



HAL
open science

Etude d'un convertisseur DC-DC-20kW pour application aéronautique : prototypages et développement d'un algorithme d'optimisation Mixed-Discrete Geometric Programming

Bastien Pasquet

► To cite this version:

Bastien Pasquet. Etude d'un convertisseur DC-DC-20kW pour application aéronautique : prototypages et développement d'un algorithme d'optimisation Mixed-Discrete Geometric Programming. Energie électrique. Université de Toulouse, 2024. Français. NNT : 2024TLSEP124 . tel-04947953

HAL Id: tel-04947953

<https://theses.hal.science/tel-04947953v1>

Submitted on 14 Feb 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Doctorat de l'Université de Toulouse

préparé à Toulouse INP

Étude d'un Convertisseur DC-DC 20kW pour application
aéronautique : Prototypages et Développement d'un
algorithme d'optimisation Mixed-Discrete Geometric
Programming

Thèse présentée et soutenue, le 6 décembre 2024 par

Bastien PASQUET

École doctorale

GEETS - Génie Electrique Electronique, Télécommunications et Santé : du système au nanosystème

Spécialité

Génie Electrique

Unité de recherche

LAPLACE - Laboratoire PLAsma et Conversion d'Énergie

Thèse dirigée par

Thierry MEYNARD

Composition du jury

M. Christophe TURPIN, Président, Toulouse INP

M. Jean-Luc SCHANEN, Rapporteur, Grenoble INP

M. Arnaud VIDET, Rapporteur, Université de Lille

M. Frédéric RICHARDEAU, Examineur, Toulouse INP

M. Timothy MCRAE, Examineur, University of Southern Denmark

Mme Anne LIENHARDT, Examinatrice, Airbus

M. Thierry MEYNARD, Directeur de thèse, Toulouse INP

Membres invités

M. Sébastien SANCHEZ, Icam

Résumé

Version Française

Dans un contexte industriel de réduction de la consommation d'énergies fossiles et d'émission de gaz à effet de serre, l'électrification des moyens de transports joue un rôle important dans la transition écologique. Les Piles à Combustible hydrogène sont le sujet de nombreux projets d'études dans le secteur de l'aéronautique. Les travaux ici présentés s'intéressent à l'implémentation d'une Pile à Combustible comme alimentation électrique pour un réseau embarqué d'aéronef $28VDC$, $20kW$ nominal. Plus spécifiquement, l'objet de cette thèse est le développement d'un Convertisseur Électrique moyenne puissance, forts courants, adaptant la puissance en sortie d'une Pile à Combustible au réseau électrique embarqué. Les enjeux de l'aéronautique imposent des critères exigeants d'optimisation de la masse et du rendement du convertisseur. Une comparaison entre plusieurs topologies de convertisseurs a été réalisée, un convertisseur à capacités flottantes 3-niveaux a été retenu. Des transistors de type GaN ont été sélectionnés pour leur faible résistance à l'état passant et leurs vitesses de commutation élevées, permettant de minimiser les pertes du convertisseur, ainsi que la taille des composants passifs et de refroidissement du convertisseur. Un soin particulier a été apporté au routage des circuits imprimés pour réduire les impédances parasites et permettre une mise en parallèle de deux transistors pour chaque interrupteur. Un prototype présentant un rendement expérimental moyen de 98.25% a été produit.

Afin d'améliorer la rapidité et la pertinence des travaux d'optimisation des design des convertisseurs électriques, un nouvel algorithme d'optimisation a été développé à partir de travaux pré-existants. Cet algorithme *Mixed-Discrete Geometric Programming* (MDGP) permet de résoudre des problèmes physiques, dotés d'un mélange de nombreuses variables d'optimisation continues et discrètes, dans un temps raisonnable tout en garantissant une convergence vers leur optimum global. Il est capable de gérer des variables discrètes multidimensionnelles, ainsi que des fonctions substituables. L'algorithme permet d'évaluer et de sélectionner des composants électriques spécifiques, plutôt que de représenter une gamme entière de composants par une modélisation continue imprécise. Les transistors ont notamment pu être modélisés sous forme de jeux de fonctions, différents pour chaque composant, *fittées* sur des résultats de simulation. Un front de Pareto suivant les critères de la masse et du rendement du convertisseur a été obtenu en $25h$ de calcul sur un ordinateur portable, pour un problème comportant 161.28 millions de combinaisons discrètes possibles.

English Version

As the industry strives to reduce its consumption of fossil fuels and the associated greenhouse gas emissions, vehicle electrification plays a major role in the Ecological Transition. There is a growing number of studies aiming at evaluating and increasing Hydrogen Fuel Cells viability as a power supply for aircraft. This work aims to study the implementation of a hydrogen Fuel Cell as an electrical power supply for an aircraft $28VDC$, $20kW$ on-board electrical network. More specifically, the subject of this thesis is the development of a low-voltage, high current Power Converter adapting the power flowing from the Fuel Cell, to the on-board electrical network. Converter mass, volume and efficiency are major optimization criteria in aircraft design. A comparison between several converter topologies has been realized, and a 3-level Flying Capacitor Converter has been selected. GaN transistors were selected for their low on-state resistance and their high switching speed, allowing the reduction of the converter power losses and of the passive components and heat sink sizes. Particular attention was provided to printed circuit board routing in order to reduce parasitic impedance and to parallelize two transistors for each power switch. A prototype has been realized, measurements show an experimental average efficiency of 98.25%.

In order to enhance the speed and pertinence of optimization studies for electrical converters design, a new optimization algorithm has been developed based on pre-existing works. This *Mixed-Discrete Geometric Programming* algorithm (MDGP) can solve physics problems composed of a mix of many continuous and discrete optimization variables, in a reasonable calculation time while still guaranteeing convergence to the global optimum. It can handle multidimensional discrete variables, as well as substitutable functions. The algorithm allows to evaluate and select specific electrical components, instead of representing a range of components by a single continuous and imprecise modeling. Specifically, transistors are modeled as collections of functions, different for each component, fitted on simulation results. A Pareto Front for the converter mass and efficiency was generated in a $25h$ calculation time on a laptop, for an optimization problem involving 161.28 million possible discrete combinations.

Remerciements

Les travaux présentés dans ce manuscrit ont été réalisés au Laboratoire Plasma et Conversion d’Energie (LAPLACE - UMR5213), sur le site de l’ENSEEIH, au sein du groupe Convertisseurs Statiques. Ces travaux s’inscrivent dans le cadre du projet PIPAA, dont les principaux collaborateurs sont le *Commissariat à l’Energie Atomique* (CEA), l’*Institut National Polytechnique de Toulouse* (Toulouse INP), l’*Ecole Nationale de l’Aviation Civile* (ENAC), piloté par *Safran Power Units* et cofinancé par la *Banque Publique d’Investissement* (BPI).

Je souhaite tout d’abord remercier mes directeurs de thèse *Thierry MEYNARD*, directeur de recherche CNRS - Laplace, et *Sébastien SANCHEZ*, enseignant-chercheur rattaché au Laplace et responsable enseignement supérieur à l’Icam Toulouse. Merci de m’avoir offert mon Everest. Les faux-plats et les détours étaient nombreux, j’ai de nombreuses fois cru m’épuiser pour rien et, finalement, après la dernière pente, l’horizon s’est dégagé et la vue était superbe. Merci, Sébastien, d’avoir cru en moi dès le début, dès mes premiers pas dans l’électronique de puissance, à l’Icam. Merci pour ton soutien indéfectible et sincère. J’aimerais aussi remercier l’ensemble des membres de mon Jury, à commencer par *Christophe TURPIN*, directeur de recherche CNRS et responsable des activités hydrogène au Laplace, qui est à l’origine du projet PIPAA côté laboratoire, auquel se rattache ma thèse. Merci à mes rapporteurs *Jean-Luc SCHANEN*, professeur à Grenoble INP, et *Arnaud VIDET*, maître de conférences à l’Université de Lille, pour votre patience et vos rapports consciencieux, qui m’auront été d’une aide précieuse dans la finalisation du manuscrit. Merci à *Timothy MCRAE*, assistant professor à l’Université du Danemark Sud, d’avoir fait le déplacement depuis le Danemark. Merci enfin à *Anne LIENHARDT*, R&T electrical architect à Airbus, et *Frédéric RICHARDEAU*, directeur de recherche CNRS - Laplace, pour leurs points de vue respectivement industriels aéronautique, et hardware. Nos quelques échanges, Frédéric, dans les couloirs du laboratoire ont tous été d’une grande qualité.

Un travail expérimental n’est rien sans les individus qui le mettent en œuvre. Il me faut spécialement remercier *Sébastien VINNAC*, alors technicien, et *Jean-Marc BLAQUIERE*, ingénieur d’étude au Laplace. Sans eux, rien de ce qui suit n’aurait pu aboutir. Sébastien aura été mon mentor dans la conception des circuits imprimés et la programmation FPGA, mon camarade d’expérimentation tout au long de cette thèse, malgré un emploi du temps surchargé, et un très bon ami. Jean-Marc a lui aussi grandement contribué à mes travaux expérimentaux, merci pour ses conseils, son aide et ses coups de main dans la conception du banc d’essai de la mesure du rendement par méthode d’opposition, ainsi que pour m’avoir guidé dans la conception de l’assemblage du convertisseur entrainé $20kW$ final (qui n’aura malheureusement jamais vu le jour faute de temps). Merci aussi à *Vincent BLEY*, qui a pris sur son temps pour sauver l’un de mes prototypes en remplaçant avec minutie l’un des transistors GaN défectueux. J’étais vraiment à un prototype près, certaines expériences n’auraient pas eu être réalisées à temps sans cela. Merci, merci, merci !

C'est maintenant le moment de saluer mes compagnons de voyage, doctorants et docteurs au Laplace, avec qui j'ai eu la joie de rire et partager des conversations et des moments géniaux. Anas, Léo, Daniel, Mohamed, Thomas, Mathieu, ainsi que Matthieu (!!!), Luc et Luc (1 et 2?), Corentin, Noé, Paul, Joseph, Dov, Evelise, Lucas, Nicolas, Rokhaya, Ryan, Tifenn, et tous les autres (comment ça j'ai oublié des noms?!), salut à vous! Courage aux uns, bonne route aux autres, et que nos chemins se croisent à nouveau! Et mes frères Lotois et musiciens à la *Bandarrêtdurgence*, *Simon GARRIGOU*, *Hugo BELLY*, *Benjamin BARRE*, *Martin MONTAGNAC*, je vous salue! Merci à mon père Pierre, ma mère Valérie, ma sœur Justine et mes grands-parents. J'aimerais en quelque sorte dédier cette thèse à mon grand-père *Jean*. Jusqu'à très récemment j'ignorais qu'il avait été ingénieur électronicien mais le destin m'a envoyé vers la même profession sans jamais dévoiler son dessein, jusqu'au moment où les jeux étaient déjà faits. La pomme ne tombe jamais loin de l'arbre.

Et pour toi, *Maxime PAIN*, mon alter ego extraverti, mon camarade de la première heure de colle à l'Icam jusqu'à la dernière heure du dernier jour, une virile poignée de main, un rire entendu, quatre pintes de bière et une gueule de bois. Même quand tu es en retard. Même quand je suis aigri.

Tschüs!

Table des matières

Introduction générale	14
1 Première approche de conception	18
1.1 Description de la méthodologie	19
1.2 Topologies de convertisseurs DC-DC non-isolées	19
1.2.1 Convertisseur BUCK	20
1.2.2 Convertisseurs entrelacés	22
1.2.3 Convertisseur Multilevel Flying Capacitor	23
1.2.4 Convertisseurs Double BUCK & Double FCML	25
1.2.5 Variante ZVS de la topologie FC3L	27
1.3 Transistors MOSFET	28
1.4 Sélection des Transistors commerciaux	30
1.5 Extraction des énergies de commutation via LTSpice	32
1.6 Elements Passifs	36
1.7 Inductances	37
1.8 Calcul des performances de plusieurs solutions	39
1.9 Comparaison des résultats	41
1.10 Conclusion du chapitre	43
2 Conception & Expérimentation de prototypes	45
2.1 Topologie & Composants sélectionnés	46
2.2 Circuit de commande des transistors	47
2.2.1 Influence de la résistance de grille sur la commutation	47
2.2.2 Caractéristiques principales d'un driver	51
2.2.3 Commutations spontanées de <i>Miller</i>	53
2.2.4 Sélection du driver	54
2.3 Routage d'un prototype de convertisseur FC3L avec transistors parallélisés	57
2.3.1 Parallélisation des transistors	57
2.3.2 Couplage inductif entre les boucles de puissance et de grille	59
2.3.3 Comparaison de deux stratégies de routage	60
2.3.3.1 Routage Adjacent	63
2.3.3.2 Routage Traversant	64
2.3.4 Simulation par éléments finis des prototypes	65
2.4 Expérimentations sur les prototypes	69
2.4.1 Mesure de l'oscillation du v_{ds} à la commutation	70
2.4.2 Assemblage du dissipateur Thermique	74
2.4.3 Mesure du rendement du prototype <i>Adjacent</i>	78

2.4.3.1	Principe de la Méthode par Opposition	78
2.4.3.2	Montage Expérimental & erreurs relatives	83
2.4.3.3	Intérêt des capacités C_{out} sur le circuit en Opposition	85
2.4.3.4	Programmation du FPGA de contrôle	90
2.4.3.5	Mesures des pertes et du rendement & déduction d'un modèle em- pirique des pertes	92
2.5	Conclusion du chapitre	95
3	Approche de conception par optimisation algorithmique Geometric Program- ming	97
3.1	Problématiques de la phase de design	98
3.2	Formuler et résoudre un Problème d'Optimisation de design d'un convertisseur	98
3.3	Présentation du Geometric Programming	104
3.3.1	Principes généraux	104
3.3.2	Problèmes GP étendus : Generalized Geometric Programming	105
3.3.3	Intérêts et Usages	105
3.4	Résoudre un problème GP contenant des variables discrètes	106
3.4.1	Algorithme Branch & Bound (BB)	107
3.4.2	Algorithme Branch & Cut (BC)	112
3.4.3	Le cas des Tuples de variables discrètes	114
3.5	Résoudre un problème à Fonctions Substituables	115
3.6	Conclusion du chapitre	117
4	Optimisation du design d'un convertisseur FCML multiphase par un algorithme MDGP	119
4.1	Avant-Propos	120
4.2	Problématique	120
4.3	Implémentation de la Topologie et des Condensateurs	121
4.4	Modélisation des Inductances Vishay	125
4.5	Modélisation des Transistors	130
4.5.1	Génération des modèles de transistors	130
4.5.2	Implémentation des variables de Φ_{MOSFET}	134
4.5.3	Gestion de la température de jonction, dissipation thermique	136
4.5.4	Améliorer l'encadrement des variables de substitution	138
4.6	Modélisation de la structure du convertisseur : PCB, Busbars, assemblage	139
4.7	Optimisation pour plusieurs points de fonctionnement	141
4.8	Optimisation Multi-Objectifs	143
4.9	Présentation des Résultats	144
4.10	Nombre d'opérations et Temps de calculs	149
4.11	Conclusion du chapitre	150
	Conclusion générale & Perspectives	151
	Annexe A	153
A.1	.lib SPICE component files for transistors	153
A.2	Prises de vue à l'oscilloscope	155
	Bibliographie	158

Table des figures

1	Représentation des spécifications du convertisseur	15
2	Caractéristiques Puissance vs Tension de la PaC en fonction de son vieillissement (extrait du cahier des charges)	16
1.1	Schéma électrique de principe d'un <i>BUCK</i>	20
1.2	Chronogrammes d'un convertisseur <i>BUCK</i>	21
1.3	Schéma électrique de principe d'un <i>BUCK</i> entrelacé	22
1.4	Rendement d'un convertisseur entrelacé à 4 phases, en fonction de la puissance de sortie, pour un nombre de phases actives variable [1]	23
1.5	Schéma électrique de principe et chronogramme d'un FC3L	24
1.6	Comparaison des taux d'ondulation de courant Δi_L (normalisés) en fonction du rapport cyclique D , pour des convertisseurs BUCK et FCML	26
1.7	Schémas électriques de principe des topologies <i>Double</i>	26
1.8	variante ZVS de la topologie <i>FC3L</i>	27
1.9	Résistance spécifique des composants Si, SiC, GaN et diamant en fonction de leur tension de claquage [2]	29
1.10	Domaines d'application des différentes technologies de transistors en fonction de la fréquence de commutation et de la puissance commutée [3]	30
1.11	Circuit de test en commutation des modèles transistors <i>LTSpice</i>	33
1.12	Modélisation <i>LTSpice</i> des drivers du circuit de test (fig. 1.11)	34
1.13	Simulation de commutations sous <i>LTSpice</i> avec des transistors <i>EPC2022</i> , $V_{in} = 37.5V$, $I_{load} = 35A$	35
1.14	Résultats <i>LTSpice</i> pour E_{off} du composant <i>EPC2022</i> , $t_{dt} = 20ns$, $R_{g, SRC} = 0.8\Omega$...	36
1.15	Densité d'énergie spécifique des inductances et capacités étudiées par [4]	37
1.16	Pertes joules et Volume des inductances <i>Vishay</i> en fonction de leur énergie à courant de saturation	38
1.17	Schéma de principe des géométries possibles pour une <i>phase</i> de conversion, en fonction de la topologie	40
1.18	Puissances massiques et Rendement des solutions évaluées (pas de remplissage : transistors commercialisés après 2020)	41
2.1	Variation de capacitance en fonction de la tension DC aux bornes de <i>GRM32EC72A106KE05</i> (datasheet)	47
2.2	Circuit équivalent de la commande de la grille d'un transistor	48
2.3	Chronogrammes de la commande de la grille d'un transistor	48
2.4	Circuit équivalent d'une cellule de commutation au <i>turn-on</i> du transistor <i>High Side</i> ..	51
2.5	Perturbation de la grille du transistor low-side lors du <i>turn-on</i> du high-side	53

2.6	Comparaison des v_{gs} mesurés sur un transistor <i>EPC2022</i> pour les drivers <i>Si27GB-IS</i> et <i>2EDF7275KXUMA1</i> , pour $V_{ds} = 0V$, $I_{ds} = 0A$	55
2.7	Schéma de principe d'une alimentation en <i>bootstrap</i> des drivers [5]	56
2.8	Impédances parasites sur un interrupteur composé de deux transistors parallélisés . . .	58
2.9	Configurations de couplage entre les boucles de puissance et de grille [6]	60
2.10	Boucles de commutation typiques [7]	61
2.11	Circuit d'une cellule de commutation à condensateurs fractionnés	61
2.12	Schématisation du routage de deux transistors <i>EPC2022</i> parallélisés	61
2.13	Vue en coupe de la cellule de commutation 1 pour les deux stratégies de routage étudiées	62
2.14	Vue du dessus (partielle) de la cellule de commutation 1 du routage <i>Adjacent</i> , circulation du courant i_L en fonction du chemin emprunté	63
2.15	Aperçus du routage <i>Adjacent</i> sous <i>Altium</i>	64
2.16	Aperçus du routage <i>Traversant</i> sous <i>Altium</i>	65
2.17	Densité de courants surfacique de la cellule de commutation 2 du Prototype <i>Adjacent</i> , simulation sous <i>Ansys Q3D</i> pour une stimulation $f_{Q3D} = 100MHz$ (vue du dessus) . .	67
2.18	Boucle de commutation problématique de la cellule 2, sur le prototype <i>Traversant</i> . . .	68
2.19	Ébauche du convertisseur entrelacé $20kW$	69
2.20	Circuit <i>LTSpice</i> incluant les inductances et capacités parasites de la boucle de commutation	72
2.21	Oscillations de la tension v_{ds} de l'interrupteur <i>Low-Side</i> des cellules 1 et 2 au <i>turn-off</i> du routage <i>Adjacent</i> , à $V_{in} = 75V$, $I_{out} = 70A$ (vues oscilloscope : Figures A.1 et A.2 en Annexe)	73
2.22	Oscillations de la tension v_{ds} de l'interrupteur <i>Low-Side</i> des cellules 1 et 2 au <i>turn-off</i> du routage <i>Traversant</i> , à $V_{in} = 20V$, $I_{out} = 1A$ (vues oscilloscope : Figures A.3 et A.4 en Annexe)	73
2.23	Modélisation de la dissipation thermique des 8 transistors du prototype	76
2.24	Assemblage de l'interface thermique sur un transistor	76
2.25	Prise de vue de l'assemblage de l'interface thermique sur un dissipateur	76
2.26	Images thermiques du prototype A, pour $V_{in} = 75V$, $V_{out} = 28V$, $P_{in} = 2kW$, $T_{amb} = 25^\circ C$, flux d'air $4m/s$	77
2.27	Principe de la méthode par Opposition	79
2.28	Description du contrôle du circuit en Opposition dans ses deux modes de fonctionnement	80
2.29	Formes d'ondes du circuit en Opposition dans ses deux modes de fonctionnement . . .	80
2.30	Formes d'ondes du mode de fonctionnement $E_{on} + E_{off}$ pour un <i>deadtime</i> $t_{dt} \neq 0$. . .	82
2.31	Circuit électrique du montage expérimental de la méthode d'Opposition	83
2.32	Montage expérimental de la méthode d'Opposition	85
2.33	Illustration des tensions $v_{L,i}$ et courants $i_{pm,i}$ et $i_{pm,f}$ pour le mode de fonctionnement $E_{on} + E_{off}$ en présence de $C_{out,i}$	86
2.34	Influence de ΔV_{Cfly} sur les harmoniques du courant i_{Cfly} (harmoniques i_L en pointillés) pour un convertisseur <i>FC3L</i> alimentant une charge résistive	89
2.35	Convergence de $V_{Cfly,1}$ et $V_{Cfly,2}$ à $\frac{V_{in}}{4}$ (simulation <i>PLECS</i>)	90
2.36	Rendement du prototype v2.1V (pertes driver et ventilation non incluses)	92
2.37	Pertes P_{loss} du prototype v2.1V (pertes driver et ventilation non incluses)	93
2.38	Energies de commutation E_{sw} de l' <i>EPC2022</i> déduites de P_{loss} en fonction de V_{ds} et I_{ds}	94
2.39	Comparaison de la mesure P_{loss} et de la fonction <i>fittée</i>	95

2.40	Rendement du convertisseur entrelacé, en fonction du vieillissement de la Pile à Combustible	96
3.1	Représentation des différents espaces d'un problème d'optimisation	100
3.2	Représentation d'une fonction convexe $f : \mathbb{R} \rightarrow \mathbb{R}$ et du segment entre les deux points $f(x_1)$ et $f(x_2)$	101
3.3	Exemple d'un arbre de décision <i>Branch & Bound</i> montrant l'ensemble des combinaisons possibles	107
3.4	Exemple d'exploration d'un algorithme <i>Branch & Bound</i>	108
3.5	Organigramme de programmation d'un algorithme <i>Branch & Bound</i>	111
3.6	Exemples de découpages de l'espace des solutions relaxé \mathcal{S}_0 par l'algorithme <i>Branch & Cut</i> jusqu'à convergence vers l'optimum global discret	113
3.7	Exemple de l'exploration des solutions pour un problème MDGP quelconque par l'algorithme MDGP	118
4.1	Variation de l'ESR du condensateur <i>GRM32EC72A106KE05</i> en fonction de la fréquence de stimulation (datasheet)	125
4.2	<i>Fitting</i> de E_{on} l'énergie de commutation au <i>turn-on</i> du transistor <i>EPC2022</i> en fonction de \tilde{V}_{ds} , \tilde{I}_{ds} et $\tilde{R}_{g,src}(\Omega)$, pour $\tilde{t}_{dt} = 30ns$ et $\tilde{R}_{g,snk} = 0.4\Omega$. En bleu les résultats de simulation <i>LTSPICE</i> , en rouge le <i>fit</i>	135
4.3	<i>Fitting</i> de $E_{diode,off}$ et $t_{diode,off}$ le temps et l'énergie dissipée en conduction inverse du transistor <i>EPC2022</i> en fonction de \tilde{V}_{ds} , \tilde{I}_{ds} et \tilde{t}_{dt} , pour $\tilde{R}_{g,src} = 0.8\Omega$ et $\tilde{R}_{g,snk} = 0.4\Omega$. En bleu les résultats de simulation <i>LTSPICE</i> , en rouge le <i>fit</i>	135
4.4	Représentation schématique d'un busbar pour $n_{phase} = 6$	140
4.5	Répartition des différentes sources de pertes pour chaque solution optimale et pour le Prototype, aux 3 points de fonctionnement évalués	146
4.6	Répartition des différentes sources de masse pour chaque solution optimale et pour le Prototype	147
4.7	Rendement moyen pondéré η_{avg} en fonction de la masse m_{conv} du Front de Pareto et des solutions sous-optimales rencontrées les plus performantes, groupées par transistor et niveau N de la topologie (zoomé pour $\eta_{avg} \geq 98\%$)	148
A.1	Mesure de $v_{ds,LS1}$ au <i>turn-off</i> sur le Prototype <i>Adjacent</i> , pour $V_{in} = 75V$, $I_{out} = 70A$	155
A.2	Mesure de $v_{ds,LS2}$ (pris sur v_{pm}) au <i>turn-off</i> sur le Prototype <i>Adjacent</i> , pour $V_{in} = 75V$, $I_{out} = 70A$	155
A.3	Mesure de $v_{ds,LS1}$ au <i>turn-off</i> sur le Prototype <i>Traversant</i> , pour $V_{in} = 20V$, $I_{out} = 1A$	156
A.4	Mesure de $v_{ds,LS2}$ (pris sur v_{pm}) au <i>turn-off</i> sur le Prototype <i>Adjacent</i> , pour $V_{in} = 20V$, $I_{out} = 1A$	156
A.5	Mesure des tensions et courants du circuit d'Opposition du prototype <i>Adjacent</i> (sans le moyennage sur 50 échantillons) pour $V_{in} = 80V$, $I_{pmf} = 69A$	157

Liste des tableaux

1.1	Formules d'intérêt dans le dimensionnement d'un convertisseur <i>FCML</i>	25
1.2	Propriétés des semiconducteurs Grands Gap	28
1.3	Propriétés typiques des transistors sélectionnés, pour $T_j = 25^\circ C$	31
1.4	Comparaison des performances des solutions présentant le meilleur rendement pour chaque transistor & topologie, pour $V_{in} = 80V$, $V_{out} = 28V$ et $P_{in} = 20kW$	42
1.5	Impact de la parallélisation directe de transistors sur la Puissance massique, $V_{in} = 80V$, $V_{out} = 28V$ et $P_{in} = 20kW$	42
2.1	Caractéristiques du prototype développé	46
2.2	Comparaison des performances de deux drivers d'après leurs datasheets	55
2.3	Résultats des simulations <i>Ansys Q3D</i> pour une stimulation $f_{Q3D} = 100MHz$	66
2.4	Équipement expérimental	69
2.5	Résultats des simulations <i>Ansys Q3D</i> pour une stimulation $f_{Q3D} = 100MHz$	71
2.6	Comparaison des valeurs d'inductance L_{loop} obtenues par simulation et mesure	74
2.7	Comparaison des valeurs de surtension OS_{ds} obtenues par simulation et mesure	74
2.8	Calcul des erreurs relatives de la méthode d'opposition pour 2 points de fonctionnement extrêmes	85
3.1	Comparaison des différentes méthodes de résolution de problèmes d'optimisation	103
4.1	Coefficients K_f et \mathring{K}_b associés à chaque matériau de ferrite des inductances <i>Vishay</i> ..	128
4.2	Transistors implémentés dans le <i>FuncStruct</i> Φ_{MOSFET}	131
4.3	Erreurs des fonctions <i>fittées</i> pour le transistor <i>EPC2022</i> , par rapport aux simulations <i>LTSPICE</i>	134
4.4	Caractéristiques des matériaux considérés pour les busbars, dans le <i>Tuple</i> Θ_{busbar}	140
4.5	Points de Fonctionnement étudiés dans notre problème MDGP	142
4.6	Rendement moyen pondéré η_{avg} , masse m_{conv} , niveau de la topologie N et transistors des solutions optimales sur le Front de Pareto	145
4.7	Performances de l'algorithme <i>MDGP</i> sur notre problème d'optimisation, par rapport à une estimation pour un algorithme <i>Brute Force + GP</i>	149

Introduction générale

Dans le contexte d'une raréfaction des ressources en énergies fossiles, et de l'impact de ces énergies sur le réchauffement climatique, l'industrie aéronautique doit faire face à de nombreux défis en ce début de XXI^{ème} siècle. D'après la Feuille de Route de décarbonation de l'aérien [8], le secteur de l'aéronautique a contribué à 5.6% des émissions totales de CO_2 de l'économie Française en 2019 (1.2% en ne comptabilisant que les vols domestiques), soit 24.2Mt de CO_2 émises dans l'année. Ces rejets devraient encore croître d'année en année, avec une croissance du trafic aérien de 4.7%/an pour la France, et de 5.7%/an au niveau mondial (taux moyen annuel sur la plage 2000-2019). Deux scénarios sont proposés par [8], visant respectivement à une diminution des émissions de CO_2 du secteur aérien entre 63% et 80% d'ici à 2050 (en prenant en compte la croissance du secteur aérien). Sont entre autres considérés le développement et déploiement de nouvelles technologies basées sur l'hydrogène. Notamment, l'électrification des aéronefs en utilisant comme source d'énergie électrique des Piles à Combustible hydrogène. Dans une étude [9], il est estimé qu'une propulsion électrique alimentée par des piles à combustibles pourrait permettre de réduire l'impact aéronautique (équivalent CO_2) d'un avion de 75% à 90%, par rapport à une propulsion kérosène actuelle.

Ces problématiques et les perspectives prometteuses d'amélioration du bilan carbone de l'aéronautique ont motivé le financement de projets de recherche et développement des technologies hydrogènes dans ce secteur. Parmi eux, le Projet PIPAA ("Piles à combustibles Pour Applications Aéronautiques") a été lancé en 2017. Il s'agit d'une collaboration entre plusieurs entreprises et organismes publics, dont le *Commissariat à l'Energie Atomique* (CEA), l'*Institut National Polytechnique de Toulouse* (Toulouse INP), l'*Ecole Nationale de l'Aviation Civile* (ENAC), piloté par *Safran Power Units* et cofinancé par la *Banque Publique d'Investissement* (BPI). Le Projet PIPAA fait partie du méta-projet HyPort de la région Occitanie, visant à développer l'aéroport de Blagnac (Toulouse) pour y déployer les technologies hydrogène et réduire son bilan carbone. La mission du Projet PIPAA consiste à étudier l'implémentation d'une Pile à Combustible hydrogène à bord d'un avion, dans le but de remplacer le Groupe Auxiliaire de Puissance, ou *Auxiliary Power Unit* (APU) et d'alimenter le réseau électrique embarqué notamment pendant les phases de "taxi" où les alternateurs des turbines sont éteints ou tournent à bas régime.

Dans le cadre de ce projet, 4 autres thèses ont été produites :

- *Etude de l'impact des caractérisations et des modes opératoires sur le vieillissement d'une pile à combustible à membrane échangeuse de protons haute température (PEM-HT)*, menée par M. Baudy et soutenue en Juin 2022 [10].
- *Étude et optimisation des phases de démarrage et d'arrêt des piles à combustible de type PEM haute température en vue de maximiser leur durée de vie*, menée par M. Durand et soutenue en Janvier 2022 [11].

- Dimensionnement optimal d'un système hybride direct pile à combustible PEM Haute Température (PEM-HT) - batterie et impact d'un tel système sur les interactions avec le convertisseur de puissance associé, menée par T. Jarry et soutenue en Juillet 2022 [12].
- Comportement des piles à combustibles haute température sans gaz actifs (y compris le pré-chauffage avec des courants alternatifs), menée par W. Rosinski.

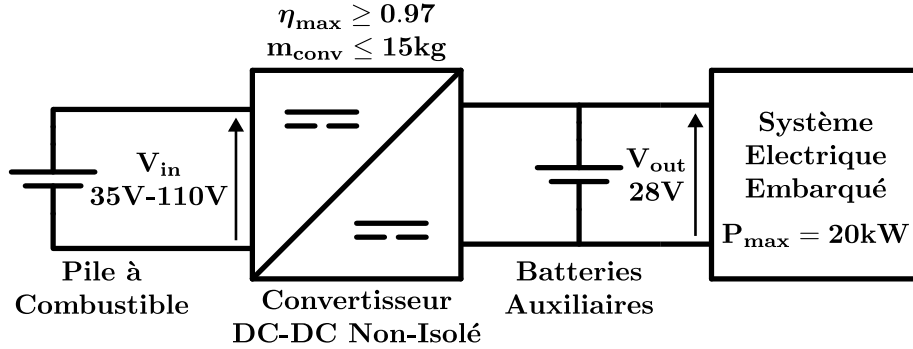


FIGURE 1 – Représentation des spécifications du convertisseur

Les travaux présentés dans ce manuscrit s'intéressent au développement du convertisseur statique DC-DC assurant l'interfaçage entre la Pile à Combustible du projet PIPAA, et le réseau électrique de l'avion. Le convertisseur statique doit présenter le meilleur rendement de conversion η et la plus faible masse m_{conv} possibles, en respectant les spécifications suivantes :

- $V_{in} \in [35VDC - 115VDC]$ La tension d'entrée, de la Pile à Combustible
- $V_{out} = 28VDC$ La tension de sortie, du réseau électrique embarqué
- $P_{in} \in [1kW - 20kW]$ La plage de puissance de la Pile à Combustible
- $I_{out} \in [35A - 715A]$ La plage de courants de charge admissibles

Et les contraintes suivantes :

- $\eta \geq 97\%$ Le rendement typique minimal (idéalement $\eta \geq 98\%$)
- $m_{conv} \leq 15kg$ La masse maximale admissible (idéalement $m_{conv} \leq 5kg$)
- $V_{conv} \leq 15L$ Le volume maximal admissible
- $\Delta v_{in,max} \leq \pm 5\%$ Le taux d'ondulation de la tension d'entrée V_{in} (idéalement $\Delta v_{in,max} \leq \pm 1\%$)
- $\Delta v_{out,max} \leq \pm 10\%$ Le taux d'ondulation de la tension de sortie V_{out}
- $10kHz \leq f_{sw}$ La fréquence minimale de l'ondulation du courant I_{in}
- Dissipation thermique par un système de refroidissement par air
- Convertisseur DC-DC non-isolé

Ces spécifications sont résumées en Figure 1. La tension V_{in} dépend du courant I_{in} et de l'état de vieillissement de la Pile. La Figure 2 décrit cette caractéristique, nous nous intéresserons particulièrement aux points de fonctionnements 1, 2 et 3 indiqués. Le point 1 correspond à une

puissance moyenne de $15kW$ pour une pile en fin de rodage, soit $V_{in} = 80V$. Le point 2 correspond à un fonctionnement à puissance nominale $20kW$, $\approx 80V$ en début de vie de la pile. Enfin, le point 3 est le point de fonctionnement à tension maximale $V_{in} = 110V$ pour une puissance minimale de $1kW$. La puissance maximale admissible par la Pile à Combustible est $P_{in,max} = 20kW$, soit un courant maximal en sortie de convertisseur $I_{out,max} = 710A$ très élevé. La gestion de ce courant va représenter un enjeu majeur dans la conception du convertisseur.

