

Simulation Hardware-In-the-Loop d'un convertisseur entrelacé haute fréquence sur une plateforme FPGA abordable

Téo ROBERT, Romain MONTHEARD, Valentin COMBET, Mathieu GAVELLE
Commissariat à l'énergie atomique et aux énergies alternatives (CEA), Labège, France

RÉSUMÉ – La simulation Hardware-In-the-Loop (HIL) est couramment utilisée pour la validation de systèmes de contrôle-commande. Néanmoins, l'application de cette méthode aux convertisseurs de puissance à haute fréquence de commutation se heurte à l'exigence de très faibles pas de temps. Ainsi, cet article a pour objet la mise en œuvre d'un banc de simulation temps-réel HIL, pour le test en boucle fermée d'un convertisseur PFC entrelacé cadencé à 100 kHz, et piloté par un microcontrôleur C2000. Le solveur utilisé, basé sur une représentation d'état et des matrices précalculées, est implémenté sur une cible FPGA Artix-7 *low-cost*. Avec un pas de temps atteignant 50 ns, la simulation permet d'obtenir des résultats fidèles à la fois en boucle ouverte et en boucle fermée. Une analyse détaillée des différents facteurs d'erreur est proposée, y compris dans le domaine fréquentiel. L'étude du comportement dynamique en boucle fermée permet de comparer les résultats HIL avec ceux du modèle petits signaux, ainsi qu'avec le convertisseur réel. La solution proposée montre ainsi que la méthode des matrices précalculées est fiable et suffisamment précise, avec un ratio performance-coût excellent.

Mots-clés – *HIL, simulation temps-réel, pas de temps, FPGA, haute fréquence, convertisseurs de puissance*

1. INTRODUCTION

Dans de nombreux domaines de l'ingénierie, la simulation HIL est utilisée pour tester les systèmes de contrôle afin d'accélérer le cycle de développement, d'accroître la flexibilité des scénarios testés, ainsi que la sécurité en supprimant le risque de dommages matériels sur le système contrôlé, puisque celui-ci est simulé. Dans le domaine de l'électronique de puissance, la simulation HIL est à la fois utile et difficile à développer, en particulier avec les transistors à large bande interdite qui permettent des fréquences de fonctionnement plus élevées. En effet, un pas de temps de simulation nettement inférieur à la période de commutation (au moins 20 fois inférieur et donc sous la microseconde) est nécessaire pour garantir la fidélité de la simulation. De plus, une faible latence est nécessaire pour simuler avec précision la dynamique du système, qui est particulièrement rapide dans les convertisseurs de puissance.

Dans ce contexte, un certain nombre de solutions ont été proposées pour répondre à ce besoin. Des simulateurs temps réel commerciaux (RTS) sont proposés par plusieurs sociétés et sont basés sur des plateformes FPGA ou CPU. Le pas de temps atteint est généralement limité à environ 200 ns sur les plates-formes les plus performantes. Le solveur est souvent basé sur la modélisation par circuit discret associé, initialement proposée par [2], qui présente l'inconvénient d'introduire des oscillations irréalistes et d'avoir une précision limitée. Des méthodes de compensation visant à atténuer ce problème ont toutefois été proposées dans [3][4]. Dans la littérature, plusieurs méthodes ont été proposées, comme la représentation détaillée des transitoires des interrupteurs avec une complexité accrue [5][6][7], et des modèles d'interrupteurs idéaux ou résistifs qui modélisent le convertisseur avec plusieurs matrices en fonction de la configuration des interrupteurs. Cette dernière méthode est similaire

aux logiciels de simulation hors ligne tels que *PLECS*, et nécessite le calcul en ligne des matrices comme proposé dans [8], ou le précalcul hors ligne et le stockage de toutes les matrices [9][10][11]. Des techniques de découpage ont également été proposées, divisant le modèle numérique discrétré en plusieurs parties afin de réduire la charge de calcul [12].

L'approche proposée dans cet article consiste à mettre en œuvre une simulation de convertisseur PFC à haute fréquence en utilisant des matrices précalculées, car elle permet d'obtenir des pas de temps faibles avec une grande précision, bien qu'elle nécessite davantage de ressources logiques. Une analyse simple de la configuration des convertisseurs est effectuée pour réduire le nombre de matrices à stocker. Les matrices sont également simplifiées à l'aide de méthodes similaires à celles utilisées pour les matrices creuses. Le solveur est implémenté sur un FPGA Xilinx Artix-7 pour démontrer la faisabilité de la simulation en temps réel de convertisseurs à haute fréquence en utilisant du matériel peu coûteux. Les performances du simulateur sont évaluées à l'aide d'une mesure de l'erreur moyenne et d'une analyse dans le domaine des fréquences. La fidélité de la réponse dynamique est également analysée et comparée avec un modèle petits signaux, et des essais en boucle fermée sont effectués avec le convertisseur piloté par un microcontrôleur C2000.

