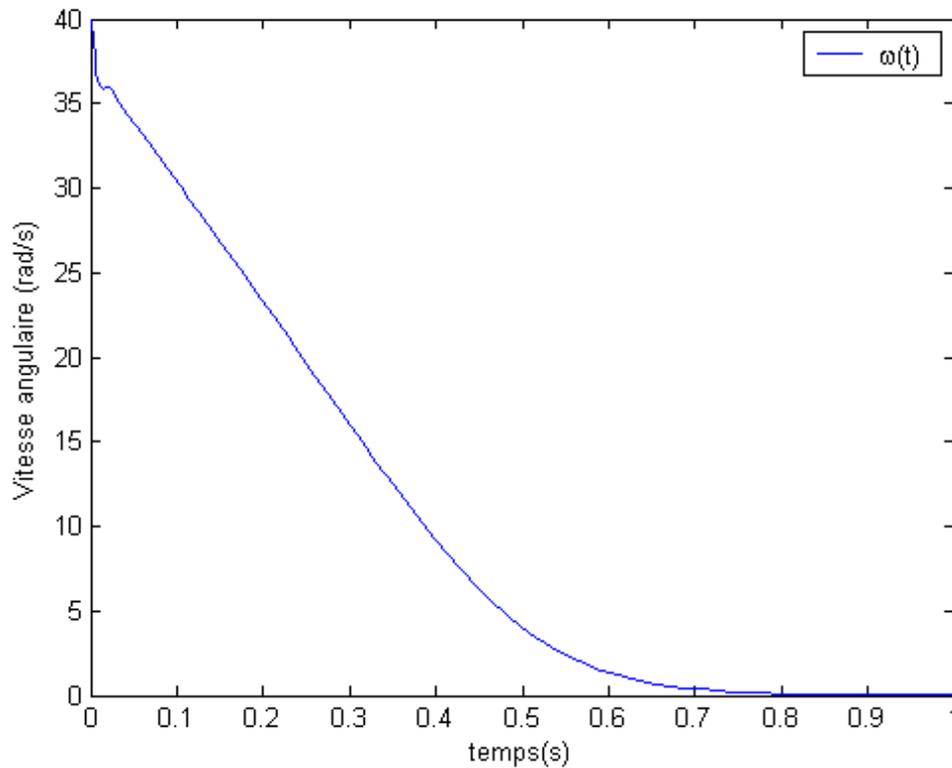


Figure 3.6 : Variation de la vitesse angulaire ω

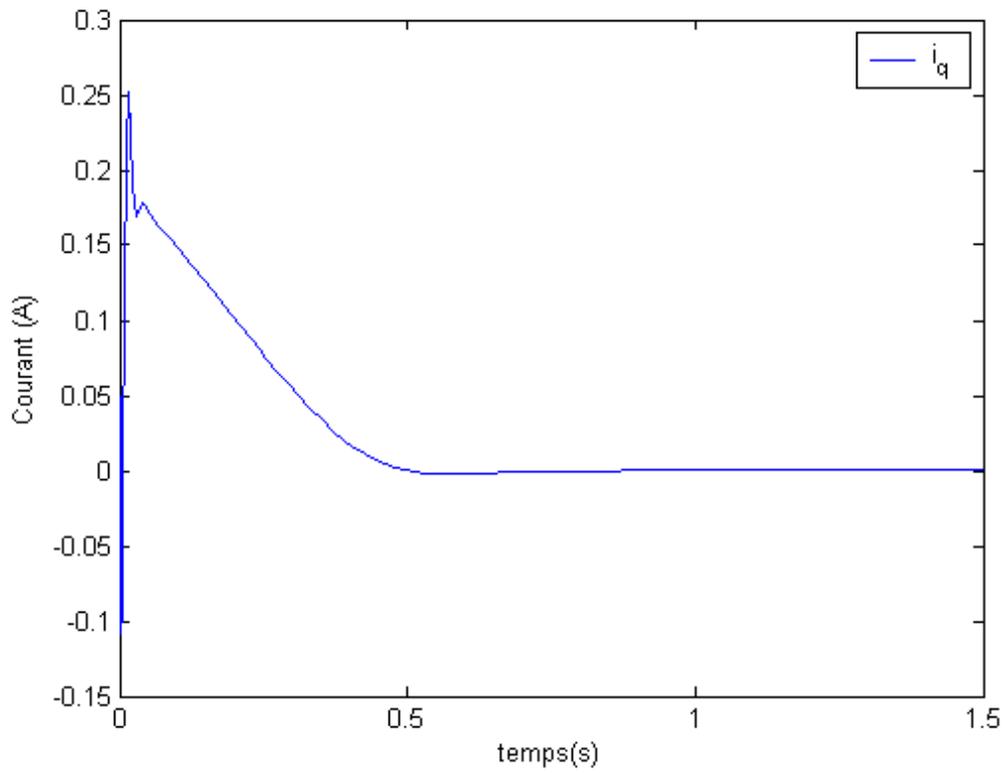


Figure 3.7 : Variation du courant i_q

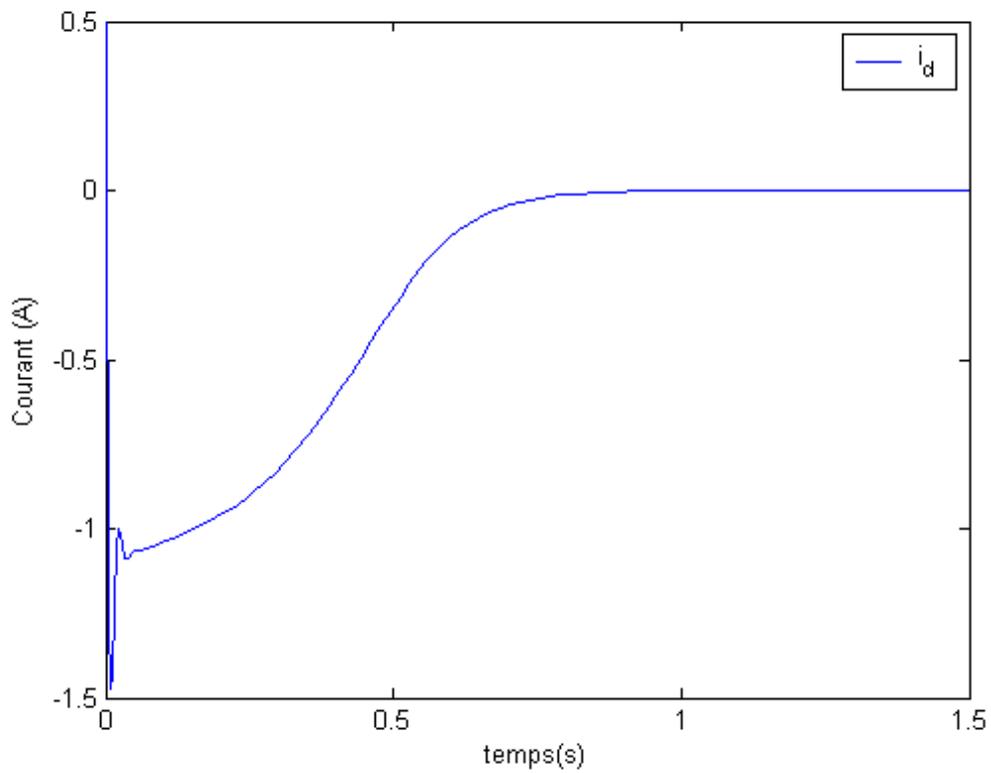


Figure 3.8 : Variation du courant i_d

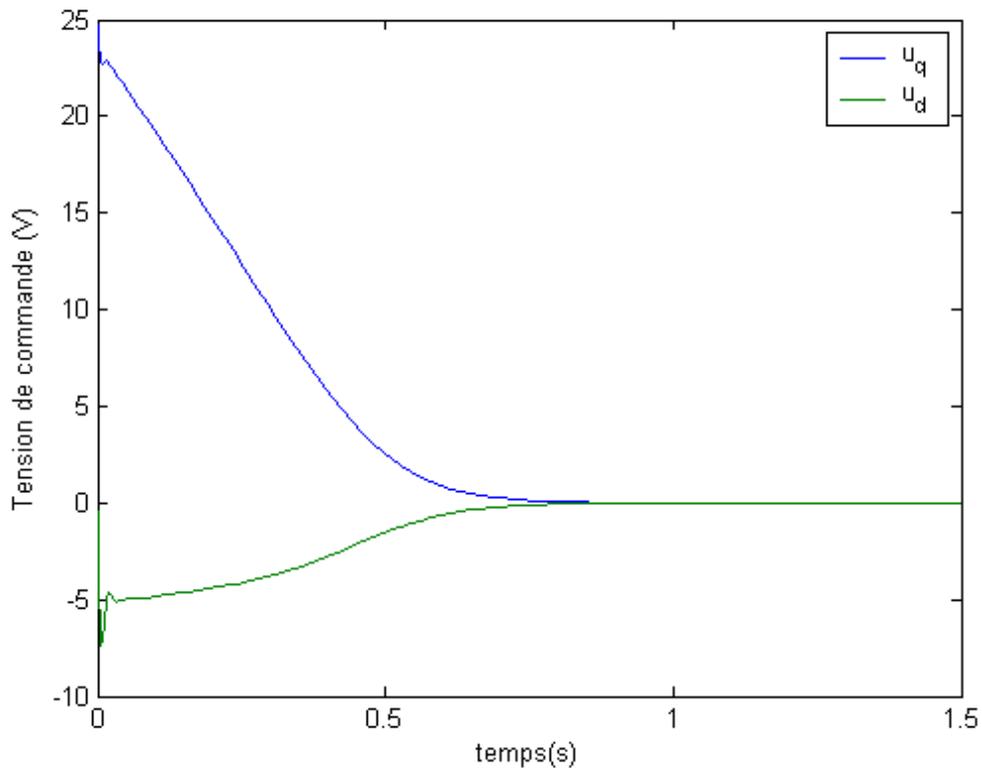


Figure 3.9 : Variation des tensions de commande

4. Suivi de trajectoire des modèles flous TS

Dans cette section, nous présentons la loi de commande qui garantit le suivi de trajectoire ainsi que l'analyse de la stabilité du suivi de trajectoire, nous supposons que tous les états sont mesurables.

Le problème de suivi de trajectoire peut être considéré comme une généralisation du problème de stabilisation.

Considérons le système décrit par [28] :

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t)) + \mathbf{g}(\mathbf{x}(t))\mathbf{u}(t) \\ \mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t)) \\ \bar{\mathbf{y}} = \boldsymbol{\varphi}(\mathbf{x}(t)) \end{cases} \quad (1.13)$$

où $\mathbf{x} \in \mathbf{R}^n$ est le vecteur d'état, $\mathbf{y}, \bar{\mathbf{y}} \in \mathbf{R}^m$ sont respectivement les vecteurs de sorties mesurées et contrôlées, $\mathbf{u} \in \mathbf{R}^m$ est le vecteur d'entrée, $\mathbf{f}(\mathbf{x}), \mathbf{g}(\mathbf{x}), \mathbf{h}(\mathbf{x})$ et $\boldsymbol{\varphi}(\mathbf{x})$ sont des fonctions non-linéaires avec des dimensions appropriées.

Le système non-linéaire (3.13) peut être exprimé par le modèle flou suivant :

Règles i : Si $z_1(t)$ est F_{1i} et ... $z_g(t)$ est F_{gi} alors

$$\dot{x} = A_i x(t) + B_i u(t), i = 1, 2, \dots, r$$

où $z_1(t) \sim z_g(t)$ sont les variables prémisses F_{ij} ($j = 1, 2, \dots, g$) sont les ensembles flous, r est le nombre de règles floues, A_i ($i = 1, 2, \dots, r$) sont les matrices d'état, B_i ($i = 1, 2, \dots, r$) sont les matrices de commande.