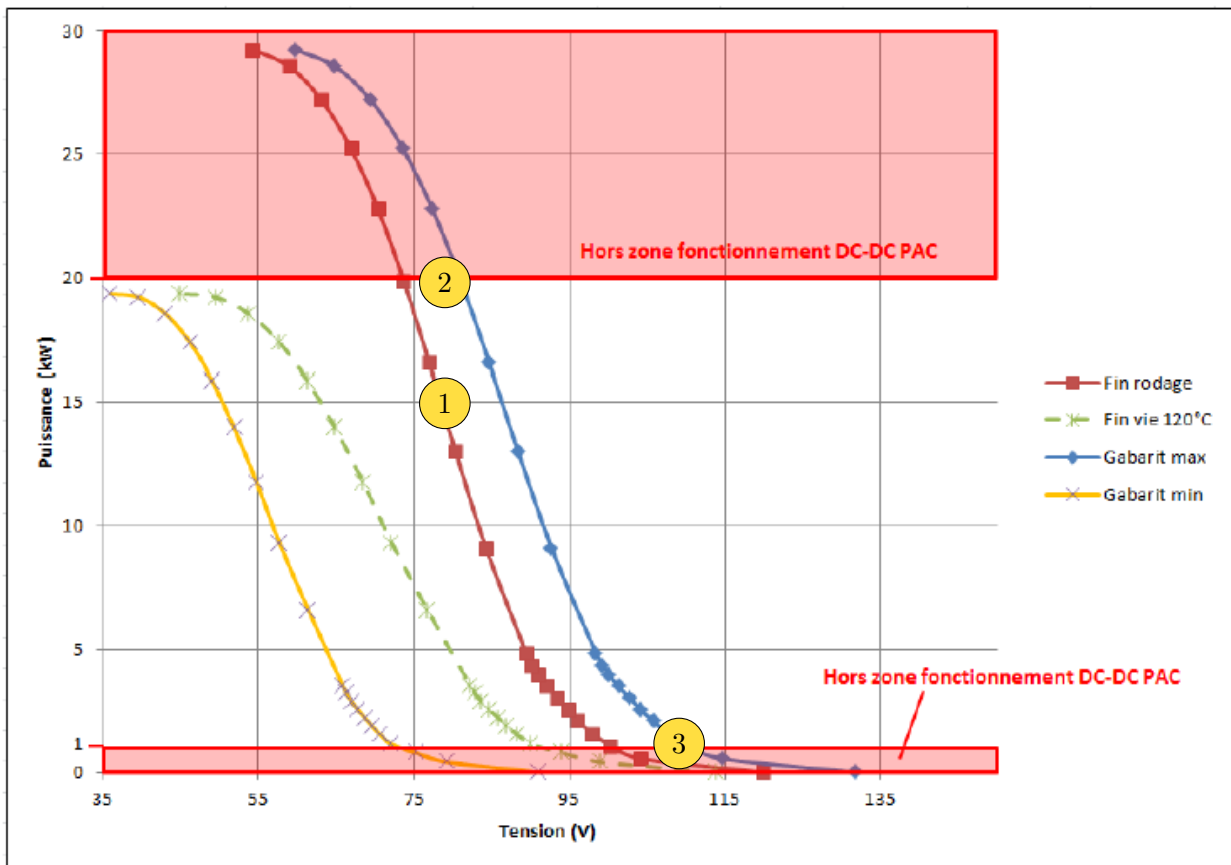


FIGURE 2 – Caractéristiques Puissance vs Tension de la PaC en fonction de son vieillissement (extrait du cahier des charges)

L'objet de cette thèse est donc le développement d'un Convertisseur Electrique DC-DC moyenne puissance, forts courants, pour une application aéronautique décrite par le cahier des charges du projet PIPAA. Les contraintes élevées sur la masse, le volume et le rendement du convertisseur vont guider nos choix de conception. Nous présenterons d'abord une première approche de design du convertisseur au **Chapitre 1**. Les grands choix de topologie, de transistors, des composants passifs, etc. y seront listés et une évaluation des performances en masse et rendement y sera réalisée afin de pouvoir déterminer le design le plus intéressant pour notre application. Pour compléter cette approche théorique, un prototype basé sur ce design jugé optimal sera conçu. Les principaux enjeux dans la conception matérielle du prototype seront présentés, et une série de simulations par éléments finis et de mesures expérimentales visant à évaluer ses caractéristiques électriques et son rendement seront effectuées au **Chapitre 2**.

Il apparaîtra que la méthodologie de conception décrite au Chapitre 1 est imparfaite. Son manque de flexibilité et d'exhaustivité ne permet pas de correctement estimer les meilleurs choix de conception d'un convertisseur. Le **Chapitre 3** s'intéressera aux différents types d'algorithmes d'optimisation. Nous y développerons un nouvel algorithme d'optimisation *Mixed-Discrete Geometric Programming*, permettant de résoudre dans un temps raisonnable des problèmes physiques, similaires à notre problème d'optimisation du design d'un convertisseur électrique, composés de nombreuses variables discrètes et continues. L'implémentation de notre problème d'optimisation de design dans cet algorithme MDGP sera décrite au **Chapitre 4**. Nous y décrirons les formulations à adopter pour modéliser un convertisseur DC-DC, les processus pour générer des modèles de composants électriques compatibles avec l'algorithme, ainsi que les résultats produits pour notre problème d'optimisation.

Chapitre 1

Première approche de conception

1.1 Description de la méthodologie

Lors de la conception d'un convertisseur, le Cahier des Charges liste les grandeurs importantes à optimiser (en vue de sélectionner la meilleure solution), et les grandeurs contraignantes à respecter (cf. Introduction).

Variables objectif :

- Rendement η [sans unité]
- Masse m_{conv} [kg], Puissance massique $\rho_{conv} = P_{nominal}/m_{conv}$ [kW/kg]
- Volume V_{conv} [L], Puissance volumique $p = P_{nominal}/V_{conv}$ [kW/L]

Variables contraintes :

- Tensions maximales
- Courants maximaux
- Taux d'ondulation maximal de la tension de sortie $\Delta v_{out,max}$ [sans unité]

Une fois les objectifs et contraintes du projet correctement formulés, nous retiendrons quelques topologies de convertisseurs adaptées au besoin. Puis nous sélectionnerons une liste de composants passifs et de transistors adaptés à chacune de ces topologies. Nous dresserons un modèle d'évaluation des pertes pour chaque topologie. Enfin, nous calculerons les performances pour chaque combinaisons de topologies et de composants. Une comparaison à la fois quantitative et qualitative des performances selon les différents critères permet ensuite de retenir une ou plusieurs solutions intéressantes pour une mise en œuvre expérimentale.

Dans le cadre de cette étude, nous nous limiterons à une sélection restreinte de composants et topologies de convertisseurs, l'objectif à ce stade de nos travaux consistant à trouver une solution intéressante dans un temps court, afin de pouvoir rapidement développer des prototypes. Les choix de conception seront justifiés tout au long du chapitre. Puis une comparaison entre plusieurs solutions évaluées à partir de cette sélection permettra de mettre en évidence certaines tendances afin de justifier le développement d'un prototype alliant haut rendement et forte compacité.

1.2 Topologies de convertisseurs DC-DC non-isolées

Les topologies de convertisseurs DC-DC explorées se limiteront aux convertisseurs non-isolés. Il s'agit d'abord d'une exigence du cahier des charges. Étant donné les fortes contraintes sur la compacité et densité de puissance du convertisseur, les topologies isolées présentent l'inconvénient de nécessiter des éléments passifs (transformateurs) qui viendraient dégrader ces deux propriétés.

Les convertisseurs à résonance ne seront pas non plus explorés. Comme nous pouvons l'observer sur les caractéristiques de la Pile à Combustible (Fig.2), la plage de fonctionnement du convertisseur varie fortement en fonction de la puissance absorbée par le réseau électrique embarqué et par le vieillissement de la Pile. Or, les convertisseurs à résonance sont dimensionnés par le choix de leurs éléments passifs, qui ont un impact majeur sur le rendement. De fait, ces convertisseurs sont davantage conçus pour fonctionner à un point de fonctionnement précis car leurs performances se dégradent rapidement en dehors du point pour lequel ils ont été optimisés.

Nous rappellerons ici le fonctionnement général du convertisseur *BUCK*, puis explorerons les topologies dites *Flying Capacitors Multilevel Converters* (FCML) et *Dual*.

1.2.1 Convertisseur BUCK

Il existe de multiples topologies de convertisseurs capables de réaliser une conversion DC-DC. La plupart de ces structures reposent sur le principe du *Hacheur*. Dans sa configuration abaisseur de tension ($V_{in} > V_{out}$), on parle de topologie *BUCK* (Fig. 1.1). Le convertisseur *BUCK* est largement employé car il présente l'avantage d'être très simple à développer et à mettre en œuvre, avec une littérature très fournie à son sujet.

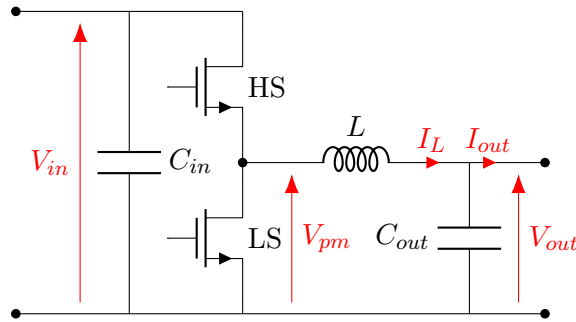


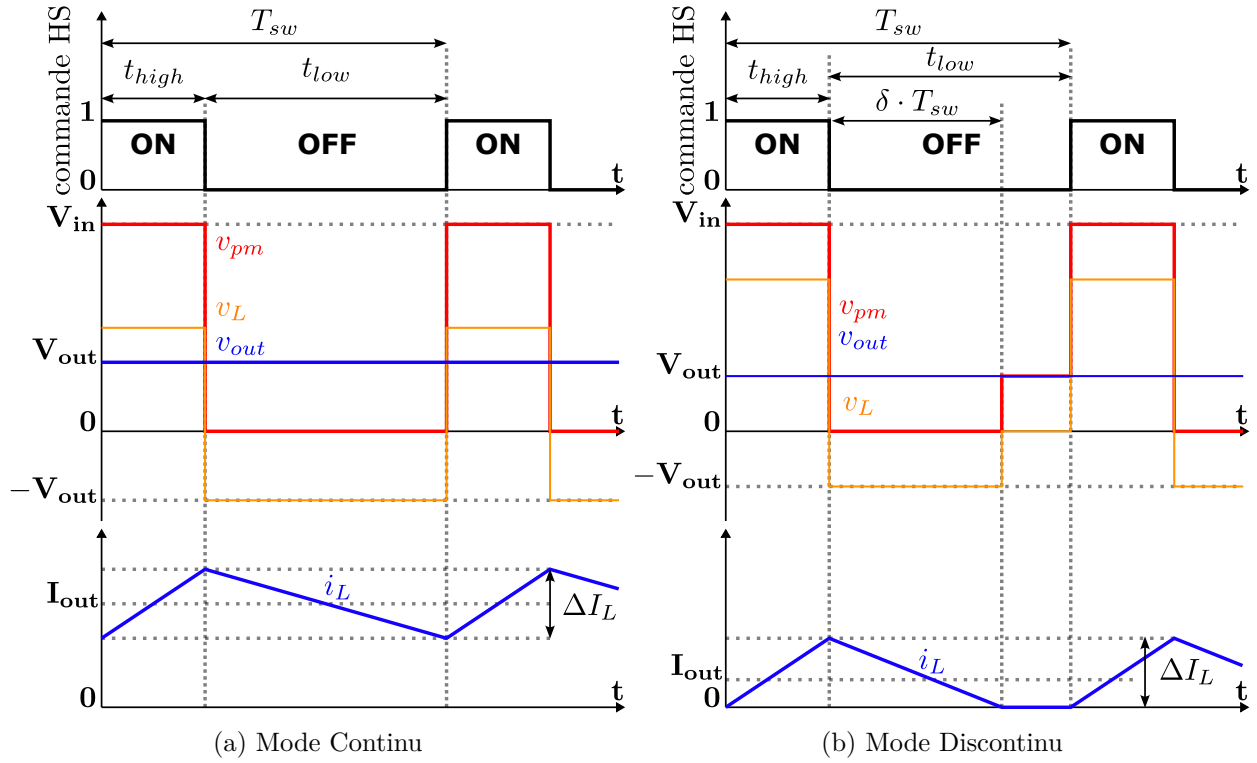
FIGURE 1.1 – Schéma électrique de principe d'un *BUCK*

Le principe de fonctionnement est représenté par le chronogramme Fig. 1.2a. Deux interrupteurs, High Side (HS) et Low Side (LS), sont commandés en opposition : quand HS est passant, LS est bloquant et inversement. La paire d'interrupteurs forme une unité appelée "*cellule de commutation*". La cellule de commutation est commandée par un signal binaire périodique à la période T_{sw} , de fréquence f_{sw} , c'est la commande par *Modulation de largeur d'impulsion* (PWM). Quand le signal PWM est à la valeur 1 pour une durée t_{high} , l'interrupteur HS est passant. L'interrupteur LS est ensuite passant pour une durée t_{low} . On qualifie de *Rapport Cyclique* de la commande PWM le ratio $D = t_{high}/T_{sw}$.

La cellule de commutation "découpe" la tension d'entrée du convertisseur V_{in} à la fréquence f_{sw} , générant une tension alternative carrée V_{pm} entre le *point milieu* (source de HS, drain de LS) et la masse. Un filtre LC passe-bas permet de générer la tension continue de sortie V_{out} à partir de la valeur moyenne de V_{pm} tel que : $V_{out} = D \cdot V_{in}$.

Le dimensionnement d'un convertisseur *BUCK* implique de choisir des transistors dont la tension de claquage BV_{ds} est supérieure à $V_{in,max}$. Dans des applications à tension d'entrée V_{in} relativement élevée, cette contrainte peut rapidement devenir problématique. Tendanciellement, plus le BV_{ds} d'un transistor est élevé, plus ses autres caractéristiques (résistance à l'état passant R_{dsON} , énergies perdues à la commutation E_{sw} , volume...) se voient dégradées [13]. Le choix du transistor est déterminant dans la minimisation des pertes du convertisseur, donc dans l'optimisation de son rendement η . Un découpage d'une tension élevée provoque des difficultés de Compatibilité Électromagnétique (CEM), cette problématique n'est toutefois pas l'objet de nos travaux.

Les pertes du convertisseur vont déterminer le dimensionnement de son système de refroidissement. Des pertes plus élevées signifient un dissipateur thermique plus volumineux et lourd, et l'emploi d'un ventilateur qui vient ajouter sa propre consommation électrique aux pertes purement dissipatives du convertisseur.


 FIGURE 1.2 – Chronogrammes d'un convertisseur *BUCK*

Enfin, le dimensionnement du filtre LC pour respecter les contraintes de taux d'ondulation de tension et courant de sortie est déterminant dans l'optimisation de la masse et du volume du convertisseur. Il est généralement admis que minimiser les valeurs de capacitance et l'inductance requises pour respecter les contraintes d'ondulation permet de réduire la masse du convertisseur. Dans le cadre de ces travaux, nous nous intéresserons au fonctionnement du convertisseur *BUCK* en Mode Continu (Fig. 1.2a), ce qui impose que, à un courant de charge I_{out} déterminé, Δi_L le taux d'ondulation normalisé du courant d'inducteur respecte le critère suivant :

$$\frac{\Delta i_L \cdot I_{out}}{2} \leq I_{out} \Leftrightarrow \Delta i_L \leq 2 \quad (1.1)$$

Les valeurs d'inductance et de capacitance de sortie minimales requises en Mode Continu (Fig. 1.2a), en fonction du point de fonctionnement du convertisseur, de la fréquence de commutation et des taux d'ondulation désirés, sont données par les relations suivantes :

$$L \geq \frac{V_{in} \cdot D(1-D)}{f_{sw} \cdot I_{out} \cdot \Delta i_L} \quad (1.2)$$

$$L_{inf} = \frac{V_{in}}{4f_{sw} \cdot I_{out} \cdot \Delta i_L} \quad (1.3)$$

$$C_{out} \geq \frac{V_{in} \cdot D(1-D)}{8L \cdot f_{sw}^2 \cdot V_{out} \cdot \Delta v_{out}} \quad (1.4)$$

$$C_{out,inf} = \frac{V_{in}}{32L \cdot f_{sw}^2 \cdot V_{out} \cdot \Delta v_{out}} \quad (1.5)$$

A gauche, les valeurs minimales de L et C_{out} pour un rapport cyclique D donné. A droite, la plus grande des valeurs minimales possibles en fonction de D , aussi appelé *infimum*, atteintes pour $D = 0.5$.

La caractérisation de la pile à combustible (Fig. 2) indique que la tension d'entrée du convertisseur V_{in} peut varier de 35V à 110V, tandis que la tension de sortie V_{out} doit être maintenue à 28V. Le rapport cyclique D peut donc varier de 0.25 à 0.8, une plage qui inclue la valeur critique $D = 0.5$. Le dimensionnement des éléments passifs doit donc se baser sur les équations 1.3 et 1.5, tel que $L \geq L_{inf}$ et $C_{out} \geq C_{out,inf}$.

1.2.2 Convertisseurs entrelacés

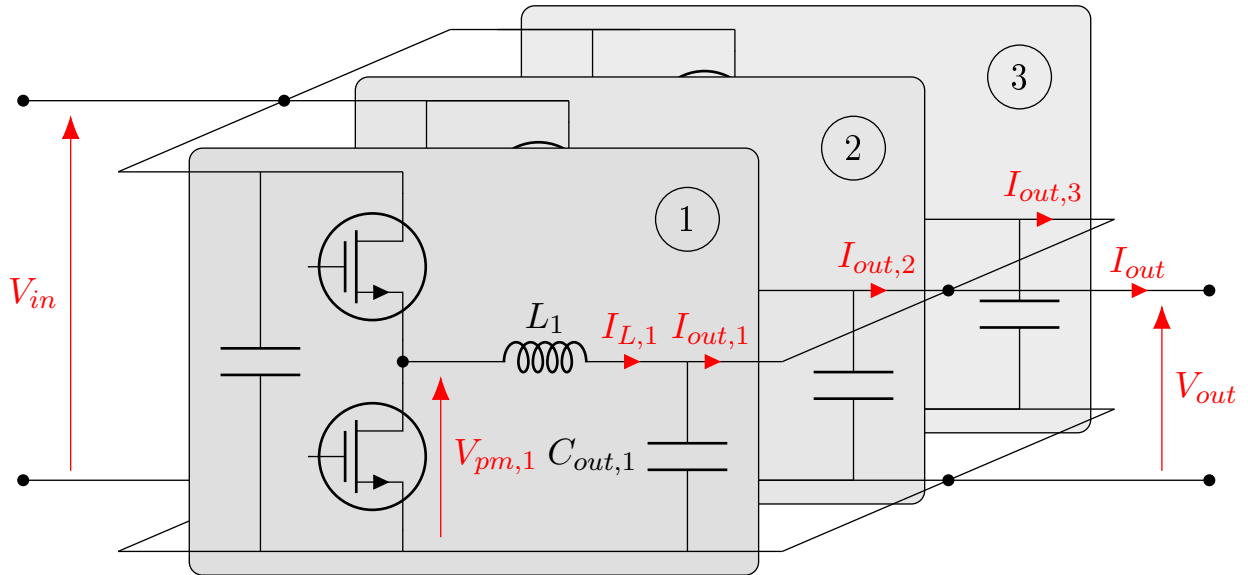


FIGURE 1.3 – Schéma électrique de principe d'un *BUCK* entrelacé

Paralléliser plusieurs circuits de conversion permet de répartir le courant entre eux, de diminuer les pertes joules et le stress thermique qu'une structure *BUCK* simple engendrerait. On parlera ici de *phases* pour nommer les sections parallélisées. Ces structures peuvent aussi bien paralléliser des circuits *BUCK* que d'autres topologies de convertisseurs. Si le filtre Capacitif C_{out} est placé en sortie de la structure parallélisée, en déphasant la commande PWM de chaque phase de $2\pi/n_{phase}$ (soit n_{phase} le nombre de phases du circuit), les ondulations des courants de sortie $I_{out,i}$ de chacune des phases auront tendance à s'annuler entre elles, une fois additionnées dans I_{out} [1, 14]. L'équation 1.6 donnera le taux d'ondulation normalisé pour une topologie *BUCK* entrelacé, en fonction du nombre de phases n_{phase} . Cet effet peut permettre de réduire le besoin en inductance de filtrage L , et donc d'alléger le convertisseur.

$$\Delta i_{out} = \frac{V_{in} \cdot \left(D - \frac{i_{phase} - 1}{n_{phase}} \right) \left(\frac{i_{phase}}{n_{phase}} - D \right)}{n_{phase} \cdot f_{sw} \cdot I_{out,phase} \cdot L_{phase}} \quad (1.6)$$

Où L_{phase} et $I_{out,phase}$ sont respectivement les valeurs de l'inductance et du courant de sortie moyen d'une phase telles que $L = L_{phase}/n_{phase}$ et $I_{out,phase} = I_{out}/n_{phase}$, et l'indice i_{phase} est un entier tel que :

$$\frac{i_{phase} - 1}{n_{phase}} < D \leq \frac{i_{phase}}{n_{phase}} \quad (1.7)$$

La Figure 1.3 représente un convertisseur entrelacé composé de 3 circuits BUCK parallélisés. Chaque circuit BUCK est doté de son propre filtre capacitif $C_{out,i}$ (donc $i_{out} \neq i_L$), ce qui facilitera la mise en œuvre d'un tel convertisseur lors de la phase de prototypage décrite au Chapitre 2 (plus modulaire, commande simplifiée) mais ne bénéficie pas des phénomènes décrits ci-dessus. Il est aussi possible de coupler les inductances entre phases, pour en réduire encore la masse [1]. Le convertisseur multiphase permet aussi une plus grande adaptabilité aux points de fonctionnement du convertisseur. Dans le cas d'une utilisation sur une charge variable, en désactivant un nombre de phases adapté, nous pouvons améliorer le rendement global du convertisseur par rapport à un fonctionnement statique avec toutes les phases actives (Fig. 1.4, [1, 15–17]). On parle de stratégies de *phase shedding*. Néanmoins, l'emploi d'une stratégie de phase shedding limite les bénéfices des structures multiphase précédemment évoqués (besoin de respecter la contrainte d'oscillation même quand une seule phase est active, difficultés à implémenter des inductances couplées).

Étant donné le courant de charge I_{out} très élevé que doit pouvoir gérer le convertisseur (jusqu'à 710A), les topologies entrelacées sont très attrayantes. En parallélisant jusqu'à 20 phases, nous pouvons réduire le courant de charge maximum par phase à 36A, ce qui permet d'employer des transistors de plus petits calibres, de diminuer la concentration du flux thermique (donc de faciliter le refroidissement), et globalement de réduire les pertes par conduction du convertisseur.

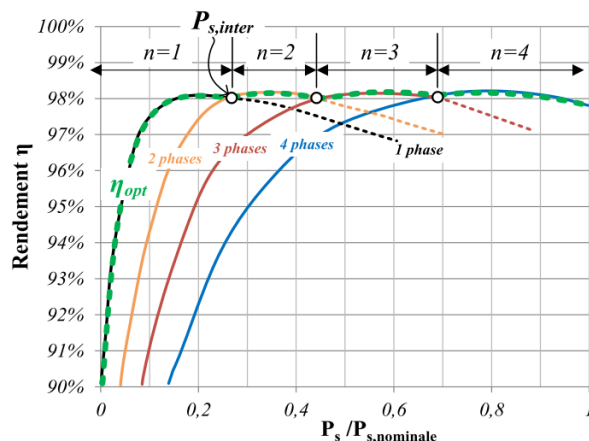


FIGURE 1.4 – Rendement d'un convertisseur entrelacé à 4 phases, en fonction de la puissance de sortie, pour un nombre de phases actives variable [1]

1.2.3 Convertisseur Multilevel Flying Capacitor

Les topologies de *Convertisseurs Multiniveaux à Capacités Flottantes* (Flying Capacitor Multilevel Converters (FCML)), décrites pour la première fois en 1992 [18], et largement employées aujourd'hui [14, 19, 20], permettent de réduire la tension V_{ds} aux bornes des transistors du convertisseur, ce qui permet ensuite de sélectionner des transistors ayant une tension de claquage BV_{ds} plus basse et des pertes par commutation et conduction plus faibles.

Le principe de fonctionnement des FCML consiste à connecter en série plusieurs cellules de commutation, les capacités flottantes $C_{fly,i}$ sont placées entre chaque cellule pour assurer la réduction effective de la tension V_{ds} aux bornes des transistors. Le niveau N du convertisseur est égal au nombre de cellules + 1. Un *BUCK* simple ne possède qu'une seule cellule de commutation, soit $N = 2$. La topologie *FC3L* (Fig. 1.5) en a 2, soit $N = 3$.

Soit i l'index d'une cellule de commutation tel que $i = 1$ désigne la cellule dont la tension d'entrée est V_{in} , et $i = N - 1$ la cellule dont la tension de sortie est V_{pm} (et qui n'a pas de capacité flottante).

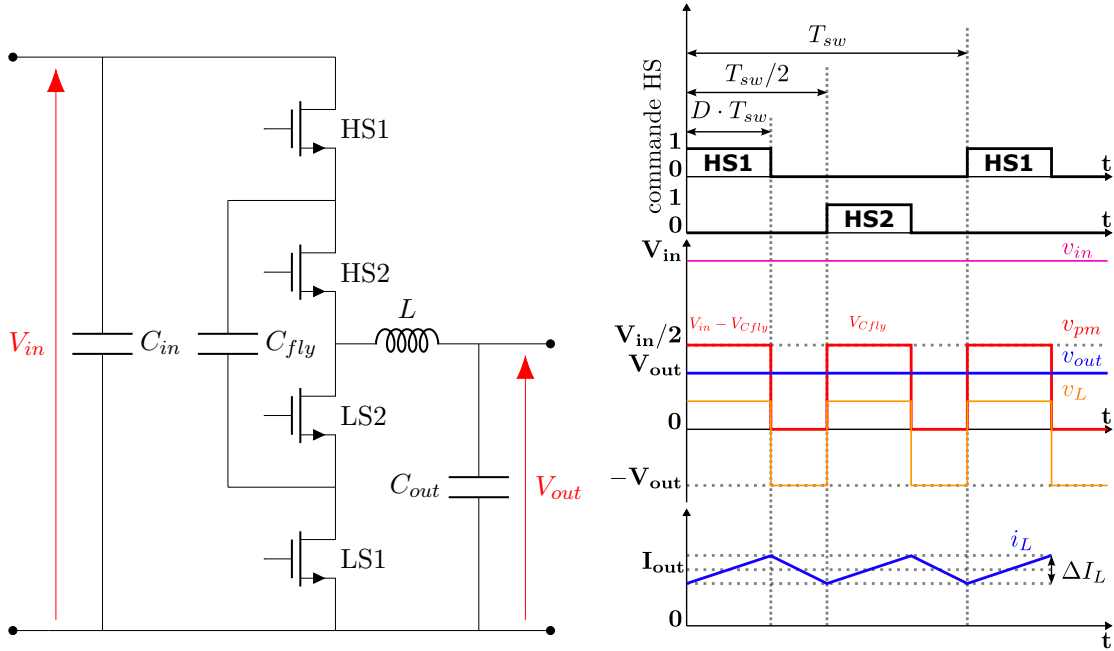


FIGURE 1.5 – Schéma électrique de principe et chronogramme d'un FC3L

Pour un convertisseur *FCML* de niveau N , le rapport cyclique D est inchangé par rapport au *BUCK*, soit $D = V_{out}/V_{in}$. En revanche, chaque cellule de commutation est commandée avec un déphasage de $\phi = 2\pi/(N - 1)$ l'une par rapport à l'autre. Ce déphasage permet d'augmenter f_{pm} la fréquence de la tension découpée au point milieu V_{pm} , et donc de l'ondulation de I_L que doit tenir l'inductance L , tel que $f_{pm} = (N - 1)f_{sw}$.

Le taux d'ondulation du courant I_L en fonction du niveau du convertisseur *FCML* est représenté Fig. 1.6. Nous observons une réduction très forte (d'un facteur 4) de Δi_L , à paramètres de dimensionnement constants entre une topologie *BUCK* ($N = 2$) et *FC3L* ($N = 3$). Les convertisseurs *FCML* présentent plusieurs régions de fonctionnement, désignées par l'indice i_D tel que :

$$\frac{i_D - 1}{N - 1} < D \leq \frac{i_D}{N - 1} \quad (1.8)$$

La région de fonctionnement a une influence sur $I_{Cfly,i}$ le courant efficace traversant chaque capacité flottante. Cette influence est représentée par le coefficient X tel que :

$$X = \begin{cases} D, & \text{si } i_D = 1 \\ \frac{1}{N-1}, & \text{si } 1 < i_D < N - 1 \\ 1 - D, & \text{si } i_D = N - 1 \end{cases} \quad (1.9)$$

Les différentes grandeurs d'intérêt sont listées en Table 1.1.

TABLE 1.1 – Formules d'intérêt dans le dimensionnement d'un convertisseur *FCML*

Grandeurs	Définition	
Fonction de transfert	$V_{out} = D \cdot V_{in}$	(1.10)
Déphasage des cellules de commutation	$\phi_{1,i} = 2\pi \cdot \frac{i-1}{N-1}$	(1.11)
Tension drain-source	$V_{ds} = \frac{V_{in}}{N-1}$	(1.12)
Tension des Capacités Flottantes	$V_{Cfly,i} = V_{in} \left(1 - \frac{i}{N-1}\right)$	$i < N - 1$ (1.13)
Inductance de sortie L	$L \geq \frac{V_{in} \cdot (D - \frac{i_D-1}{N-1}) (\frac{i_D}{N-1} - D)}{f_{sw} \cdot I_{out} \cdot \Delta i_L}$	(1.14)
Infimum de L	$L_{inf} = \frac{V_{in}}{4(N-1)^2 \cdot f_{sw} \cdot I_{out} \cdot \Delta i_L}$	(1.15)
Capacitance d'entrée C_{in}	$C_{in} \geq \frac{I_L \cdot D(1-D)}{f_{sw} \cdot V_{in} \cdot \Delta v_{in}}$	(1.16)
Infimum de C_{in}	$C_{in,inf} = \frac{I_{out}}{4f_{sw} \cdot V_{in} \cdot \Delta v_{in}}$	(1.17)
Capacitance d'une Capacité Flottante	$C_{fly,i} = \frac{I_{out} \cdot X}{f_{sw} \cdot V_{Cfly,i} \cdot \Delta v_{Cfly}}$	$i < N - 1$ (1.18)
Capacitance de sortie C_{out}	$C_{out} \geq \frac{V_{in} \cdot (D - \frac{i_D-1}{N-1}) (\frac{i_D}{N-1} - D)}{8f_{sw}^2 (N-1) \cdot L \cdot V_{out} \cdot \Delta v_{out}}$	(1.19)
Infimum de C_{out}	$C_{out,inf} = \frac{V_{in}}{32f_{sw}^2 (N-1)^3 \cdot L \cdot V_{out} \cdot \Delta v_{out}}$	(1.20)

1.2.4 Convertisseurs Double BUCK & Double FCML

Une seconde approche pour augmenter le niveau de conversion d'une topologie consiste en la division de la tension d'entrée entre deux circuits de conversion superposés [21, 22]. Le cas d'un *Double BUCK* est représenté en Figure 1.7a. Pour une topologie *simple* de niveau de conversion N , sa variante *double* est de niveau $2N - 1$. La topologie *Double BUCK* permet donc de diviser par 2 la contrainte en tension sur les interrupteurs par rapport à un *BUCK* simple, tandis que sa variante *Double FC3L* (Fig. 1.7b) la divise par 4.

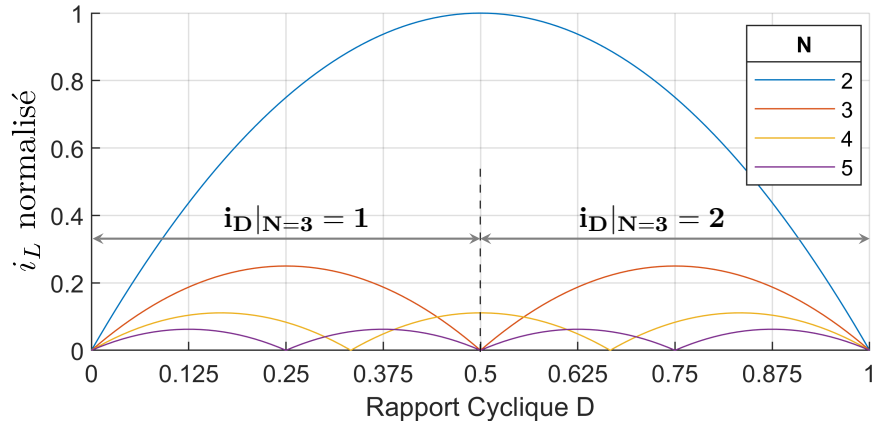


FIGURE 1.6 – Comparaison des taux d’ondulation de courant Δi_L (normalisés) en fonction du rapport cyclique D , pour des convertisseurs BUCK et FCML

Les deux circuits de conversion sont contrôlés de la même manière que leur topologie *simple* d’origine, mais avec un déphasage de π entre les circuits supérieur et inférieur. Le dimensionnement des éléments capacitifs et inductifs répond toujours aux équations 1.14 à 1.20. L est la somme des deux inductances de la topologie *Double*, ce qui revient à diviser par deux la valeur d’inductance de chaque bobine. Néanmoins, comme expliqué en section 1.7, les boîtiers des inductances disponibles à l’achat sont souvent standardisés pour une plage de valeurs inductives, donc multiplier par 2 le nombre d’éléments inductifs peut se traduire par un doublement de la masse et du volume total lié aux inductances.

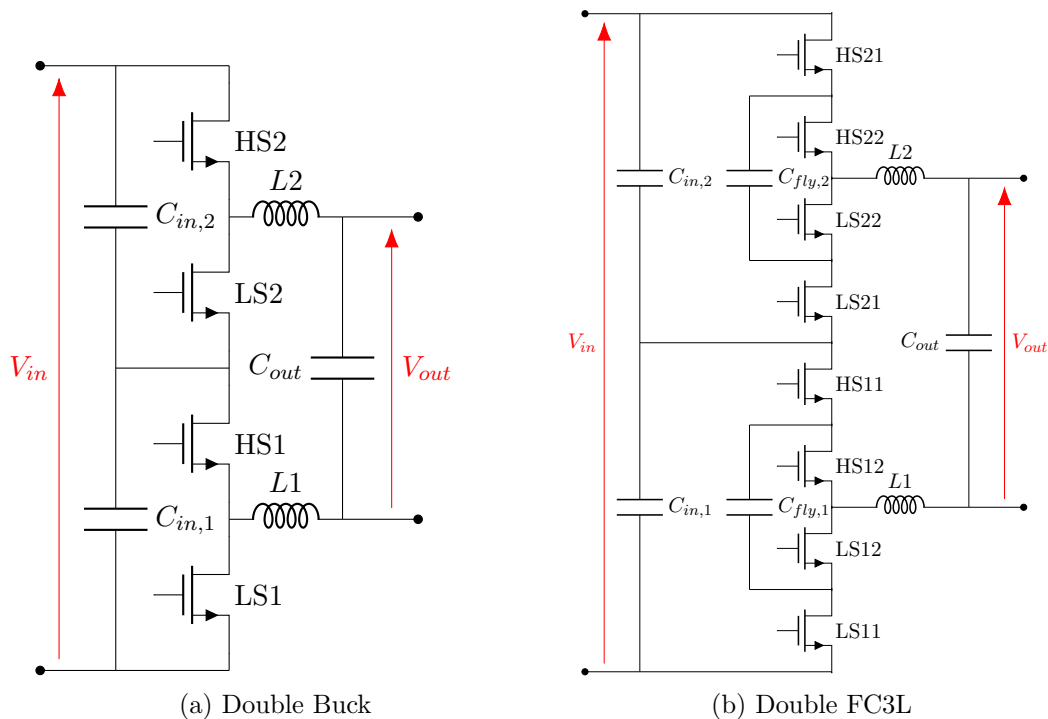


FIGURE 1.7 – Schémas électriques de principe des topologies *Double*

1.2.5 Variante ZVS de la topologie FC3L

Une variante de la topologie *Flying Capacitor* à 3 niveaux a été proposée par [23]. En ajoutant un circuit auxiliaire au *point milieu*, la répartition du courant de charge à travers le convertisseur est modifiée. Le circuit auxiliaire fournit au *point milieu* un courant alternatif i_{aux} à la fréquence $f_{sw} \cdot (N - 1)$, tel que le courant parcourant les interrupteurs $i_{ds} = i_L - i_{aux}$ soit négatif au moment de la commutation *turn-on* des transistors *High Side*. Ce courant négatif permet, avant la commutation, de décharger la tension V_{ds} desdits transistors à travers leurs diodes de corps. Lorsque la commutation est enclenchée, V_{ds} est déjà nulle, ce qui évite une commutation *hard-switching* et réduit les pertes par commutation.