2. MODÉLISATION DU CONVERTISSEUR TOTEM-POLE PFC

Le système considéré est un convertisseur PFC totem-pole entrelacé à base d'interrupteurs GaN, dont le schéma est donné sur la Fig. 1. Celui-ci est prévu pour fonctionner à une fréquence de découpage de 100 kHz et à une puissance nominale fixée à 3.3 kW, en étant raccordé à un réseau 230 V / 50 Hz. Cette topologie est généralement employée pour le raccordement au réseau alternatif de systèmes de stockage d'énergie et de chargeurs de véhicules électriques. La tension d'entrée AC est redressée par la cellule basse fréquence (cellule 0) commutée à la fréquence du réseau, tandis que les cellules haute fréquence (cellules 1, 2, 3) sont entrelacées pour réduire l'ondulation du courant d'entrée et le taux de distorsion harmonique.

Le schéma de contrôle est basé sur des boucles de courant et de tension imbriquées, régulant la tension de sortie à un niveau nominal de 380 V. L'analyse de la fidélité du modèle en temps réel se concentrera sur la boucle de courant, car elle est beaucoup plus rapide que la boucle de tension. La représentation d'état du convertisseur est établie, en tenant compte des multiples configurations du convertisseur en fonction des états des interrupteurs. Pour comparer la réponse en fréquence du simulateur HIL avec le convertisseur réel, le modèle petits signaux sera dérivé dans la section 6.3.

Basé sur une représentation d'état du convertisseur, le modèle temps-réel est intimement lié à l'état des interrupteurs, qui doit être déterminé par le solveur, comme détaillé dans la Section 3.2.

En appliquant les lois de Kirchhoff au circuit dans les diffé-

rentes configurations, la représentation d'état est établie comme indiqué dans l'équation (1), G_x étant le signal de commande de l'interrupteur x . Le vecteur d'état X contient les courants des inductances et la tension du condensateur, tandis que le vecteur d'entrée U contient la tension d'entrée et le courant de sortie. Dans le vecteur de sortie Y , nous pouvons récupérer le courant d'entrée total en additionnant les courants des inductances.

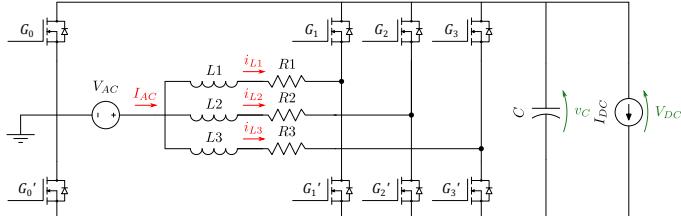


FIG. 1. Schéma électrique du convertisseur PFC totem-pole entrelacé

$$\begin{cases} \dot{X} = M_r A_r X + M_r B_r U \\ Y = CX \end{cases} \quad (1)$$

$$X = \begin{bmatrix} i_{L1} \\ i_{L2} \\ i_{L3} \\ v_C \end{bmatrix} \quad U = \begin{bmatrix} v_{AC} \\ i_{DC} \end{bmatrix} \quad Y = \begin{bmatrix} i_{L1} \\ i_{L2} \\ i_{L3} \\ v_C \\ i_{AC} \end{bmatrix} \quad (2)$$

$$A_r = \begin{bmatrix} -r_1 & 0 & 0 & G_0 - G_1 \\ 0 & -r_2 & 0 & G_0 - G_2 \\ 0 & 0 & -r_3 & G_0 - G_3 \\ G_1 - G_0 & G_2 - G_0 & G_3 - G_0 & 0 \end{bmatrix} \quad (3)$$

$$M_r = \left[\begin{array}{ccc|c} L_1^{-1} & 0 & 0 & 0 \\ 0 & L_2^{-1} & 0 & 0 \\ 0 & 0 & L_3^{-1} & 0 \\ \hline 0 & 0 & 0 & C^{-1} \end{array} \right] \quad (4)$$

$$B_r = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & -1 \end{bmatrix} \quad (5)$$

3. SOLVEUR TEMPS-RÉEL

3.1. Discréétisation du modèle

La représentation d'état du convertisseur est discréétisée à l'aide d'un schéma d'Euler implicite du premier ordre. Cette méthode est plus stable (A-stable) et plus précise que le schéma d'Euler explicite, tout en étant moins exigeante en termes de calcul que les méthodes d'ordre supérieur [14]. Elle est également particulièrement adaptée au traitement des discontinuités de commutation, contrairement au schéma trapézoïdal qui introduit des oscillations irréalistes autour de ces discontinuités (bien qu'il soit généralement plus précis) [15]. En appliquant le schéma d'Euler implicite à la représentation d'état, on obtient les équations du système discréétisé dans l'équation (6).

$$\begin{cases} v = \mathcal{M}^{-1} b \\ y = Cv \end{cases} \quad (6)$$

$v = X_{k+1}$	vecteur d'état
$y = Y_{k+1}$	vecteur de sortie
$b = X_k + hM_r B_r U_{k+1}$	vecteur source
$\mathcal{M} = I - hM_r A_r$	matrice d'état discréète

Le vecteur source b contient à la fois les termes historiques (vecteur d'état au pas de temps précédent) et les termes d'entrée (à partir du vecteur d'entrée au pas de temps actuel). La matrice

\mathcal{M} , appelée matrice d'état discréète, dépend de la configuration du convertisseur (liée aux états des interrupteurs). Par conséquent, elle est inversée et stockée dans toutes ses configurations pour être utilisée par le solveur.