4.1 Synthèse de loi de commande

L'objectif d'un contrôleur flou qui garanti le suivi de trajectoire est de satisfaire la condition suivante :

$$\bar{y}(t) - r(t) \rightarrow 0, t \rightarrow \infty$$

où $r(t)$ représente la trajectoire désirée ou le signal de référence. Afin de convertir le problème de suivi de trajectoire en un problème de stabilisation, on introduit un ensemble de variables virtuelles désirées x_d qui doivent être suivies par les variables d'états x telle que :

$$\bar{y}(t) = \varphi(x)$$

$$r(t) = \varphi(x_d)$$

En posant $\tilde{x}(t) = x(t) - x_d(t)$ qui désigne l'erreur de poursuite, la dérivée de $\tilde{x}(t)$ s'écrit comme suit :

$$\dot{\tilde{x}}(t) = \dot{x}(t) - \dot{x}_d(t)$$

$$\begin{aligned} \dot{\tilde{x}}(t) &= \sum_{i=1}^r h_i(z(t)) \{A_i x(t) + B_i u(t)\} - \dot{x}_d(t) \\ \dot{\tilde{x}}(t) &= \sum_{i=1}^r h_i(z(t)) A_i x(t) - \sum_{i=1}^r h_i(z(t)) A_i x_d(t) + \sum_{i=1}^r h_i(z(t)) B_i u(t) \\ &\quad + \sum_{i=1}^r h_i(z(t)) A_i x_d(t) - \dot{x}_d(t) \end{aligned}$$

En posant :

$$\sum_{i=1}^r \mathbf{h}_i(\mathbf{z}(t)) \mathbf{B}_i \boldsymbol{\tau} = \sum_{i=1}^r \mathbf{h}_i(\mathbf{z}(t)) \mathbf{B}_i \mathbf{u}(t) + \sum_{i=1}^r \mathbf{h}_i(\mathbf{z}(t)) \mathbf{A}_i \mathbf{x}_d(t) - \dot{\mathbf{x}}_d(t) \quad (3.14)$$

On peut écrire l'erreur de poursuite comme suit :

$$\dot{\tilde{\mathbf{x}}}(t) = \sum_{i=1}^r \mathbf{h}_i(\mathbf{z}(t)) \{ \mathbf{A}_i \tilde{\mathbf{x}}(t) + \mathbf{B}_i \boldsymbol{\tau}(t) \} \quad (3.15)$$

Selon la description ci-dessus, on constate que le problème de suivi de trajectoire est similaire au problème de la stabilisation de l'erreur de suivi $\tilde{\mathbf{x}}$. Dans ce qui suit notre but est d'obtenir un contrôleur pour rendre le nouvel état $\tilde{\mathbf{x}}(t) = \mathbf{0}$. Alors on cherche à déterminer le nouveau contrôleur flou.