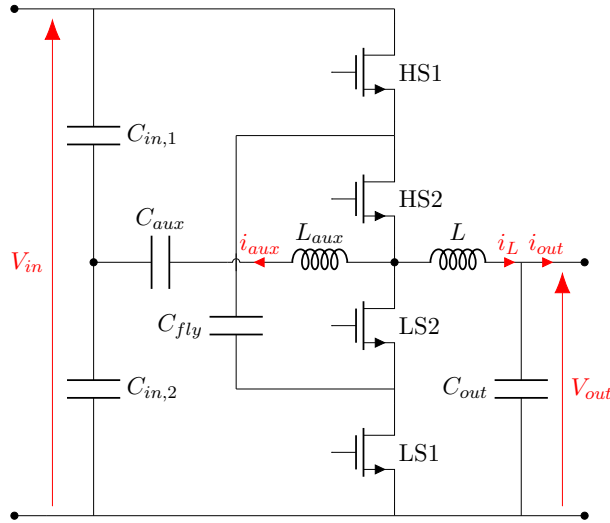


FIGURE 1.8 – variante ZVS de la topologie FC3L

Finalement, cette topologie permet de complètement supprimer les pertes *turn-on* de tous les transistors. La Topologie *BUCK* (et ses dérivées) produisent déjà des commutations *ZVS* au *turn-on* des transistors *Low Side*, cette topologie permet donc d'étendre cet effet aux transistors *High Side*. En revanche, elle ne supprime ni les pertes au *turn-off*, ni les pertes de recouvrement de la diode de corps. Il faut aussi anticiper le dimensionnement et les pertes propres associées à l'inductance auxiliaire. En effet, pour que le courant i_{ds} soit négatif lors du *turn-on* des transistors *High Side*, il faut respecter la condition de *ZVS* suivante :

$$I_{out} - \frac{\Delta I_L}{2} < \frac{\Delta I_{aux}}{2} \quad (1.21)$$

Ce qui implique :

$$L_{aux} < \frac{V_{in} \cdot (D - \frac{i_D - 1}{N-1})(\frac{i_D}{N-1} - D)}{f_{sw} \cdot (2I_{out} - \Delta I_L)} \quad (1.22)$$

L'équation 1.22 nous montre que plus le courant de charge I_{out} est élevé, et l'ondulation de l'inducteur en sortie Δi_L est faible, plus il faut réduire la valeur de L_{aux} pour augmenter l'amplitude du courant I_{aux} . Cela revient à augmenter considérablement les pertes par conduction alternatives et les pertes fer de l'inductance auxiliaire, sans n'avoir aucun impact réducteur sur les pertes de

l'inductance principale. L'augmentation de l'amplitude de I_{aux} se répercute aussi sur l'amplitude du courant I_{ds} traversant les transistors tel que $\Delta I_{ds} = 2I_{out}$. Cette topologie est adaptée pour des convertisseurs où le courant reste faible et où les principales pertes sont liées à la commutation des transistors. Étant donné les courants de charge I_{out} très élevés avec lesquels nous travaillons, cette topologie sera écartée de la suite de l'étude.

1.3 Transistors MOSFET

Le principe de fonctionnement de tous les convertisseurs statiques repose sur l'utilisation de semi-conducteurs, et plus spécifiquement de transistors. Ces transistors remplissent la fonction d'interrupteurs commandés électriquement. Nous nous intéresserons ici uniquement aux technologies *MOSFET* (Metal-Oxyde-Semiconductor Field-Effect Transistor), qui s'avéreront les plus adaptées à notre application basse tension.

Un MOSFET est principalement composé d'un substrat semi-conducteur et de trois terminaux électriques : la *source*, le *drain* et la *grille* (gate). Le courant traverse le composant du drain à la source à travers le *canal*, si la tension grille-source V_{gs} est supérieure à la tension de seuil V_{th} du composant. Le courant peut aussi circuler en sens inverse, de la source au drain, à travers une diode de corps de tension V_f , ou via le canal si la grille est active.

Les premiers transistors de puissance MOSFET à base de Silicium sont apparus à la fin des années 1970 et ont profondément changé l'industrie de l'électronique de puissance. Aujourd'hui, cette technologie est bien connue et employée dans de nombreuses applications. Néanmoins, les applications industrielles contemporaines requièrent des composants toujours plus performants, fiables et peu chers. Bien que les MOSFETs Silicium répondent parfaitement à ces derniers critères, le développement de composants Silicium approche aujourd'hui des limites physiques du matériau. L'utilisation de nouveaux matériaux dans la conception de transistors est un enjeu majeur de l'électronique de puissance.

TABLE 1.2 – Propriétés des semiconducteurs Grands Gap

	Si	SiC	GaN
Bande Interdite $E_g(eV)$	1.12	3.2	3.4
Champ Critique $E_{crit}(MV \cdot cm^{-1})$	0.3	3.5	3.3
Mobilité des électrons $\mu_n(cm^2 \cdot V^{-1} \cdot s^{-1})$	1500	650	990 ~ 2000
Permittivité ϵ_r	11.8	9.7	9

Les semi-conducteurs à larges bande interdite (*Wide Band-Gap semiconductors*, WBG) tels que le Carbure de Silicium (SiC) ou le Nitrure de Gallium (GaN) présentent des caractéristiques attractives résumées en Table 1.2. Grâce à leur valeur de bande interdite E_g , trois fois plus grande que pour le Silicium, il est possible d'utiliser le SiC et le GaN à des températures plus élevées ($\sim 100^\circ C$ pour le Si, contre $200^\circ C$ à $300^\circ C$ pour le SiC et le GaN, si leur packaging le permet). La tension de claquage BV_{ds} est corrélée avec l'énergie cinétique minimale requise pour que les électrons libèrent des porteurs supplémentaires en cas d'impact avec les atomes du réseau cristallin

du matériau. Une valeur de E_g plus élevée signifie qu'il faut plus d'énergie, donc une tension au borne du composant plus forte, pour engendrer le claquage du composant. Cela se traduit par une meilleure densité de puissance pour les composants SiC et GaN.

Le Champ Critique E_{crit} très élevé des semi-conducteurs WBG joue aussi un rôle majeur dans le gain en densité de puissance pour ces matériaux. Le SiC et le GaN peuvent supporter jusqu'à dix fois le champ électrique maximal toléré par des composants Si, ce qui peut grossièrement se traduire par une réduction de la distance entre le drain et la source d'un MOSFET d'un facteur 10, pour des composants conçus pour des tensions et des surface de conduction égales.

Des designs prometteurs de transistors SiC et GaN ont été développés et commercialisés ces deux dernières décennies. Bien que le transistor Si reste la technologie la moins coûteuse, ses performances sont déjà égalées et dépassées par ces nouveaux composants WBG, et ce malgré le manque de maturité sur ces technologies [2,24]. La Figure 1.9, issue de [2] présente la résistance spécifique à l'état passant de plusieurs composants, en fonction de leur tension de claquage. Chaque courbe montre la limite théorique d'un matériau. La limite de Nakagawa concerne les Si-IGBT [25]. Nous constatons la maturité des composants Si, dont plusieurs d'entre eux excèdent déjà les limites théoriques de leur matériau. A l'inverse, les composants GaN et SiC n'atteignent pas encore leurs limites théoriques, notamment à cause des difficultés liées au développement de *packaging* et de circuits adaptés aux performances extrêmes de ces matériaux [26]. Les composants diamant, quant à eux, présentent des caractéristiques encore plus intéressantes, mais la technologie est encore en phase de développement préliminaires, et rencontre notamment des difficultés pour tenir des courants supérieurs à 1A.

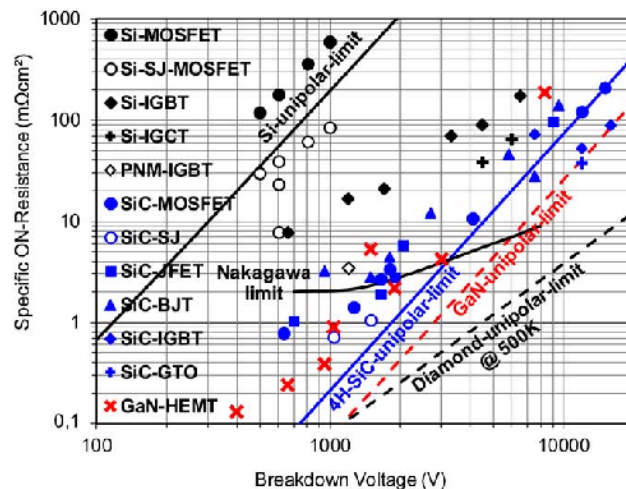


FIGURE 1.9 – Résistance spécifique des composants Si, SiC, GaN et diamant en fonction de leur tension de claquage [2]

Les matériaux SiC et GaN présentent des propriétés électriques supérieures à celles du Silicium, mais ces deux matériaux semi-conducteurs n'entrent eux-mêmes que rarement en compétition dans leurs applications. Les composants GaN sont généralement utilisés pour des applications où les fréquences de commutation de la puissance sont élevées (jusqu'au MHz), car ils permettent des vitesses de commutation des transistors très élevées (et donc des pertes par commutation faibles). Mais leurs designs actuels (HEMT, Cascode...) présentent plusieurs défauts qui les rendent impropres aux applications hautes tension, où le SiC est généralement préféré. La Figure 1.10 résume les usages de ces matériaux et du type de composants associés, en fonction de la puissance commutée et de la fréquence de découpage du convertisseur.

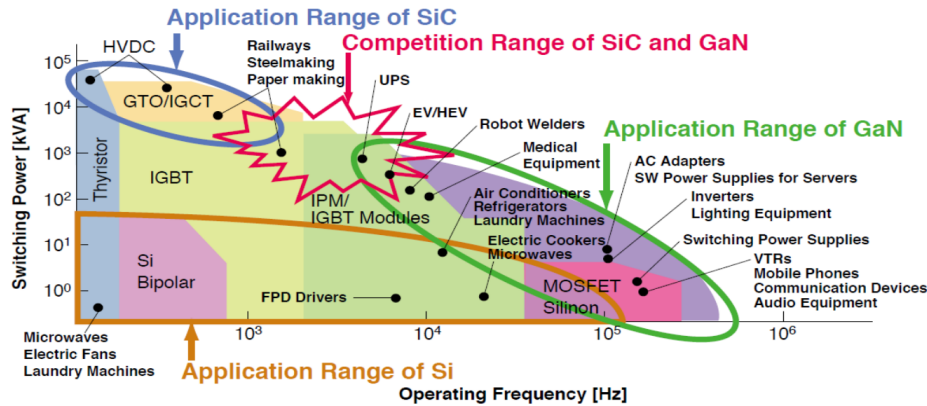


FIGURE 1.10 – Domaines d’application des différentes technologies de transistors en fonction de la fréquence de commutation et de la puissance commutée [3]

Dans le cadre de notre application basse tension, forts courants, décrite en Introduction, nous nous intéresserons uniquement au cas des transistors GaN. En plus de présenter des temps de commutation (et ses pertes associées) plus rapides que leurs équivalents Si, les transistors GaN HEMT ont aussi pour particularité de ne pas posséder de diode de corps comme les MOSFET Si. Quand une tension V_{ds} négative est imposée à ses bornes, le drain peut se comporter comme une source et le “canal” (un gaz 2D d’électrons) se forme. Dans les faits, ce comportement peut être assimilé à celui d’une diode de corps avec une très faible (voire nulle) charge de recouvrement inverse Q_{rr} , ce qui implique significativement moins de pertes par commutation dure. Néanmoins, la tension directe de cette pseudo-diode est une tension de seuil grille-drain $V_{th,gd}$, généralement supérieure au seuil grille-source V_{th} , soit autour de 2V pour les transistors GaN du fabricant *EPC*. Dans son fonctionnement “diode”, le transistor GaN est donc bien plus dissipatif qu’un transistor Si, ce qui rend d’autant plus important d’assurer un contrôle fin du composant et notamment de minimiser autant que possible les durées de *deadtime*.

1.4 Sélection des Transistors commerciaux

Dans nos plages de courant ($I_{ds} \sim 100A$) et de tension ($V_{ds} < 200V$), il est rare que le fabricant fournisse dans ses datasheets des informations sur les pertes par commutation de leurs transistors. Il existe des méthodes pour les estimer à partir de leurs éléments parasites, certaines sont purement calculatoires tandis que d’autres reposent sur la création d’un modèle équivalent sous forme de circuit simulable par un logiciel circuit de type *SPICE*. Les modèles les plus complexes requièrent la connaissance de tous les éléments parasites capacitifs et inductifs du composant [27], il est extrêmement compliqué de les mettre en œuvre sans obtenir soi-même expérimentalement ces valeurs, faute de les trouver dans les datasheets. Des modèles simplifiés n’employant que des paramètres connus des datasheets ont aussi été développés [28–31].

Utiliser les modèles *SPICE* directement conçus et fournis par le constructeur est souvent l’approche la plus pratique et flexible. Cette approche présente néanmoins deux grandes limites. La première est que beaucoup de constructeurs ne fournissent aucun modèle *SPICE*, ou bien leurs modèles sont adaptés à un logiciel spécifique (généralement *PSPICE*) incompatible avec les logiciels

à notre disposition. Nous utilisons *LTSPICE*, les composants *PSPICE* peuvent parfois être compatibles quand ils n'utilisent pas dans leurs modèles certaines fonctions exclusives à *PSPICE*, mais d'une manière générale les modèles complexes de transistors sont incompatibles.

La seconde limite est la qualité des modèles. Aucune modélisation théorique du fonctionnement d'un transistor n'est parfaitement fidèle au comportement réel. C'est aussi le cas pour les modèles *SPICE* des transistors *EPC*, comme illustré dans leur application note [32], où nous "discernons" (faute d'avoir plus d'informations) que la simulation produit des temps de commutation beaucoup plus courts que ce qui est observé expérimentalement, ce qui tend certainement à minimiser les pertes simulées. La qualité des modèles est extrêmement variable. Au cours de ces travaux il est devenu apparent que certains modèles *LTSPICE* sont bien trop approximatifs pour être pertinents. La comparaison entre le modèle *SPICE* du composant GaN *EPC2022* (Annexe A.2), et du composant silicium *SiDR668DP* (Annexe A.1) est frappante.

Le composant *SiDR668DP* n'est modélisé que par des composants élémentaires de *LTSPICE*, un *nmos* et un *pmos* dont les paramètres ont été ajustés. La valeur de résistance $R_{dsON} = 4.8m\Omega$ à $25^\circ C$ correspond à la datasheet. En revanche, les capacités parasites C_{gs} , C_{gd} et C_{ds} (CJO dans *nmos*) ne varient pas en fonction de la tension drain-source et leurs valeurs ne correspondent pas non plus aux valeurs retenues pour $V_{ds} = 50V$ par la datasheet. En simulation, le composant présente des comportements erratiques pendant certaines commutations, qui empêchent la simulation de converger.

TABLE 1.3 – Propriétés typiques des transistors sélectionnés, pour $T_j = 25^\circ C$

	EPC2034C	EPC2022	EPC2024	GS61008T	EPC2302	EPC2066
BV_{ds}	200V	100V	40V	100V	100V	40V
I_{ds,max}	48A	90A	90A	100A	101A	90A
R_{dsON}	6mΩ	2.4mΩ	1.2mΩ	7mΩ	1.4mΩ	0.8mΩ
C_{oss}	641pF	840pF	1620pF	250pF	1000pF	1670pF
C_{iss}	1155pF	1400pF	1920pF	610pF	3200pF	3539pF
R_{θj,c}	0.3°C/W	0.4°C/W	0.4°C/W	0.55°C/W	0.2°C/W	0.3°C/W

Pour ces raisons de disponibilité et de pertinence des modèles *LTSpice*, nous n'avons retenu dans cette étude que quelques transistors produits par *EPC*, et un transistor GaNSystem, listés en Table 1.3. L'étude a été d'abord réalisée en 2020 et n'a retenu que les composants *EPC2034C*, *EPC2022*, *GS61008T* et *EPC2024*. De nouveaux composants plus performants ont été commercialisés à l'heure où ces lignes sont écrites (2023) et les plus intéressants ont été ajoutés postérieurement à l'étude (grisés dans les différentes tables).

Il est possible, dès la sélection des transistors effectuée, de réaliser des comparaisons entre des composants de même calibre (BV_{ds} et $I_{ds,max}$) à partir des valeurs R_{dsON} , C_{iss} et C_{oss} données par leurs datasheets. La résistance à l'état passant R_{dsON} permet de rapidement calculer les pertes par conduction du transistor à un courant de charge donné $P_{cond} = R_{dsON} \cdot I_{ds}^2$. Ce critère est extrêmement important dans notre application où le courant de charge en sortie du convertisseur peut atteindre 700A, et les courants I_{ds} traversant les différents transistors sont de plusieurs dizaines d'ampères. Les capacités parasites C_{iss} et C_{oss} du transistor sont liées aux pertes par commutation du transistor et, bien qu'il ne soit pas possible de directement estimer ces dernières à partir de ces valeurs (qui ne sont elles-mêmes que des approximations de grandeurs évoluant durant la commu-

tation), nous pouvons prendre comme hypothèse simplificatrice que, à calibres égaux, un transistor affichant un C_{iss} et un C_{oss} bien plus faibles qu'un autre composant devrait générer moins de pertes par commutations. Il faut néanmoins être prudent avec cette approche.

Dans la sélection de composants présentée en Table 1.3, nous observons la tendance décrite en section 1.2.1. Dans la gamme des transistors *EPC* commercialisés avant 2020, nous constatons un meilleur R_{dsON} pour les BV_{ds} plus faibles. Le composant *GaNSystems GS61008T* doit être comparé à l'*EPC2022*, il semble présenter de meilleures performances en commutation, mais son R_{dsON} est trois fois plus élevé ($7,4m\Omega$ contre $2.4m\Omega$). *GaNSystems* n'ont commercialisé que deux autres transistors GaN basse tension (100V), *GS61008P* qui n'est qu'une variante de *GS61008T* conçue pour un refroidissement par le bas (à travers le PCB) sans différence de performance notable, et *GS61004B* est à $16m\Omega$. A tensions de claquage BV_{ds} égales, nous remarquons une amélioration du R_{dsON} pour les composants les plus récents (en gris), pour des valeurs C_{iss} et C_{oss} relativement inchangées.

1.5 Extraction des énergies de commutation via LTSpice

Afin d'évaluer les énergies de commutation de chacun de ces transistors, leur modèle *SPICE* fourni par les fabricants est testé aux tensions et courants adéquats sur un circuit *LTSpice*, présenté en Figure 1.11. Il s'agit d'un circuit *BUCK* idéalisé, nous appellerons *HS* le transistor *High Side*, et *LS* le transistor *Low Side*. L'alimentation d'entrée et la charge sont modélisées par des sources de tension et courant idéales, ce qui permet d'imposer directement aux transistors les valeurs V_{ds} et I_{ds} . Le circuit ne présente aucune inductance parasite. Les deux circuits à droite servent à évaluer les chutes de tensions V_{RdsON} et V_{diode} liées à la résistance à l'état passant et à la diode de corps, qui dépendent toutes deux de la température de jonction T_j du composant et du courant drain-source I_{ds} .

Les drivers commandant les transistors (Fig. 1.12) ont été simplifiés au maximum pour n'être que de simples *Totem pole* directement contrôlés par le signal de commande. La tension d'alimentation des drivers V_{alim} et de leurs résistances internes $R_{gridSRC}$ et $R_{gridSNK}$ sont adaptés aux informations disponibles sur les datasheets de chaque transistor. Typiquement, la gamme des transistors *EPC* a été testée avec $V_{alim} = 5V$, $R_{gridSRC} = 0.8m\Omega$ et $R_{gridSNK} = 0.4m\Omega$. Ces valeurs de résistance sont inspirées des caractéristiques du composant Gate Driver *2EDF7275Kd* d'*Infineon*, que nous exploiterons au chapitre 2.

La simplicité du circuit permet des temps de simulation rapides ($\sim 2.5s$), avec une extraction des informations importantes directement intégrée à la simulation via les commandes *SPICE .meas*. Les énergies de commutation peuvent être directement lues par l'utilisateur dès que la simulation est terminée. L'intérêt de la rapidité de la simulation et de l'automatisme de cette procédure se révélera au chapitre 4 où de vastes campagnes de simulations seront réalisées. La mesure des énergies de commutation est réalisée par une intégration du produit $V_{ds} \times I_{ds}$, qui donne la puissance instantanée d'un composant. Le choix du composant observé et la plage de temps de l'intégration permettent d'isoler les différentes pertes. Le relevé de ces énergies de commutation est décrit par les équations 1.23 à 1.25, et représenté en Figure 1.13.

L'énergie au *turn-on* E_{on} :

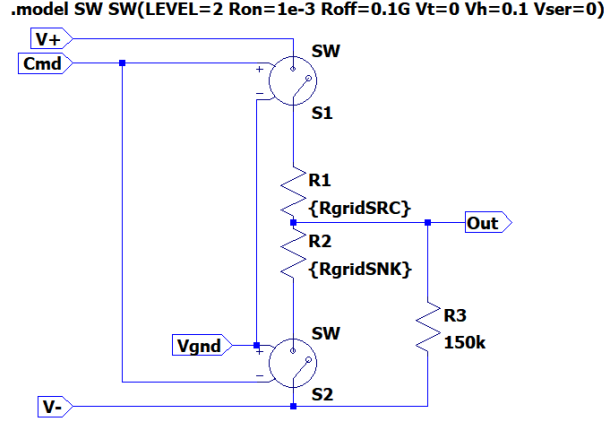


FIGURE 1.12 – Modélisation *LTSpice* des drivers du circuit de test (fig. 1.11)

$$\left\{ \begin{array}{l} E_{off} = \left(\int_{t_{off,1}}^{t_{off,2}} (V_{ds,HS}(t) \cdot I_{ds,HS}(t) + V_{ds,LS}(t) \cdot I_{ds,LS}(t)) dt \right) - R_{dsON} I_{ds,HS}^2 \cdot (t_{off,2} - t_{off,1}) \\ t_{off,1} \text{ le temps où } V_{gs,HS} < 0.99 \cdot V_{alim} \text{ pour la première fois} \\ t_{off,2} \text{ le temps où } I_{ds,HS} < 0.01 \cdot I_{load} \text{ pour la première fois} \end{array} \right. \quad (1.24)$$

L'énergie de recouvrement de la diode de corps E_{rr} :

$$\left\{ \begin{array}{l} E_{rr} = \left(\int_{t_{rr,1}}^{t_{rr,2}} V_{ds,LS}(t) \cdot I_{ds,LS}(t) dt \right) \\ t_{rr,1} \text{ le temps où } I_{ds,LS} > -0.9 \cdot I_{load} \text{ pour la première fois} \\ t_{rr,2} \text{ le temps où } V_{ds,LS} > 0.99 \cdot V_{in} - V_{rdson} \text{ pour la première fois} \end{array} \right. \quad (1.25)$$

E_{off} est relevé non pas sur la seule puissance instantanée du transistor *High Side HS*, mais sur la somme des puissances instantanées de *HS* et *LS*. En effet, une partie de l'énergie observée sur la commutation de *HS* n'est pas dissipée, mais transférée à *LS* (Fig. 1.13a). La somme de ces deux énergies permet d'annuler le transfert et de ne garder que la perte nette du système au Turn-off de *HS*. Bien que les transistors GaN aient la réputation de n'avoir "aucune" charge de recouvrement Q_{rr} (les datasheets d'*EPC* indiquent $Q_{rr} = 0nC$), Nous observons tout de même dans le comportement des modèles *LTSpice* fournis par les fabricants un courant de recouvrement sur $-I_{ds,D}$ et une énergie dissipée par *LS* lorsque *HS* devient passant et que la "diode de corps" devient bloquante (Fig. 1.13b). Puisqu'elle en a tous les traits, Nous appellerons cette énergie E_{rr} .

Les lignes de commande *.meas* de *LTSpice* ne permettent pas de bien gérer les oscillations de commutation que peuvent provoquer les inductances parasites des transistors. Les commandes décrites par les équations 1.23-1.25 intègrent un produit $V \times I$ entre deux événements détectés par *LTSpice*. L'événement de début de l'intégration ne pose pas de problème particulier à situer. En revanche, l'événement de fin peut être compliqué à situer en cas d'inductances parasites, car les tensions et courants qui servent à le situer peuvent osciller et franchir le seuil de déclenchement de l'événement sans que la commutation soit réellement achevée. L'énergie calculée est alors incomplète et faussée. Les commandes *.meas* nous offrent la possibilité de détecter l'événement après le $n^{ième}$

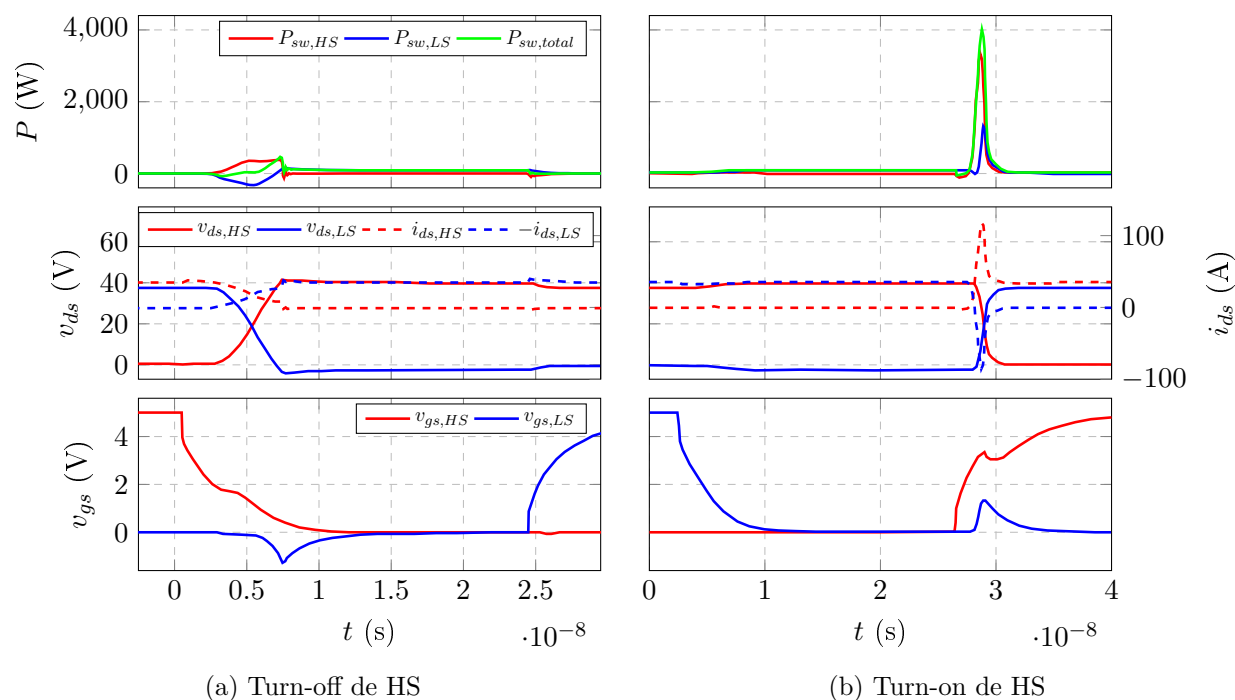


FIGURE 1.13 – Simulation de commutations sous *LTSpice* avec des transistors *EPC2022*, $V_{in} = 37.5V$, $I_{load} = 35A$

franchissement de la condition, n étant un nombre qui doit être manuellement fixé par l'utilisateur avant la simulation, ce qui n'est pas une méthode viable. Une autre méthode consisterait à intégrer le produit $V \times I$ sur une durée fixée par l'utilisateur, plus longue que la commutation, mais en procédant ainsi nous risquerions d'ajouter aux pertes par commutation une quantité non-négligeable et difficile à déterminer de pertes par conduction ou conduction inverse.

Étant donné ces sévères limitations, il est préférable de complètement éliminer toutes les inductances parasites, y compris celles incluses dans le modèle *Spice* du composant. Ces inductances se limitent généralement aux inductances de packaging, directement adjacentes aux terminaux de drain et de source et représentées en Figure 2.2 (Chapitre 2), il est assez aisé de les repérer dans le fichier *.lib* du composant et de les retirer. Les modèles *Spice* fournis par *EPC* ne contiennent aucune inductance, mais ce n'est pas le cas de la plupart des modèles de transistors, dont notamment le *SiDR668DP*. L'impact des inductances parasites sur les pertes a été évalué par [6, 33], il apparaît que seule l'inductance de source commune L_{ss} a une influence notable sur les pertes, car elle tend à freiner la commutation du transistor d'une manière similaire à la résistance de grille. Les autres inductances parasites augmentent le dépassement de tension de V_{ds} et les oscillations, mais n'ont finalement que peu d'effet dissipatif. De nombreux transistors disposent d'un terminal source dédié à la commande, distinct de leur terminal source dédié au passage de la puissance, permettant d'éliminer les problèmes causés par L_{ss} . C'est le cas des transistors *EPC* et du composant *GS61008T*. Dans ce cas de figure, Nous nous permettons d'éliminer les autres inductances (drain et source, sur la puissance) du modèle *Spice*, de sorte que la simulation ne génère pas d'oscillation parasite qui viendrait fausser la mesure des énergies de commutation.

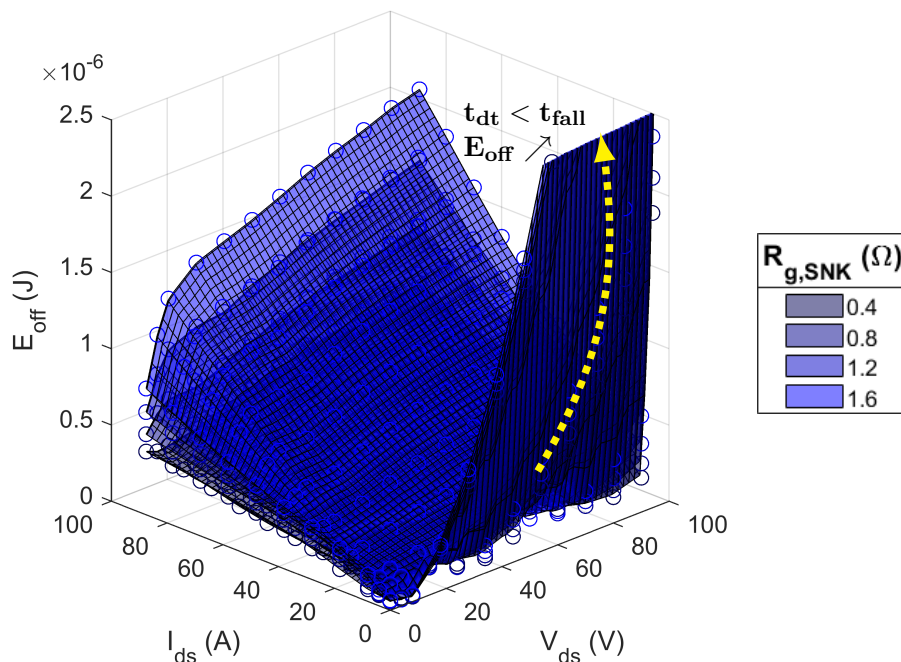


FIGURE 1.14 – Résultats *LTSpice* pour E_{off} du composant *EPC2022*, $t_{dt} = 20ns$, $R_{g,SRC} = 0.8\Omega$

Une fois le circuit établi et tous ces éléments pris en compte, nous pouvons automatiser l'extraction des pertes pour chaque composant, en fonction de V_{ds} , I_{ds} et T_j , mais aussi des résistances de grille de *pull-up* $R_{g,SRC}$ et *pull-down* $R_{g,SNK}$ et du *deadtime*. Ces trois derniers paramètres peuvent permettre d'observer l'influence de la résistance de grille sur les pertes et les temps de commutation du transistor, et le *deadtime* t_{dt} peut permettre, quand il est trop court, d'observer l'impact du *hard-switching* de *LS* sur E_{off} (Fig. 1.14).

1.6 Elements Passifs

Le choix des éléments passifs est primordial dans la conception d'un convertisseur de puissance. La topologie du convertisseur, les tensions et courants en entrée et sortie, et la fréquence de commutation des interrupteurs influent grandement sur les besoins en éléments passifs. Ces derniers sont nécessaires pour que les formes d'onde respectent le cahier des charges, mais représentent une partie importante du volume et de la masse du convertisseur, et génèrent des pertes additionnelles. De nombreux travaux se sont intéressés aux performances des composants disponibles sur le marché pour ces différents critères [4, 34, 35].

Dans une étude exhaustive portant sur 500 000 inductances et capacités commercialisées [4], il apparaît les tendances suivantes, illustrées en Figure 1.15. La densité d'énergie spécifique des inductances croît avec leur courant nominal, et celle des capacités croît avec leur tension nominale. Globalement, les capacités présentent une meilleure densité énergétique que les inductances. Choisir des topologies de convertisseur favorisant le stockage de l'énergie par des capacités peut contribuer à améliorer la densité de puissance globale du convertisseur.

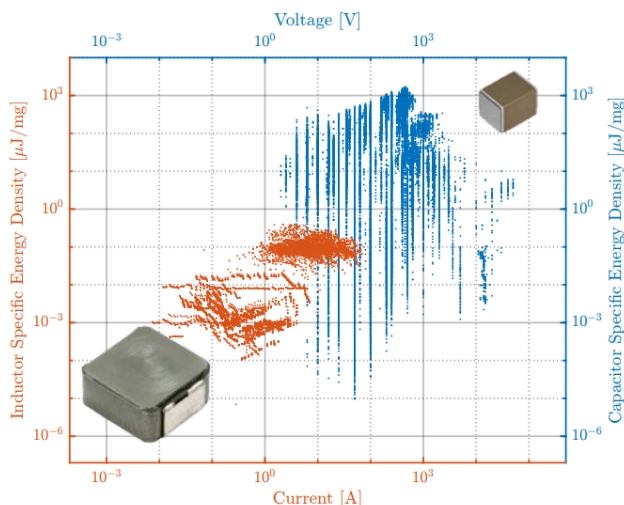


FIGURE 1.15 – Densité d’énergie spécifique des inductances et capacités étudiées par [4]

Un modèle pour estimer la densité énergétique des éléments passifs à partir de leur impédance en fonction de leurs technologies est proposé par [4]. Néanmoins, la précision du modèle reste limitée, avec des erreurs relatives moyennes du *fitting* de l’ordre de ~ 0.15 , et des R^2 compris entre 0.2 et 0.62 en fonction des technologies. Ces modèles permettent d’estimer des tendances et d’aider à la sélection, mais ne peuvent pas se soustraire à l’intégration des données datasheet du composant sélectionné dans le calcul final des performances du convertisseur.