3.2. Détermination de l'état des interrupteurs et sélection de la matrice

La Fig. 2 décrit la structure du solveur telle qu'elle est implémentée sur la cible FPGA. La valeur du pas de temps est de 50 ns. Cela représente 200 pas par période de découpage à 100 kHz, ce qui est au-delà de la norme (généralement de 20 à 100 pas par période).

Pour sélectionner la matrice appropriée, le solveur doit déterminer les états des interrupteurs K_X , sur la base des signaux de contrôle G_X et du vecteur d'état au pas de temps précédent. Un problème majeur dans la simulation en temps réel des convertisseurs d'électronique de puissance est le calcul de l'état des diodes, en raison de l'imprévisibilité de l'instant de commutation. Plusieurs méthodes ont été proposées pour déterminer l'état des diodes. Par exemple, une méthode utilisant le mapping du courant avec les états des diodes a été proposée dans [14]. Comme les diodes fonctionnent principalement pendant les temps morts dans la présente application, une détermination de l'état basée sur la polarité du courant dans les inductances est suffisante.

Pour réduire le nombre de matrices à stocker, certaines configurations sont exclues ou redirigées vers des états équivalents, ce qui donne finalement 15 matrices. Les états dans lesquels les cellules sont court-circuitées sont exclus parce qu'ils sont interdits par la conception du driver, tandis que les états dans lesquels les deux interrupteurs d'une cellule sont ouverts sont redirigés vers la conduction de l'une des diodes des interrupteurs, en fonction d'état de ces dernières. En outre, les uns et les zéros dans les matrices sont identifiés et simplifiés pour réduire la consommation de DSP sur le FPGA.

Une fois que le solveur a déterminé les états des interrupteurs K_X , la matrice appropriée est extraite d'une table de correspondance. Simultanément, le solveur met à jour le vecteur b en utilisant les entrées et les termes historiques. Enfin, le solveur calcule le nouvel état et le vecteur de sortie à l'aide de produits matrice-vecteur.

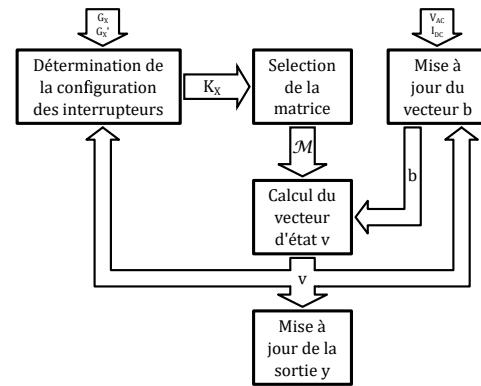


FIG. 2. Structure du solveur proposée

4. IMPLÉMENTATION

Alors que le pas de temps typique d'une simulation temps-réel sur une cible CPU se situe au-delà de la microseconde, posant ainsi une limite sur la fréquence de découpage autour de 10 kHz, une plateforme de type FPGA est la cible de choix pour atteindre les 50 ns comme exigé dans notre application. Le solveur est ainsi implémenté sur une cible d'entrée de gamme Xilinx Artix-7 XC7A100T, dont les spécifications logiques sont données dans le Tab. 1.

4.1. Représentation des nombres

Le solveur utilise une représentation des nombres en virgule fixe sur 32 bits, comprenant 20 bits pour la partie fractionnaire. Bien qu'une représentation en virgule flottante permet une plus grande plage de valeurs représentées ainsi qu'une précision supérieure, l'implémentation sur FPGA d'une unité de calcul en virgule flottante est complexe et exigeante en ressources. A l'inverse, le calcul en virgule fixe est robuste et économique en ressource. Cependant, il nécessite une attention particulière quant à la mise à l'échelle. Celle-ci est nécessaire pour augmenter la précision tout en évitant les dépassements de format, en particulier du fait de la plage de valeurs plus restreinte. Ainsi, un facteur d'échelle binaire de $\frac{1}{2^{20}}$ est choisi. Bien que moins intuitif qu'un facteur d'échelle décimal, un facteur d'échelle binaire est particulièrement commode pour effectuer les mises à l'échelle et appliquer les offsets requis notamment par les interfaces. En effet, ceux-ci peuvent être aisément effectués par simple sélection et décalage de bits. Avec ce format, les données dans le solveur peuvent être écrites sous la forme de l'équation (7). La plage de données obtenue est $[-2048, 2048]$, avec une précision de $5 \cdot 10^{-7}$.

$$x = c_{31} 2^{11} + \cdots + c_{21} 2^1 + c_{20} + c_{19} 2^{-1} + \cdots + c_0 2^{-20} \quad (7)$$