Règle du contrôleur :

Règle i : Si $\mathbf{z}_1(t)$ est \mathbf{F}_{1i} et ... $\mathbf{z}_g(t)$ est \mathbf{F}_{gi} alors

$$\boldsymbol{\tau}(t) = -\mathbf{K}_i \tilde{\mathbf{x}}(t), i = 1, 2, \dots, r$$

La sortie inférée par le contrôleur PDC est déterminée par la somme suivante :

$$\boldsymbol{\tau}(t) = - \sum_{i=1}^r \mathbf{h}_i(\mathbf{z}(t)) \mathbf{F}_i \tilde{\mathbf{x}}(t) \quad (3.16)$$

En utilisant (3.14) et (3.15), on obtient le système en boucle fermée suivant :

$$\dot{\tilde{\mathbf{x}}}(t) = \sum_{i=1}^r \sum_{j=1}^r \mathbf{h}_i(\mathbf{z}(t)) \mathbf{h}_j(\mathbf{z}(t)) \mathbf{G}_{ij} \tilde{\mathbf{x}}(t) \quad (3.17)$$

avec

$$\mathbf{G}_{ij} = \mathbf{A}_i - \mathbf{B}_i \mathbf{F}_j$$

Remarque : Les conditions de stabilité pour le suivi de trajectoire sont les mêmes que le problème de stabilisation des modèles flous TS par une loi de commande type PDC. Cela signifie que les gains de retour d'état \mathbf{F}_i peuvent être obtenus en faisant directement la résolution du problème de stabilisation.

Il nous reste alors qu'à déterminer les variables désirées $\mathbf{x}_d(t)$ et obtenir ensuite la loi de commande $\mathbf{u}(t)$. Pour atteindre cet objectif, on utilise l'expression suivante [29] :

$$\mathbf{g}(\mathbf{x}) = \sum_{i=1}^r \mathbf{h}_i(\mathbf{z}(t)) \mathbf{B}_i$$

et on réécrit (3.14) comme suit :

$$\mathbf{g}(\mathbf{x})(\mathbf{u}(t) - \boldsymbol{\tau}(t)) = -\mathbf{A}(\mathbf{x}) \mathbf{x}_d(t) + \dot{\mathbf{x}}_d(t) \quad (3.18)$$

où

$$A(x) = \sum_{i=1}^r h_i(z(t)) A_i$$

L'existence de la commande $u(t)$ dépend de la forme de $g(x)$. La matrice d'entrée $g(x)$ est supposée de plein rang colonne. On commence par les décompositions suivantes :

$$g(x) = \begin{bmatrix} \mathbf{0}_{n-m} \\ \mathbf{B}(x) \end{bmatrix}, A(x) = \begin{bmatrix} A_{n-m} \\ A_m(x) \end{bmatrix}, x_d(x) = \begin{bmatrix} x_{d_{n-m}} \\ x_{d_m}(x) \end{bmatrix} \quad (3.19)$$

Où $\mathbf{0}_{n-m} \in \mathbf{R}^{(n-m) \times m}$ est une matrice nulle et $\mathbf{B}(x) \in \mathbf{R}^{m \times m}$.

Par conséquent, l'état désiré est donné par la formule suivant :

$$\begin{bmatrix} \mathbf{0}_{n-m} \\ \mathbf{B}(x)(u - \tau) \end{bmatrix} = \begin{bmatrix} \dot{x}_d(t)_{n-m} - A(x)_{n-m} x_d(t) \\ \dot{x}_d(t)_m - A_m(x) x_d(t) \end{bmatrix} \quad (3.20)$$

Les variables désirées sont déterminées selon les contraintes suivantes :

$$r(t) = \varphi(x_d) \quad (3.21)$$

$$\dot{x}(t)_{n-m} = A_{n-m}(x) x_d(t) \quad (3.22)$$

De (3.20) on peut déduire la commande $u(t)$ comme suit :

$$u = -\sum_{i=1}^r h_i(z(t)) F_i \tilde{x} + \mathbf{B}^{-1}(x) [\dot{x}_d(t)_m A_m(x) x_d] \quad (3.23)$$

4.2 Application à la machine synchrone à aimants permanents

Les variables désirées x_d sont nécessaires pour satisfaire (3.18), qui est réécrite ci-dessous :

$$g(x)(u(t) - \tau(t)) = -A(x)x_d(t) + \dot{x}_d(t)$$

On obtient alors la forme matricielle suivante :

$$\begin{bmatrix} 0 & 0 \\ 1 & 0 \\ L_s & 0 \\ 0 & 1 \\ & L_s \end{bmatrix} (\mathbf{u}(t) - \boldsymbol{\tau}(t)) = - \begin{bmatrix} -\frac{B}{J} & \frac{3p\phi_v}{2J} & 0 \\ p\phi_v & -\frac{R_s}{L_s} & -p\omega \\ 0 & p\omega & \frac{R_s}{L_s} \end{bmatrix} \begin{bmatrix} \omega_d \\ i_{qd} \\ i_{dd} \end{bmatrix} + \begin{bmatrix} \dot{\omega}_d \\ \dot{i}_{qd} \\ \dot{i}_{dd} \end{bmatrix}$$

Le second indice d désigne les états désirés. Selon la première équation, on obtient :

$$\dot{\omega}_d = -\frac{B}{J}\omega_d + \frac{3p\phi_v}{2J}i_{qd}$$

Ce qui montre que : $i_{qd} = (\dot{\omega}_d + \frac{B}{J}\omega_d) \frac{2J}{3p\phi_v}$

$$\dot{i}_{qd} = (\ddot{\omega}_d + \frac{B}{J}\dot{\omega}_d) \frac{2J}{3p\phi_v}$$

Puis, à partir de la deuxième et la troisième équation, nous pouvons obtenir la loi de commande suivante :

$$\mathbf{u}_q = p\phi_v\omega_d + R_s i_{qd} + L_s \frac{d}{dt}(i_{qd}) + L_s p\omega i_{dd} + \tau_q$$

$$\mathbf{u}_d = -pL_s\omega i_{qd} + R_s i_{dd} + L_s \frac{d}{dt}(i_{dd}) + \tau_d$$

Puisque le rotor est un aimant permanent alors $i_{dd} = 0$.

$$\mathbf{u}_q = p\phi_v\omega_d + R_s i_{qd} + L_s \dot{i}_{qd} + \tau_q$$

$$\mathbf{u}_d = -pL_s\omega i_{qd} + \tau_d$$

où τ_q, τ_d sont les nouvelles commandes. La commande $\boldsymbol{\tau}$ est donnée par :

$$\boldsymbol{\tau} = - \sum_{i=1}^2 \mathbf{h}_i F_i(\mathbf{x} - \mathbf{x}_d)$$

où $\boldsymbol{\tau} = [\tau_q \ \tau_d]^T$

$$\mathbf{x} = [\omega \ i_q \ i_d]^T$$

$$\mathbf{x}_d = [\omega_d \ i_{qd} \ i_{dd}]^T$$

La simulation du système, pour les conditions initiales $[\mathbf{0rd/s}, \mathbf{0A}, \mathbf{0A}]^T$ et la vitesse désirée $\omega_r(t) = 25 \sin t + 25$, nous a donné les résultats schématisés par les figures

3.10, 3.11, 3.12, 3.13 et 3.14 qui montrent respectivement la variation des vitesses angulaires ω_d et ω , la variation des courants i_{qd} et i_q , la variation des courants i_{dd} et i_d , et la variation des tensions de commande u_q et u_d .