1.7 Inductances

Il est relativement courant de développer des inductances sur mesure pour un convertisseur. [36, 37] synthétisent les technologies, les méthodes de dimensionnement et de fabrication. Mais pour des applications industrielles, nous préférons l’usage d’inductances commercialisées. Parmi les principaux fabricants, *Vishay* produit une très vaste gamme d’inductances couvrant une plage allant de $10nH$ à $100\mu H$ pour des courants de plusieurs dizaines d’ampères. La figure 1.16 synthétise les pertes joules et le volume d’inductances *Vishay* issues de 10 séries différentes, en fonction de leur énergie stockée à courant de saturation tel que :

$$E_{sat} = \frac{1}{2} L I_{sat}^2 \quad (1.26)$$

Utiliser l’énergie E_{sat} plutôt que le courant nominal en abscisse permet de représenter l’effet de la valeur d’inductance L , en plus de celui du courant I_{sat} . Nous pouvons observer sur la Figure 1.16 que chaque série a été conçue pour une énergie de stockage donnée, et leurs composants sont chacun un compromis entre I_{sat} et L autour de cette énergie. L’énergie stockée est logarithmiquement proportionnelle au volume du composant. Pour ce qui est des pertes par effet joules, on distingue deux groupes obéissant à la même tendance, à savoir une augmentation des pertes joules $P_{L,DC}$ avec l’énergie stockée E_{sat} . Les séries entourées sur la Figure 1.16 ont cependant été optimisées pour produire moins de pertes à des courants plus élevés, au détriment de leurs volumes V_L .

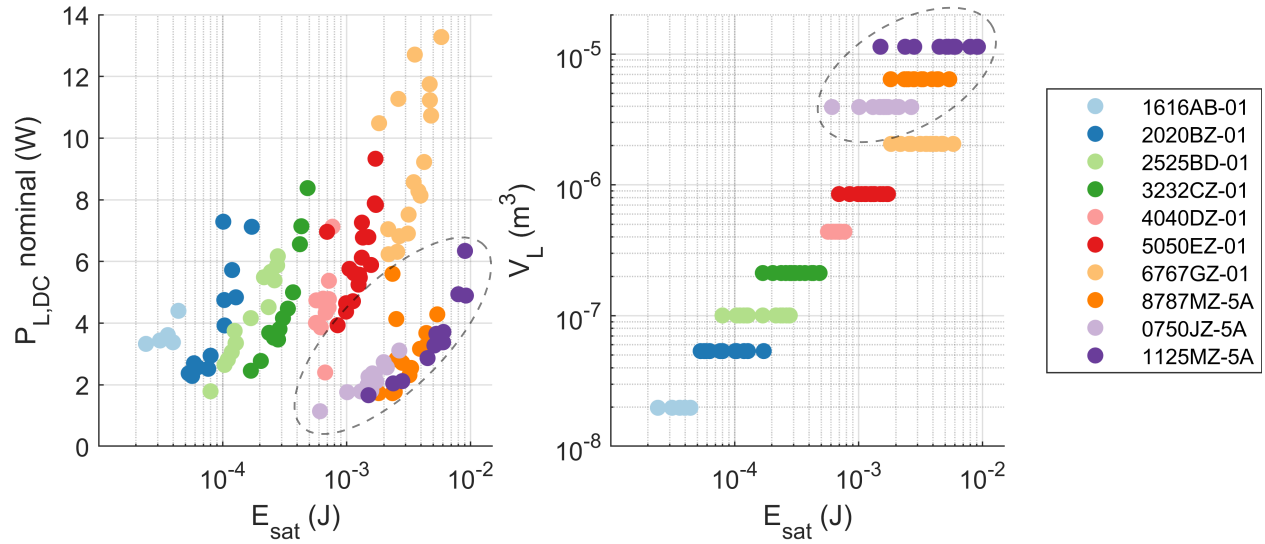


FIGURE 1.16 – Pertes joules et Volume des inductances *Vishay* en fonction de leur énergie à courant de saturation

Vishay propose une modélisation de leurs pertes [38,39] pour chacun de leurs composants [40]. Cette modélisation n'est pas valable en cas de dépassement du courant de saturation I_{sat} par I_L . Les pertes fer P_{core} sont modélisées à partir de l'équation de Steinmetz modifiée [41] :

$$P_{core} = K_0 \cdot f_e^{K_f - 1} \cdot B_{pk}^{K_b} \cdot f_L \cdot 10^{-14} \quad (1.27)$$

Les pertes joules alternatives P_{ac} , causées par effet de peau, sont décrites par l'Equation 1.28 :

$$P_{ac} = K_1 \cdot \Delta I_L^2 \cdot \sqrt{f_L} \cdot R_L \quad (1.28)$$

K_0 et K_1 sont deux constantes liées à l'inductance *Vishay* sélectionnée. La constante de densité de flux K_b et de fréquence K_f sont des constantes liées au matériau utilisé dans le noyau et ne changent pas d'une valeur d'inductance à une autre pour une même référence. Ces quatre constantes ont été établies empiriquement par *Vishay* pour chaque référence, leurs valeurs sont répertoriées dans [40]. Dans le cadre d'une inductance de filtrage en sortie d'un convertisseur DC-DC non-isolé, la fréquence f_L du courant I_L parcourant l'inductance est directement liée à la topologie du convertisseur et à la fréquence de découpage f_{sw} des transistors, tel que $f_L = (N - 1)f_{sw}$. La fréquence effective f_e sert à prendre en compte le caractère non-sinusoidal du courant I_L (généralement triangulaire), tel que :

$$f_e = \frac{f_L}{2\pi \cdot (D_{eff} - D_{eff}^2)} \quad (1.29)$$

Où D_{eff} est le rapport cyclique effectif à l'intérieur de la région de fonctionnement d'indice i_D du convertisseur multiniveaux (défini en section 1.2.3) :

$$D_{eff} = \left(D - \frac{i_D - 1}{N - 1} \right) \cdot (N - 1) \quad (1.30)$$

La densité de flux maximale B_{pk} dépend d'une constante ET_{100} qui est le produit $V \cdot \mu s$ du circuit pour une induction magnétique de 100G (répertorié dans [40]), et du produit $V \cdot \mu s$ effectif ET_{ckt} du circuit tels que :

$$B_{pk} = \frac{ET_{ckt}}{ET_{100}} \cdot 100 \quad (1.31)$$

$$ET_{ckt} = \frac{V_{in} \cdot \left(\frac{i_D}{N - 1} - D \right) \left(D - \frac{i_D - 1}{N - 1} \right)}{f_L} \times 10^6 \quad (1.32)$$

Enfin, la résistance R_L dépend de la résistance typique DCR_{typ} et de la température de l'inductance T_L tel que :

$$R_L = DCR_{typ} \cdot \frac{234.5 + T_L}{259.5} \quad (1.33)$$

Cette formule implique une boucle de rétroaction entre l'augmentation des pertes, l'élévation de T_L et l'augmentation de R_L . Dans leur simulateur en ligne et leur document exemple [39], Vishay néglige cette rétroaction et calcule directement R_L en supposant que T_L est une constante reflétant le pire cas espéré.

1.8 Calcul des performances de plusieurs solutions

Dans le cadre de cette étude, nous nous limiterons à une sélection restreinte de composants et topologies. Nous étudierons le point de fonctionnement $V_{in} = 80V$, $V_{out} = 28V$ pour une puissance de charge $P_{load} = 20kW$.

Nous utiliserons des capacités $10\mu F$, $100V$ *GRM32EC72A106KE05* de *Murata*, la variable de design étant le nombre de capacités à placer en parallèle et série sur le design pour produire les valeurs de capacitance désirées. Les inductances seront restreintes à la série des *IHLP8787MZ* de *Vishay*, qui sont conçues pour de plus forts courants que le reste de leurs composants, tout en étant montés en surface, contrairement aux *IHTH* traversants. Il sera envisagé la mise en parallèle ou série de ces inductances pour respecter les besoins en induction et en courant. Les topologies *BUCK*, *FC3L* et leurs variantes *Double* seront comparées entre elles. Le choix du transistor pour chaque topologie est restreint aux composants listés en Table 1.3, et adapté par rapport à la tension V_{ds} imposée par chaque topologie.

Les pertes par commutation P_{sw} et conduction P_{cond} sont calculées à partir des énergies de commutation et résistances R_{dsON} des modèles *Spice* des transistors. Un modèle thermique de refroidissement des transistors, décrit plus en détail dans la suite de ce document en section 2.4.2,

permet d'estimer la température T_j du transistor et d'estimer plus finement les pertes P_{cond} . Ce modèle utilise le dissipateur 960-31-15-D-AB-0 de Wakefield-Vette associé au matelas thermique SARCON XR-Um de Fujipoly et à un flux d'air de 4m/s.

Les différentes pertes associées aux inductances Vishay sont calculées à l'aide de leurs modélisations empiriques (Eq. 1.27 - 1.33). La sélection des inductances se fait selon la logique suivante. L'inductance L_{inf} (Eq. 1.15) est calculée pour une solution donnée, pour un taux d'ondulation $\Delta i_L = \pm 10\% = 20\%$. Ce taux d'ondulation appliqué au point de fonctionnement 20kW permet de garantir un fonctionnement de l'inductance en Mode Continu même à des courants de charge bien plus faibles. Est sélectionné parmi les inductances IHLP-8787MZ le composant ayant la plus petite valeur L satisfaisant cette condition, et dont le courant de saturation I_{sat} est supérieur au courant I_{out} . Si aucun composant ne remplit ces conditions, une réévaluation est effectuée pour deux inductances placées en parallèle (divise la valeur L effective, et la contrainte en courant), ou en série (multiplie par 2 la valeur L effective sans changer la contrainte en courant). Nous limiterons les recherches à des configurations n'impliquant que 2 inductances au maximum.

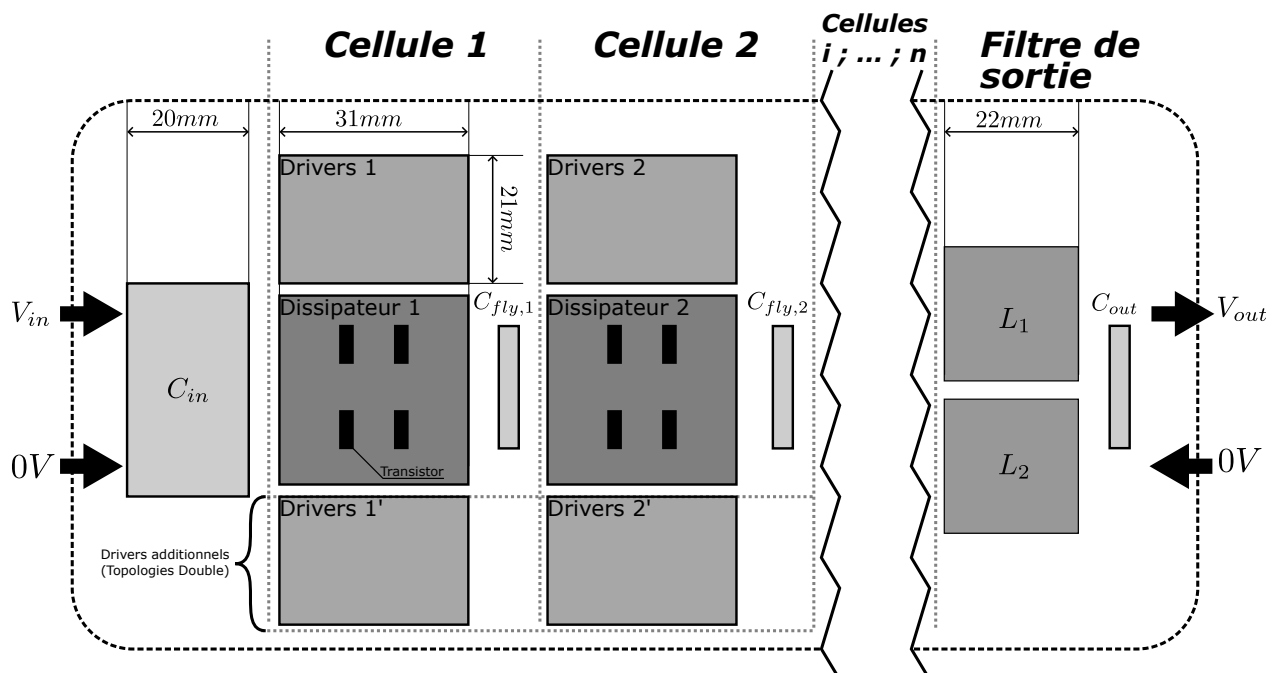


FIGURE 1.17 – Schéma de principe des géométries possibles pour une *phase* de conversion, en fonction de la topologie

Pour estimer la masse et le volume du convertisseur, nous établirons une règle de calcul de la surface de circuit imprimé nécessaire pour une *phase* de conversion, qui permet ensuite d'estimer le volume et la masse de cette *phase* et, *in fine*, du convertisseur complet hors circuit de commande. La géométrie d'une *phase* est représentée en Figure 1.17. Un dissipateur thermique peut couvrir jusqu'à 4 transistors. Les cellules de commutation de la topologie FCML se succèdent sur la longueur du PCB, et les topologies Doubles nécessitent de doubler l'ensemble du PCB "simple" sur la largeur du PCB.

La masse du PCB (hors composants) peut se calculer à partir de sa surface, du nombre de couche de cuivre et de l'épaisseur des couches. Sur des convertisseurs de puissance à forts courants, le nombre de couches est généralement de 6 ou 8, permettant de répartir le courant sur plusieurs

couches et diminuer la résistance du PCB. L'épaisseur standard des couches est de $35\mu m$, mais nous opterons ici pour une épaisseur de $70\mu m$. L'épaisseur totale du PCB est fixée à $1mm$, la proportion de cuivre par couche est estimée de 80%. La masse volumique du cuivre est de $8930kg/m^3$, et celle du matériau isolant *FR-4* qui compose le reste du PCB est de $1850kg/m^3$.

Nous prendrons aussi en compte la masse et le volume des ventilateurs. Nous nous baserons sur la référence *8484-NL* de *EBMpapst*, capable de fournir un flux d'air de $5m/s$ à quatre dissipateurs thermiques. Son poids est de $95g$, et son volume de $80mm \times 80mm \times 25mm$, auquel il faut ajouter $50mm$ de distance entre le ventilateur et les dissipateurs pour représenter la canalisation d'air.

Pour chaque combinaison d'une topologie et d'un transistor, nous évaluerons plusieurs solutions, chacune à une fréquence de commutation $f_{sw} \in [100kHz; 500kHz]$ et un nombre de phases $n_{phase} \in [10; 25]$ différent.

1.9 Comparaison des résultats

Nous comparons la Puissance Massique ρ_{conv} et le Rendement η de chaque solution. Les résultats sont présentés en Figure 1.18, et un échantillon représentatif en Table 1.4 (où ne sont présentées que les solutions réalisables au moment de l'étude, en 2020, et présentant les meilleurs rendements). Chaque point correspond à une solution différente. La couleur indique le transistor utilisé (si le remplissage est vide, le transistor a été commercialisé après 2020), et la forme la topologie mise en oeuvre.

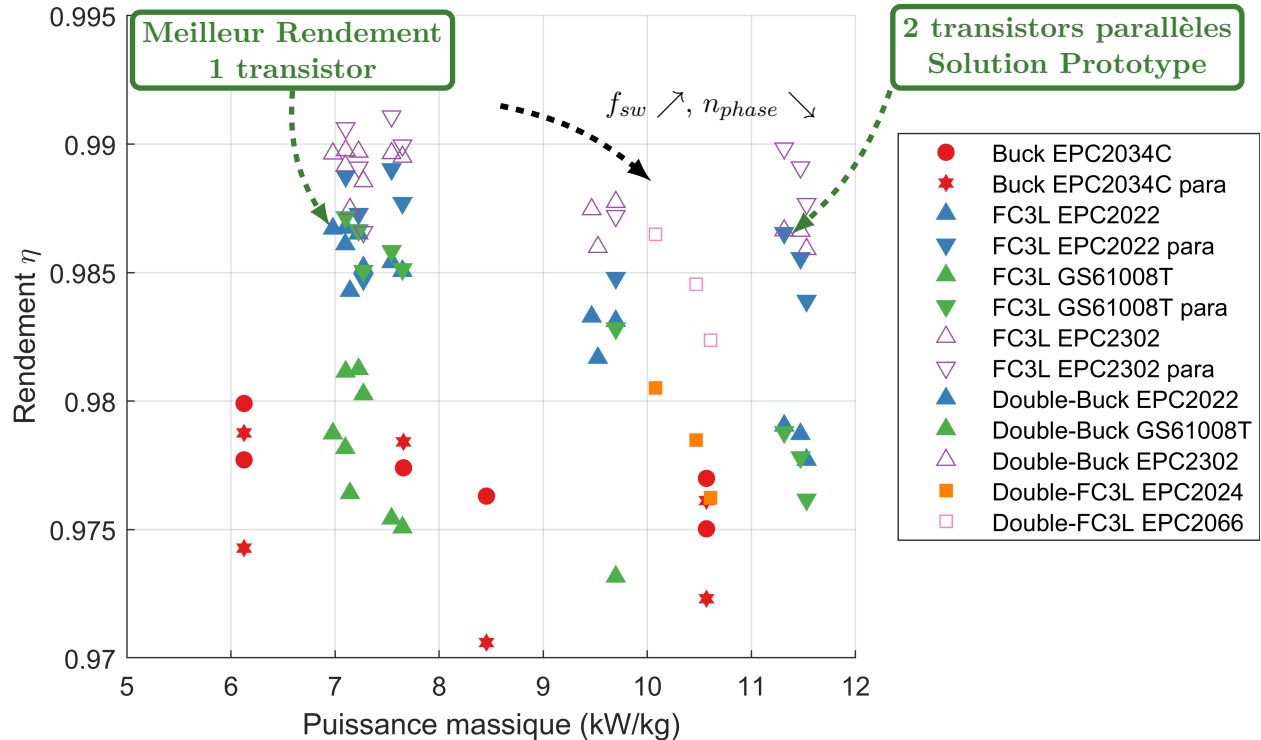


FIGURE 1.18 – Puissances massiques et Rendement des solutions évaluées (pas de remplissage : transistors commercialisés après 2020)

Nous constatons une corrélation forte entre le choix du transistor et le rendement, tandis que la topologie semble davantage influencer la densité de puissance du convertisseur. Cette observation peut s'expliquer par l'importance des pertes par conduction, du fait des courants importants traversant le convertisseur (700A divisés entre les phases). La résistance à l'état passant R_{dsON} du transistor a une forte influence sur ces pertes. Typiquement, et malgré ses pertes par commutation plus faibles à V_{ds} et I_{ds} égaux, le transistor *GS61008T* produit des solutions dont le rendement est nettement inférieur aux solutions équivalentes employant l'*EPC2022* (-0.57% sur η en Table 1.4), à cause de son R_{dsON} de $5.04m\Omega$, contre $2.58m\Omega$ pour l'*EPC2022*, soit $+95\%$ de pertes par conduction (en prenant en compte l'influence de la température de jonction T_j sur le R_{dsON}).

TABLE 1.4 – Comparaison des performances des solutions présentant le meilleur rendement pour chaque transistor & topologie, pour $V_{in} = 80V$, $V_{out} = 28V$ et $P_{in} = 20kW$

	Buck EPC2034C	FC3L EPC2022	FC3L GS61008T	Double Buck EPC2022	Double FC3L EPC2024
f_{sw}	500kHz	200kHz	200kHz	200kHz	200kHz
n_{phase}	25	20	20	20	10
$P_{cond}(W)$	194	132	258	132	294
$P_{sw}(W)$	99.9	40.2	29.4	40.2	41.9
$P_L(W)$	146	96	96	98	62
$T_{j,max}$	60°C	37°C	44°C	47°C	68°C
η	97.99%	98.67%	98.1%	98.67%	98.05%
$\rho_{conv}(kW/kg)$	6.13	7.10	7.10	6.98	10.08

La mise en parallèle directe de deux transistors permet de diviser la résistance effective de l'élément "interrupteur" par 2 et, malgré l'augmentation des pertes par commutations, l'effet sur le rendement reste bénéfique. Nous pouvons aussi utiliser cette réduction du R_{dsON} pour augmenter la compacité du convertisseur. La Table 1.5 montre que l'on peut diviser par 2 le nombre de phases d'une solution sans diminuer le rendement de manière significative, pour une réduction de -60% de la masse finale du convertisseur, et une division par 2 de son volume. Que ce soit pour augmenter le rendement ou réduire la masse du convertisseur, paralléliser les transistors présente donc un gain clair.

TABLE 1.5 – Impact de la parallélisation directe de transistors sur la Puissance massique, $V_{in} = 80V$, $V_{out} = 28V$ et $P_{in} = 20kW$

	FC3L EPC2022	FC3L 2xEPC2022
f_{sw}	200kHz	200kHz
$n_{sw,para}$	1	2
n_{phase}	20	10
$P_{cond}(W)$	132	140
$P_{sw}(W)$	40.2	45.1
$P_L(W)$	96	88
$T_{j,max}$	37°C	48°C
η	98.67%	98.65%
$\rho_{conv}(kW/kg)$	7.10	11.31

D'une manière générale, augmenter le nombre de *phases* en parallèle permet d'augmenter le rendement en divisant la résistance équivalente globale du convertisseur, mais dégrade la densité de puissance. La réduction du courant par *phase* ne permet pas de réduire la taille d'une sous-unité avec notre méthode de conception, l'augmentation du nombre de *phase* revient donc à augmenter d'autant la masse du convertisseur complet. Il aurait été possible de prendre en compte l'utilisation de dissipateurs thermiques plus petits (l'élément dimensionnant pour une cellule de commutation), ce qui aurait un petit impact sur le volume du convertisseur, donc sur la masse du PCB. Mais cela ne changerait pas les deux principales sources de masse que sont les busbars et ventilateurs (nécessairement plus longs et nombreux pour couvrir l'ensemble des *phases*).

L'influence de la fréquence de commutation f_{sw} est plus ambiguë. Si l'on suppose que la masse de l'inducteur est directement corrélée à son inductance, l'équation 1.15 indique qu'une augmentation de la fréquence apparente f_L du courant traversant l'inductance devrait permettre de sélectionner des inductances plus petites et légères. Mais les inductances commercialisées par *Vishay* correspondant à nos applications ne suivent pas réellement cette tendance, bien que la Figure 1.16 puisse le laisser penser. Dans les faits, à moins d'augmenter le nombre de *phases* en parallèle au delà de 25 (résultant en $I_L < 29A$), seule la série *IHLP-8787MZ* permet de répondre à la fois aux besoins en inductance et en tenue de courants. Dès lors, comme tous les composants d'une même série partagent le même packaging, changer la valeur L requise n'a d'effet sur la masse du convertisseur que s'il est requis de paralléliser ou mettre en série deux inductances (dégrade la masse). Les composants de cette série à plus faible inductance produisent toutefois moins de pertes (conduction et fer), mais le gain n'est pas suffisant pour compenser l'augmentation des pertes par commutation P_{sw} des transistors causée par l'augmentation de f_{sw} requise.

Il apparaît donc préférable de sélectionner la fréquence de commutation f_{sw} la plus faible étudiée, afin de réduire au maximum les pertes P_{sw} des transistors. Augmenter f_{sw} ne permet pas de réduire la masse du convertisseur à cause de l'absence de composants plus petits adaptés à notre application. En revanche, diminuer f_{sw} en dessous de $100kHz$ impliquerait l'utilisation d'inducteurs bien plus volumineux (augmentation de l'inductance requise) ou la parallélisation de davantage de *phases*, menant à une dégradation drastique de la densité de puissance.

Il faut noter, par ailleurs, les progrès technologiques réalisés par *EPC* dans les performances de leurs composants. Les solutions employant des transistors commercialisés après 2020 présentent des rendements supérieurs à toutes les solutions "pré-2020". Particulièrement, la réduction du R_{dsON} , de $2.4m\Omega$ à $1.4m\Omega$, entre l'*EPC2022* et l'*EPC2302* (tous deux des transistors $100V - 100A$) en est la principale raison.

1.10 Conclusion du chapitre

Lors de ce chapitre, nous avons pu mettre en œuvre une stratégie de conception standard pour un convertisseur de puissance. Différentes topologies de conversion de puissance ont été présentées, ainsi que l'intérêt des transistors WBG, et particulièrement des transistors de type GaN. Afin d'estimer les pertes des différents transistors et des inductances, nous avons utilisé les modélisations (*LTSpice* et formules paramétriques) fournies par les fabricants. En anticipant la forme que pourrait prendre le routage des PCB, nous avons pu estimer le volume et la masse du convertisseur pour chaque solution envisagée. Finalement, il nous a fallu restreindre le nombre de solutions évaluées

pour limiter le temps passé à cette tâche. Une partie importante des pistes écartées l'ont été de manière relativement arbitraires, justifiables par des tendances observées sur les quelques solutions testées. Nous reviendrons sur les erreurs que ce procédé peut engendrer au Chapitre 3.

Une fois ce travail réalisé, il nous est maintenant possible d'envisager la conception d'un prototype de *phase* afin d'étudier la mise en œuvre d'une solution et ses performances réelles. Cette tâche ayant été réalisée en 2020, nous prendrons seulement en compte les données disponibles à l'époque. Les résultats présentés en Figure 1.18 soulignent les meilleures performances, aussi bien en rendement qu'en densité de puissance, des topologies *Flying Capacitor* à 3 niveaux associées au transistor GaN *EPC2022*. La suite de nos travaux s'est donc portée sur la conception de prototypes basés sur ces solutions.

Chapitre 2

Conception & Expérimentation de prototypes

2.1 Topologie & Composants sélectionnés

Les travaux préalables nous ont permis de sélectionner pour le prototypage la topologie *Flying Capacitor* à 3 niveaux (*FC3L*) et les transistors *EPC2022*. Deux transistors seront mis en parallèle pour former un interrupteur, afin de diviser par 2 la résistance à l'état passant effective. La fréquence de découpage f_{sw} retenue est de $200kHz$, et le nombre de *phases* sera $n_{phase} = 10$. Une *phase* devra donc supporter un courant en sortie $I_{out,max}$ de $71A$, pour une tension de sortie $V_{out} = 28V$, soit une puissance nominale de *phase* $P_{max} = 2kW$. La tension de bus en entrée V_{in} dépend de la puissance absorbée sur la pile à combustible, et peut varier de $35V$ à $115V$. Néanmoins, la puissance de charge associée à la tension $115V$ est inférieure à $1kW$, la limite de fonctionnement du convertisseur. Dans les faits, la tension V_{in} maximale en fonctionnement est de $110V$ pour $1kW$. Ces caractéristiques sont résumées en Table 2.1.

Ce chapitre va traiter de plusieurs aspects importants lors de la réalisation d'une carte de puissance, à savoir la commande rapprochée des transistors, les problématiques de routage de carte avec la parallélisation de transistors et les couplages inductifs, les signaux de commande et la mesure expérimentale. Les travaux qui suivent ne porteront que sur le développement de cette *phase* de $2kW$, dont les caractéristiques sont résumées en Table 2.1. Le prototypage d'une mise en parallèle de plusieurs *phases* afin d'augmenter la puissance totale du prototype n'a pas pu être effectué, faute de temps.

TABLE 2.1 – Caractéristiques du prototype développé

Topologie	FC3L
Transistor	$2 \times EPC2022$
Inductance	$2 \times IHL P-8787MZ-51$ $L_{out} = 4.7\mu H / 2 = 2.35\mu H$
Condensateurs	<i>GRM32EC72A106KE05</i> $C_{in} = 10\mu F \times 25 \times 2 \div 2$ $C_{fly} = 10\mu F \times 25$ $C_{out} = 10\mu F \times 8$
f_{sw}	$200kHz$
P_{max}	$2kW$
$I_{out,max}$	$71A$
V_{in}	$[35V - 115V]$
V_{out}	$28V$

En utilisant l'Eq. 1.15 pour déterminer l'inductance nécessaire pour un taux d'ondulation du courant de sortie $\Delta i_L \leq 20\%$ à $I_{out} = 70A$ et $V_{in} = 110V$, nous retiendrons le composant *IHL P-8787MZ-51* de $4.7\mu H$. Son courant de saturation I_{sat} est de $37A$. Nous utiliserons toutefois deux inducteurs en parallèle, ce qui abaisse aussi l'inductance à $2.35\mu H$, donnant un taux d'ondulation $\Delta i_L = 20\%$, des pertes $P_L = 12.5W$, et une résistance DC équivalente de $0.815m\Omega$. Pour le point d'intérêt $V_{in} = 80V$, $I_{out} = 70A$, nous obtenons $\Delta i_L = 15\%$ et $P_L = 7.84W$. Le courant de charge I_{out} peut donc être abaissé jusqu'à $7.14A$ ($P = 200W$) avant que le convertisseur ne fonctionne en Mode Discontinu. Etant donné que le convertisseur final de $20kW$ composé de 10 phases de $2kW$ n'est pas amené à fonctionner en dessous de $1kW$, le Mode Discontinu ne sera pas étudié.

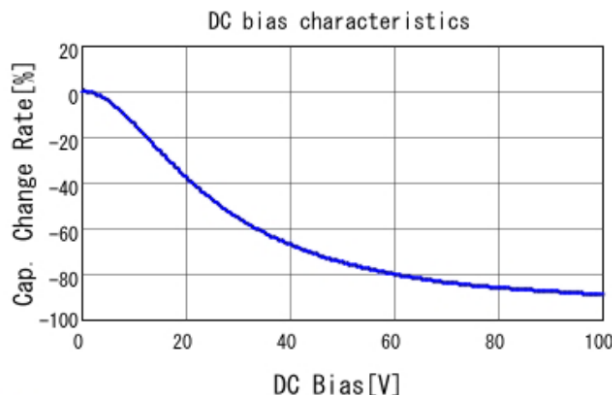


FIGURE 2.1 – Variation de capacitance en fonction de la tension DC aux bornes de *GRM32EC72A106KE05* (datasheet)

L'Eq. 1.18 nous donne la valeur requise pour l'unique capacité flottante de notre topologie *FC3L*. Pour un courant $I_{out} = 70A$ et un taux d'ondulation de la capacité flottante $\Delta v_{Cfly} = 10\%$, la valeur d'infimum est atteinte pour le rapport cyclique $D = 0.5$, résultant en $C_{fly} \geq 62.5\mu F$. Nous utiliserons les condensateurs céramique *GRM32EC72A106KE05*, de capacitance $10\mu F$ pour une tension maximale de $100V$. La valeur de capacitance indiquée se dégrade lorsque la tension imposée à ses bornes augmente suivant la courbe décrite en Fig. 2.1 tel que, pour la tension de capacité flottante maximale $V_{Cfly,max} = 55V$, cette valeur soit de $2.5\mu F$. Il faut donc paralléliser 25 condensateurs pour maintenir le taux d'ondulation $\Delta v_{Cfly} \leq 10\%$.

Le même raisonnement est employé pour calculer la capacité de filtrage en sortie C_{out} , à partir de l'Eq. 1.20, pour $L = 2.35\mu H$, $V_{in} = 110V$ et une ondulation de la tension en sortie $\Delta v_{out} = 1\%$. On en déduit que $C_{out} \geq 16.3\mu F$, soit 7 condensateurs en parallèle. Nous déciderons d'en placer 8 pour des raisons de symétrie.

2.2 Circuit de commande des transistors

Si la sélection des transistors est un élément majeur dans le développement d'un convertisseur, il faut ensuite les mettre en œuvre. Le routage du circuit de puissance et du circuit de commande du transistor sont déterminants dans le bon fonctionnement d'un transistor et peuvent grandement influencer la qualité des commutations. Si ces problématiques sont toujours présentes lors du développement d'un convertisseur électrique, elles sont particulièrement importantes pour la bonne mise en œuvre de transistors GaN. En effet, leurs faibles pertes par commutation sont l'un des atouts majeurs de ce type de transistors, obtenues grâce à une vitesse de commutation extrêmement élevée de l'ordre de $2V/ns$. Or une mauvaise mise en œuvre de ces transistors peut venir ralentir la commutation, donc augmenter les pertes, ou générer des surtensions et des oscillations parasites lors des commutations, pouvant entraîner un vieillissement accéléré voire une destruction des composants.

2.2.1 Influence de la résistance de grille sur la commutation

Le paramètre principal à prendre en compte dans le choix du driver pour maximiser la vitesse de commutation du transistor est la capacité du driver à fournir un courant élevé à la grille du transistor. La Figure 2.2 représente le circuit équivalent de commande de la grille d'un transistor, et la Figure 2.3 l'évolution de v_{gs} , v_{ds} et i_{ds} lors de commutations *turn-on* et *turn-off*. C'est le

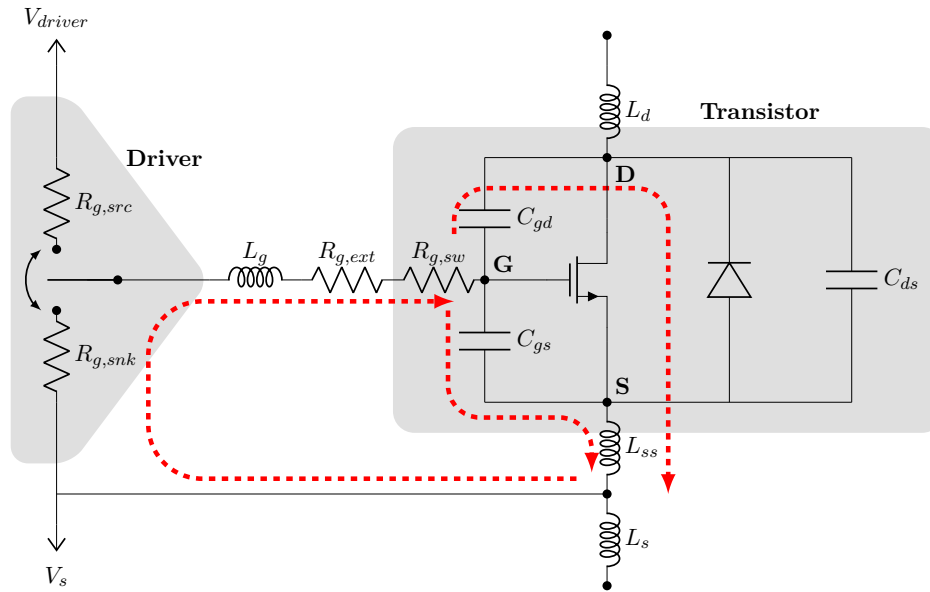


FIGURE 2.2 – Circuit équivalent de la commande de la grille d'un transistor

courant i_g qui vient charger ou décharger la capacité C_{gs} qui contrôle la conduction du transistor. En prenant en compte la capacité parasite C_{gd} qui vient elle aussi absorber une partie du courant i_g , nous obtenons une capacité équivalente $C_{iss} = C_{gd} + C_{gs}$. Le driver, la grille du transistor ainsi que la piste connectant les deux terminaux présentent chacun une résistance et une inductance. En général, la résistance de la piste est très faible, mais elle est la source principale d'inductance car elle forme une boucle de courant à travers le PCB. Les imperfections de packaging et de routage du PCB provoquent l'apparition de ces inductances parasites. Une résistance externe $R_{g,ext}$ est généralement rajoutée sur ce circuit. Nous nommerons la somme de toutes ces résistances $R_{g,tot}$. La charge de C_{iss} par le driver forme donc un circuit RLC.

