Commande Optimale d'un convertisseur de puissance Buck Boost

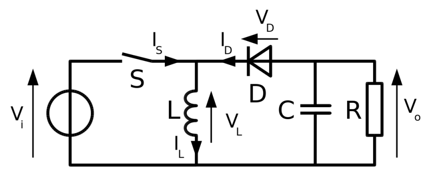

Figure 1 – Buck Boost

Un convertisseur de puissance continu-continu est un dispositif permettant d'adapter la tension délivrée à une charge à partir d'une tension d'entrée constante. On considère (Figure1) un convertisseur Buck-Boost (abaisseur-survolteur). La charge en sortie est assimilée à une résistance R.

Le pilotage de ce dispositif s'effectue par l'ouverture et la fermeture à fréquence élevée de l'interrupteur S (semi conducteur). Le système a donc 2 configurations de fonctionnement (Figure 2). Lorsque $s = 1$, l'interrupteur

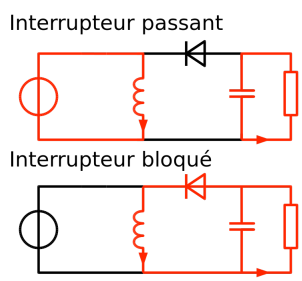

Figure 2 – Modes

est fermée, l'énergie est stocké dans l'inductance et la capacité se décharge dans la résistance. Lorsque $s = 0$, l'interrupteur est ouvert, l'inductance est reliée à la charge et à la capacité. Il en résulte un transfert de l'énergie accumulée dans l'inductance vers la capacité et la charge.

En considérant les variables d'états : $x_1 = i_L$ le courant dans l'inductance et $x_2 = v_0$ la tension dans la capacité et en notant V_i la tension d'entrée supposée constante, les équations de ce modèle non linéaire peuvent alors s'écrire $(x = (x_1, x_2))$:

$$\dot{x} = s(A_1 x + B_1 V_i) + (1 - s)(A_2 x + B_2 V_i)$$

avec $s \in \{0, 1\}$ la variable de commande et

$$A_1 = \begin{pmatrix} 0 & 0 \\ 0 & -1/(RC) \end{pmatrix} \quad B_1 = \begin{pmatrix} 1/L \\ 0 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} 0 & -1/L \\ 1/C & -1/(RC) \end{pmatrix} \quad B_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Paramètres : $V_i = 10$ volts, $R = 2\Omega$, $L = 0.3H$, $C = 0.1F$.

Analyse du fonctionnement :
Le pilotage du circuit s'effectue via une MLI (modulation de largeur d'impulsion)(Figure 3). Pour une période de

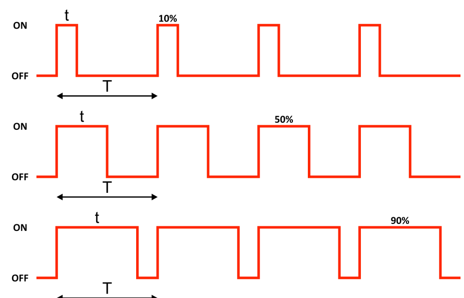

Figure 3 – Modulation de largeur d'impulsion

découpage T_s fixée et petite devant les constantes de temps du circuit, la commande est déterminée par un rapport cyclique $\alpha(t) \in [0\ 1]$ suivant la loi :

$$s(t) = \begin{cases} 1 \text{ si } kT_s \leq t \leq kT_s + \alpha T_s \\ 0 \text{ si } kT_s + \alpha T_s < t \leq (k+1)T_s \end{cases}$$

Un simple calcul montre que la valeur moyenne de $s(t)$ sur la période T_s est égale à $\alpha \in [0\ 1]$. On voit donc que la commande s peut être assimilée à une variable continue comprise entre les valeurs 0 et 1 si T_s est choisi suffisamment petit. Par la suite, on peut donc relaxer le domaine de commande $\{0, 1\}$ à l'intervalle $[0\ 1]$ et simplifier le problème par la recherche une commande $s(t) \in [0\ 1]$.