4.2. Interfaces

Le solveur est interfacé avec le contrôleur C2000 via des signaux numériques et analogiques, comme présenté dans la Fig. 3. Les sources (tension d'entrée et courant de sortie) sont produites par un générateur de signaux externe (pour permettre leur modification durant la simulation) et échantillonnes par des Convertisseurs Analogique-Numérique (CANs) sur le FPGA. La tension d'entrée est aussi mesurée par le contrôleur pour la synchronisation de la référence de courant alternative et pour le *feedforward*. Les signaux numériques de contrôle des interrupteurs générés par le contrôleur sont échantillonnes par le FPGA à chaque pas de temps de la simulation. Les sorties du simulateur (courant d'entrée, courants dans les inductances et tension de sortie) sont converties par des Convertisseurs Numérique-Analogique (CNAs) pour être mesurées par le contrôleur en boucle fermée. Des mises à l'échelle sont appliquées sur tous les signaux analogiques pour exploiter la pleine échelle des CANs et des CNAs, ceci afin d'augmenter la précision et le rapport signal-bruit. Les CANs du contrôleur étant unipolaires, des offsets sont appliqués numériquement sur les signaux alternatifs.

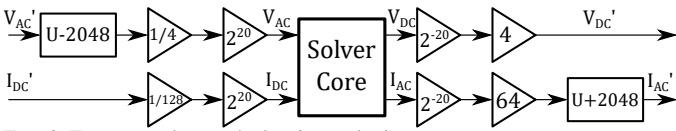


FIG. 3. Formatage des entrées/sorties analogiques

4.3. FPGA

Le solveur est décrit en VHDL et implémenté sur le FPGA. Les paramètres de synthèse et d'implémentation sont gardés standards et aucune optimisation du design RTL n'est effectuée. L'implémentation sur Vivado prédit le respect des contraintes temporelles avec un *worst negative slack* de l'ordre de 5 ns. Les ressources logiques utilisées sont données dans le Tab. 1.

Bien qu'étant en général une ressource critique dans le cas des matrices précalculées, l'utilisation des LUTs est très faible. Les ressources les plus limitantes sont les DSPs, qui effectuent les produits matrice-vecteur, ainsi que les entrées-sorties qui limitent la potentielle utilisation de bus parallèles (pour les CNAs notamment).

TABLEAU 1. Ressources FPGA

Ressource	Utilisation	Total	Pourcentage
LUT	1656	63400	2,61 %
FF	601	126800	0,47 %
DSP	48	240	20 %
IO	25	210	11,90 %
BUFG	3	32	9,38 %
MMCM	1	6	16,67 %

5. PERFORMANCE DU SIMULATEUR

5.1. Estimation de l'erreur

Afin d'estimer la fidélité du simulateur HIL, les résultats expérimentaux sont comparés avec une simulation *offline* sous *Matlab Simscape SPS*. Cette dernière est effectuée avec un pas de temps variable et un solveur adapté aux équations différentielles raides (ode23tb), et représente un bon point de référence pour l'analyse du simulateur HIL. A titre d'illustration, les formes d'onde du courant d'entrée sont données sur la Fig. 5.

En moyennant les valeurs du courant données par ces deux simulations sur 1 ms, l'erreur mesurée est inférieure à 2%, ce qui est conforme aux attentes, compte tenu des erreurs introduites par le solveur et les interfaces.

En complément de cette estimation sur la valeur moyenne, et afin d'identifier les différentes sources d'erreur, les formes d'ondes du courant sont analysées dans le domaine fréquentiel, par une transformée de Fourier rapide (FFT). Les résultats sont donnés sur la Fig. 4.

Différents groupes de raies sont identifiés. A basse fréquence, on observe la résonance de la capacité et des inductances autour de 200 Hz. C'est un comportement réaliste du convertisseur, dû aux faibles variations du rapport cyclique. Des composantes à 1.6 kHz et ses multiples sont également identifiées. Ce sont des oscillations non réalistes dues à un effet d'*aliasing* lors de l'échantillonnage des signaux de contrôle par le FPGA, qui sera discuté par la suite. Bien que leur amplitude soit faible au regard de la composante CC qui est de l'ordre de 14 A, ces pics sont plus importants que les raies à 300 kHz et ses multiples, qui correspondent à l'ondulation de courant dans les inductances à trois fois la fréquence de commutation (du fait de l'entrelacement). Finalement, des raies peuvent être identifiées à haute fréquence, au delà de 3 MHz, et sont dues à la mise à jour des CNAs série sur 17 cycle à une fréquence de 50 MHz. La fidélité de la réponse dynamique sera discuté par la suite dans la Section 6.