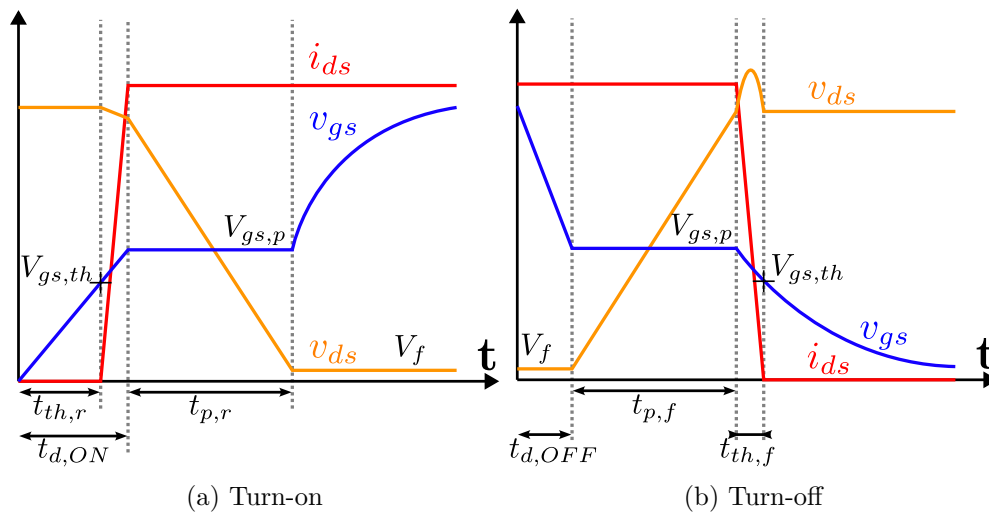


FIGURE 2.3 – Chronogrammes de la commande de la grille d'un transistor

En négligeant l'inductance de boucle du circuit de commande, la charge de C_{iss} se fait approximativement selon un circuit RC suivant une loi d'équation différentielle du premier ordre pendant la durée $t_{d,ON}$ (Fig. 2.3a), tel que :

$$v_{gs} = V_{driver} \left(1 - \exp \left(\frac{-t}{R_{g,tot} \cdot C_{iss}} \right) \right) \quad (2.1)$$

$$t_{d,ON} = R_{g,tot} \cdot C_{iss} \times \ln \left(\frac{V_{driver}}{V_{driver} - V_{gs,p}} \right) \quad (2.2)$$

$$t_{rise} = t_{d,ON} - t_{th,r} = R_{g,tot} \cdot C_{iss} \times \ln \left(\frac{V_{driver} - V_{gs,th}}{V_{gs,th} - V_{gs,p}} \right) \quad (2.3)$$

Comme représenté en Figure 2.3a, lors de la fermeture du transistor, la tension v_{gs} monte jusqu'à passer la tension de seuil $V_{gs,th}$ à partir de laquelle le canal conducteur du transistor commence à se former. A une certaine tension $V_{gs,p}$ apparaît le plateau de *Miller*, la tension v_{gs} est bloquée à la valeur $V_{gs,p}$, souvent estimée à $V_{gs,th} + I_{ds}/g_{fs}$, où g_{fs} est la transconductance du canal. $V_{gs,p}$ peut aussi être retrouvée sur la datasheet en se référant aux courbes de la charge Q_g de la grille en fonction de v_{gs} . Il est provoqué par la rapide chute de la tension drain-source V_{ds} (C_{ds} se décharge à travers le canal nouvellement formé), qui entraîne une décharge tout aussi rapide de C_{gd} tirant à lui la majorité du courant i_g et freinant la charge de C_{gs} .

$$t_{p,r} = R_{g,tot} \cdot C_{gd} \times \frac{V_{ds}}{V_{driver} - V_{gs,p}} \quad (2.4)$$

$$t_{p,r} = R_{g,tot} \cdot \frac{Q_{gd}}{V_{ds}} \cdot \frac{V_{ds}}{V_{driver} - V_{gs,p}} \quad (2.5)$$

Le phénomène exact est complexe, mais en admettant que v_{gs} est constant sur toute la durée $t_{p,r}$, et que v_{ds} passe de sa tension pleine initiale V_{ds} à 0V, nous obtenons l'équation 2.4. Sachant que C_{gs} est une fonction de V_{ds} qui décroît sur cette période, il est plus simple de directement exprimer la durée $t_{p,r}$ en employant la charge Q_{gd} indiquée dans la datasheet [42], ce qui donne l'équation 2.5. Une fois la capacité C_{gd} déchargée, le plateau Miller prend fin et la charge de C_{gs} reprend suivant une loi d'équation différentielle du premier ordre similaire à l'équation 2.1. Les phénomènes dissipatifs se produisent lorsque i_{ds} et v_{ds} sont non nuls. Pour diminuer les pertes par conduction, il faut chercher à réduire au maximum la durée $t_{sw,loss} = t_{rise} + t_{p,r}$. Dans chacune des formules présentées ci-dessus, nous observons que cette durée totale $t_{sw,loss}$ est proportionnelle à $R_{g,tot}$.

En prenant cette fois ci en compte les inductances parasites, le circuit RLC de charge de la grille peut générer une surtension et une oscillation du v_{gs} à a commutation. Nous pouvons estimer la fréquence propre $f_{g,0}$ et le coefficient d'amortissement ζ_g de la charge de la grille :

$$f_{g,0} = \frac{1}{2\pi\sqrt{(L_g + L_{ss})C_{iss}}} \quad (2.6)$$

$$\zeta_g = \frac{R_{g,tot}}{4\pi f_{g,0}(L_g + L_{ss})} = \pi f_{g,0} \cdot R_{g,tot} \cdot C_{iss} = \frac{R_{g,tot}}{2} \cdot \sqrt{\frac{C_{iss}}{L_g + L_{ss}}} \quad (2.7)$$

Quand le système est pseudo-périodique ($\zeta_g < 1$), une oscillation du signal à la pseudo-période T_g apparaît. Le ratio OS_g du dépassement de la tension v_{gs} par rapport à la valeur cible V_{driver} peut être calculé par l'équation 2.9 (dont découle la relation 2.10) :

$$T_g = \frac{1}{f_{g,0}\sqrt{1 - \zeta_g^2}} \quad (2.8)$$

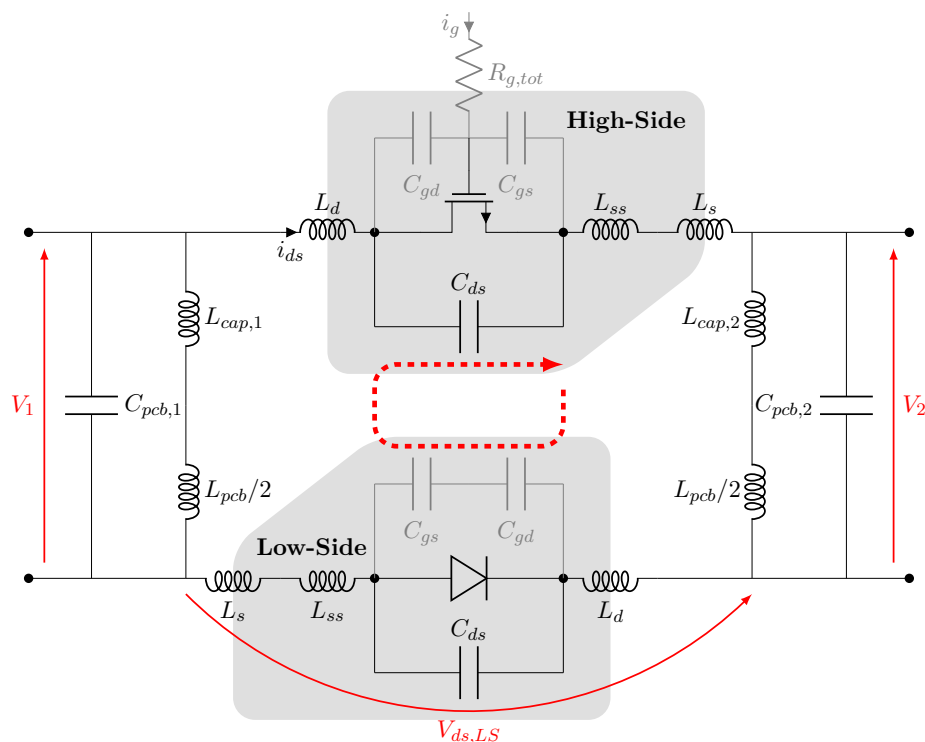
$$OS_g = \exp\left(-\frac{\zeta_g\pi}{\sqrt{1 - \zeta_g^2}}\right) \quad (2.9) \quad \zeta_g = \frac{-\ln(OS_g)}{\sqrt{\ln(OS_g)^2 + \pi^2}} \quad (2.10)$$

Dans le pire scénario, lorsque $\zeta \rightarrow 0$, la surtension peut atteindre 100% de la tension V_{driver} , soit une tension $\max(v_{gs}) = 2 \cdot V_{driver}$. Or, la tension de grille maximale admissible pour le transistor *EPC2022* est de 6V. En imposant $V_{driver} = 5V$, la surtension maximale admissible est de 20%, soit $\zeta \geq 0.456$. L'équation 2.7 montre que le coefficient d'amortissement ζ_g est proportionnel à la résistance de grille $R_{g,tot}$. Une réduction du $R_{g,tot}$ permet certes de réduire les pertes par commutation, mais va aussi diminuer l'amortissement du système, et donc augmenter la valeur de la surtension. Il est possible de soulager cette contrainte sur le $R_{g,tot}$ en réduisant plutôt l'inductance de boucle $L_g + L_{ss}$ lors de la phase de conception du routage du PCB, permettant d'optimiser à la fois la vitesse de commutation du composant et de réduire les risques associés aux surtensions.

La dynamique du circuit de charge de la grille influe aussi sur le circuit de puissance. La Figure 2.4 représente une cellule de commutation et ses éléments parasites lors du *turn-on* de l'interrupteur *High-Side*, soit le blocage de la diode de corps de l'interrupteur *Low-Side* à la fin du *deadtime*. La boucle de puissance forme un pseudo-circuit RLC, mais la description de sa dynamique est complexe lors du phénomène de commutation. L_{pcb} représente l'inductance parasite de la boucle de courant à travers le PCB. $C_{pcb,i}$ est la capacité parasite entre les plans de cuivre de différents potentiels. Pour simplifier le circuit équivalent, nous n'avons représenté que les capacités entre les plans situés du même côté de la cellule de commutation. Aux fréquences élevées des perturbations de la boucle de puissance, les condensateurs utilisés pour tenir les tensions en entrée et sortie de la cellule de commutation présentent un comportement inductif représenté par $L_{cap,i}$. A chaque inductance est associée une résistance équivalente série (ESR). Des modélisations analytiques de l'évolution et de l'oscillation du v_{ds} ont été explorées par [27, 43, 44]. Une estimation du ratio OS_{ds} du dépassement de la tension v_{ds} par rapport à la valeur cible V_{ds} est notamment proposée par [44] :

$$OS_{ds} \approx \frac{1}{V_{ds}} \sqrt{\frac{I_{ds} \cdot V_{gs,p}}{R_{g,tot} C_{gd}} \cdot 2L_{loop}} \quad (2.11)$$

Où L_{loop} est la somme de toutes les inductances présentes sur la boucle de puissance. L'équation 2.11 n'est valable que pour un système extrêmement sous-amorti. Elle suppose en outre que la première oscillation de v_{ds} (au franchissement de V_{ds}) est entièrement déterminée par le dv_{ds}/dt


 FIGURE 2.4 – Circuit équivalent d'une cellule de commutation au *turn-on* du transistor *High Side*

(durées $t_{p,r}$ et $t_{p,f}$ Fig. 2.3), c'est à dire par le phénomène présenté par les équations 2.4 et 2.5. On y retrouve l'influence de C_{gd} , mais aussi de $R_{g,tot}$. La surtension de v_{ds} est inversement proportionnelle à la racine de la résistance de grille, nous observons là encore la nécessité d'optimiser l'inductance de boucle de la puissance, et d'adapter $R_{g,tot}$ en conséquence.

2.2.2 Caractéristiques principales d'un driver

Dans un premier temps, la sélection du driver consiste à trouver un composant capable d'imposer une tension V_{gs} adaptée au transistor. Les transistors *EPC2022* sélectionnés peuvent soutenir un V_{gs} compris entre $-4V$ et $6V$, sa tension de seuil d'activation $V_{gs,th}$ est de $1.4V$. Les composants GaN *normally-off* ont une grille très fragile, la plage du V_{gs} est très restreinte comparée aux transistors Si. Nous imposerons une tension $V_{gs,on} = 5V$ au *turn-on*, ce qui laisse une marge de sécurité de $1V$ avec la limite haute. A cause du seuil d'activation $V_{gs,th}$ très faible, il peut être souhaitable d'imposer une tension légèrement négative pour le *turn-off* du transistor, pour notamment se protéger des commutations spontanées de *Miller* (que nous détaillons plus bas). Mais pour des raisons de complexification du circuit de commande, nous nous contenterons d'imposer un $V_{gs,off} = 0V$, c'est à dire de court-circuiter la grille du transistor sur sa source à travers le driver.

Comme nous avons pu le montrer ci-dessus, la résistance de grille $R_{g,tot}$ est l'un des paramètres déterminants de la vitesse de commutation du transistor. Dans cette somme de résistances, le transistor impose sa résistance interne $R_{g,sw}$. C'est la sélection du driver qui déterminera la valeur minimale $R_{g,tot}$ du circuit de commande de la grille, et nous ajouterons ensuite une résistance externe $R_{g,ext}$ pour ajuster la valeur $R_{g,tot}$ à nos besoins. Un driver peut être représenté par un circuit *totem-pole* : en fonction de l'ordre de commande reçu par le composant, la tension en sortie est

assignée à la valeur de *turn-on* du transistor, ou à la valeur de *turn-off*. Le courant de grille passe alors, respectivement, à travers la résistance interne au driver $R_{g,src}$ ou $R_{g,snk}$. Ces résistances sont des paramètres disponibles dans la datasheet du driver, elles sont généralement différentes l'une de l'autre et sont corrélées aux indications de courant maximal admissible $I_{src,peak}$ et $I_{snk,peak}$, respectivement pour la *source* et le *sink* du driver. Ces deux informations tendent à indiquer la capacité du driver à faire commuter rapidement le transistor, nous chercherons des drivers admettant les courants les plus élevés et dotés des résistances les plus faibles possibles. En supposant l'inductance parasite de grille L_g nulle, nous pouvons calculer la valeur maximale $I_{g,max}$ du courant de grille, qui ne doit pas excéder les courants admissibles par le driver :

$$I_{g,max} = \frac{V_{gs,on}}{\min(R_{g,tot})} \quad (2.12)$$

Un autre facteur important à prendre en considération pour estimer l'impact du driver sur les performances du transistor sont ses caractéristiques dynamiques. Le circuit interne du driver n'est évidemment pas idéal, il est composé de circuits logiques, de transistors, d'éléments capacitifs, d'une isolation, etc. Lorsque le driver reçoit pour consigne de faire commuter le transistor, il lui faut d'abord un certain *temps de propagation* $t_{drv,PD}$ entre la réception de la consigne et le début de son exécution. Ce temps de propagation peut être de plusieurs dizaines de nanosecondes. Dans l'absolu, cette valeur peut poser problème en cas d'implémentation d'une boucle fermée sur la commande du convertisseur, car elle vient retarder la commutation du transistor. Dans les faits, ce délai est généralement trop faible pour avoir un impact sur la régulation en courant ou en tension d'un convertisseur, car il est bien inférieur aux périodes de commutation T_{sw} usuelles des transistors pour nos plages de puissance (qui ne descendent que très rarement sous $1\mu s$).

En revanche, la variation $\Delta t_{drv,PD}$ du $t_{drv,PD}$ entre les puces d'une même référence de driver, et la stabilité de cette valeur sur une même puce entre deux commutations ne peuvent pas être négligées, on parle de *jitter*. Ces irrégularités sont naturellement propagées aux transistors. Les deux transistors d'une même cellule de commutation sont commutés successivement après un temps mort (*deadtime*, ou t_{dt}) pendant lequel ils sont tous deux commandés à l'état bloquant. Le *deadtime* permet d'éviter un court-circuit de la cellule de commutation, qui entraînerait une destruction du convertisseur, mais il engendre aussi des pertes supplémentaires engendrées par le courant I_L traversant la diode de corps de l'un des transistors. Il est donc souhaitable de garder ce *deadtime* à une valeur réduite. Or, si le *jitter* du driver est trop important, il risque de réduire le *deadtime* effectif de la cellule de commutation à des valeurs dangereuses. Si le *deadtime* est à peu près inférieur ou égal au temps de commutation $t_{sw,off}$ du transistor, il n'y a pas destruction de composant mais il provoque une commutation *dure* du transistor *low-side* (au lieu d'un *ZVS*), ce qui augmente considérablement les pertes du convertisseur et le stress sur les transistors. Dans le cas extrême où le *jitter* entraîne un *deadtime* effectif nul ou "négatif" ($t_{dt} \leq \Delta t_{drv,PD}$), la cellule de commutation est court-circuitée et détruite. Nous cherchons donc la variation $\Delta t_{drv,PD}$ la plus faible possible, afin de pouvoir minimiser le *deadtime* requis.

Toutes les caractéristiques ci-dessus aboutissent à une limite de la puissance moyenne du driver. Cette puissance moyenne maximale, trouvable dans la datasheet, dépend du courant tiré par le transistor à chaque commutation, des résistances $R_{g,src}$ et $R_{g,snk}$, de la fréquence de commutation f_{sw} à laquelle travaille le driver et des caractéristiques thermiques de la puce. A nos échelles, il est courant que cette puissance soit estimée à environ $1W$ pour un montage du driver sans dissipateur

thermique ni flux d'air pour aider à son refroidissement. La puissance traversant les drivers de notre convertisseur peut être estimée en employant la valeur Q_g de la charge de la grille du transistor *EPC2022*, la tension V_{drv} délivrée par le driver et la fréquence de commutation f_{sw} , telles que :

$$P_{drv,out} = V_{drv} \cdot Q_g \cdot f_{sw} \quad (2.13)$$

Les transistors *EPC2022* ont une charge de grille $Q_g = 13nC$. Pour $V_{drv} = 5V$ et $f_{sw} = 200kHz$, d'après l'équation 2.13, la puissance traversant un driver vers un unique transistor est $P_{drv,out} = 13mW$, soit bien inférieur à $1W$.

2.2.3 Commutations spontanées de *Miller*

Pendant la phase de temps mort précédant la commutation *turn-on* du transistor *high-side*, les deux transistors de la cellule de commutation sont commandés *bloquants*. L'inductance de sortie force le courant i_L à passer à travers la diode de corps du transistor *low-side*, la tension aux bornes du transistor est $V_{ds,LS} = V_f \approx 0V$. C'est le transistor *high-side* qui tient toute la tension $V_{in}/(N-1)$. Lorsque le transistor *high-side* effectue sa commutation *turn-on*, la diode de corps du transistor *low-side* se bloque et sa tension $v_{ds,LS}$ monte rapidement de $0V$ à $V_{in}/(N-1)$. Il en résulte l'apparition d'un courant de charge à travers C_{gd} , proportionnel à la pente dv_{ds}/dt , qui part du drain vers la grille, et atteint la source à travers le driver, comme représenté en Fig. 2.5.

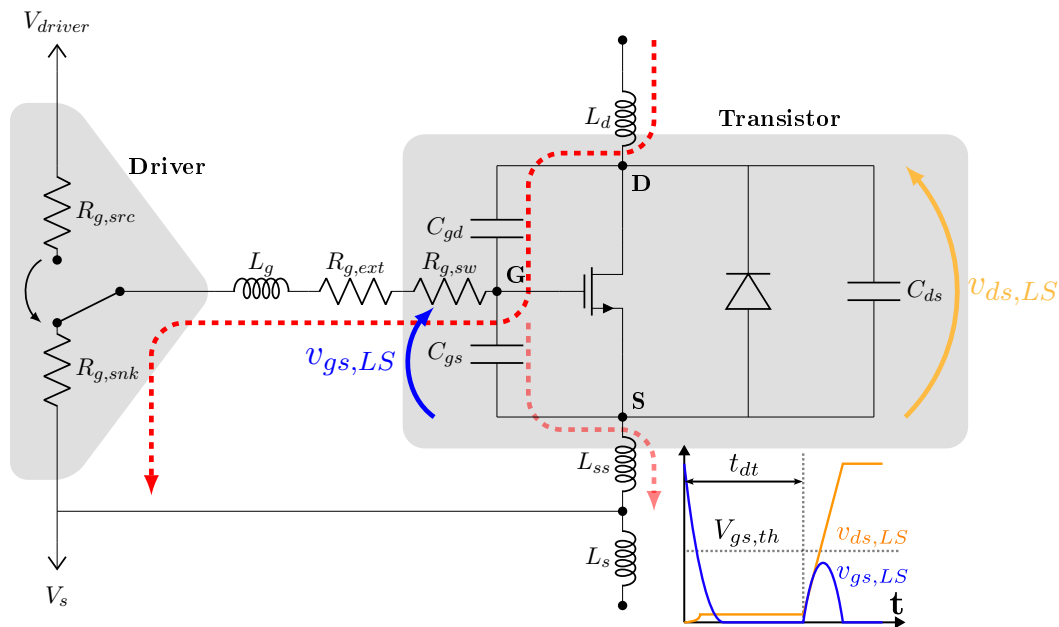


FIGURE 2.5 – Perturbation de la grille du transistor *low-side* lors du *turn-on* du *high-side*

Or, sur le circuit de commande de la grille se trouve la résistance $R_{g,tot}$. Le passage du courant de charge de C_{gd} vient générer une tension proportionnelle à la valeur de cette résistance, et forcer la charge de C_{gs} . L'apparition d'une impulsion positive sur le $v_{gs,LS}$ du transistor *low-side* peut donc être observée au moment la commutation du transistor *high-side*. Bien que brève, si cette impulsion atteint une tension supérieure à la tension de seuil $V_{gs,th}$ du transistor, ce dernier peut

entrer partiellement en conduction, perturber la commutation de la cellule et augmenter les pertes du système, voire entraîner la destruction de la cellule de commutation. Ce phénomène est appelé la commutation spontanée de *Miller*.

La plage de fonctionnement de la grille du transistor *EPC2022*, et d’une manière générale de tous les transistors GaN *normally-off* est extrêmement restreinte [45]. Les composants produits par *EPC* ont tous pour contrainte $V_{gs} \in [-4V; 6V]$ et une tension de seuil $V_{gs,th} \approx 1.4V$. Ils sont de fait très sensibles à tout phénomène pouvant perturber la tension de grille v_{gs} , et notamment à la commutation spontanée de *Miller*. Leur vitesse de commutation dv_{ds}/dt particulièrement élevée engendre un pic de courant sur la grille d’autant plus grand, et donc un risque accru.

La solution directe à ce problème est de minimiser le $R_{g,tot}$ du transistor *low-side*. Il peut être suffisant d’abaisser la résistance $R_{g,ext}$ pour se protéger de cet effet *Miller*, mais ce n’est pas toujours possible (dv_{ds}/dt excessif, courant i_g supérieur aux limites du driver, etc.). Un driver “standard” ne dispose que d’un unique terminal de contrôle de la grille, qui sert de chemin au courant i_g indifféremment lorsque le transistor est commandé *ON* ou *OFF*. Il est possible d’utiliser deux résistances $R_{g,extON}$ et $R_{g,extOFF}$ distinctes pour l’amorçage ou le blocage du transistor en les couplant avec des diodes. Certains drivers assurent eux-mêmes la division des chemins *ON* et *OFF*. La solution idéale consiste à employer un driver proposant un terminal additionnel de *clamp* : une fois la commutation *turn-off* du transistor achevée, la résistance $R_{g,extOFF}$ est ensuite court-circuitée par le terminal de *clamp*, ce qui permet d’abaisser drastiquement la résistance de grille du circuit et de se protéger des commutations spontanées de *Miller*. La résistance $R_{g,extOFF}$ peut ainsi être dimensionnée indépendamment de cet effet.

2.2.4 Sélection du driver

Afin de répondre au mieux à nos besoins, le driver doit être en mesure d’alimenter la grille d’un transistor *EPC2022* à $V_{gs} = 5V$, à une fréquence de $200kHz$, avec une stabilité du signal permettant de mettre en œuvre un temps mort de quelques dizaines de nanosecondes. Il doit être capable de fournir un courant $I_{g,peak}$ le plus élevé possible, c’est à dire d’assurer des temps de commutation $t_{sw,rise}$ et $t_{sw,fall}$ les plus brefs possibles pour minimiser les pertes par commutation du transistor. Le driver doit disposer d’une isolation entre son circuit de commande et le circuit d’alimentation de la grille du transistor. Enfin, il doit idéalement disposer d’un terminal de *clamp*, pour se protéger des commutations spontanées de *Miller*.

Le driver *2EDF7275KXUMA1* fabriqué par *Infineon* a été retenu. Ses caractéristiques importantes ont été résumées en Table 2.2. Au moment de nos recherches, il s’agissait du driver le plus rapide que nous ayons trouvé disposant d’une isolation entrée-sortie et capable de délivrer une tension de $5V$ à la grille du transistor. Il possède deux canaux isolés l’un de l’autre, permettant ainsi de commander deux interrupteurs (soit une cellule de commutation complète) avec un seul composant, avec un *jitter* de $3ns$ entre les deux canaux. En revanche, chaque canal ne dispose que d’un seul terminal de sortie, les chemins *SRC* et *SNK* sont indifférenciés, ce qui nous contraint à utiliser une résistance externe $R_{g,ext}$ unique adaptée à la charge comme à la décharge de Q_g . Le driver ne dispose pas non plus de terminal de *clamp*.

L’importance du choix du driver a pu être démontré expérimentalement. Au lancement de cette thèse, un premier prototype a été réalisé employant un driver différent, le *Si827GB-IS* fabriqué par *Silicon Labs*, dont les caractéristiques sont présentées en Table 2.2. Le composant semble a priori moins performant que le *2EDF7275KXUMA1*. Une comparaison des signaux v_{gs} générés par les

TABLE 2.2 – Comparaison des performances de deux drivers d’après leurs datasheets

	Si827GB-IS	2EDF7275KXUMA1
$V_{dd,out}$	4.2V – 30V	4.5V – 20V
$I_{src,peak}$	1.8A	4A
$I_{snk,peak}$	4A	8A
$R_{g,src}$	2.7 Ω	0.85 Ω
$R_{g,snk}$	1 Ω	0.35 Ω
$t_{drv,rise}$	10.5ns	6.5ns
$t_{drv,fall}$	13.3ns	4.5ns
$\Delta t_{drv,PD}$	$\pm 20ns$	$\pm 6.5ns$

deux drivers est présentée en Fig. 2.6. L’équation 2.12 nous permet de déduire que, théoriquement, la résistance externe $R_{g,ext}$ n’est pas requise pour le *Si827GB-IS*, et qu’une résistance $R_{g,ext} \geq 0.53\Omega$ serait adaptée au *2EDF7275KXUMA1* (calculé à partir de la valeur minimale de $R_{g,src} = 0.42\Omega$ donnée par la datasheet). Nous utiliserons pour ce second montage une résistance $R_{g,ext} = 0.67\Omega$. Nous observons une diminution de 60% des temps de montée et descente du v_{gs} avec le driver *2EDF7275KXUMA1*, par rapport au *Si827GB-IS*.

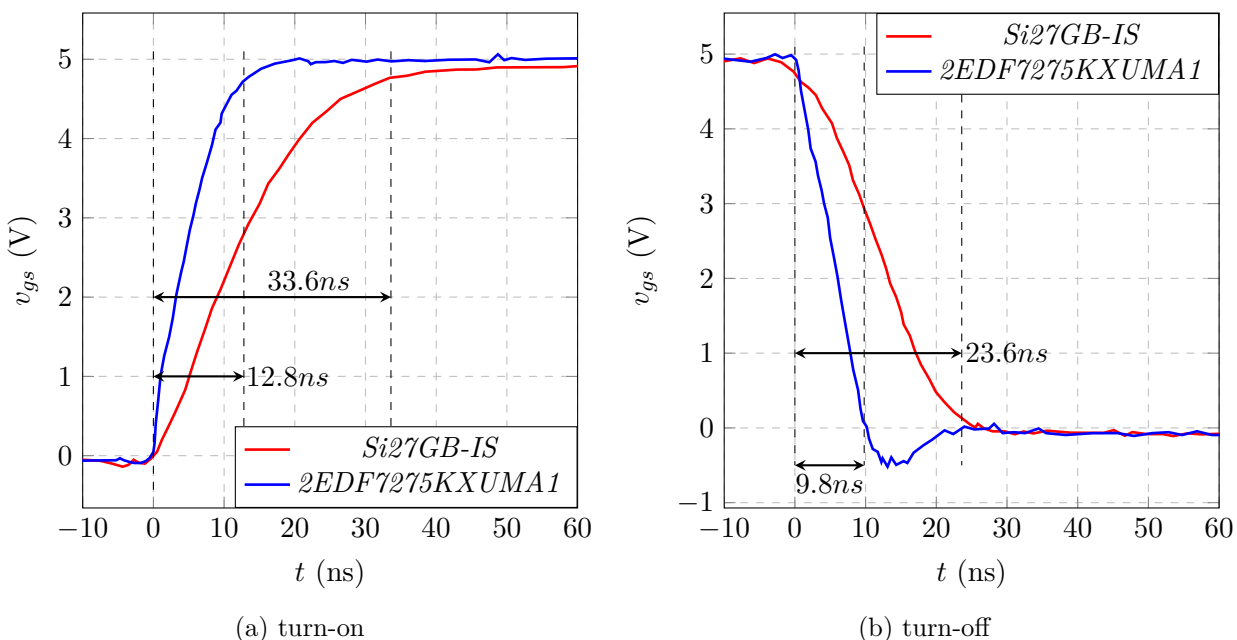


FIGURE 2.6 – Comparaison des v_{gs} mesurés sur un transistor *EPC2022* pour les drivers *Si827GB-IS* et *2EDF7275KXUMA1*, pour $V_{ds} = 0V$, $I_{ds} = 0A$

A l’exception du transistor *low-side* de la première cellule de commutation, la source de tous les transistors est à une tension flottante. C’est la raison pour laquelle les drivers mis en œuvre incluent une isolation entrée-sortie. Il faut à cela associer une alimentation DC-DC isolée chargée de délivrer la puissance chargeant la grille du transistor. Il existe comme alternative le circuit *bootstrap*, permettant d’employer unique alimentation non-isolée pour l’ensemble des drivers, représenté en Fig. 2.7 [5]. La tension d’alimentation de chaque driver est tenue par un condensateur. Les condensateurs sont montés en cascade tels que chacun prélève sa charge sur le condensateur précédent lorsque le

transistor affilié à ce dernier est passant (la source des deux transistors sont au même potentiel pendant ce laps de temps). Lorsque ce transistor est bloquant, la différence de potentiel entre les deux circuits de commandes est tenu par une diode.

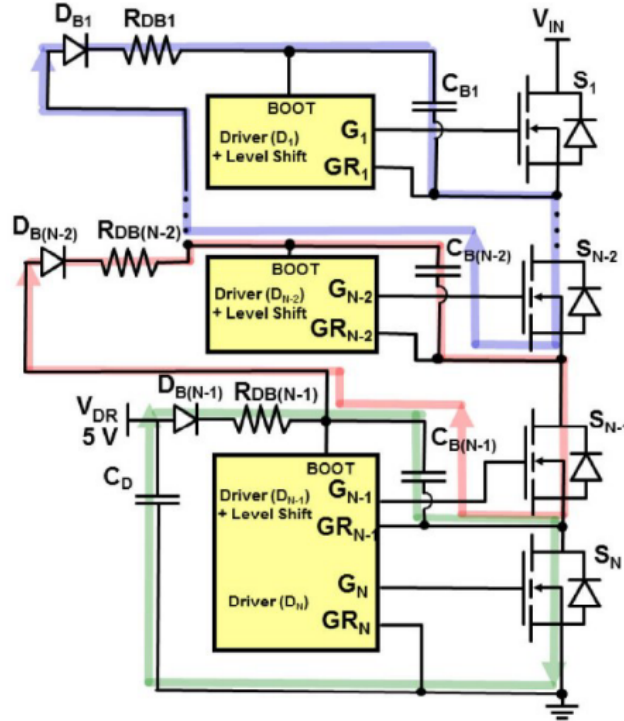


FIGURE 2.7 – Schéma de principe d'une alimentation en *bootstrap* des drivers [5]

Le principal inconvénient du circuit *bootstrap* est lié à la chute de tension causée par la diode de *bootstrap*. Cette dernière implique qu'un condensateur se charge à une tension légèrement inférieure à celle du condensateur le précédant dans le système, de telle sorte que le condensateur en bout de chaîne est chargé à la tension d'alimentation de la base auquel il faut soustraire la somme de toutes les chutes de tension des diodes. Les transistors GaN ayant une plage de tension v_{gs} très faible (dans notre cas, $[0V; 5V]$), cette chute de tension de l'alimentation de la grille est problématique. Pour minimiser ce phénomène, il est d'usage d'employer des diodes *Schottky*, dont la chute de tension est plus faible. Il est possible d'associer à cette approche un circuit *charge-pump*, exploitant la capacité flottante de la topologie de convertisseur *FCML* pour directement charger un condensateur *bootstrap* de potentiel élevé, à l'aide d'une seule diode soit une chute de tension minimale [46]. Il a été aussi proposé dans [5] de remplacer les diodes du circuit *bootstrap* par des transistors GaN, dont la chute de tension est extrêmement faible, et contrôlés de manière synchrone, sans signal de commande additionnel.

Il a été décidé de ne pas utiliser les stratégies décrites ci-dessus pour notre prototype, afin de simplifier son développement. Le driver *2EDF7275KXUMA1* comporte deux canaux, et peut donc contrôler deux interrupteurs. A chaque interrupteur est associée une alimentation DC-DC isolée de puissance $1W$, capable de transposer une tension $5V$ en entrée, en une tension $5V$ en sortie mise au potentiel de la source de l'interrupteur. Au lancement du développement du prototype en 2020, l'alimentation la plus adaptée à cette fonction était le composant *NXE1S0505MC* de *Murata*. Son atout principal tient en sa petite taille ($12.7mm \times 10.41mm \times 4.36mm$), qui lui permet d'être

facilement intégré à un PCB sans nuire à sa densité de puissance. Le composant est largement utilisé dans l'industrie, en témoignent les pénuries survenues sur cette référence tout au long de la thèse. *Würth Elektronik* a commercialisé en 2022 un nouveau composant, le *1769205132 MagI³C*, proposant de meilleures performances pour un volume réduit ($9\text{mm} \times 7\text{mm} \times 3.1\text{mm}$). Nous n'avons toutefois pas pu tester ce composants, tous nos prototypes ayant été développés pour le *NXE1S0505MC*.

2.3 Routage d'un prototype de convertisseur FC3L avec transistors parallélisés

Afin de pouvoir augmenter la densité de puissance du convertisseur *FC3L*, comme observé en Section 1.9, nous avons opté pour la parallélisation directe de deux transistors *EPC2022* pour former un interrupteur. La division par 2 de la résistance à l'état passant effective de l'interrupteur obtenue en parallélisant des transistors permet d'admettre un courant I_{out} deux fois plus élevé sur un unique PCB, soit de réduire d'autant le nombre de *phases* à entrelacer pour obtenir un convertisseur *20kW*. Néanmoins, sa mise en œuvre soulève des problématiques additionnelles de routage de la grille et de la cellule de commutation.

2.3.1 Parallélisation des transistors

Connecter deux transistors en parallèle n'est pas anodin. Le premier effet est une mise en parallèle de leurs capacités parasites, notamment leurs C_{iss} . La valeur effective au niveau de l'interrupteur est donc doublée. Au regard des équations 2.1 - 2.4, nous pouvons donc nous attendre à un doublement du temps de commutation, et donc des pertes par commutation de l'interrupteur par rapport à un transistor unique, à tension V_{ds} égale et pour un courant I_{load} doublé (soit $I_{ds} = I_{load}/2$ dans une configuration parallèle).

Il est important d'assurer la simultanéité de la commutation des deux transistors parallélisés. A cause du doublement de C_{iss} , il peut être tentant de commander les deux transistors séparément, en attribuant un driver individuel à chacun afin de ne pas doubler la chute de tension causée par la résistance interne du driver ($R_{g,src}$ ou $R_{g,snk}$), donc de ne pas ralentir la commutation. Néanmoins, le *jitter* $\Delta t_{drv,PD}$ peut causer la commutation d'un transistor pairé avant l'autre. Lors d'un *turn-on*, quand le $v_{gs,1}$ du transistor 1 en avance atteint $V_{gs,p}$, son canal se forme et la tension v_{ds} de l'interrupteur chute. Le transistor 2 en retard n'a pas encore formé son canal, sa capacité $C_{ds,2}$ n'a d'autre choix que de se décharger à travers le canal du transistor 1. Le transistor doit par ailleurs commuter le courant complet I_{load} seul. Le transistor 1 va donc subir des pertes par commutation drastiquement supérieures. Cette commutation asymétrique peut se produire de manière partielle, si le transistor 2 commence sa commutation avant que le transistor 1 n'ait fini la sienne. Les transistors *EPC2022* commutent en une dizaine de nanosecondes. Dans une configuration où les drivers sont individualisés, le *jitter* $\Delta t_{drv,PD}$ ne doit pas excéder les quelques nanosecondes. Il apparaît donc bien plus prudent de connecter la grille des deux transistors à un unique driver pour assurer la simultanéité des commutations.