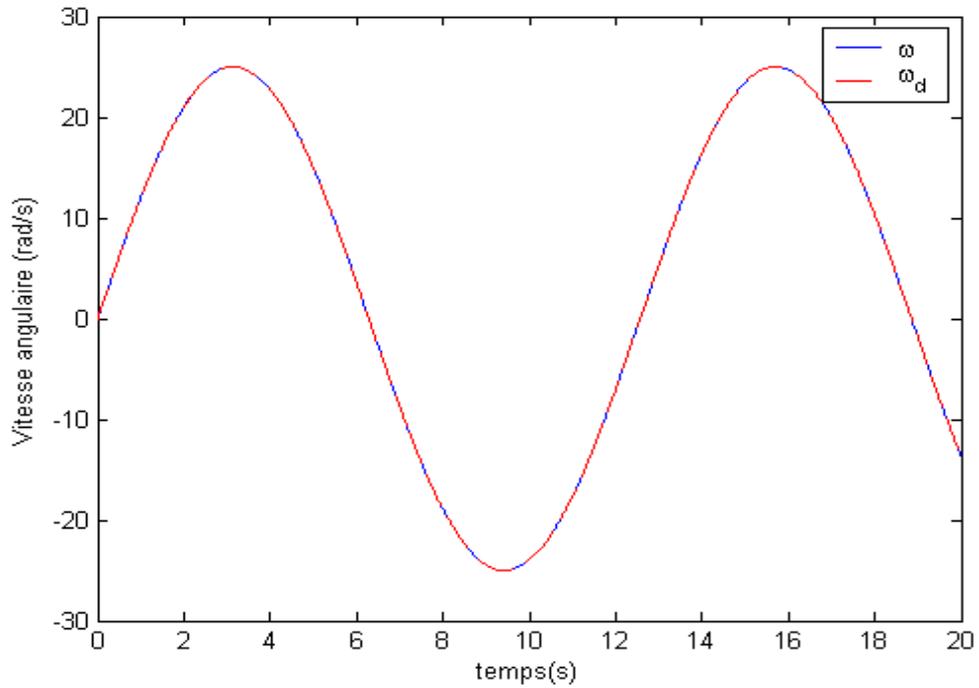


Figure 3.10 : Variation des vitesses angulaires ω_d et ω

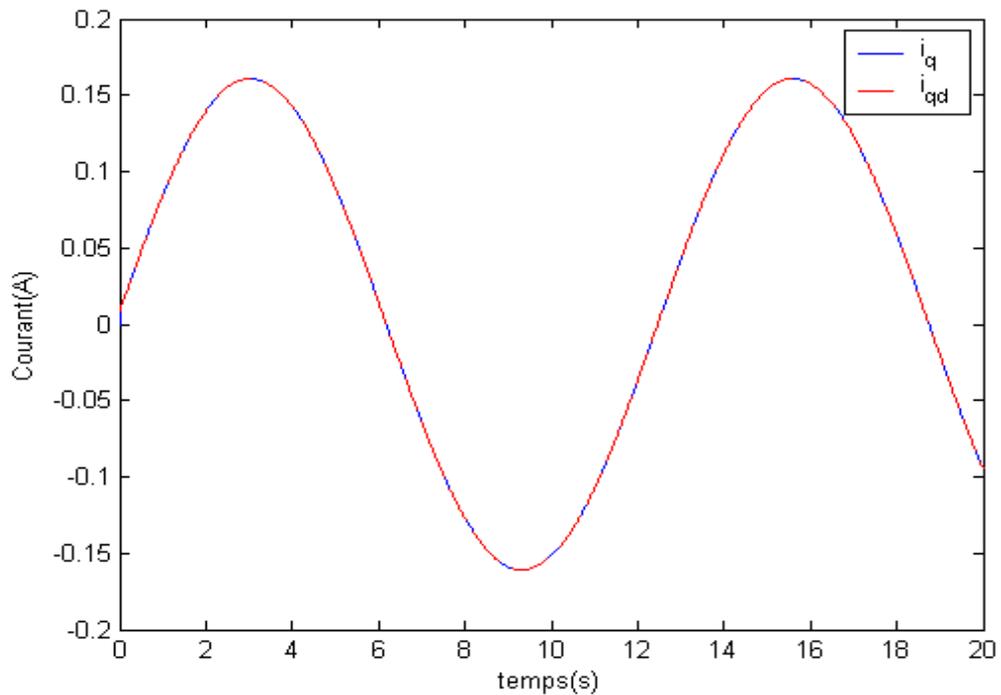


Figure 3.11 : Variation des courants i_{qd} et i_q

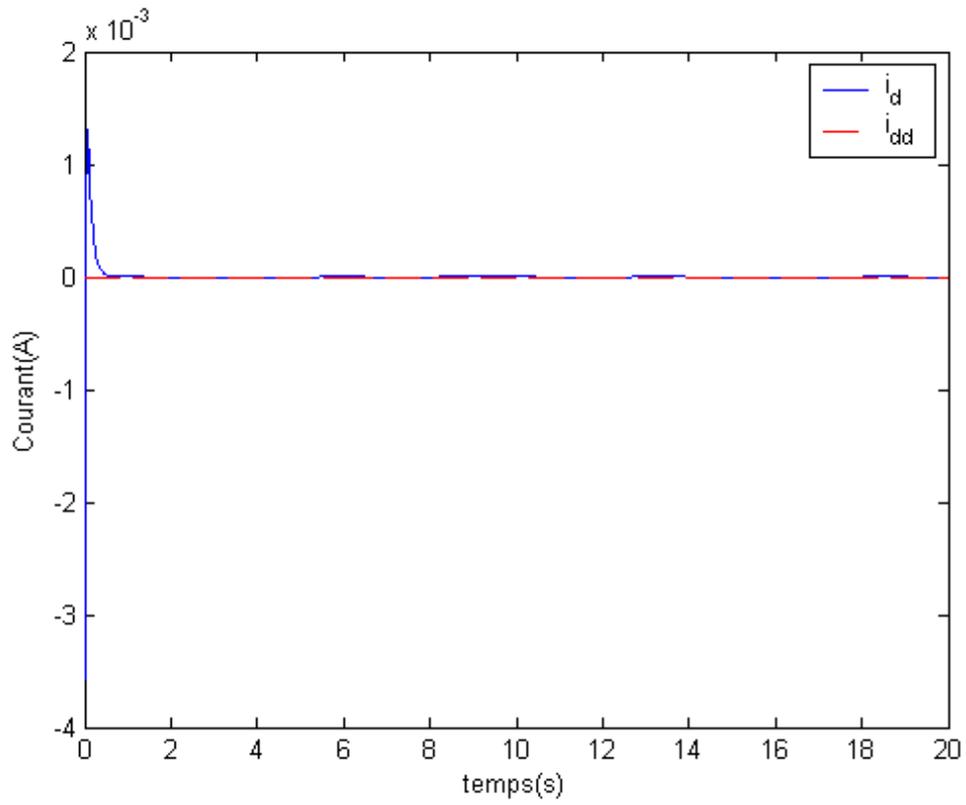


Figure 3.12 : Variation des courants i_{dd} et i_d

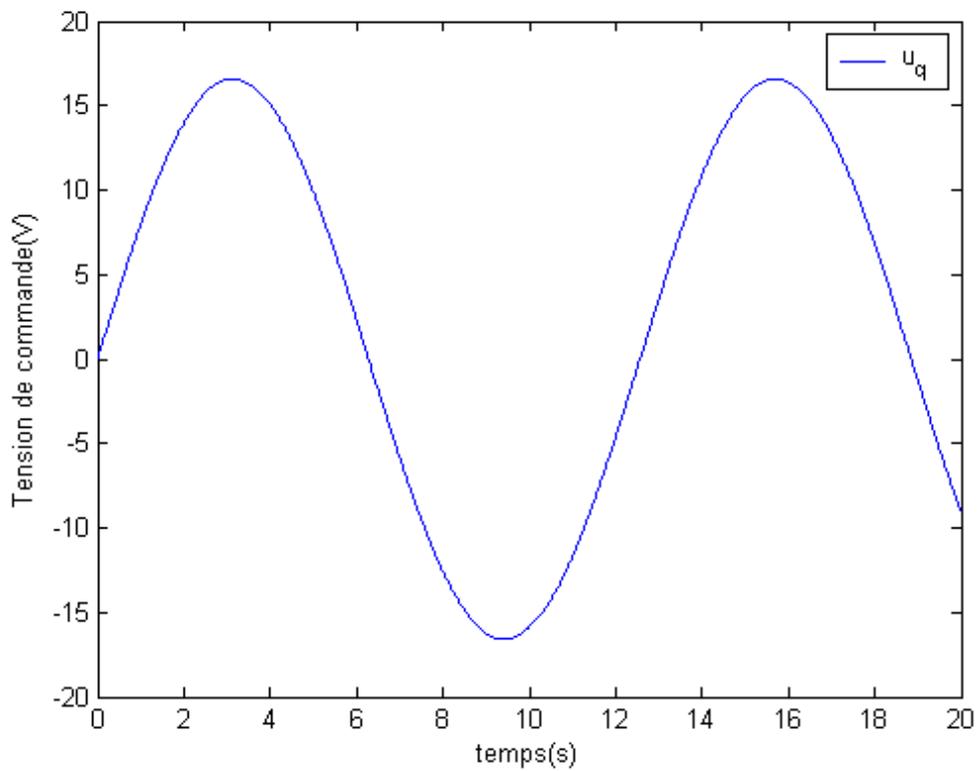


Figure 3.13 : Variation de la commande u_q

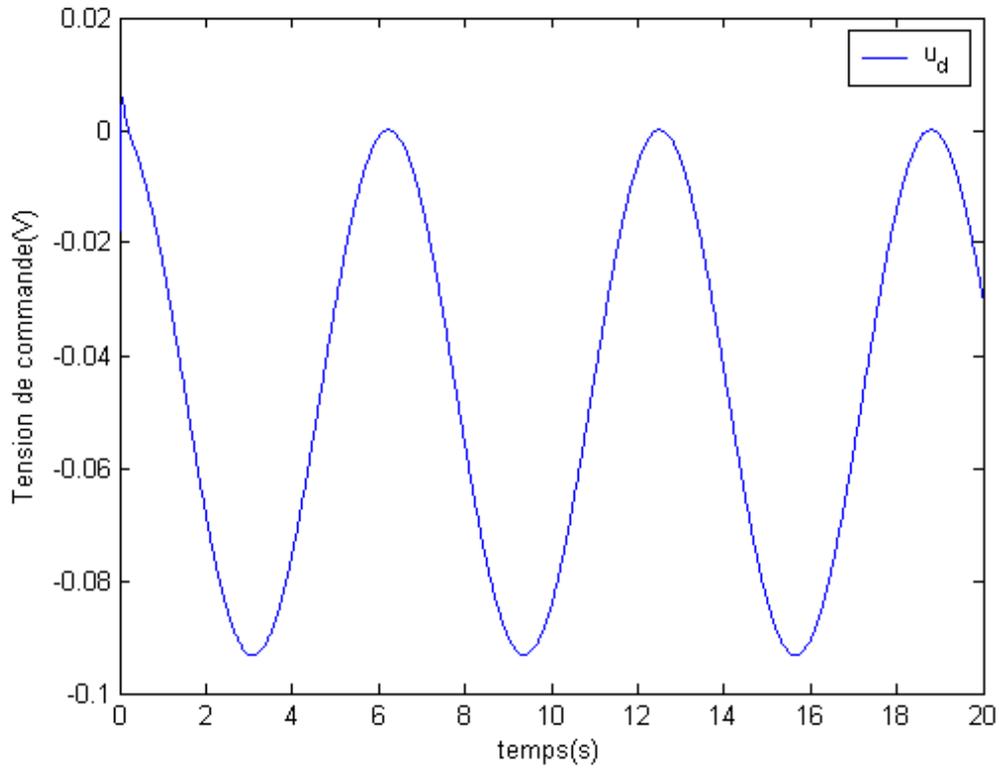


Figure 3.14 : Variation de la commande u_d

La simulation du système, pour les conditions initiales $[0 \text{ rad/s}, 0\text{A}, 0\text{A}]^T$ et la vitesse désirée $\omega_d = 50 \text{ rad/s}$, nous a donné les résultats schématisés par les figures 3.15, 3.16, 3.17 et 3.18 qui montrent respectivement la variation des vitesses angulaires ω_d et ω , la variation des courants i_{qd} et i_q , la variation des courants i_{dd} et i_d , et la variation des tensions de commande u_q et u_d .