5.2. Effet d'aliasing sur l'échantillonnage des signaux de contrôle

Comme le montre la FFT, un contenu harmonique important est présent à 1.6 kHz et ses multiples. Celui-ci s'explique par le phénomène d'*aliasing* lors de l'échantillonnage des signaux de contrôle, qui introduit des harmoniques que l'on retrouve sur le courant en sortie du simulateur. Des travaux précédents ont décrit ce phénomène [16]. Du fait de l'absence de synchronisation entre le contrôleur et le FPGA, et des tolérances sur les fréquences d'horloge, les signaux de contrôle à 100 kHz des interrupteurs sont échantillonnes par le solveur de manière asynchrone, à une fréquence finie de 20 Mhz. Cela introduit des oscillations subharmoniques, comme illustré sur la Fig. 6. Leur fréquence est mesurée à 1.6 kHz, ce qui correspond à une répétition de motif tous les 62 cycle de commutation, ce qui est vérifié sur les signaux échantillonnes, confirmant donc l'origine de ces harmoniques. Des travaux futurs sont à considérer pour limiter l'impact de ce phénomène sur le simulateur et ses sorties.

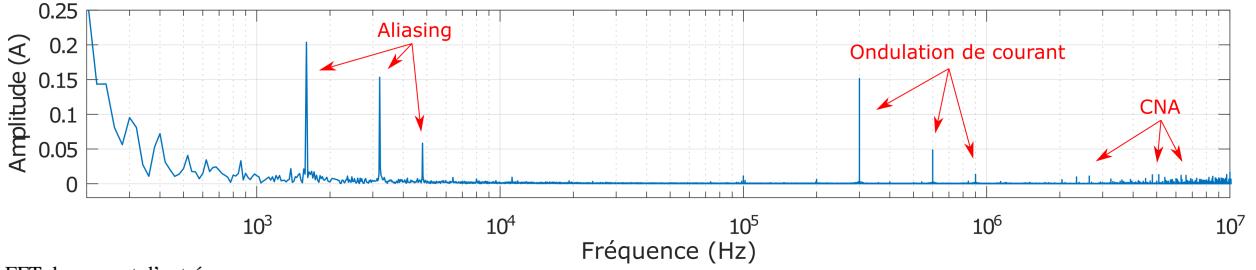


FIG. 4. FFT du courant d'entrée

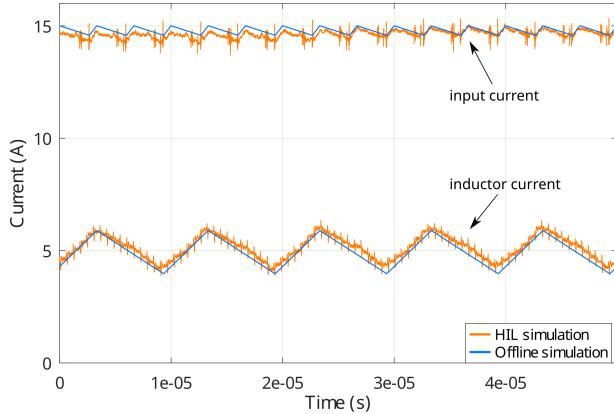


FIG. 5. Formes d'onde du courant d'entrée en simulation HIL et offline

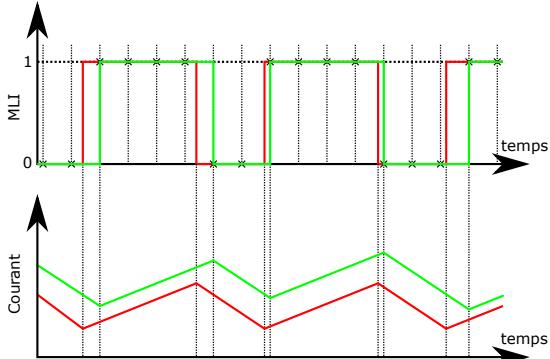


FIG. 6. Illustration de l'effet d'aliasing sur l'échantillonage du signal PWM

6. PERFORMANCE DU SYSTÈME BOUCLÉ

6.1. Prototype réel du convertisseur

La simulation HIL présentée dans cet article a pour but la validation du logiciel embarqué développé pour le contrôle-commande d'un prototype réel, d'un convertisseur PFC totem-pole à base d'interrupteurs GaN, illustré sur la Fig. 7. Dans le cadre du travail présenté ici, ce convertisseur physique de 3.3 kW est utilisé pour fournir un point de comparaison croisée supplémentaire, dont les résultats sont présentés et discutés dans la Section 6.4.

6.2. Caractérisation de la latence du simulateur

Si la latence du simulateur n'a pas d'impact sur les performances en boucle ouverte, elle peut en revanche dégrader la réponse dynamique du système en boucle fermée. Les différentes sources de contribution à la latence sont résumées dans la Fig 8. Les entrées analogiques (sources de tension et de courant) ne sont pas représentées, car elles ne font pas partie de la boucle de courant et ont une dynamique très lente (continue pour I_{DC} et 50 Hz pour V_{AC}). Dans leur cas, le temps de conversion des CANs doit être considéré.