En Section 2.2, les mécanismes de charge et décharge de la grille d'un transistor ont été présentés dans le but de pouvoir sélectionner un driver et d'adapter la résistance externe de grille $R_{g,ext}$. La présence d'inductances parasites liées au routage et aux imperfections des composants a été négligée afin de simplifier la présentation des phénomènes. En outre, ces inductances parasites sont difficiles à prendre en compte dans le développement d'un prototype avant la phase de routage, car

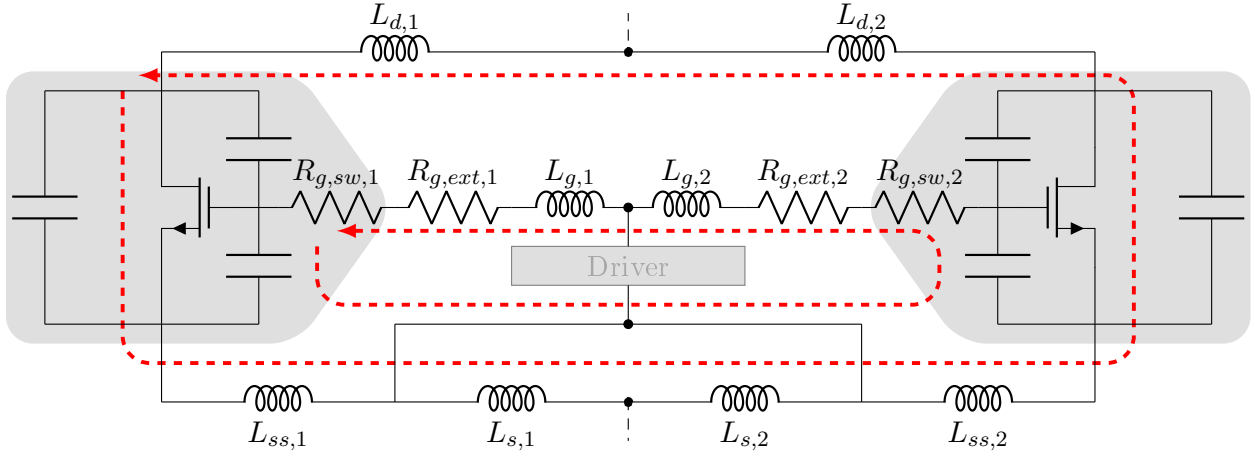


FIGURE 2.8 – Impédances parasites sur un interrupteur composé de deux transistors parallélisés

leur ordre de grandeur peut varier fortement en fonction de la qualité du routage. De nombreuses recherches ont été menées sur l'influence de ces éléments parasites sur la commutation de transistors parallélisés [45, 47–54], que nous résumerons ci-après.

Le circuit équivalent en Figure 2.8 représente l'ensemble des éléments parasites au niveau d'un interrupteur composé de deux MOSFETs parallélisés. Nous nous référerons à la Figure 2.4 pour décrire la cellule de commutation. La somme des inductances $L_{d,i} + L_{s,i}$ inclue les inductances parasites liées au packaging et au routage du PCB pour un unique composant, sur la piste dédiée au passage de la puissance uniquement. L'inductance de source commune $L_{ss,i}$ représente les phénomènes inductifs provoqués sur la piste partagée par le courant commuté et le courant de charge de la grille du transistor. Enfin, $L_{g,i}$ représente l'inductance propre à la piste de charge de la grille.

La mise en parallèle de deux transistors vient former de nouvelles boucles RLC, représentées en Figure 2.8. Si les circuits des deux transistors sont parfaitement symétriques ($L_{d,1} = L_{d,2}$, etc.), les inductances parasites causent des oscillations au même titre que pour un transistor unique, mais elles sont elles-mêmes symétriques entre elles et les boucles RLC liées au parallélisme ne sont pas stimulées. En revanche, en cas d'inégalité des impédances, un phénomène d'oscillation entre les deux transistors peut être observé sur leurs tensions v_{ds} et v_{gs} . Il est possible d'estimer le risque d'oscillation de la grille quand l'interrupteur est en état *ON* ou *OFF*, et en négligeant les effets complexes se produisant pendant la commutation, à partir des coefficients d'amortissement suivants :

$$\zeta_{para,gON} = (R_{g,sw} + R_{g,ext}) \cdot \sqrt{\frac{C_{iss}}{2 \cdot \sum_i^j (L_{g,i} + L_{ss,i} + L_{s,i})}} \quad (2.14)$$

$$\zeta_{para,gOFF} = (R_{g,sw} + R_{g,ext}) \cdot \sqrt{\frac{C_{gd}}{2 \cdot \sum_i^j (L_{g,i} + L_{d,i})}} \quad (2.15)$$

Nous constatons encore une fois l'importance de la résistance de grille externe $R_{g,ext}$, cette fois ci pour limiter les oscillations entre des transistors parallèles. Il peut être tentant de connecter directement les deux grilles des transistors parallélisés sans placer de résistance entre elles, et de placer une unique résistance externe entre cette grille commune et le driver. Mais cela reviendrait à drastiquement abaisser la résistance du circuit RLC à la valeur $2 \cdot R_{g,sw}$, ce qui diminuerait d'autant l'amortissement et augmenterait le risque d'une surtension sur la grille. Afin de minimiser $L_{g,i}$, nous opterons donc pour un routage où la piste de grille se scinde en deux au plus près des transistors, et les résistances $R_{g,ext}$ sont individuelles à chaque grille, comme représenté en Figure 2.8. Les oscillations peuvent être réduites en augmentant la valeur $R_{g,ext}$, ce qui augmente cependant le risque de commutations spontanées de *Miller* [45].

[48] propose une approche analytique pour décrire ces phénomènes pour des transistors GaN *HEMT*, et met en évidence des problématiques de ralentissement des commutations. D'une manière générale, l'inductance totale de la grille $L_{ss} + L_g$ cause un ralentissement de la commutation. Une asymétrie de cette dernière va donc provoquer un retard à la commutation de l'un des deux interrupteurs et engendrer une asymétrie dans la répartition du courant et des pertes. Étonnamment, une asymétrie $L_{s,1} - L_{s,2}$ a le même effet ralentissant sur la vitesse de commutation que l'inductance de source commune L_{ss} .

Une autre approche décrite par [47] suggère au contraire de minimiser les oscillations de parallélisme du v_{gs} , en forçant un transistor à commuter avant l'autre. Deux résistances de grille externes différentes $R_{g,ext2} > R_{g,ext1}$ sont employées, tel que $R_{g,ext2}$ ralentisse suffisamment la commutation du transistor 2 pour que le transistor 1 ait fini sa commutation quand 2 la commence. Ainsi, le transistor 2 commute dans des conditions *ZVS*, ce qui élimine les oscillations en question. Toutefois, cela suppose de sélectionner un transistor suffisamment robuste pour pouvoir commuter le plein courant seul, ainsi que la décharge du transistor retardé comme décrit plus haut. L'utilisation d'un $R_{g,ext}$ plus élevé expose le transistor à un risque plus important de commutations spontanées de *Miller*, il faut donc associer cette stratégie à l'emploi d'un driver doté d'un terminal de *clamp*, ou imposer une tension de grille à l'état bloquant $V_{gs,off}$ négative. Pour ne pas augmenter nos pertes par commutation, nous avons écarté cette approche dans notre conception. En outre, le driver *2EDF7275KXUMA1* sélectionné pour notre projet ne possède pas de terminal de *clamp*.

2.3.2 Couplage inductif entre les boucles de puissance et de grille

Les boucles de courant formées par le circuit de charge de la grille et par le courant de puissance sont nécessairement à proximité l'une de l'autre. Une interaction entre ces deux boucles peut se produire sous la forme d'un couplage inductif M . Ce couplage a une influence importante sur la commutation, pouvant être représenté par une modification des inductances parasites L_d , L_g et L_{ss} [6] telle que :

$$\begin{cases} L'_{ss} = L_{ss} + M \\ L'_g = L_g + M \\ L'_d = L_d - M \end{cases} \quad (2.16)$$

Le signe et la magnitude du couplage M est déterminé par l'orientation des boucles entre elles, comme représenté en Figure 2.9. Le couplage est nul lorsque les boucles sont perpendiculaires l'une à l'autre. Lors de la commutation, un couplage négatif va réduire les oscillations de v_{ds} et v_{gs} ,

mais aussi ralentir la commutation et augmenter les pertes. Et inversement un couplage positif va accélérer la commutation et aggraver le risque de surtensions. Nos transistors *EPC2022* disposent de terminaux de source commune, ce qui implique que L_{ss} est très faible. Dans ce cas de figure, M devient la cause majeure de “pseudo-inductance” de source commune, et L'_{ss} peut même prendre une valeur négative.

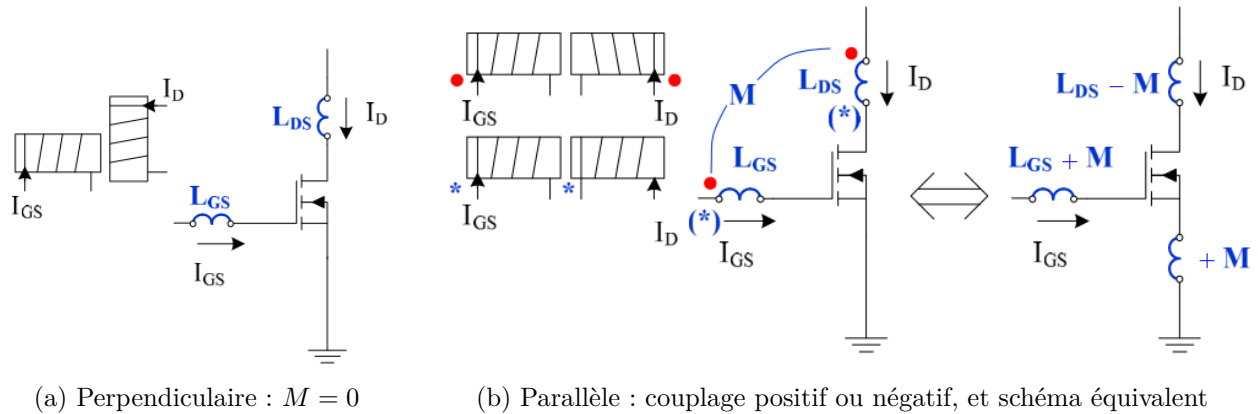


FIGURE 2.9 – Configurations de couplage entre les boucles de puissance et de grille [6]

2.3.3 Comparaison de deux stratégies de routage

Les inductances issues du packaging des transistors sont incompressibles, mais il est possible d’améliorer les inductances causées par le routage du PCB en suivant quelques principes simples. Ce sont les boucles de courant à l’intérieur du PCB qui produisent ces inductances. La valeur de l’inductance parasite est intimement liée à la surface de la boucle, comme le montre l’approximation suivante [55], valable pour une boucle circulant entre deux plans parallèles :

$$L_{pcb} = \mu_0 \frac{e \cdot l}{w} \quad (2.17)$$

Où e est la distance inter-plan, l est la longueur des plans (soit $e \times l$ la surface de la boucle de courant), w est la largeur du plan par rapport au sens du courant, et μ_0 la perméabilité du vide. A noter que cette formule n’est une bonne approximation que si $w \gg e$, mais elle illustre une tendance. Il est possible de réduire L_{pcb} en minimisant la surface de la boucle. Le paramètre l n’est pas facilement réductible à cause de la longueur (dans le sens du courant) incompressible des transistors et condensateurs utilisés formant la boucle. Il est courant de placer les condensateurs d’une cellule de commutation sur la face du PCB opposée à celle des transistors (Fig. 2.10a), ce qui permet de minimiser l autant que possible. L’épaisseur totale du PCB impose alors la valeur de e (entre $1mm$ et $1.6mm$). En plaçant les condensateurs sur la même face que les transistors (Fig. 2.10b), il devient possible d’utiliser la couche de cuivre adjacente à la face du PCB. La longueur l est augmentée pour permettre une réduction drastique de e (environ $100\mu m$, soit une réduction par un facteur 10) [7].

Il est relativement facile de placer les condensateurs à proximité des transistors dans le cas d’un convertisseur *BUCK* n’ayant qu’une seule cellule de commutation. En revanche, pour une topologie multiniveau telle que le *FC3L*, un compromis doit être trouvé pour les capacités flottantes qui doivent être placées entre deux cellules de commutation. Il est possible de placer quelques condensateurs

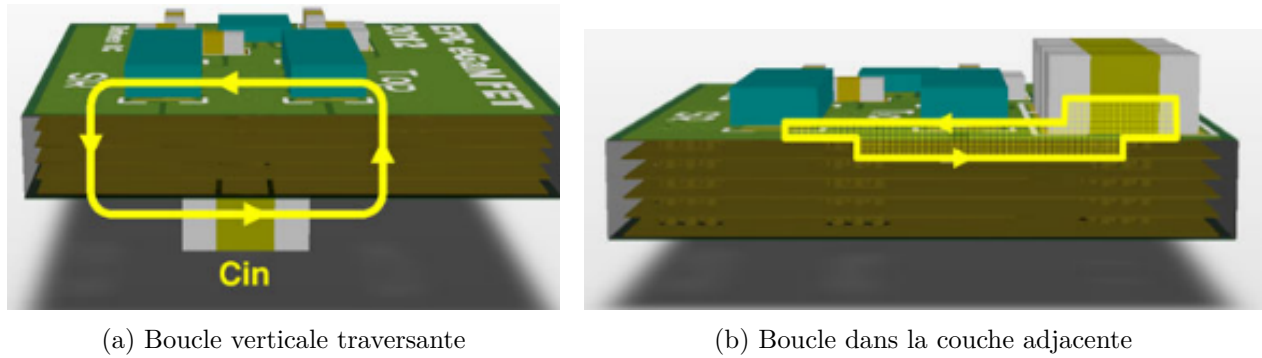


FIGURE 2.10 – Boucles de commutation typiques [7]

au plus près de l’une ou l’autre cellule de commutation, et donc de fractionner l’élément capacitif du circuit pour créer des boucles de commutation aussi réduites que possibles [52]. Dans le cadre d’un convertisseur *FC3L*, quelques condensateurs de la capacité flottante sont placés au plus près des condensateurs de la tension de bus, comme présenté en Figure 2.11. En plus de réduire la taille de la boucle de commutation de la cellule, l’enchevêtrement de ces condensateurs crée localement des boucles de courant de sens opposé, ce qui permet de réduire l’inductance totale de la boucle par effet d’annulation du flux magnétique.

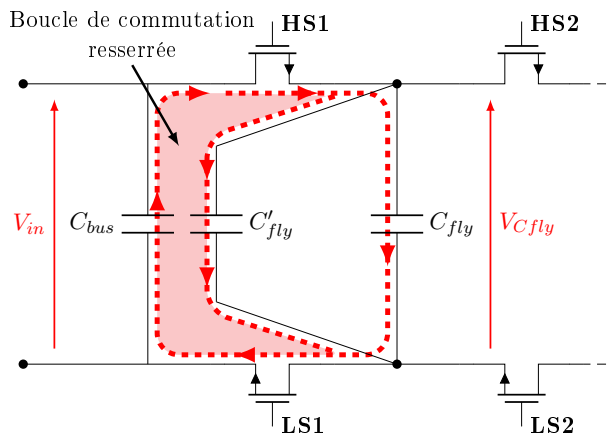


FIGURE 2.11 – Circuit d’une cellule de commutation à condensateurs fractionnés

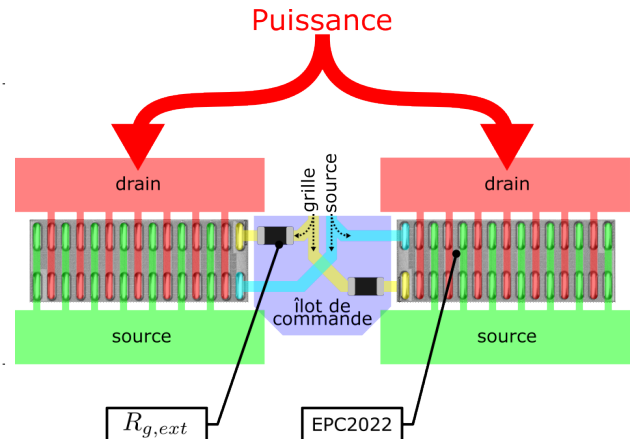
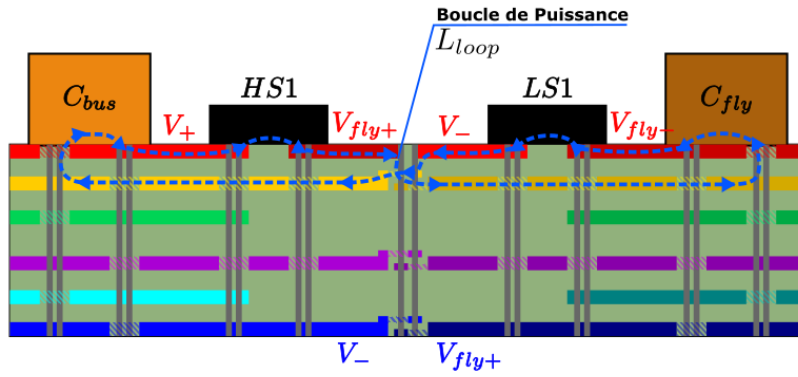


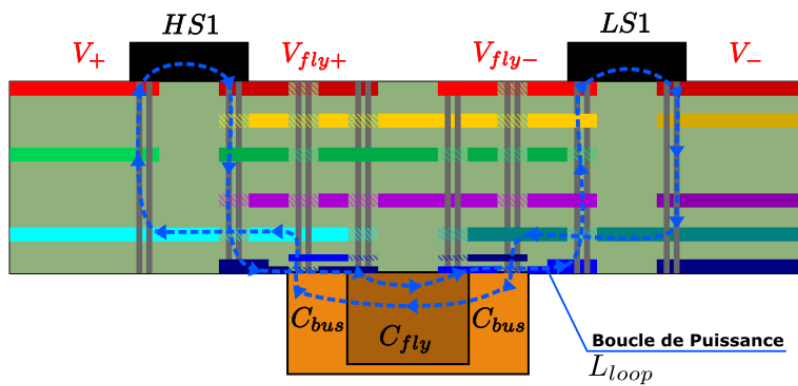
FIGURE 2.12 – Schématisation du routage de deux transistors *EPC2022* parallélisés

Le routage des transistors en parallèle demande une attention particulière. Le packaging *LGA* (Land Grid Array) utilisé pour les transistors *EPC2022* présente une valeur d’inductance drain-source extrêmement faible inférieure à $0.2nH$ [50], mais la disposition de ses terminaux ne facilite pas la parallélisation. Afin d’assurer une symétrie maximale entre les deux transistors, aussi bien pour leurs boucles de puissance que de grille, il faut les placer “dos-à-dos” tel que représenté en Figure 2.12. Le signal de grille se retrouve alors encastré sur un îlot au milieu du passage de la puissance, où il est inenvisageable de placer le circuit de driver. Cela implique que le driver doit être déporté en dehors du chemin dédié à la puissance. Les transistors (et donc le circuit de puissance) sont placés sur la face *top* du PCB, il faut donc faire passer le signal de grille par la face *bottom*. Afin de protéger le signal de grille de la puissance circulant sur les couches supérieures, [52] propose de l’enserrer entre deux couches de cuivre au potentiel de la source, connectées entre elles par une

série de vias borgnes. Il a été décidé pour nos prototypes d'en réduire la taille en retirant la couche extérieure de blindage, remplacée par deux traces à la source encadrant la piste de grille, et en se passant des vias borgnes pour réduire les coûts de production du PCB (exemple en Figure 2.15c).



(a) Routage Adjacent



(b) Routage Traversant

FIGURE 2.13 – Vue en coupe de la cellule de commutation 1 pour les deux stratégies de routage étudiées

Le dissipateur thermique sélectionné, le 960-31-15-D-AB-0 de Wakefield-Vette, couvre une surface de $31\text{mm} \times 31\text{mm}$ sur la face *top* du PCB. Comme nous souhaitons avoir la résistance thermique la plus faible possible, et les transistors EPC2022 ayant une épaisseur de 0.8mm , l'espace entre le dissipateur et le PCB fera donc 1mm . Il n'est donc pas possible de placer les condensateurs au plus près des transistors comme représenté en Figure 2.10b, car ils ne tiendraient pas sous le dissipateur. Pour résoudre ce problème, deux stratégies de routage ont été explorées. L'une, que nous appellerons *routage A* (Adjacent), consiste à placer les condensateurs sur la même face que les transistors mais à les repousser un peu plus loin sur le PCB, en dehors de la surface de contact du dissipateur thermique. L'autre, le *routage T* (Traversant), opte pour placer les condensateurs sur la face opposée, juste en dessous des transistors, et d'entrelacer les condensateurs de tension V_{in} et V_{Cfly} comme décrit par [52]. Les vues en coupe de la cellule de commutation pour les deux dispositions sont représentées en Figure 2.13. Deux prototypes ont été développés sur le logiciel de conception de PCB Altium, représentés en Figures 2.15a et 2.16a. Ils utilisent les mêmes composants, et leur PCB est composé de 6 couches de cuivre d'une épaisseur de $70\mu\text{m}$, pour une épaisseur totale de 1mm .

2.3.3.1 Routage Adjacent

La surface du prototype est de $72 \times 147 \text{mm}^2$, soit une masse d'approximativement 142g . La longueur l de la boucle de commutation est allongée, mais nous bénéficions encore du faible espacement inter-couche de $90 \mu\text{m}$ pour minimiser e . Toutes les boucles de courant sont formées dans l'épaisseur du PCB. Cela est rendu possible par un routage en "fermeture éclair" décrit en Figure 2.14. Lorsque l'interrupteur High-Side est passant, le courant circule sur la couche de cuivre supérieure à travers l'interrupteur, puis descend dans les couches inférieures par une ligne de vias perpendiculaire au courant. Ce même réseau de vias permet de remonter le courant des couches inférieures jusqu'à l'interrupteur Low-Side lorsque High-Side est bloquant, une via sur deux assurant l'une ou l'autre connexion. Cette configuration permet de réduire la boucle de courant dans le plan (et donc l'inductance parasite associée) par rapport aux routages habituels, où les différents plans de cuivre sont interconnectés par des quadrillages de vias occupant une surface donnée. Mais il faut prendre précaution à disposer suffisamment de vias (et donc allonger la ligne de vias) et à maximiser la quantité de cuivre de chaque via pour ne pas dégrader la résistance électrique du PCB.

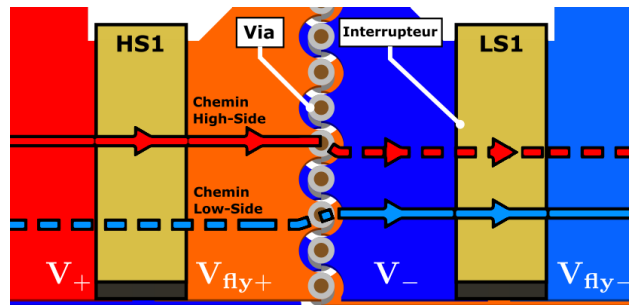
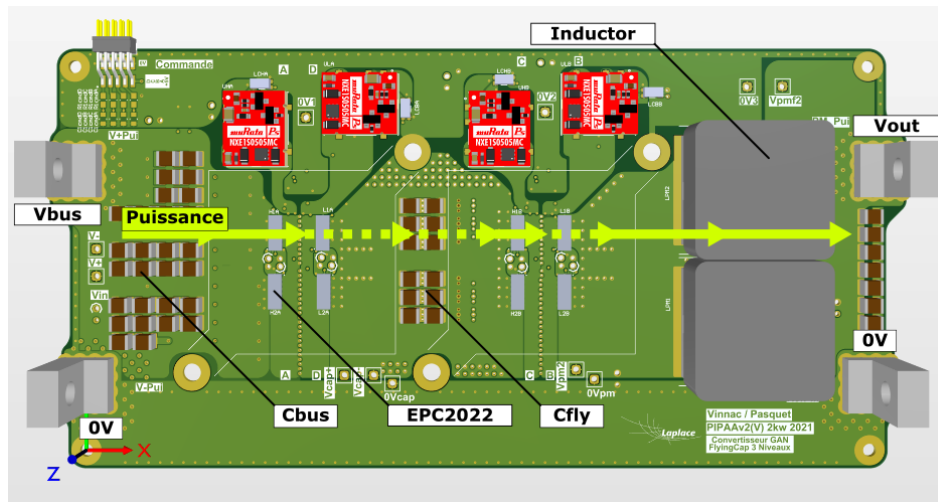
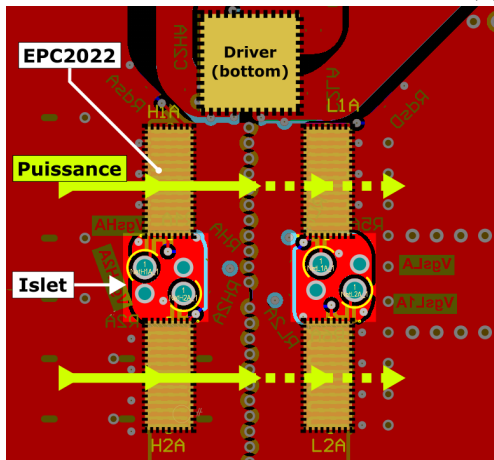


FIGURE 2.14 – Vue du dessus (partielle) de la cellule de commutation 1 du routage *Adjacent*, circulation du courant i_L en fonction du chemin emprunté

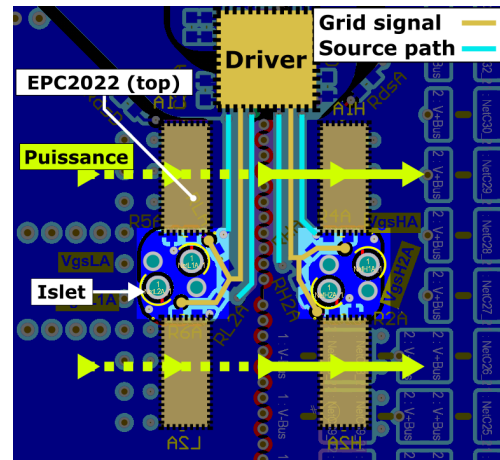
La Figure 2.15c présente le routage des signaux de grille sur la couche *bottom* du PCB. La zone située entre les deux transistors appariés forme un îlot (sur toute l'épaisseur du PCB) maintenu au potentiel de la source kelvin des transistors. La boucle de charge de la grille des transistors est perpendiculaire à la boucle de commutation, ce qui devrait minimiser leur inductance mutuelle M .



(a) Vue du dessus



(b) Routage au *top* de la cellule de commutation

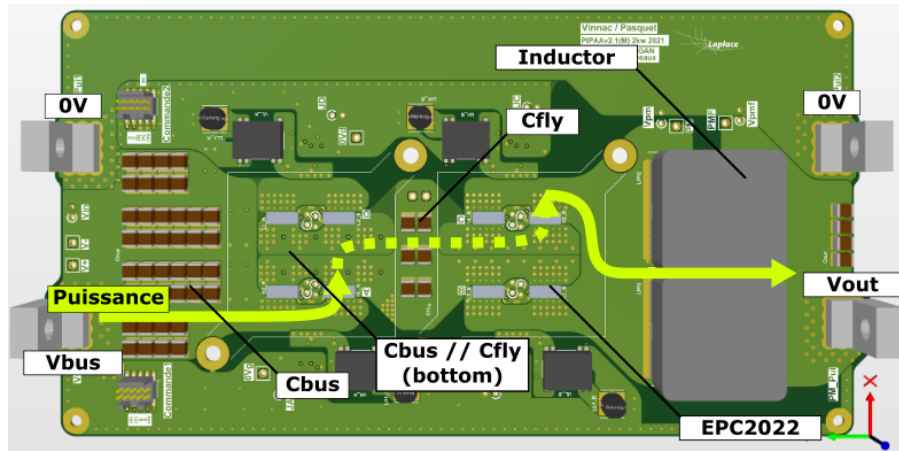


(c) Routage au *bottom* des signaux de grille

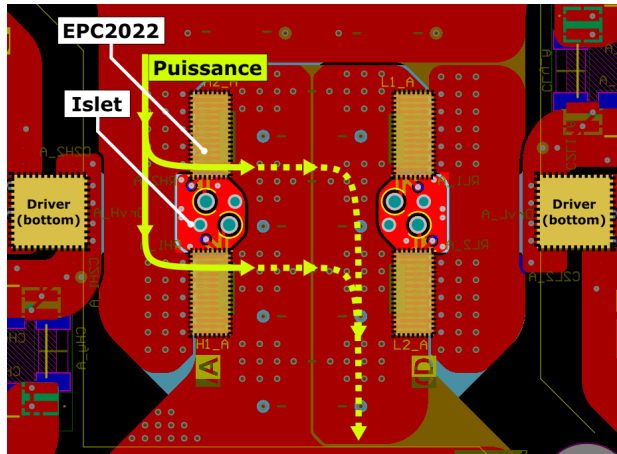
FIGURE 2.15 – Aperçus du routage Adjacent sous *Altium*

2.3.3.2 Routage Traversant

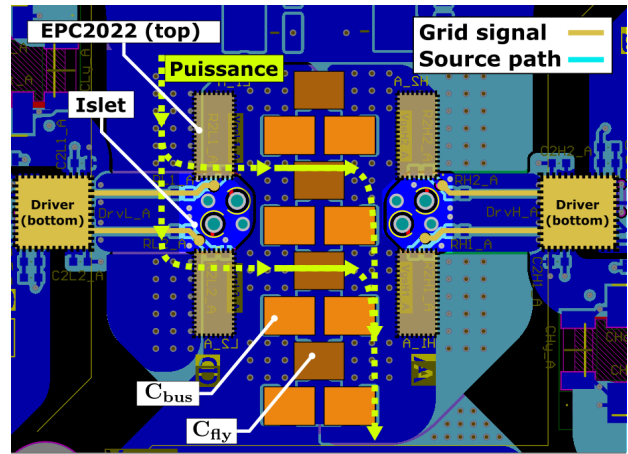
La surface du prototype est de $81 \times 158 \text{mm}^2$, soit une masse d'approximativement 151g . La longueur l de la boucle de commutation est minimisée, mais sa largeur e est fixée par l'épaisseur de 1mm du PCB. Une partie des condensateurs de C_{in} et C_{fly} a été placée entre les deux interrupteurs de la cellule de commutation, et entrelacés afin de créer l'effet d'annulation de flux censé réduire un peu plus l'inductance parasite de boucle. La cellule de commutation est cette fois-ci orientée perpendiculairement à l'arrivée du courant de la puissance, une configuration standard pour ce genre de convertisseurs. Le routage de la grille est représenté en Figure 2.16c. Les îlots au potentiel de source sont à nouveau employés, mais la boucle de charge de la grille est cette fois-ci parallèle à la boucle de puissance, ce qui risque d'engendrer une inductance mutuelle M .



(a) Routage Traversant



(b) Routage au *top* de la cellule de commutation



(c) Routage au *bottom* des signaux de grille

FIGURE 2.16 – Aperçus du routage Traversant sous *Altium*

2.3.4 Simulation par éléments finis des prototypes

Afin de déterminer les inductances parasites des boucles de puissance et de grille des deux routages, des simulations par éléments finis ont été réalisées à l'aide du logiciel *Ansys Q3D*. Les fichiers *Altium* ont d'abord été exportés sous format *.step* sous le logiciel de CAO *Inventor* pour les épurer et formater le *.step* de manière à le rendre exploitable sous *Ansys Q3D*. A partir de la mesure du v_{gs} à vide d'un *EPC2022* réalisée en section 2.2.4 (Fig. 2.6), nous supposons que la commutation se fait sur une durée de l'ordre de la dizaine de nanosecondes. Les simulations *Q3D* se font donc pour une fréquence de stimulation $f_{Q3D} = 100MHz$.

Les résultats de ces simulations pour les deux topologies sont présentés en Table 2.3. L'équation 2.18 [56] permet d'anticiper l'effet de ces éléments parasites sur la fréquence propre de l'oscillation du v_{ds} à la commutation :

$$f_{loop,0} = \frac{1}{2\pi\sqrt{L_{loop} \cdot C_{loop}}} \quad (2.18)$$

$$\zeta_{loop} = \frac{R_{loop}}{2} \cdot \sqrt{\frac{C_{loop}}{L_{loop}}} \quad (2.19)$$

TABLE 2.3 – Résultats des simulations *Ansys Q3D* pour une stimulation $f_{Q3D} = 100MHz$

	Adjacent	Traversant
$L_{pcb,1}$	$0.183nH$	$0.381nH$
$L_{pcb,2}$	$0.151nH$	$2.616nH$
$ESR_{pcb,1}$	$7.37m\Omega$	$10.4m\Omega$
$ESR_{pcb,2}$	$6.44m\Omega$	$56.9m\Omega$
L_g	$[2.67nH; 3.09nH]$	$[2.33nH; 2.34nH]$
M	$[-0.00265nH; 0.00155nH]$	$[-0.00029nH; 0.00161nH]$
$C_{pcb,in}$	$2080pF$	$2200pF$
$C_{pcb,fly}$	$1911pF$	$1220pF$
$C_{pcb,gs}$	$[147pF; 300pF]$	$130pF$
$\zeta_{loop,1}$	0.032	0.027
$f_{loop,0,1}$	$144.6MHz$	$109.2MHz$
$OS_{ds,1}$	24.5%	41.3%
$\zeta_{loop,2}$	0.027	0.033
$f_{loop,0,2}$	$203.9MHz$	$56.48MHz$
$OS_{ds,2}$	22.5%	105.3%

Où $L_{loop,i} = L_d + L_s + L_{ss} + L_{pcb,i} + L_{cap}$ est l'inductance parasite totale de la boucle de puissance, et $R_{loop,i} = R_{dsON}|_{100MHz} + ESR_{pcb,i} + ESR_{cap}$ la résistance de la boucle à $100MHz$. ESR_{pcb} est la résistance équivalente série du PCB sur la boucle de commutation, pour la fréquence $100MHz$. ESR_{cap} et L_{cap} sont la résistance et l'inductance à $100MHz$ de l'ensemble des condensateurs sur la boucle. Nous reviendrons plus en détail sur le calcul de ESR_{cap} et L_{cap} en section 2.4.1, soit $ESR_{cap,1} = 2.5m\Omega$ et $L_{cap,1} = 26.24pH$ pour la cellule 1, et $ESR_{cap,2} = 1.25m\Omega$ et $L_{cap,2} = 13.12pH$ pour la cellule 2. La capacitance C_{loop} est la somme des capacitances C_{oss} , et des capacitances parasites entre les plans de cuivre aux potentiels drain et source, tel que :

$$C_{loop,1} = 2C_{oss}|_{\frac{V_{in}}{2}} + C_{pcb,in} + C_{pcb,fly} \quad (2.20)$$

$$C_{loop,2} = 2C_{oss}|_{\frac{V_{in}}{2}} + C_{pcb,fly2pm} + C_{pcb,fly} \quad (2.21)$$

Où $C_{pcb,in}$ est la capacitance entre le plan de cuivre au potentiel V_{in} et le plan $0V$, $C_{pcb,fly}$ est la capacitance interplanaire entre les deux potentiels de la capacité flottante, et $C_{pcb,fly2pm}$ est la capacitance interplanaire entre les potentiels de la capacité flottante et le plan de point milieu au potentiel V_{pm} . Pour nos routages, cette dernière est négligeable, soit $C_{pcb,fly2pm} \approx 0$. La capacitance C_{oss} effective de l'interrupteur est doublée par la mise en parallèle des deux transistors, nous utiliserons la valeur extraite de la datasheet de l'*EPC2022* pour la tension à l'état ouvert, soit $V_{in}/2$ pour la topologie *FC3L*. Les capacitances C_{pcb} ont été évaluée sous *Ansys Q3D* précédemment. $C_{oss} \approx 900pF \times 2$ pour $V_{ds} = 37.5V$ en prenant en compte la parallélisation des deux transistors. La valeur l'inductance de packaging des *EPC2022* n'est étonnamment pas documentée par *EPC*. La seule estimation disponible du packaging *LGA* d'*EPC* a été réalisée par [57] en 2012, sur des composants plus anciens dont le packaging est moins optimisé (pads plus larges et moins nombreux). Ils ont estimé la valeur de l'inductance de packaging, pour ces composants plus anciens, inférieure à $0.2nH$ à partir de simulations des pistes et terminaux du transistor sur le logiciel *Ansys Maxwell*.