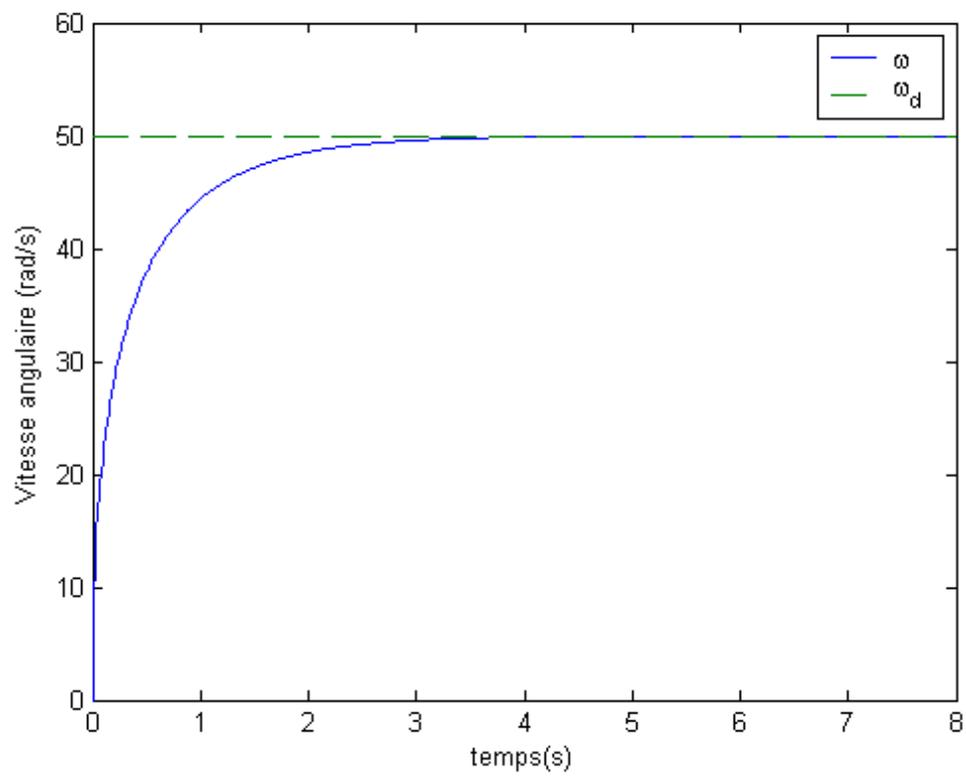


Figure 3.15 : Variation des vitesses angulaires ω et ω_d

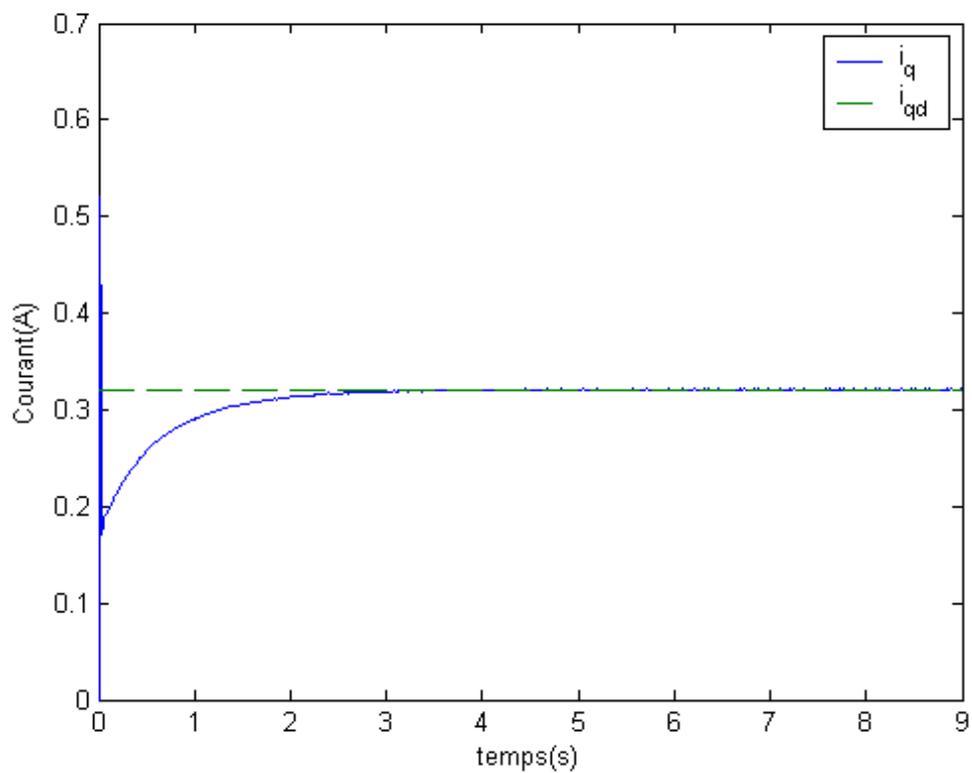


Figure 3.16 : Variation des courants i_q et i_{qd}

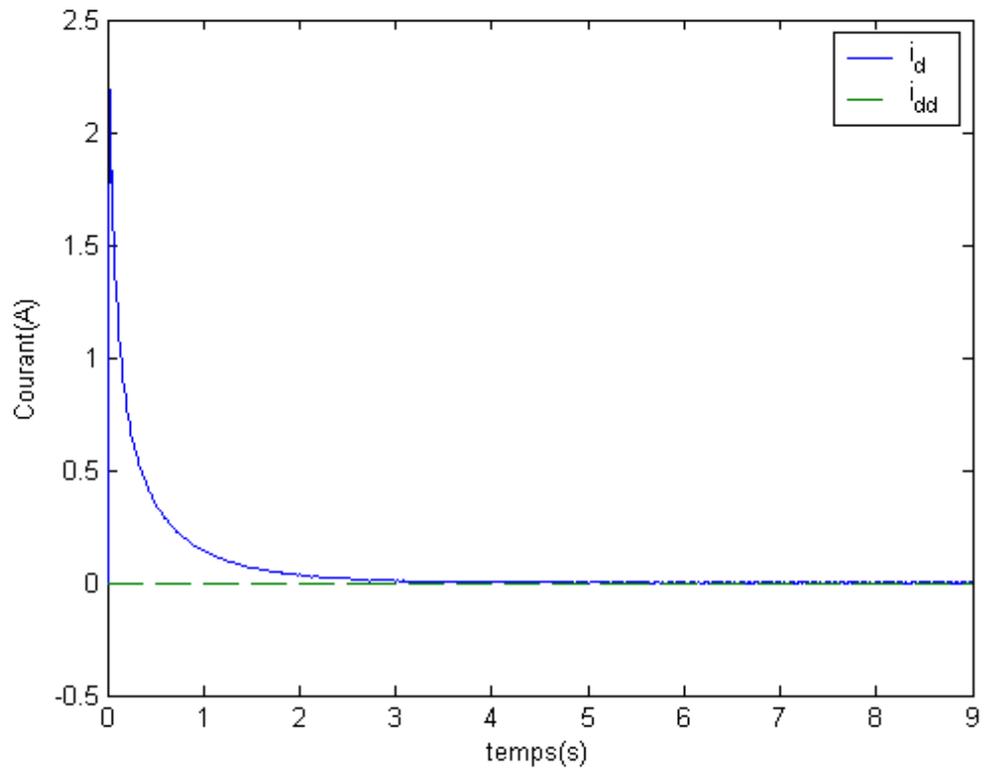


Figure 3.17 : Variation des courants i_d et i_{dd}

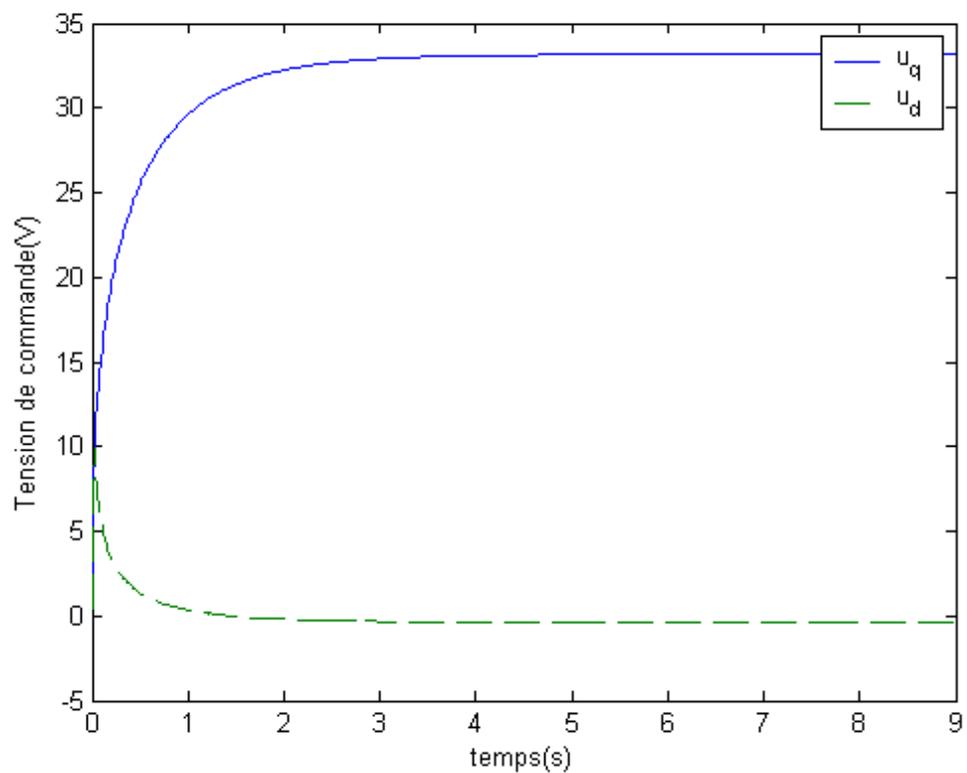


Figure 3.18 : Variation des tensions de commande u_q et u_d

La simulation du système, pour les conditions initiales $[15 \text{ rad/s}, 0.2 \text{ A}, 0.5 \text{ A}]^T$ et la vitesse désirée $\omega_d(t) = \frac{50}{3}t$, pour $t < 3$ et $\omega_d(t) = 50 \text{ rad/s}$ pour $t \geq 3$, nous a donné les résultats schématisés par les figures 3.19, 3.20, 3.21 et 3.22 qui montrent respectivement la variation des vitesses angulaires ω_d et ω , la variation des courants i_{qd} et i_q , la variation des courants i_{dd} et i_d , et la variation des tensions de commande u_q et u_d .