La latence de synchronisation des signaux numériques d'en-

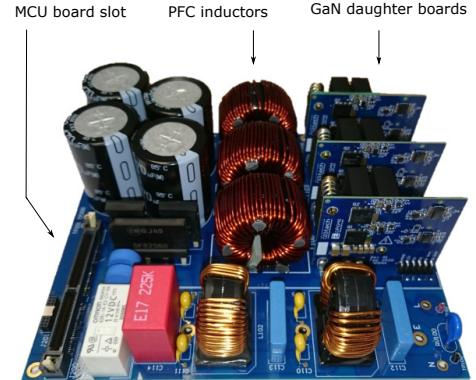


FIG. 7. Photographie du prototype physique du PFC totem-pole

trée et celle du solveur sont chacune de 50 ns, soit un pas de temps de simulation. D'autre part, les résultats de mesure de la latence de l'interface de sortie sont donnés sur la Fig 9. Celle-ci se décompose ainsi :

- sérialisation des données, ce qui compte pour 340 ns (17 cycles sur un DAC 50 MHz) ;
- temps de conversion, voisin de 100 ns ;
- temps de stabilisation, de 600 ns dans le pire cas (à savoir un échelon de pleine échelle).

La latence totale du simulateur peut ainsi être établie à environ 1.14 µs.

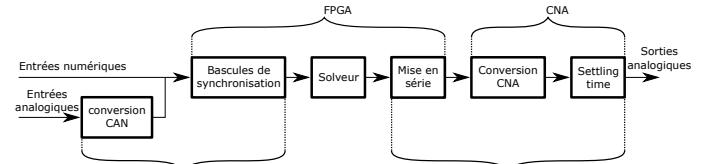


FIG. 8. Contributions à la latence

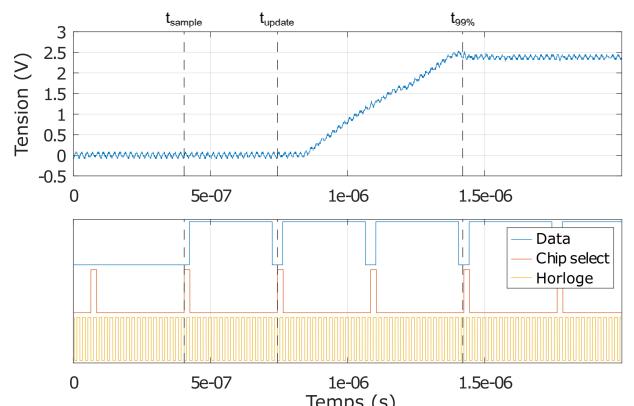


FIG. 9. Latence du DAC

6.3. Modèle petits signaux

Afin de caractériser la fidélité du simulateur en termes de comportement dynamique, un modèle petits signaux linéarisé est établi dans l'équation (8). Celui-ci permet d'obtenir notamment la fonction de transfert liant le rapport cyclique au courant d'entrée, dont l'expression est donnée dans l'équation (13), avec H , la fonction de transfert extraite du modèle petits signaux, et G_d qui modélise le délai associé à la MLI.

$$\begin{cases} \tilde{\dot{X}} = MA\tilde{X} + MB\tilde{U} \\ \tilde{Y} = C\tilde{X} \end{cases} \quad (8)$$

$$\tilde{X} = \begin{bmatrix} \tilde{i}_L \\ v_C \end{bmatrix} \quad \tilde{U} = \tilde{d} \quad \tilde{Y} = i_{AC} \quad (9)$$

$$A = \begin{bmatrix} -r & -D \\ 3D & \frac{-I_{DC}}{V_C} \end{bmatrix} \quad (10)$$

$$M = \begin{bmatrix} L^{-1} & 0 \\ 0 & C^{-1} \end{bmatrix} \quad (11)$$

$$B = \begin{bmatrix} -V_C \\ 3I_L \end{bmatrix} \quad C = \begin{bmatrix} 3 \\ 0 \end{bmatrix} \quad (12)$$

$$\frac{i_{AC}}{\tilde{d}}(s) = H(s)G_d(s) \quad (13)$$

6.4. Comparaison de la réponse dynamique

Une petite perturbation est ajoutée à la valeur du rapport cyclique en entrée du modulateur MLI, ce qui permet de tracer le diagramme de Bode en balayant la fréquence de cette perturbation, et en observant la réponse sur le courant d'entrée mesuré en sortie du simulateur. La Fig. 10 montre les résultats ainsi obtenus, superposés avec les résultats mesurés sur le convertisseur réel, ainsi que le modèle petits signaux, sans et avec la prise en compte d'une latence de 1.14 µs. La simulation HIL donne des résultats en bonne adéquation avec le modèle et le prototype réel. On remarque que la latence du simulateur a un impact plus faible qu'attendu (une explication réside dans la mesure du temps d'établissement de la sortie du CNA, qui a été effectuée dans le cas le plus défavorable). En effet, l'impact de celle-ci à la fréquence de coupure cible de l'asservissement (autour de 7 kHz) est négligeable, même par rapport au convertisseur réel. La dégradation de la marge de phase est alors de l'ordre de 5°, ce qui est acceptable compte tenu de la marge de phase cible de 45°. Il est à noter également que contrairement aux mesures réalisées sur le simulateur, et du fait des contraintes liées à l'implémentation physique, le diagramme de Bode du convertisseur réel a été mesuré avec une précision plus faible, ce qui explique la variabilité des résultats vis-à-vis de son modèle petits signaux.