— Déterminer la valeur (rapport cyclique) de $s \in [0\ 1]$ permettant de délivrer une tension de sortie de 15 volts en régime permanent. Calculer le courant correspondant à cet équilibre.

On notera x_{ref} et s_{ref} les valeurs de l'état et de la commande à cet équilibre.

On souhaite alors établir une commande optimale en minimisant le critère suivant et en partant de condition initiale nulle $x_0 = (0, 0)$:

$$J = \frac{1}{2}\int_0^T (x - x_{ref})^T Q (x - x_{ref}) + \rho(s - s_{ref})^2 dt + \frac{1}{2}(x(T) - x_{ref})^T Q_f (x(T) - x_{ref})$$

On pourra choisir comme réglage au départ : $Q_f = Q = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ et $\rho = 1$ et $T = 5$. Puis faire évoluer ces pondérations et voire l'influence sur la dynamique.

L'algorithme dit "Forward Backward Sweep" ou "balayage avant arrière" basé sur les conditions nécessaires du Principe du Minimum permet de résoudre ce problème numériquement :

Pour un problème défini par :

$$\dot{x} = f(x, u) \tag{1}$$

$$x(0) = x_0 \tag{2}$$

$$\min_u J(u) = \int_0^T L(x, u)dt + \phi(x(T)) \tag{3}$$

il s'énonce de la manière suivante :

Algorithm FBS :

A partir d'une commande u donnée et quelconque ($u(t) \in [0\ 1]$).

1. Intégration sur $[0\ T]$ du système :
$$\dot{x} = f(x, u),\ x(0) = x_0$$
2. Application de la condition de transversalité en T :
$$\lambda(T) = \frac{\partial \phi}{\partial x}(x(T))$$
3. Intégration sur $[T\ 0]$ du système adjoint (sens rétrograde) :
$$\dot{\lambda} = -\frac{\partial H(x, u, \lambda)}{\partial x}$$
4. Calcul de $v = \arg\min_u H(x, u, \lambda)$
5. Mise à jour de u :
$$u = \arg\min_{u_k}\{J(u_k), k = 0, 1, 2, \cdots\}$$
avec $u_k = \Pi(u + \beta^k(v - u))$ où $\Pi(x)$ désigne la projection sur $[0\ 1]$ et $\beta \in (0, 1)$.
$$\Pi(x) = \max(\min(1, x), 0)$$
6. Si Test arrêt ok alors fin sinon retour au point 1

Mise en oeuvre pratique :

— Pour intégrer numériquement en sens direct ou arrière, (point 1 et 3), il faut utiliser un schéma numérique d'intégration, par exemple Euler :
$$x_{k+1} = x_k + hf(x_k, u_k)$$
avec $h = T/N$ et N nombre de points de discrétisation.

— La première commande (initialisation) pourra être choisie à $s(t) = s_{ref}$ pour tout t.

— Le nombre de points de discrétisation doit être suffisant (ici au moins 50) pour limiter les erreurs d'approximation liées au pas h.
— Au point 5, on peut prendre $\beta = \frac{1}{2}$ et $k_{max} = 25$ (au delà $\beta^k < 1.510^{-8}$ et un pas plus petit n'est plus vraiment d'interêt)
— Le test d'arrêt est obtenu lorsque l'évolution de la commande ou du critère entre 2 itérations successives n'évoluent quasiment plus soit lorsque $\|u - u_{old}\| < \epsilon_1$ ou lorsque $\|J - J_{old}\| < \epsilon_2$ avec ϵ_1 et ϵ_2 suffisamment petit, par exemple $\epsilon = 10^{-6}$. On ajoutera également un nombre maximum d'itération pour éviter d'entrée sur un boucle sans fin.

Travail demandé :

1. Ecrire formellement, pour le problème à traiter, l'algorithme "FBS" et l'implémenter dans Matlab.
2. Prévoir l'affichage de la commande s et de l'état x_1 et x_2 en fonction du temps à chaque itération de l'algorithme.
3. Etudier l'influence de la pondération utilisée dans le critère sur le résultat.
4. Rédiger un compte rendu synthétique à rendre en fin de séance (4h).