Etant donné que nos simulations sous *Ansys Q3D* prennent en compte les mêmes aspects (intégrés à l'ensemble du PCB) que la méthodologie de [57], nous considérerons que la valeur L_{pcb} simulée inclue déjà les inductances de packaging, soit $L_{loop,i} = L_{pcb,i} + L_{cap,i}$.

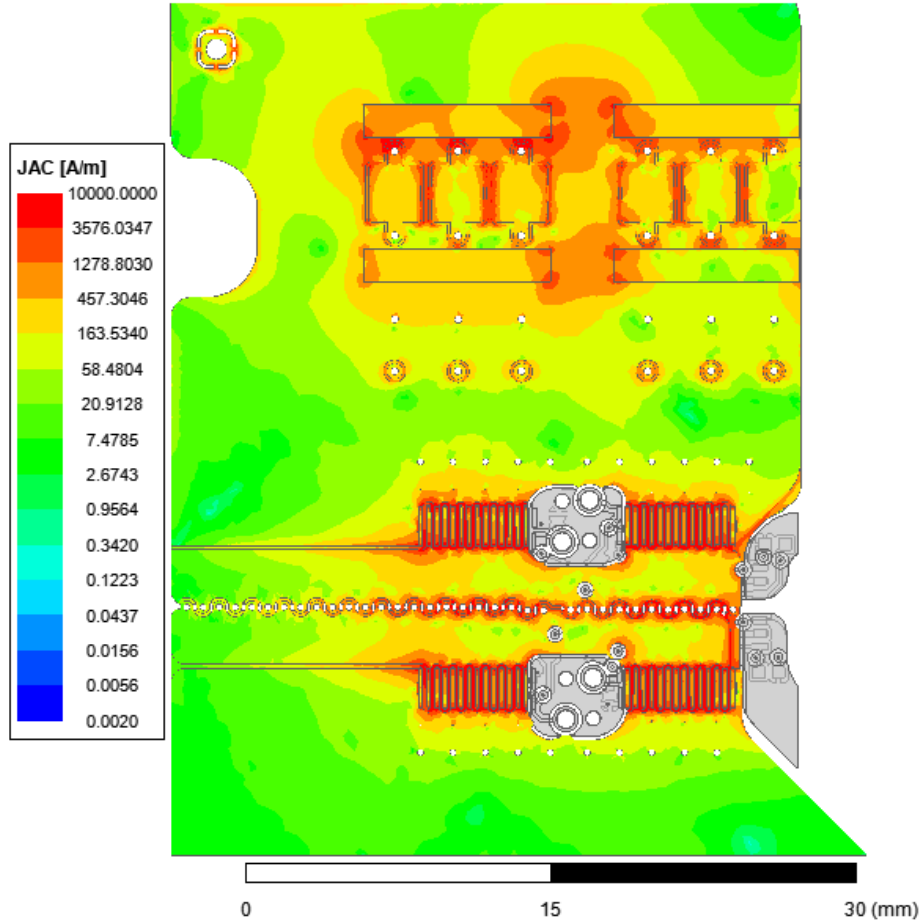
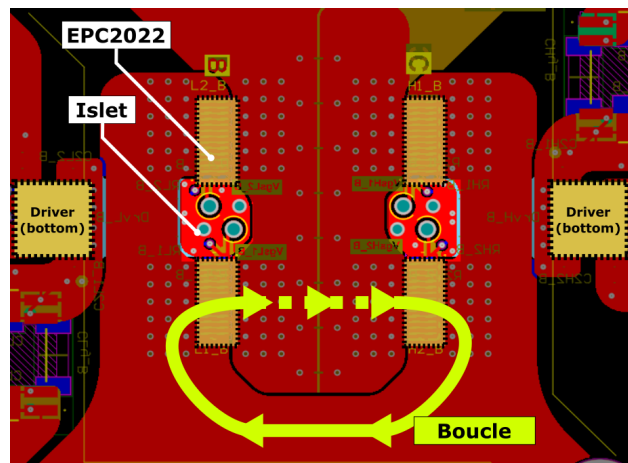


FIGURE 2.17 – Densité de courants surfacique de la cellule de commutation 2 du Prototype *Adjacent*, simulation sous *Ansys Q3D* pour une stimulation $f_{Q3D} = 100MHz$ (vue du dessus)

Les fréquences $f_{loop,0,i}$ données en Table 2.3 ont été calculées en injectant les paramètres simulés sous *Ansys Q3D* dans l'équation 2.18. Faute de disposer d'information adéquates sur sa datasheet, nous supposons que la résistance à l'état passant du transistor pour une fréquence $100MHz$ est égale à sa valeur DC, soit $R_{dsON}|_{100MHz} = 2.4m\Omega$. En injectant l'ensemble de ces paramètres dans l'équation 2.19, nous constatons que les coefficients d'amortissement des boucles de puissance sont très inférieurs à 1. Nous pourrions donc supposer que la pseudo-période $T_{loop} \approx 1/f_{loop,0}$. L'équation 2.11 nous permet d'estimer la surtension de v_{ds} par rapport à $V_{in}/2$, en supposant que la boucle RLC soit très sous-amortie. L'estimation OS_{ds} est calculée pour une commutation *turn-off* à une tension de bus $V_{in} = 75V$ et un courant de charge $I_{out} = 70A$. C_{gd} a été fixé à sa valeur de $300pF$ pour $V_{ds} = 0V$ (début de la commutation), $v_{gs,p}$ la tension de plateau *Miller* est de $2.3V$.

A partir de ces simulations, il est déjà possible d'observer une différence majeure entre le routage *Adjacent* et le routage *Traversant*. Ce dernier présente une inductance parasite sur la boucle de commutation de la cellule 2 de $2.616nH$, soit 7 fois plus importante que l'inductance de la boucle 1 à $0.381nH$. Cela est dû ni plus ni moins qu'à une erreur de routage, le plan de point milieu n'a


 FIGURE 2.18 – Boucle de commutation problématique de la cellule 2, sur le prototype *Traversant*

pas été entrelacé sous la cellule de commutation avec les plans de la capacité flottante. La boucle de commutation n'est alors pas formée dans l'épaisseur du PCB mais dans son plan (Figure 2.18), ce qui augmente drastiquement l'inductance parasite. La surtension $OS_{ds,2}$ est estimée à 105.3% de la tension nominale $V_{in}/2 = 37.5V$ pour un courant de 70A. En se fiant à ces estimations, cette erreur de routage devrait rendre le prototype *Traversant* très prompt à produire du bruit, voire inutilisable avec des oscillations excessives à la commutation.

Si nous nous intéressons uniquement à la cellule 1, qui elle est correctement routée sur les deux prototypes, nous constatons que le routage *Adjacent* présente une inductance 2 fois plus faible que celle du routage *Traversant*, pour une capacité parasite de PCB $C_{pcb,in} + C_{pcb,fly}$ seulement 1.16 fois plus grande. Il en résulte une surtension de la cellule 1 $OS_{ds,1}$ deux fois plus faible pour le routage *Adjacent* par rapport au routage *Traversant*, d'après l'approximation fournie par l'équation 2.11 [44].

Les inductances de grille L_g ont été calculées pour l'interrupteur équivalent formé par la parallélisation des deux transistors. Elles sont relativement équivalentes entre les deux routages, avec un léger avantage pour le routage *Traversant*, pour lequel chaque transistor parallélisé dispose de sa propre piste d'alimentation de la grille et de son propre circuit de driver. Le couplage inductif M entre les signaux de grille et la boucle de puissance est négligeable pour les deux routages, ce qui est idéal.

Même en ignorant l'erreur de routage sur la cellule 2 du routage *Traversant*, le routage *Adjacent* semble le plus prometteur des deux. Non seulement les caractéristiques électriques présentées ici sont meilleures, sa surface est 1.2 fois plus petite que celle du routage *Traversant* (Section 2.3.3). Leurs masses de 202g et 208g (respectivement) sont à peu près équivalentes. Monté dans un convertisseur entrelacé de 10 phases, le routage *Adjacent* résulte en une puissance massique de 7.77kW/kg (2.576kg), contre 7.59kW/kg (2.636kg) pour le routage *Traversant*.

2.4 Expérimentations sur les prototypes

Les designs A et T ont été mis en production. Les prototypes vont servir à mesurer expérimentalement les formes d'onde à la commutation, ainsi que le rendement du convertisseur. L'ensemble des tests ne visent qu'à caractériser les performances de ces prototypes $2kW$, c'est à dire d'une seule des 10 phases du convertisseur entrelacé $20kW$ qui est l'objet de cette étude. Les performances du convertisseur complet seront extrapolées à partir de celles de ses phases. Une ébauche de l'assemblage du convertisseur entrelacé est présentée en Figure 2.19. L'équipement expérimental employé tout au long de ces mesures est récapitulé en Table 2.4.

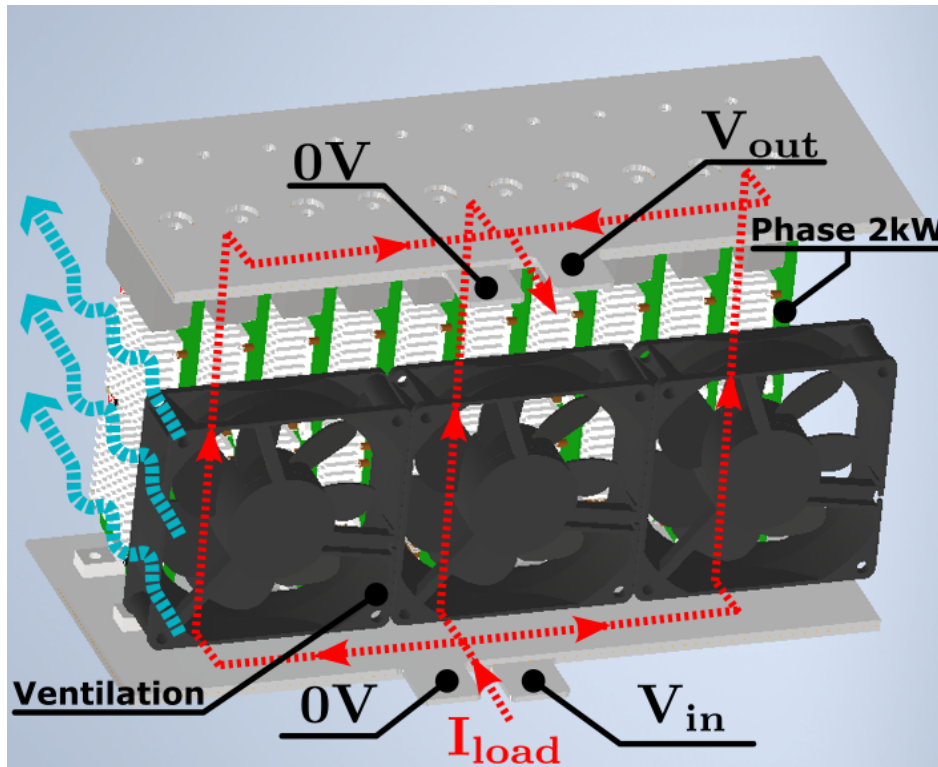


FIGURE 2.19 – Ébauche du convertisseur entrelacé $20kW$

TABLE 2.4 – Équipement expérimental

Oscilloscopes	DPO5054 (<i>Tektronix</i>) - 500MHz TDS3014 (<i>Tektronix</i>) - 100MHz
Sondes tension	pointe + ressort TPP0500 (<i>Tektronix</i>) - 500MHz - 300V _{RMS}
Alimentations	EA-PS8160-170 (<i>Elektro-Automatik</i>) - 160VDC, 170A, 10kW SM15-200D (<i>Delta Elektronika</i>) - 15V, 200A AX503 (<i>Metrix</i>) - 30VDC, 2.5A
Charge Active	EA-ELR9080-510A (<i>Elektro-Automatik</i>) - 80V, 510A, 10.5kW
Caméra thermique	FLIR-564501
Ventilateur	San Ace 80 - 109R0824G4021 - 24VDC, 0.2A, 80 × 80 × 20mm ³

2.4.1 Mesure de l'oscillation du v_{ds} à la commutation

Il est possible de vérifier la fiabilité des simulations *Ansys Q3D* (section 2.3.4) en comparant leurs résultats avec des observations expérimentales. En mesurant la tension v_{ds} d'un interrupteur lors de sa commutation, si le circuit RLC de la boucle de commutation est sous-amortie (présence d'une surtension et d'ondulations du v_{ds}), et en connaissant la capacitance parasite, nous pouvons déduire l'inductance parasite totale de la boucle $L_{loop} \approx L_{pcb}$ en mesurant la fréquence f_{loop} de l'oscillation de v_{ds} à la commutation, à un point de fonctionnement donné. Nous utilisons l'équation 2.22 ci-dessous [56], où f_{loop} est la fréquence mesurée expérimentalement, et C_{loop} reste une estimation théorique basée sur les valeurs C_{pcb} simulées et C_{oss} extraites de la datasheet du transistor *EPC2022*. L'équation 2.22 suppose que le coefficient d'amortissement du système RLC est très sous-amorti, comme montré en section 2.3.4, ce qui permet d'établir que la fréquence mesurée f_{loop} est égale à la fréquence propre $f_{loop,0}$

$$L_{loop} \approx \frac{1}{(2\pi f_{loop})^2 \cdot C_{loop}} \quad (2.22)$$

La mesure est prise sur les prototypes à l'aide d'un oscilloscope *DPO5054* de *Tektronix* (Bande Passante : 500MHz). Pour pouvoir se passer d'une sonde différentielle, elle-même source d'imprécisions supplémentaires, la mesure est prise directement à l'aide d'une sonde pointe-ressort sur la tension découpée V_{pm} . Ce signal nous permet d'observer les tensions drain-source des interrupteurs *Low-Side* de la cellule 1 $V_{ds,LS1}$ ou de la cellule 2 $V_{ds,LS2}$, à différents moments de la période de commutation $T_{sw} = 1/f_{sw}$.

A partir des caractéristiques électriques simulées sous *Ansys Q3D*, il est possible de comparer les formes d'onde, et notamment la surtension drain-source à la commutation, entre les mesures expérimentales et des simulations électriques réalisées sous *LTSPICE*. Pour ce faire, nous réutiliserons le circuit d'une cellule de commutation présenté en Figure 1.11, en lui ajoutant les éléments inductifs et capacitifs de la Table 2.3. Le circuit complet est présenté en Figure 2.20. En parallèle de l'inductance de boucle L_{loop} , il faut placer une résistance reflétant l'énergie dissipée par l'oscillation haute fréquence. Celle-ci est obtenue à partir de la simulation *Ansys Q3D* qui, en plus de calculer l'inductance de boucle L_{loop} , fournit aussi une résistance équivalente en série $ESR_{pcb,i}$. Cette dernière est transformée en résistance équivalente parallèle $EPR_{pcb,i}$ via la relation :

$$ESR_{pcb,i} \cdot I_{HF}^2 = \frac{V_{HF}^2}{EPR_{pcb,i}} \iff EPR_{pcb,i} = \frac{(2\pi L_{loop} \cdot f_{Q3D})^2}{ESR_{pcb,i}} \quad (2.23)$$

Où $f_{Q3D} = 100MHz$ est la fréquence pour laquelle le comportement électrique du PCB a été simulé sous *Ansys Q3D*. Pour une valeur résistive plus rigoureuse, il faudrait à nouveau simuler le PCB à la fréquence f_{loop} calculée à partir de la première simulation et injecter dans l'équation 2.23 la résistance ainsi obtenue et sa fréquence f_{loop} . Néanmoins, pour éviter des temps de simulation excessifs, il a été décidé de s'en tenir aux résultats de simulation à $100MHz$. La Table 2.5 présente les valeurs utilisées dans la simulation *LTSPICE* pour chaque boucle de commutation calculée préalablement en Table 2.3. Il est aussi nécessaire de prendre en compte la capacitance des condensateurs *GRM32EC72A106KE05*, dont la fréquence de coupure est à $1.5MHz$. Aux fréquences en jeu lors de la commutation, ces condensateurs se comportent principalement comme des résistances. Il était raisonnable de négliger l'impact de leur inductance sur L_{loop} , mais leur résistance haute fréquence va

avoir une influence notable sur la surtension et l'amortissement des oscillations. A $100MHz$, la résistance ESR d'un condensateur est de $30m\Omega$, et son inductance est de $0.315nH$. La capacité flottante est composée de 24 condensateurs en parallèle, et la capacité de tension d'entrée de condensateurs en série de 2, parallélisés 48 fois. L'inductance équivalente $L_{fly,100MHz} = L_{in,100MHz} = 13.12pH \approx 0$, et leur résistance parallèle équivalente est $EPR_{fly,100MHz} = EPR_{in,100MHz} = 54.4m\Omega$.

TABLE 2.5 – Résultats des simulations *Ansys Q3D* pour une stimulation $f_{Q3D} = 100MHz$

	Adjacent	Traversant
ESR_{pcb,1}	0.00737 Ω	0.01044 Ω
ESR_{pcb,2}	0.00644 Ω	0.05696 Ω
EPR_{pcb,1}	1.793 Ω	5.489 Ω
EPR_{pcb,2}	1.398 Ω	47.43 Ω

Les Figures 2.21 et 2.22 présentent le blocage des transistors *Low-Side* des deux cellules de commutation, à l'amorçage des transistors *High-Side*, observés expérimentalement ainsi que les courbes résultantes des simulations *LTSPICE*. Nous relevons la fréquence d'oscillation du signal pour en déduire l'inductance de boucle L_{loop} , ainsi que la surtension OS_{ds} à la commutation. Les inductances déduites de l'expérimentation sont comparées aux simulations *Ansys Q3D* en Table 2.6. Les surtensions OS_{ds} expérimentales sont quant à elles comparées aux estimations obtenues par l'équation 2.11 issue de [44], et aux surtensions simulées sous *LTSPICE* en Table 2.7.

Pour les mesures sur le prototype *Adjacent*, la tension d'entrée V_{in} est fixée à $75V$, et le courant de sortie I_{out} à $75A$. Le v_{ds} mesuré au *turn-off* de l'interrupteur est représentée en Figure 2.21. Nous observons deux fréquences d'oscillations. La fréquence la plus élevée correspond à l'inductance de boucle L_{loop} , l'erreur relative entre la fréquence mesurée et la fréquence théorique est de -9.63% pour la cellule 1 et de -0.098% pour la cellule 2, soit une précision satisfaisante des simulations *Ansys Q3D*. En revanche, l'origine de la seconde fréquence d'environ $60MHz$, présente à la commutation des deux cellules, n'a pas été établie. Cette oscillation secondaire rend compliquée la comparaison des surtensions mesurées et simulées, où son impact semble significatif.

Lors du test sur le prototype *Traversant*, il est apparu que l'erreur de routage de la cellule 2 dégrade très fortement la qualité de la commutation. Des relevés ont été effectués à très faible puissance, pour $V_{in} = 20V$ et $I_{out} = 1A$, et sont présentés en Figure 2.22. Nous vérifions d'abord la commutation de *LS1* (Fig. 2.22a). La surtension au dessus de $\frac{V_{in}}{2} = 10V$ est de $6.4V$. Pour de si faibles tensions, l'impact du V_{ds} au moment de la surtension sur la valeur du C_{oss} du transistor est non négligeable. Pour $V_{ds} \approx 15V$, nous considérerons $C_{oss} \approx 1750pF$. En utilisant cette valeur de C_{oss} associée à la fréquence de la première oscillation à $96.1MHz$ dans l'équation 2.22, nous en déduisons $L_{loop,1} = 0.388nH$, soit un faible écart de 1.8% de la valeur simulée.

La commutation de *LS2* (Fig. 2.22b) est, elle, particulièrement oscillante. La surtension est de 101% alors même que le courant commuté n'est que de $1A$. L'estimation de la surtension à 105.3% donnée en Table 2.3 via l'équation 2.11 avait été calculée pour un courant commuté de $70A$ et une tension $V_{ds} = 37.5V$. Pour un courant et une tension si faibles, les approximations sur lesquelles repose l'équation 2.11 deviennent invalides. Quant à l'oscillation, elle présente une fréquence de $38.8MHz$. En prenant $C_{oss} = 1400pF$ correspondant à $V_{ds} = 20V$, l'équation 2.22 nous indique une inductance parasite de boucle $L_{loop,2} = 4.19nH$, soit un écart de -37.3% avec la simulation *Q3D*.

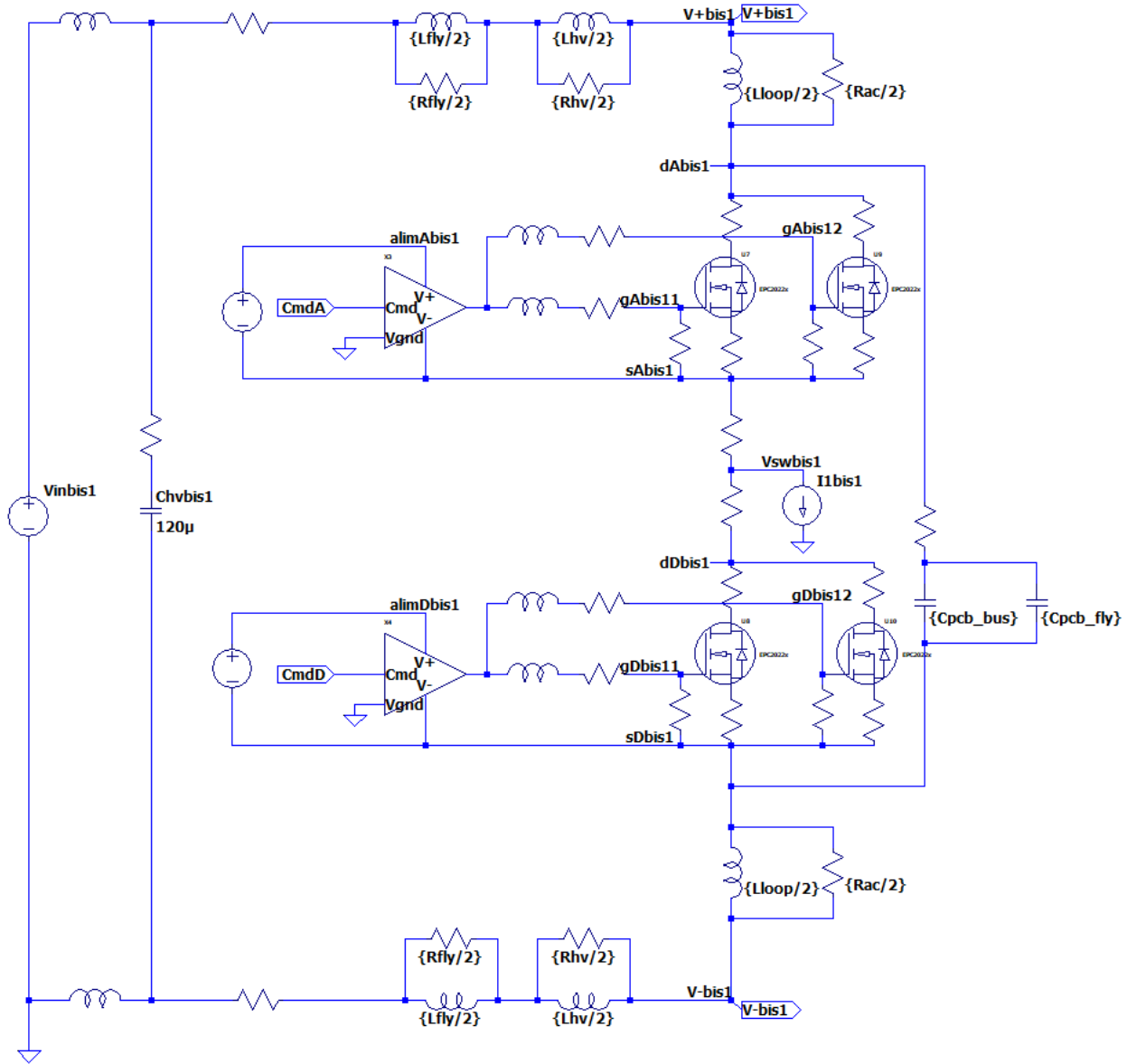


FIGURE 2.20 – Circuit *LTSpice* incluant les inductances et capacitances parasites de la boucle de commutation

Malgré les surtensions et oscillations très excessives de son $V_{ds,LS2}$, il a été tenté de monter le prototype au point de puissance $V_{in} = 75V$, $I_{out} = 70A$, mais, une fois la tension fixée, l'augmentation du courant a mené à sa destruction pour un courant $I_{out} \approx 50A$. A cause de ce défaut de routage, le prototype *Traversant* est inutilisable en puissance. L'observation expérimentale de l'impact de cette erreur de conception reste néanmoins instructive, car elle rappelle l'absolue nécessité d'optimiser autant que possible le routage des convertisseurs employant des transistors de type GaN, sous peine de dégrader son fonctionnement, voire de le rendre complètement inopérant, en cas de routage négligeant.

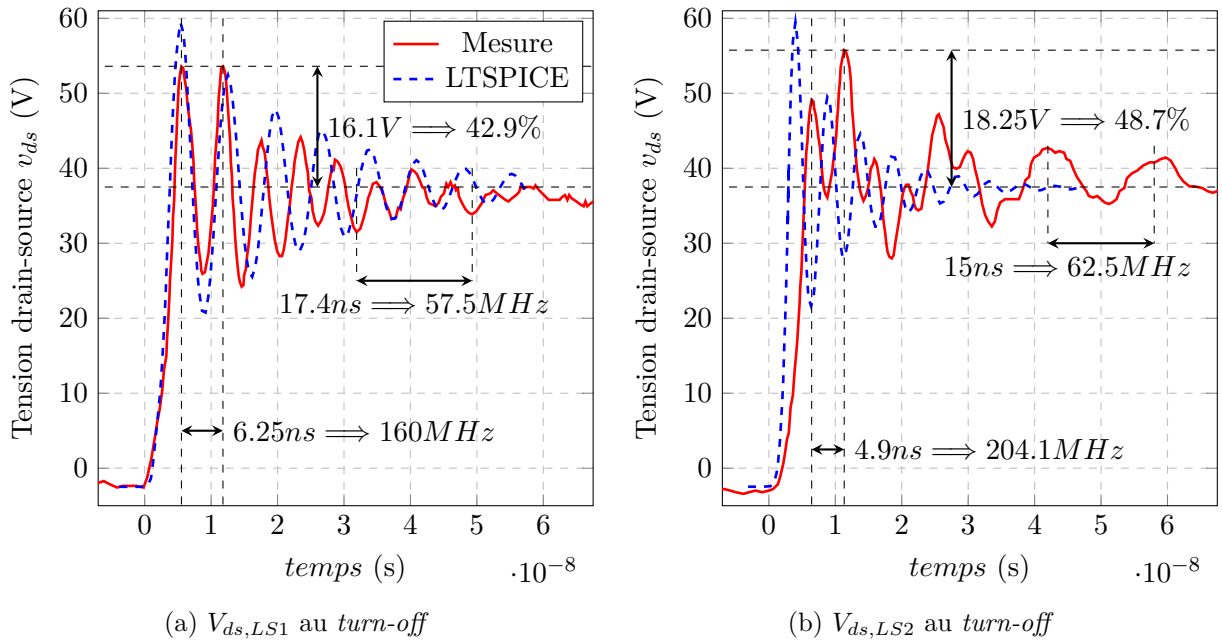


FIGURE 2.21 – Oscillations de la tension v_{ds} de l'interrupteur *Low-Side* des cellules 1 et 2 au *turn-off* du routage *Adjacent*, à $V_{in} = 75V$, $I_{out} = 70A$ (vues oscilloscope : Figures A.1 et A.2 en Annexe)

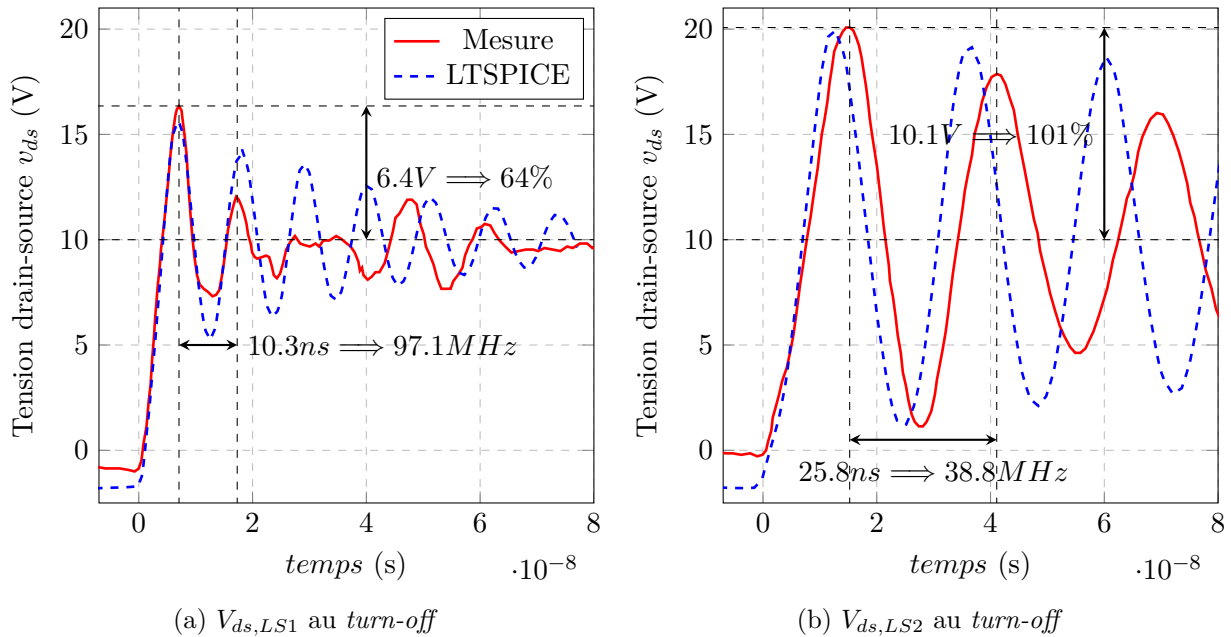


FIGURE 2.22 – Oscillations de la tension v_{ds} de l'interrupteur *Low-Side* des cellules 1 et 2 au *turn-off* du routage *Traversant*, à $V_{in} = 20V$, $I_{out} = 1A$ (vues oscilloscope : Figures A.3 et A.4 en Annexe)

Le calcul des erreurs relatives sur les surtensions (Table 2.7) nous permet de conclure que l'équation 2.11 n'est globalement pas pertinente pour nos transistors GaN à la commutation extrêmement rapide, avec des erreurs au mieux égales à 42.9%. Les surtensions estimées via ce modèle sont toutes inférieures aux surtensions mesurées, ce qui signifie que l'équation 2.11, à défaut d'être précise, ne peut pas non plus être employée comme critère de dimensionnement majorant. Bien que les simula-

TABLE 2.6 – Comparaison des valeurs d’inductance L_{loop} obtenues par simulation et mesure

	Expérimental		Simulations Q3D		Erreurs relatives	
	$L_{loop,1}$	$L_{loop,2}$	$L_{loop,1}$	$L_{loop,2}$	$\epsilon_{loop,1}$	$\epsilon_{loop,2}$
Adjacent	0.171nH	0.163nH	0.209nH	0.164nH	+22.2%	+0.61%
Traversant	0.388nH	4.19nH	0.407nH	2.629nH	+4.9%	-37.3%

TABLE 2.7 – Comparaison des valeurs de surtension OS_{ds} obtenues par simulation et mesure

Surtensions	Expérimental		Equation 2.11		LTSPICE	
	$OS_{ds,1}$	$OS_{ds,2}$	$OS_{ds,1}$	$OS_{ds,2}$	$OS_{ds,1}$	$OS_{ds,2}$
Adjacent	42.9%	48.7%	24.5%	22.5%	58.5%	58.7%
Traversant	64%	101%	18.5%	47.2%	56.2%	99.2%
Erreurs relatives	$\epsilon_{OS,1}$	$\epsilon_{OS,2}$	$\epsilon_{OS,1}$	$\epsilon_{OS,2}$	$\epsilon_{OS,1}$	$\epsilon_{OS,2}$
Adjacent			-42.9%	-53.7%	+36.4%	+20.5%
Traversant			-72%	-53.6%	-12.2%	-1.8%

tions des caractéristiques électriques du PCB par *Ansys Q3D* présentent une bonne précision (Table 2.6), leur implémentation dans des simulations de la commutation sous *LTSPICE* produit des résultats mitigés sur la précision des surtensions, avec des erreurs relatives dont les valeurs absolues sont supérieures à 10% (sauf sur la cellule défectueuse et très oscillante), jusqu’à 36.4%. Une comparaison qualitative des formes d’onde sur les Figures 2.21 et 2.22 semblent néanmoins montrer une bonne correspondance entre la mesure et la simulation. L’impact de la seconde fréquence oscillante à 60MHz est difficile à estimer, et son implémentation dans la simulation *LTSPICE* requerrait de comprendre l’origine exacte du phénomène. En conclusion, il semble donc pertinent, lors de la phase de design précédant la fabrication des prototypes, d’estimer la viabilité de ces derniers via les logiciels de simulation *Ansys Q3D* et *LTSPICE* comme présenté précédemment.

2.4.2 Assemblage du dissipateur Thermique

Les températures élevées dégradent les performances des transistors, notamment leur résistance à l’état passant R_{dsON} , et peuvent accélérer leur vieillissement, voire les détruire (la température maximale de jonction est 150°C). Les prototypes nécessitent un refroidissement actif de leurs transistors, ils ont été conçus pour fonctionner avec un dissipateur thermique par cellule de commutation. Le dissipateur thermique 960-31-15-D-AB-0 de *Wakefield-Vette* a été sélectionné pour son coût faible et sa simplicité de mise en œuvre. Il se fixe au PCB grâce à l’assemblage de taquets montés sur ressorts, ce qui permet d’appliquer une pression sur le matériau couvert, donc d’améliorer la qualité du contact et réduire la résistance thermique. L’assemblage se fait à la main, sans outil, à l’inverse d’un assemblage vissé.

Le dissipateur couvre les 4 transistors d’une cellule de commutation sous sa surface de 31mm × 31mm. Le modèle sélectionné dispose d’ailettes de 15mm de long, la résistance thermique $R_{\theta hs, amb}$ entre la base du dissipateur et l’air ambiant dépend du flux d’air le traversant. Le choix du ventilateur pour le banc d’essai n’a pas fait l’objet d’une étude. Le ventilateur à notre disposition (voir Table 2.4) est capable de fournir un débit d’air de 4m/s (mesuré à l’anémomètre) sur la section latérale de

31mm × 15mm du dissipateur thermique. Pour ce flux d'air, $R_{\theta_{hs,amb}} = 2.25^\circ C/W$. Une canalisation d'air a été dessinée sur le logiciel de CAO *Inventor*, et fabriquée par imprimante 3D. Elle guide le flux d'air de 4m/s sur 4 dissipateurs thermiques, ce qui permet de refroidir deux prototypes montés l'un au dessus de l'autre.