Le comportement du simulateur est finalement vérifié en régulation (boucle fermée de courant et de tension), le convertisseur fonctionnant en CA/CC, comme présenté sur les formes d'ondes de la Fig. 11. Comme attendu, la tension de sortie est régulée à 380 V CC, et le courant d'entrée est autour de 14 A RMS avec peu de distorsion harmonique. Concernant les courants dans les inductances, les écarts entre les cellules dûs à l'entrelacement sont observés comme dans le convertisseur réel.

7. CONCLUSION

La simulation temps-réel d'un convertisseur PFC entrelacé à haute fréquence de découpage permettant la validation HIL de stratégies de contrôle-commande est présentée dans cet article. Le convertisseur est modélisé par une représentation d'état discréétisée, implémentée sur un FPGA Artix-7 en utilisant la méthode des matrices précalculées. Les résultats expérimentaux montrent une précision satisfaisante de l'ordre de 2% et un comportement dynamique très proche du modèle théorique, démontrant ainsi la possibilité d'employer une cible FPGA abordable

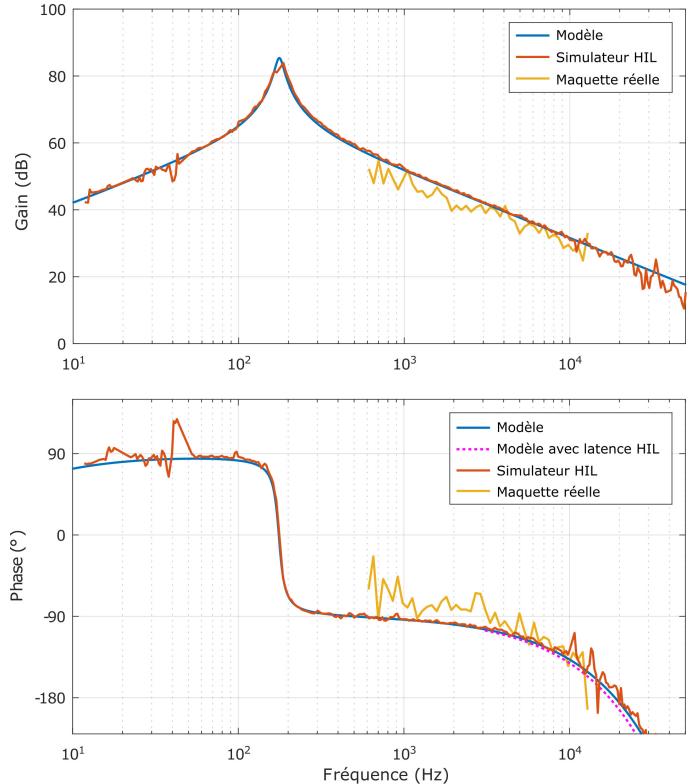


FIG. 10. Diagramme de Bode de la boucle de courant d'entrée en HIL vs. modèle petits signaux et convertisseur réel

pour le test HIL du contrôle de convertisseurs à haute fréquence de découpage. De futurs travaux pourront contribuer à réduire la latence du simulateur, ainsi qu'à investiguer des méthodes de réduction de l'aliasing sur l'échantillonnage des signaux de contrôle.

8. RÉFÉRENCES

- [1] Opal-RT, <https://www.opal-rt.com/fr/solver-ehs-2>
- [2] P. Pejovic and D. Maksimovic, "A method for fast time-domain simulation of networks with switches," in IEEE Transactions on Power Electronics, vol. 9, no. 4, pp. 449-456, July 1994, doi : 10.1109/63.318904.
- [3] K. Wang, J. Xu, G. Li, N. Tai, A. Tong and J. Hou, "A Generalized Associated Discrete Circuit Model of Power Converters in Real-Time Simulation," in IEEE Transactions on Power Electronics, vol. 34, no. 3, pp. 2220-2233, March 2019, doi : 10.1109/TPEL.2018.2845658.
- [4] C. Dufour, "Method and system for reducing power losses and state-overshoots in simulators for switched power electronic circuit," (US9665672B2).
- [5] G. Parma and V. Dinavahi, "Real-Time Digital Hardware Simulation of Power Electronics and Drives," 2007 IEEE Power Engineering Society General Meeting, 2007, pp. 1-1, doi : 10.1109/PES.2007.385592.
- [6] A. Myaing and V. Dinavahi, "FPGA-Based Real-Time Emulation of Power Electronic Systems With Detailed Representation of Device Characteristics," in IEEE Transactions on Industrial Electronics, vol. 58, no. 1, pp. 358-368, Jan. 2011, doi : 10.1109/TIE.2010.2044738.
- [7] H. Bai, H. Luo, C. Liu, D. Paire and F. Gao, "A Device-Level Transient Modeling Approach for the FPGA-Based Real-Time Simulation of Power Converters," in IEEE Transactions on Power Electronics, vol. 35, no. 2, pp. 1282-1292, Feb. 2020, doi : 10.1109/TPEL.2019.2918590.
- [8] H. F. Blanchette, T. Ould-Bachir and J. P. David, "A State-Space Modeling Approach for the FPGA-Based Real-Time Simulation of High Switching Frequency Power Converters," in IEEE Transactions on Industrial Electronics, vol. 59, no. 12, pp. 4555-4567, Dec. 2012, doi : 10.1109/TIE.2011.2182021.
- [9] D. Majstorovic, I. Celanovic, N. D. Teslic, N. Celanovic and V. A. Katic, "Ultralow-Latency Hardware-in-the-Loop Platform for Rapid Validation of Power Electronics Designs," in IEEE Transactions on In-