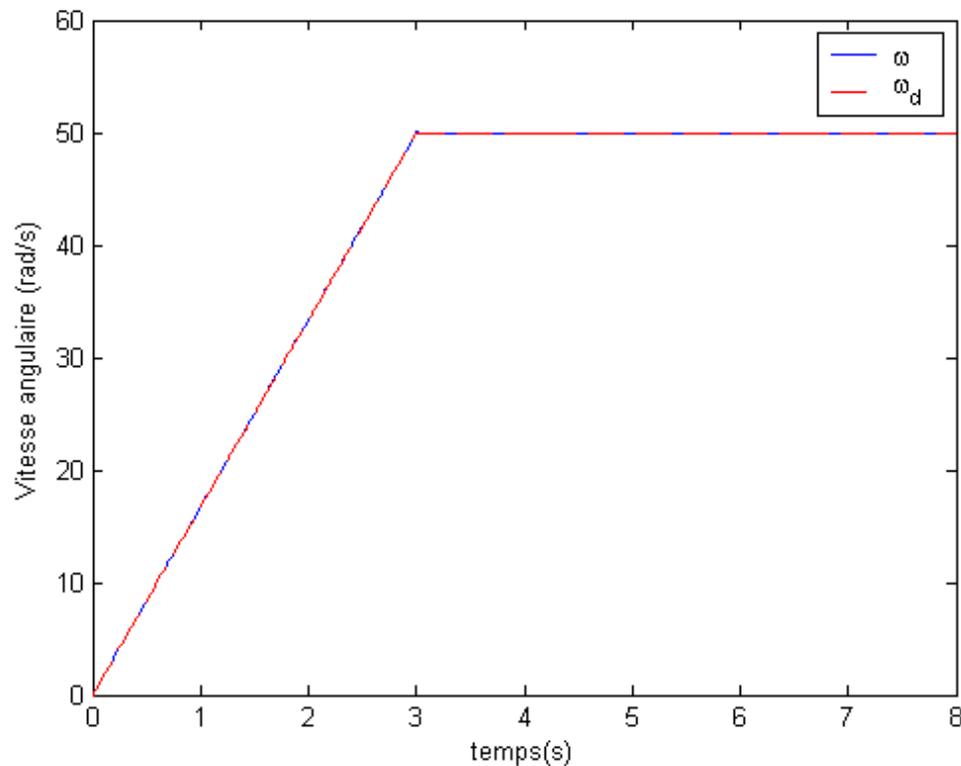


Figure 3.19 : Variation des vitesses angulaires ω et ω_d

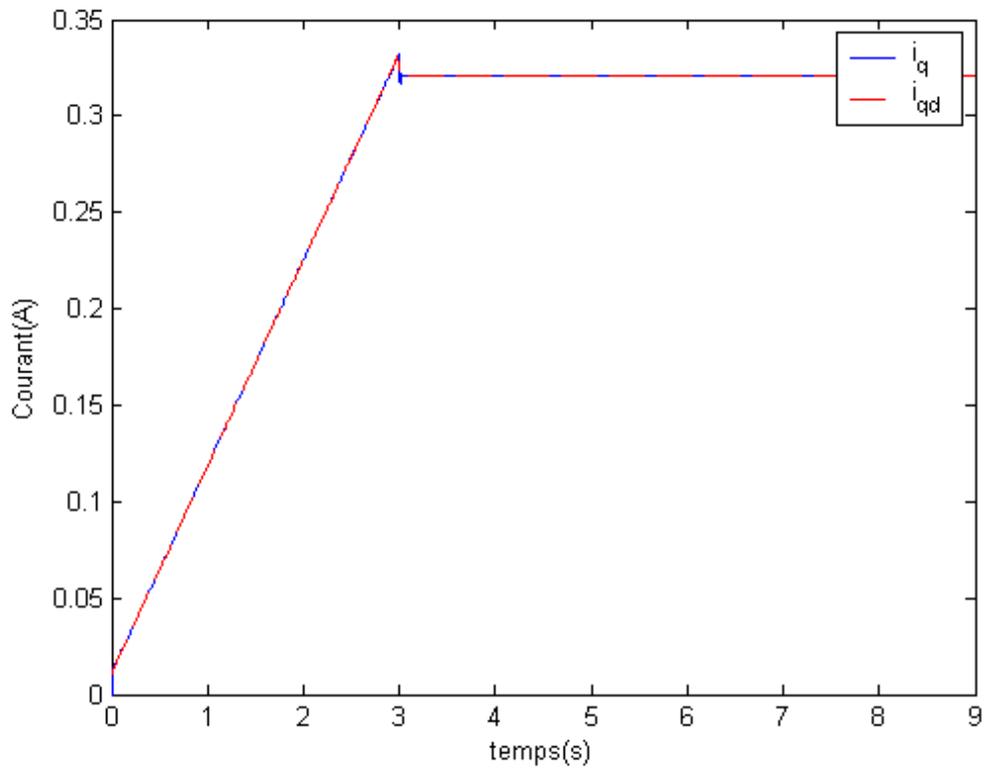


Figure 3.20 : Variation des courants i_q et i_{qd}

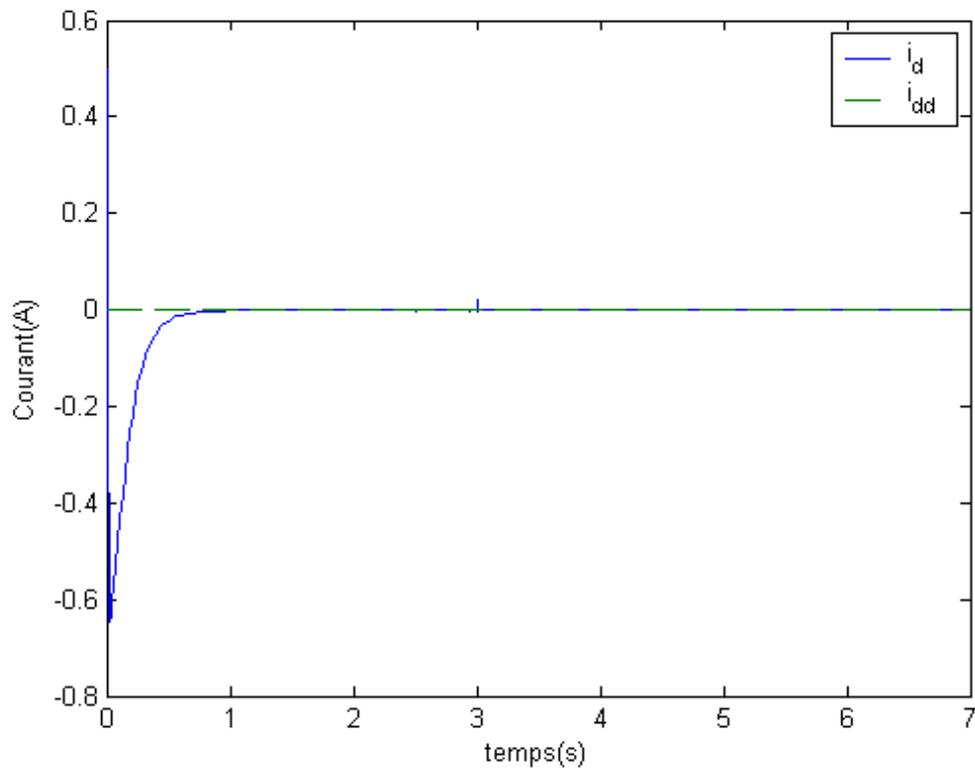


Figure 3.21 : Variation des courants i_d et i_{dd}

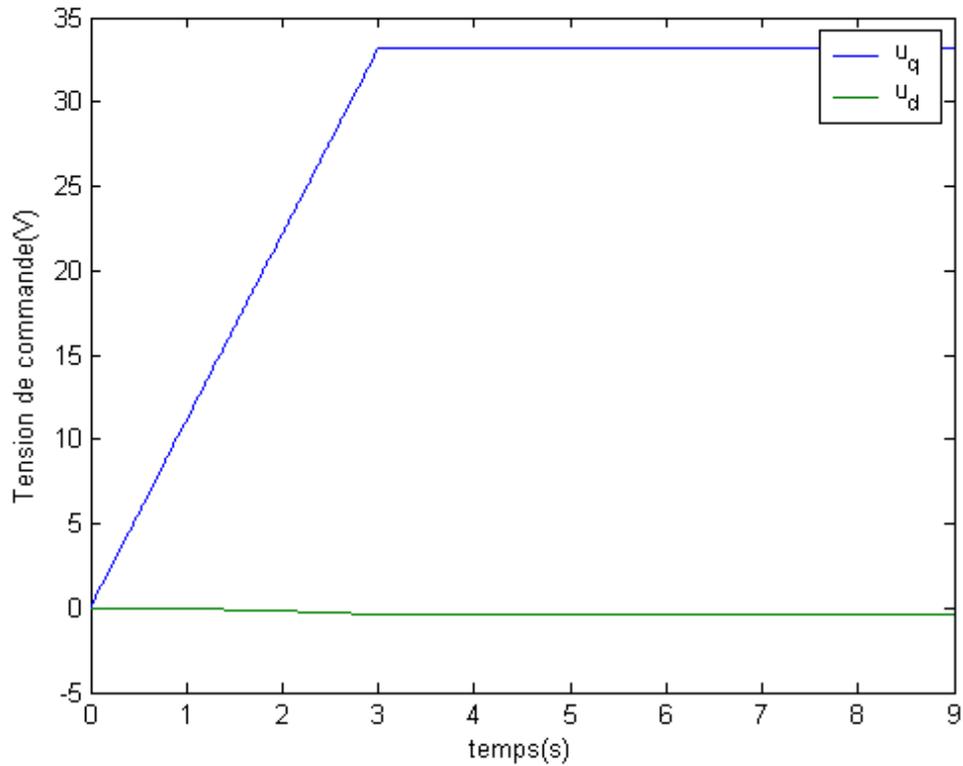


Figure 3.22 : Variation des tensions de commande u_q et u_d

5. Interprétation des résultats

En ce qui concerne la stabilisation, la simulation du système a montré que la machine synchrone à aimants permanents a bien été stabilisée du fait que toutes les trajectoires convergent vers zéro.

D'autre part, nous avons montré par trois simulations distinctes que les trajectoires du système suivent les trajectoires désirées. La première simulation concerne une vitesse désirée dont la fonction est $\omega_r(t) = 25 \sin 0.5t + 25$, La seconde simulation concerne une vitesse désirée constante dont la valeur est $\omega_d(t) = 50 \text{rad/s}$. Et la troisième concerne qu'en t'à elle une autre vitesse angulaire désirée dont la fonction est $\omega_d(t) = \frac{50}{3}t$ pour $t < 3s$ et $\omega_d(t) = 50 \text{rad/s}$ pour $t \geq 3s$.

6. Conclusion

Dans ce chapitre, nous avons étudié la stabilité des modèles TS, ainsi que la stabilisation des modèles TS par une loi de commande de type PDC. Le problème de suivi de trajectoire des modèles flous TS avec un retour d'état a aussi été abordé.

Ensuite, nous avons appliqué cette approche à la machine synchrone à aimants permanents, nous avons trouvé que toutes les trajectoires du système convergent vers zéro. Pour ce qui est du suivi de trajectoire, nous avons montré que toutes les trajectoires du système font partie des trajectoires désirées.