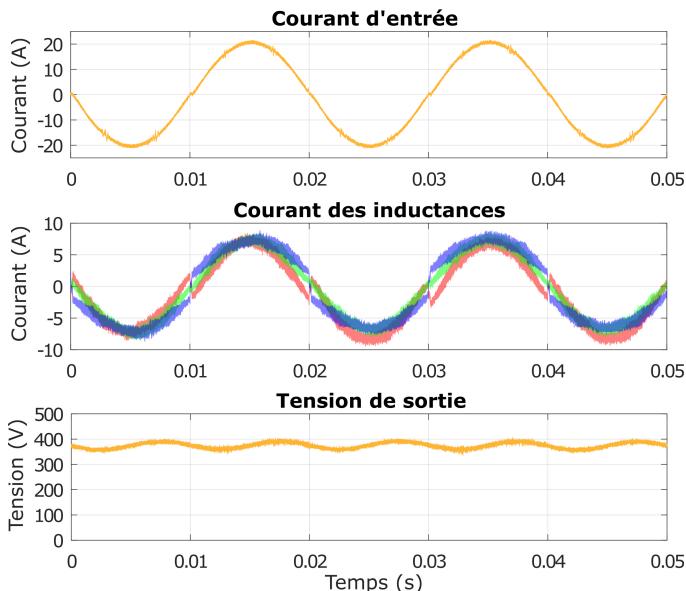


FIG. 11. Formes d'ondes CA/CC avec régulation de tension

- dustrial Electronics, vol. 58, no. 10, pp. 4708-4716, Oct. 2011, doi : 10.1109/TIE.2011.2112318.
- [10] M. S. Vekić, S. U. Grabić, D. P. Majstorović, I. L. Čelanović, N. L. Čelanović and V. A. Katić, "Ultralow Latency HIL Platform for Rapid Development of Complex Power Electronics Systems," in IEEE Transactions on Power Electronics, vol. 27, no. 11, pp. 4436-4444, Nov. 2012, doi : 10.1109/TPEL.2012.2190097.
 - [11] E. Chai, I. Celanovic and J. Poon, "Validation of Frequency- and Time-domain Fidelity of an Ultra-low Latency Hardware-in-the-Loop (HIL) Emulator," 2013 IEEE 14th Workshop on Control and Modeling for Power Electronics (COMPEL), 2013, pp. 1-5, doi : 10.1109/COMPEL.2013.6626402.
 - [12] T. Ould-Bachir, H. F. Blanchette and K. Al-Haddad, "A Network Tearing Technique for FPGA-Based Real-Time Simulation of Power Converters," in IEEE Transactions on Industrial Electronics, vol. 62, no. 6, pp. 3409-3418, June 2015, doi : 10.1109/TIE.2014.2365752.
 - [13] Texas Instruments, "Bidirectional Interleaved CCM Totem Pole Bridgeless PFC Reference Design Using C2000™ MCU"
 - [14] H. Chalangar, T. Ould-Bachir, K. Sheshyekani and J. Mahseredjian, "A Direct Mapped Method for Accurate Modeling and Real-Time Simulation of High Switching Frequency Resonant Converters," in IEEE Transactions on Industrial Electronics, vol. 68, no. 7, pp. 6348-6357, July 2021, doi : 10.1109/TIE.2020.2998746.
 - [15] W. Gao, E. Solodovnik, R. Dougal, G. Cokkinides and A. P. S. Meliopoulos, "Elimination of numerical oscillations in power system dynamic simulation," Eighteenth Annual IEEE Applied Power Electronics Conference and Exposition, 2003. APEC '03., 2003, pp. 790-794 vol.2, doi : 10.1109/APEC.2003.1179306.
 - [16] E. Zamiri, A. Sanchez, M. S. Martínez-García, et A. de Castro, "Analysis of the aliasing effect caused in hardware-in-the-loop when reading PWM inputs of power converters," in International Journal of Electrical Power and Energy Systems, vol. 136, p. 107678, mars 2022, doi : 10.1016/j.ijepes.2021.107678.