Le dissipateur thermique est associé à un matériau d'interface thermique (*Thermal Interface Material*, ou *TIM*) qui doit assurer la bonne conduction thermique entre le boîtier des transistors et le dissipateur. La Figure 2.23 décrit la modélisation de la dissipation thermique pour les 8 transistors du prototype. La jonction d'un transistor est représentée par une source de flux thermique $\Phi_{loss,i}$ qui correspond aux pertes du transistor i . $R_{\theta_{j,b}} = 1.1^\circ C/W$ et $R_{\theta_{j,c}} = 0.4^\circ C$ sont respectivement les résistances thermiques entre la jonction et le PCB, et entre la jonction et le boîtier du transistor, et sont données par la datasheet de l'*EPC2022*. La résistance thermique entre le boîtier du transistor et le dissipateur thermique, $R_{\theta_{c,hs}}$, et la résistance thermique entre le PCB et le dissipateur, $R_{\theta_{b,hs}}$, sont calculées directement à partir des dimensions du transistor et du dissipateur, et de la conductivité thermique du *TIM* κ_{TIM} , tels que :

$$R_{\theta_{c,hs}} = \frac{1}{\kappa_{TIM}} \cdot \frac{e_{c,hs}}{S_{sw}} \quad (2.24) \quad R_{\theta_{b,hs}} = \left(\frac{e_{b,hs}}{\kappa_{TIM}} + \frac{e_{mask}}{\kappa_{mask}} \right) \cdot \frac{1}{S_{hs} - 4S_{sw}} \quad (2.25)$$

Où $e_{c,hs}$ et $e_{b,hs}$ sont respectivement l'épaisseur du *TIM* boîtier-dissipateur et PCB-dissipateur. S_{sw} est la surface couverte par un transistor (soit 6.05mm × 2.3mm), et S_{hs} la surface du dissipateur (31mm × 31mm). $R_{\theta_{b,hs}}$ prend aussi en compte la résistance thermique du verni isolant qui recouvre le PCB, dont l'épaisseur e_{mask} est estimée à 10μm et la conductivité thermique κ_{mask} à 0.2W · m⁻¹ · K⁻¹. Le modèle simplifié présenté en Figure 2.23 suppose que les couches de cuivre du PCB et l'aluminium du dissipateur thermique sont des conducteurs thermiques parfaits.

Dans nos premiers essais, nous avons utilisé la pâte thermique *TGF4000* de *Bergquist*, qui présente une conductivité thermique de 4W · m⁻¹ · K⁻¹, soit de bonnes performances pour un produit standard dans l'industrie. Du fait qu'il s'agisse d'un liquide, il ne peut pas tenir mécaniquement le dissipateur. Une petite cale en plastique a été conçue à cet effet, mais les résultats l'assemblage étaient mitigés sur la qualité du remplissage du volume sous le dissipateur par le *TIM*. Finalement, un *TIM* solide sous la forme d'un pad thermique a été retenu. Il s'agit du *Sarcon XR-Um* de *Fujipoly*, qui présente une conductivité thermique $\kappa_{TIM} = 17W \cdot m^{-1} \cdot K^{-1}$ bien supérieure à celle du *TGF4000*. Ils peuvent être achetés en feuilles d'épaisseurs 0.3mm, 0.5mm et 1mm.

Après plusieurs essais, l'assemblage du *TIM* retenu est présenté en Figures 2.24 et 2.25. Un compromis a été trouvé entre l'épaisseur $e_{c,hs}$ et la qualité globale du contact thermique. Bien qu'en théorie, une simple couche d'épaisseur 0.3mm soit suffisante pour assurer le contact entre le boîtier et le dissipateur, il faudrait alors se passer de la conduction thermique à travers le PCB. L'entièreté de la pression exercée par le dissipateur thermique serait par ailleurs tenue uniquement par les boîtiers des transistors. Nous ajouterons donc une couche de 1mm d'épaisseur autour des transistors, ce qui nécessite de découper des trous de 2mm par 6mm à l'emplacement des transistors pour les encastrent dans le *TIM*, afin de répartir la pression mécanique et d'améliorer la conduction thermique sur le PCB. Ce processus demande une minutie particulière dans le découpage et à l'assemblage.

La méthode la plus simple et fiable consiste à presser une première fois un carré de 1mm d'épaisseur sur la cellule de commutation pour marquer le contour des transistors, puis de délicatement décoller le *TIM*. Les empreintes des transistors sont ensuite précautionneusement découpées à l'aide d'un cutter. Un découpage laser des trous a été expérimenté sur différents échantillons du *Sarcon*

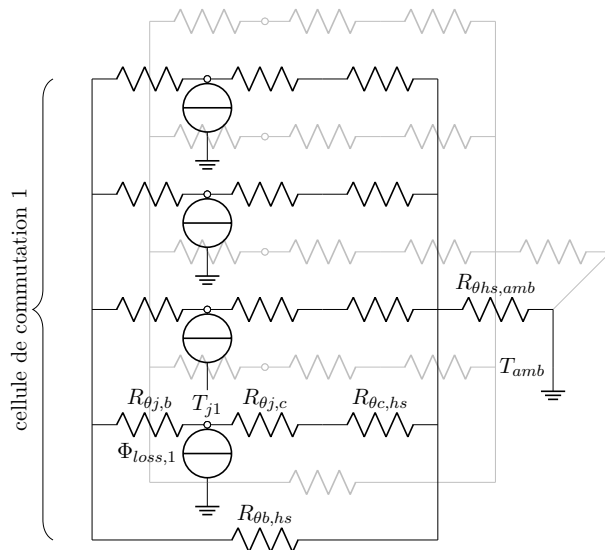


FIGURE 2.23 – Modélisation de la dissipation thermique des 8 transistors du prototype

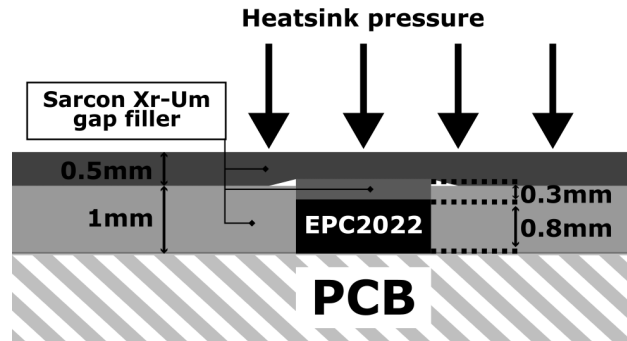


FIGURE 2.24 – Assemblage de l'interface thermique sur un transistor

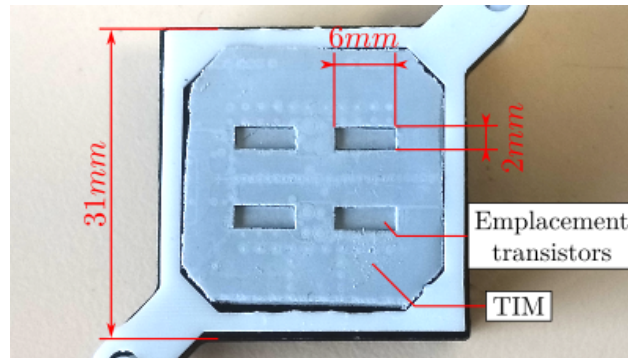
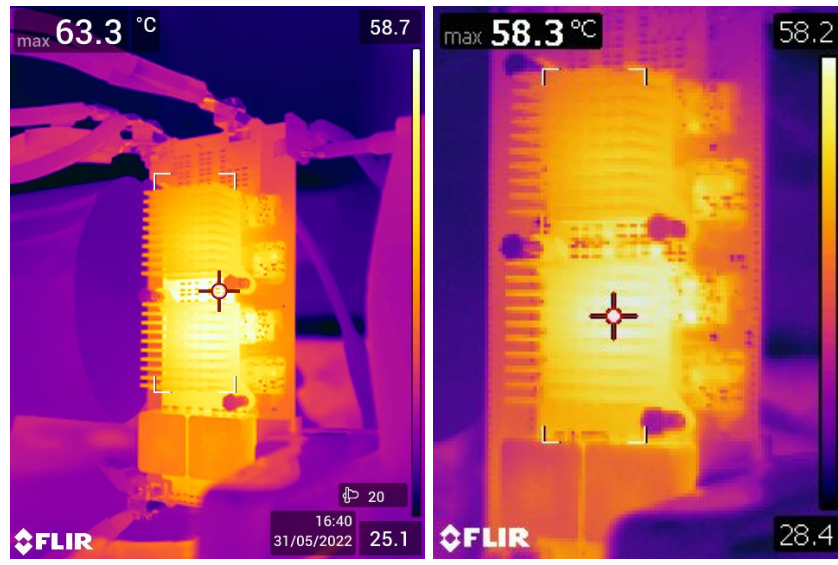


FIGURE 2.25 – Prise de vue de l'assemblage de l'interface thermique sur un dissipateur

XR-Um dans une tentative de simplifier la technique, mais le matériau tend à brûler, voire résiste complètement à la découpe pour l'épaisseur 1mm . Une fois le carré de *TIM* de 1mm d'épaisseur découpé, il est placé manuellement directement sur la cellule de commutation, en ajustant le matériau autour des transistors à la pince pour maximiser le contact latéral avec leur boîtier. Des carrés de $6\text{mm} \times 2\text{mm}$ et d'épaisseur 0.3mm sont ensuite découpés puis placés directement sur le boîtier. Le transistor *EPC2022* a une épaisseur d'environ 0.8mm auquel a été ajouté une couche de *TIM* de 0.3mm , ce qui résulte en un affleurement de 0.2mm au-dessus du *TIM* de 1mm environnant. Si l'on presse le dissipateur dans cette configuration, les transistors vont encaisser entièrement la pression exercée par l'assemblage du dissipateur, ce qui risque de les endommager en plus de dégrader le contact thermique entre le PCB et le dissipateur. Ajouter une couche de 0.5mm au-dessus de l'ensemble permet de lisser les imperfections. La résistance thermique $R_{\theta_{c,hs}}$ augmente d'autant, mais une comparaison expérimentale avec et sans cette couche supplémentaire (Fig. 2.26) montre une diminution de la résistance totale $R_{\theta_{j,amb}}$ grâce à cet ajout. Nous avons observé une diminution de 5°C de la température maximale mesurée sur le PCB. Le point chaud sur la mauvaise configuration (Fig. 2.26a) est par ailleurs sur le PCB, plutôt que sur le dissipateur, confirmant la mauvaise conduction thermique à travers le *TIM*.



(a) Sans la couche 0.5mm

(b) Avec la couche 0.5mm

FIGURE 2.26 – Images thermiques du prototype A, pour $V_{in} = 75V$, $V_{out} = 28V$, $P_{in} = 2kW$, $T_{amb} = 25^{\circ}C$, flux d'air $4m/s$

A partir du modèle décrit en Figure 2.23, la résistance thermique équivalente $R_{\theta_{sw,amb}}$ peut être évaluée entre l'ensemble des transistors et l'air ambiant, soit la capacité du convertisseur à évacuer les pertes des transistors tel que décrit par l'équation 2.26, où P_{sw} et P_{cond} sont les pertes totales par commutation et conduction. Les pertes sont supposées parfaitement réparties entre tous les transistors, soit $\forall i, \Phi_{loss,i} = (P_{sw} + P_{cond})/n_{sw}$, où n_{sw} est le nombre de transistors sur le convertisseur. La température de jonction théorique sera donc elle aussi homogène entre tous les transistors. En résulte l'équation 2.27 qui permet de calculer cette résistance équivalente :

$$\Delta T_j = R_{\theta_{sw,amb}} \cdot (P_{sw} + P_{cond}) \quad (2.26)$$

$$R_{\theta_{sw,amb}} = \frac{1}{n_{hs}} \cdot \left(R_{\theta_{hs,amb}} + \frac{1}{\frac{n_{sw}/n_{hs}}{R_{\theta_{j,c}} + R_{\theta_{c,hs}}} + \frac{1}{\frac{R_{\theta_{j,b}}}{n_{sw}/n_{hs}} + R_{\theta_{b,hs}}}} \right) \quad (2.27)$$

En utilisant les informations présentées précédemment, nous évaluons $R_{\theta_{sw,amb}} = 1.27^{\circ}C/W$ pour l'assemblage du dissipateur présenté en Figure 2.24. Pour des pertes $P_{sw} + P_{cond} = 20W$, l'élévation de température résultante sera $\Delta T_j = +25.4^{\circ}C$ soit, pour une température ambiante $T_{amb} = 25^{\circ}C$, $T_j = 50.4^{\circ}C$.

2.4.3 Mesure du rendement du prototype *Adjacent*

2.4.3.1 Principe de la Méthode par Opposition

Un point crucial dans la conception du convertisseur est la connaissance de son rendement énergétique η sur la plage de fonctionnement complète de la pile à combustible. Il nous faut mesurer ce rendement sur notre prototype *Adjacent* de $2kW$ non seulement pour vérifier qu'il ne dissipe pas de pertes excessives, mais aussi pour pouvoir programmer un algorithme de contrôle du convertisseur final de $20kW$ en *phase shedding* (section 1.2.2). En connaissant le rendement d'une phase du convertisseur entrelacé envisagé, il est possible d'estimer le nombre de phases optimal à un point de fonctionnement donné pour maximiser le rendement de l'ensemble.

La méthode classique d'évaluation du rendement du convertisseur consiste à connecter le convertisseur à une charge active, puis à mesurer la puissance en entrée P_{in} et à lui soustraire la mesure de la puissance en sortie P_{out} pour en déduire les pertes totales du convertisseur P_{loss} , c'est à dire $P_{in} - P_{out} = P_{loss}$. Nous parlerons de méthode *Entrée - Sortie*. Cette méthode montre de très fortes limites dans sa précision pour des convertisseurs à rendements très élevés [58]. L'erreur relative d'une telle méthode est donnée par :

$$\frac{\delta P_{loss}}{P_{loss}} = \frac{1}{1 - \eta} \cdot \frac{\delta P_{in}}{P_{in}} + \frac{\eta}{1 - \eta} \cdot \frac{\delta P_{out}}{P_{out}} \quad (2.28)$$

Où P et η sont respectivement la puissance et le rendement réels que nous cherchons à mesurer, δP est l'erreur absolue de la mesure de P , et $\frac{\delta P}{P}$ l'erreur relative. L'équation 2.28 montre que, pour une précision de mesure $\frac{\delta P}{P}$ constante, plus le rendement η s'approche de 1, plus l'erreur relative sur l'estimation des pertes $\frac{\delta P_{loss}}{P_{loss}}$ est élevée. Nos estimations théoriques menées en section 1.9 indiquent un rendement $\eta \approx 0.98$. À cette valeur de η , pour une précision de mesure hypothétique $\frac{\Delta P_{in}}{P_{in}} = \frac{\delta P_{out}}{P_{out}} = \pm 0.25\%$, l'erreur relative sur les pertes est estimée à $\frac{\delta P_{loss}}{P_{loss}} = \pm 25\%$. L'imprécision intrinsèque à l'outillage de mesure a été amplifiée par un facteur 100 par la méthode de mesure *Entrée - Sortie*.

Plutôt que de procéder ainsi, nous préférons à la méthode *Entrée - Sortie* une autre méthode de mesure dite *par Opposition*, présentée en détail par [58, 59]. La méthode a été spécifiquement développée pour contourner les problèmes d'imprécisions de la mesure pour les convertisseurs à forts rendements, et permet par ailleurs d'isoler les pertes des transistors au *turn-on* $E_{sw,on}$ des pertes au *turn-off* $E_{sw,off}$, ce qui en fait aussi une méthode de choix pour caractériser des transistors. Enfin, cette méthode contourne les difficultés de mise en œuvre des sources et des charges de la méthode *Entrée - Sortie*, pour les convertisseurs fortes puissances typiquement.

Le principe de la méthode par Opposition est décrit par la Figure 2.27. Deux exemplaires du convertisseur sont placés en opposition l'un par rapport à l'autre. Le convertisseur 1 fonctionne en générateur, la puissance s'écoule de ses terminaux $V_{in}|0V$ vers son terminal *point milieu*. Le convertisseur 2 fonctionne en récepteur, la puissance remonte du *point milieu* vers les terminaux $V_{in}|0V$. Le système forme une boucle dans laquelle la puissance s'écoule de manière circulaire d'un convertisseur à l'autre. En parallèle de ce système est connectée une source de tension servant à imposer la tension V_{in} , mais aussi et surtout à fournir la puissance dissipée par le système en opposition, c'est à dire $P_{in} = P_{loss,1} + P_{loss,2}$. La mesure des pertes peut alors se faire directement sur la source de tension d'entrée. La précision de la mesure des pertes est alors identique à la précision

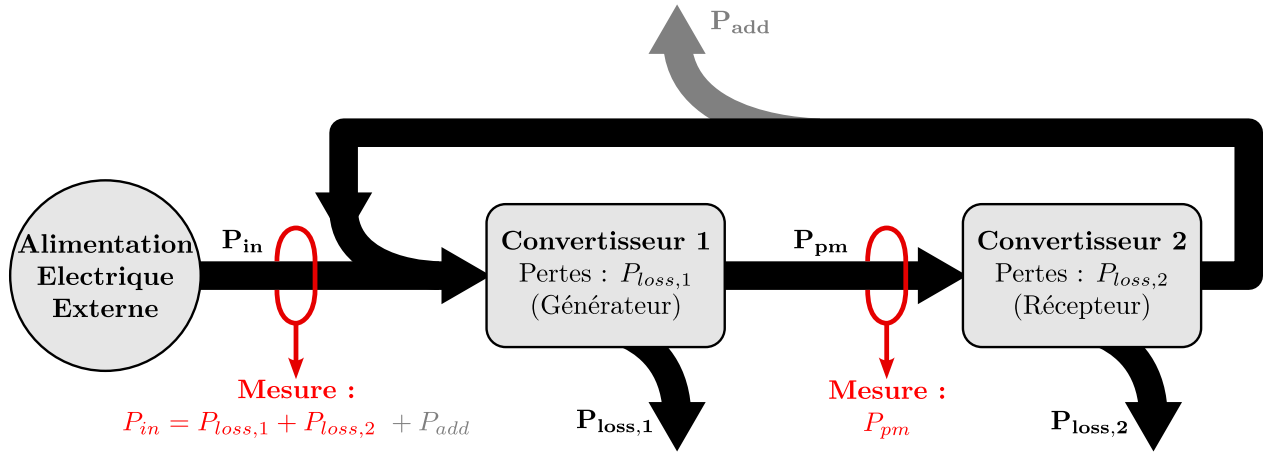


FIGURE 2.27 – Principe de la méthode par Opposition

de l’outillage de mesure, sans effet multiplicatif lié au rendement. Une mesure de la puissance au point milieu P_{pm} est ajoutée entre les deux convertisseurs, permettant de connaître le point de fonctionnement des convertisseurs et calculer le rendement.

La méthode par Opposition sera ici expliquée pour le cas d’un convertisseur DC-DC abaisseur mais, d’une manière générale, elle peut être employée pour tout autre type de convertisseurs électriques, à condition que le convertisseur respecte les règles suivantes (extraites de [58]) :

1. *Réversibilité* : Si deux convertisseurs identiques sont utilisés, l’un opérant en générateur, l’autre en récepteur, ils doivent être réversibles en puissance. A première vue, les systèmes électriques non-réversibles tels que les ponts redresseurs à diodes ou les hacheurs à transistor unique ne peuvent pas être testés avec cette méthode.
2. *Symétrie des modes de fonctionnement* : Même quand les deux convertisseurs sont identiques et réversibles, les conditions de fonctionnement peuvent être très différentes entre les modes générateur et récepteur. Les modes de fonctionnement doivent être aussi symétriques que possibles afin d’obtenir une répartition équilibrée des pertes entre les deux systèmes.

La règle de *Symétrie des modes de fonctionnement* est importante pour déduire correctement les pertes du convertisseur. En effet, la mesure sur P_{in} ne donne pas directement les pertes d’un seul convertisseur, mais les pertes totales du système en Opposition, soit $P_{loss,1} + P_{loss,2}$, auquel il faut aussi ajouter les pertes liées aux éléments ajoutés sur le circuit pour la bonne mise en œuvre du système. Ces pertes additionnelles P_{add} proviennent principalement de l’ajout d’une inductance externe entre les points milieu des deux convertisseurs, dont nous expliquerons la fonction ci-après, mais peuvent aussi provenir dans une moindre mesure des busbars dans lesquels un courant élevé peut circuler. L’ajout d’une inductance externe requière d’avoir déjà caractérisé ses pertes pour pouvoir les déduire des pertes totales du système, P_{in} . En supposant la symétrie des modes de fonctionnement parfaite, c’est à dire $P_{loss,1} = P_{loss,2}$, les pertes P_{loss} d’un unique convertisseur sont estimées au point de fonctionnement donné, dont nous déduisons le rendement :

$$P_{loss} = \frac{1}{2}(P_{in} - P_{add}) \quad (2.29) \quad \eta = 1 - \frac{P_{loss}}{P_{pm} + P_{loss}} = \frac{P_{pm}}{P_{pm} + P_{loss}} \quad (2.30)$$

L'équation 2.30 n'est vraie que si P_{add} est négligée, où si la mesure P_{pm} a été prise au plus près du point milieu du convertisseur 1, en amont (sens du courant DC) des éléments additionnels.

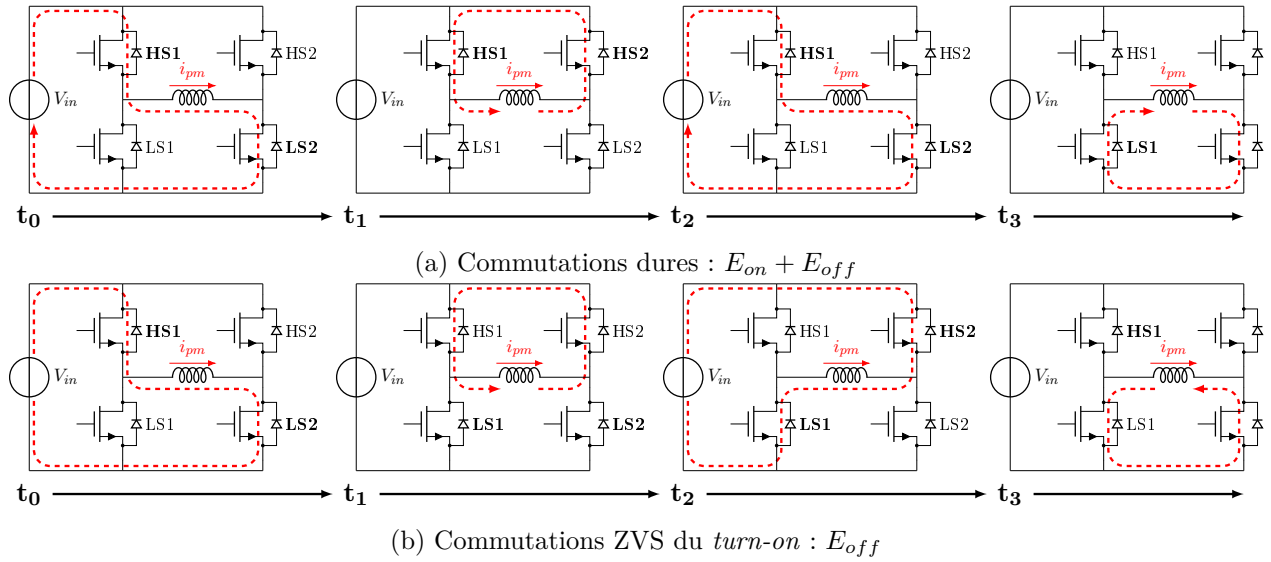


FIGURE 2.28 – Description du contrôle du circuit en Opposition dans ses deux modes de fonctionnement

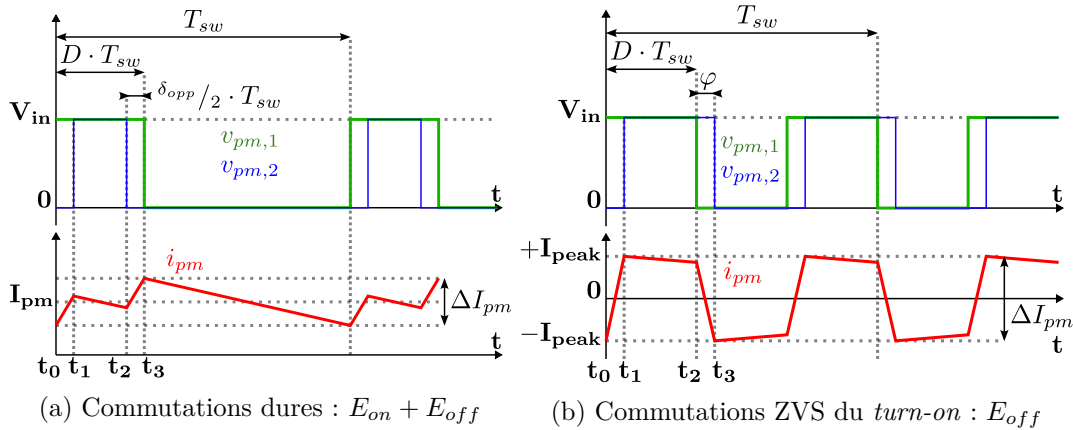


FIGURE 2.29 – Formes d'ondes du circuit en Opposition dans ses deux modes de fonctionnement

La Figure 2.28 représente les modes de fonctionnement de la Méthode par Opposition, un pont en H purement inductif est formé par deux convertisseurs *BUCK* et une inductance L connecte leurs points milieu. Le mode de fonctionnement représenté en Figure 2.28a permet de réaliser la mesure complète des pertes P_{loss} des convertisseurs. Nous parlerons de *mode de fonctionnement* $E_{on} + E_{off}$ car les pertes transistors incluent les énergies de commutation à la fermeture E_{on} et à l'ouverture E_{off} , ainsi que les pertes de recouvrement E_{rr} . Le fonctionnement est le suivant :

- Le convertisseur 1 fonctionne en générateur et le convertisseur 2 en récepteur.
- Un rapport cyclique D_1 est imposé au convertisseur 1 tel que la tension moyenne à son point milieu $V_{pm,1} = D_1 \cdot V_{in}$. Le convertisseur 2 est contrôlé sans déphasage par rapport aux convertisseur 1, avec un rapport cyclique $D_2 = D_1 - \delta_{opp}$.

- Lorsque $\delta_{opp} = 0$, $V_{pm,1} = V_{pm,2}$, et le courant circulant entre les deux convertisseur à travers l'inductance $I_{pm} = 0$, soit $P_{pm} = 0W$.
- Lorsque $\delta_{opp} > 0$, $V_{pm,1} = V_{pm,2} + V_L$. Un différentiel apparaît entre $V_{pm,1}$ et $V_{pm,2}$, ce qui provoque une montée du courant moyen I_{pm} . La Figure 2.29a montre les formes d'ondes du circuit. A l'échelle d'une période de commutation, lorsque $v_{pm,1} \neq v_{pm,2}$, le différentiel vient charger en énergie l'inductance, et le courant i_{pm} augmente. Lorsque $v_{pm,1} = v_{pm,2}$, l'inductance se décharge, i_{pm} diminue.

Le courant i_{pm} se stabilise autour de sa valeur moyenne I_{pm} , suivant une loi différentielle de premier ordre dont la constante de temps est τ_{opp} :

$$I_{pm} = \frac{\delta_{opp} \cdot V_{in}}{R_{opp}} \quad (2.31) \quad \tau_{opp} = \frac{L}{R_{opp}} \quad (2.32)$$

Où R_{opp} est la résistance totale de la boucle de courant, incluant la résistance de l'inductance, des transistors à l'état passant et des busbars. L'équation 2.31 fait l'hypothèse que $R_{opp} \cdot I_{pm}(1 - \delta_{opp}) \ll \delta_{opp} \cdot V_{in}$. Il faut noter que R_{opp} est par design extrêmement faible : nous avons cherché à minimiser les pertes par conduction, et les résistances R_L et R_{busbar} sont maintenues aussi basses que possibles pour ne pas parasiter la mesure des pertes du convertisseur. Dès lors, une très faible variation de δ_{opp} entraîne une montée rapide du courant. Il est d'usage d'implémenter une régulation du courant i_{pm} sur la commande du convertisseur 2 pour cette raison.

L'ondulation (relative) du courant Δi_{pm} est :

$$\Delta i_{pm} = \frac{R_{opp}}{L} \cdot \frac{\max\{D - \delta_{opp}; 1 - D\}}{f_{sw}} \quad (2.33)$$

$$\Delta i_{pm} = \frac{V_{in} \cdot \delta_{opp}}{L \cdot I_{pm}} \cdot \frac{\max\{D - \delta_{opp}; 1 - D\}}{f_{sw}} \quad (2.34)$$

Dans les faits, le *deadtime* t_{dt} est généralement d'une durée non négligeable devant la durée théorique de charge de l'inductance $\delta_{opp} \cdot f_{sw}$. Comme illustré en Figure 2.30, il va avoir pour effet de réduire la durée de charge de l'inductance, et de rajouter des durées de décharge à $v_L = -V_f$, où V_f est la tension de seuil de la diode de corps du transistor. Ainsi, nous obtenons les nouveaux rapport cyclique et courant nominal effectifs δ'_{opp} et I'_{pm} :

$$\delta'_{opp} = \delta_{opp} - 2 \cdot t_{dt} \cdot f_{sw} \quad (2.35) \quad I'_{pm} = \frac{\delta'_{opp} \cdot V_{in} - V_f \cdot 2t_{dt} \cdot f_{sw}}{R_{opp}} \quad (2.36)$$

L'ondulation effective $\Delta i'_{pm}$ peut toujours être calculée à partir de l'équation 2.33, à condition de substituer I_{pm} et δ_{opp} par I'_{pm} et δ'_{opp} . Lorsque $\delta'_{opp} \leq 0$, le circuit en Opposition ne génère aucun courant i_{pm} . En fonctionnement, le *deadtime* t_{dt} et la tension de seuil de la diode de corps V_f vont grandement contribuer à réduire la montée du courant. Par exemple, pour $V_{in} = 40V$, $R_{opp} = 10m\Omega$ et $\delta_{opp} = 4\%$, l'équation 2.31 indique que $I_{pm} = 160A$, soit un courant très excessif pour nos transistors *EPC2022*. Toutefois, en prenant en compte la diode $V_f = 1.8V$ (valeur issue

de la datasheet du transistor GaN *EPC2022*) et un *deadtime* $t_{dt} = 35ns$ pour une fréquence de commutation $f_{sw} = 400kHz$, les équations 2.35 et 2.36 indiquent $\delta'_{opp} = 1.2\%$ et $I_{pm}' = 42.96A$, soit une réduction de 73.15% du courant nominal théorique.

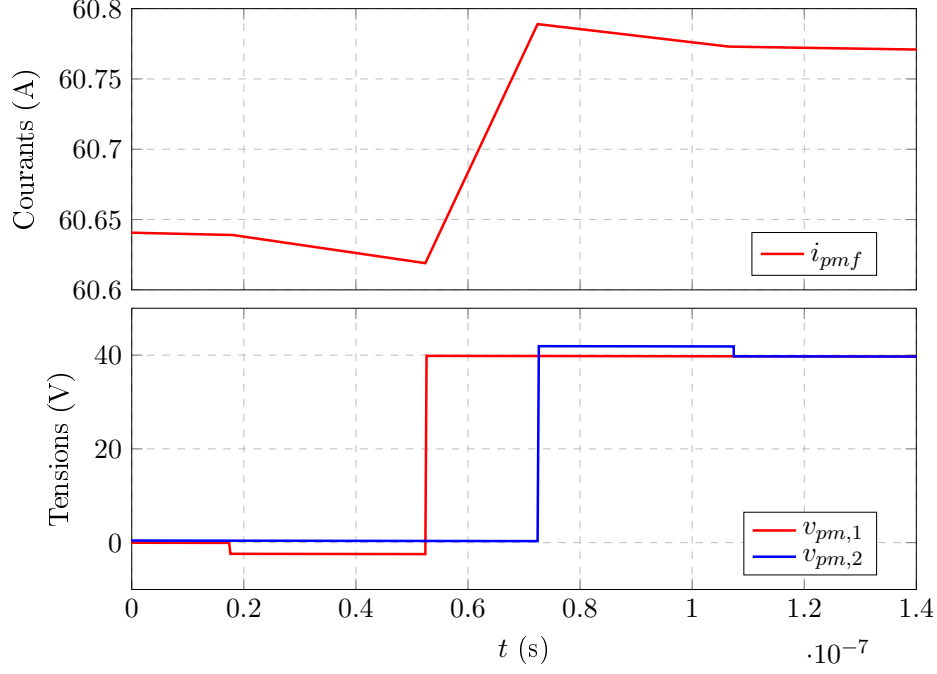


FIGURE 2.30 – Formes d’ondes du mode de fonctionnement $E_{on} + E_{off}$ pour un *deadtime* $t_{dt} \neq 0$

Les pertes P_{loss} mesurées incluent donc les pertes par conduction et commutation de manière indifférenciée. A condition de pouvoir mesurer les pertes additionnelles du circuit P_{add} dont font partie les pertes de l’inductance P_L , ou bien de pouvoir estimer via une modélisation fiable (cf. Section 1.7), il est possible d’isoler les pertes par conduction du convertisseur des pertes par commutation de ses transistors. A un point de fonctionnement donné (V_{in} et P_{pm} fixés), la fréquence de commutation f_{sw} est variée et les pertes P_{loss} sont relevées à intervalle régulier, desquelles sont déduites les pertes P_L . La variation de $P_{loss} - \frac{P_{add}}{2}$ suivant f_{sw} suit une loi affine, dont le terme constant est P_{cond} , et la pente est $E_{sw,tot}$ la somme des énergies de commutation sur une période $1/f_{sw}$.

Il est ensuite possible de différencier au sein de E_{sw} les énergies E_{on} et E_{off} en mettant en œuvre un second mode de fonctionnement de la méthode par Opposition, que nous nommerons ici *mode de fonctionnement* E_{off} , représenté en Figure 2.28b. Ce mode de fonctionnement force les transistors à réaliser leurs *turn-on* à tension v_{ds} nulle, et donc $E_{sw} = E_{off}$:

- Des rapports cycliques égaux $D_1 = D_2$ sont imposés, ainsi qu’un déphasage φ entre les deux convertisseurs. L’augmentation de φ va engendrer l’apparition d’un courant alternatif trapézoïdal. La Figure 2.29b montre les formes d’ondes du circuit.
- Lorsque $v_{pm,1} \neq v_{pm,2}$, $v_L = \pm V_{in}$ et $\frac{di_{pm}}{dt} \neq 0$, le courant change de $\pm I_{peak}$ à $\mp I_{peak}$. Autrement, lorsque $v_{pm,1} = v_{pm,2}$, $v_L = 0V$ et $\frac{di_{pm}}{dt} = 0$, le courant se maintient à $\pm I_{opp,p}$.
- à t_0 , le courant $i_{pm} < 0$ circule à travers la diode de corps du transistor HS1 pendant le *deadtime*, soit $E_{on,HS1} = 0J$ et $E_{off,LS1} > 0J$.

- à t_1 , le courant $i_{pm} > 0$ circule à travers la diode de corps du transistor HS2 pendant le deadtime, soit $E_{on,HS2} = 0J$ et $E_{off,LS2} > 0J$.
- à t_2 , le courant $i_{pm} > 0$ circule à travers la diode de corps du transistor LS1 pendant le deadtime, soit $E_{on,LS1} = 0J$ et $E_{off,HS1} > 0J$.
- à t_3 , le courant $i_{pm} < 0$ circule à travers la diode de corps du transistor LS2 pendant le deadtime, soit $E_{on,LS2} = 0J$ et $E_{off,HS2} > 0J$.

A point de fonctionnement fixe, en faisant varier la fréquence, nous déduisons les pertes E_{off} des transistors. Par simple soustraction des pertes $E_{on} + E_{off}$ précédemment mesurées, les pertes E_{on} sont alors connues.

2.4.3.2 Montage Expérimental & erreurs relatives

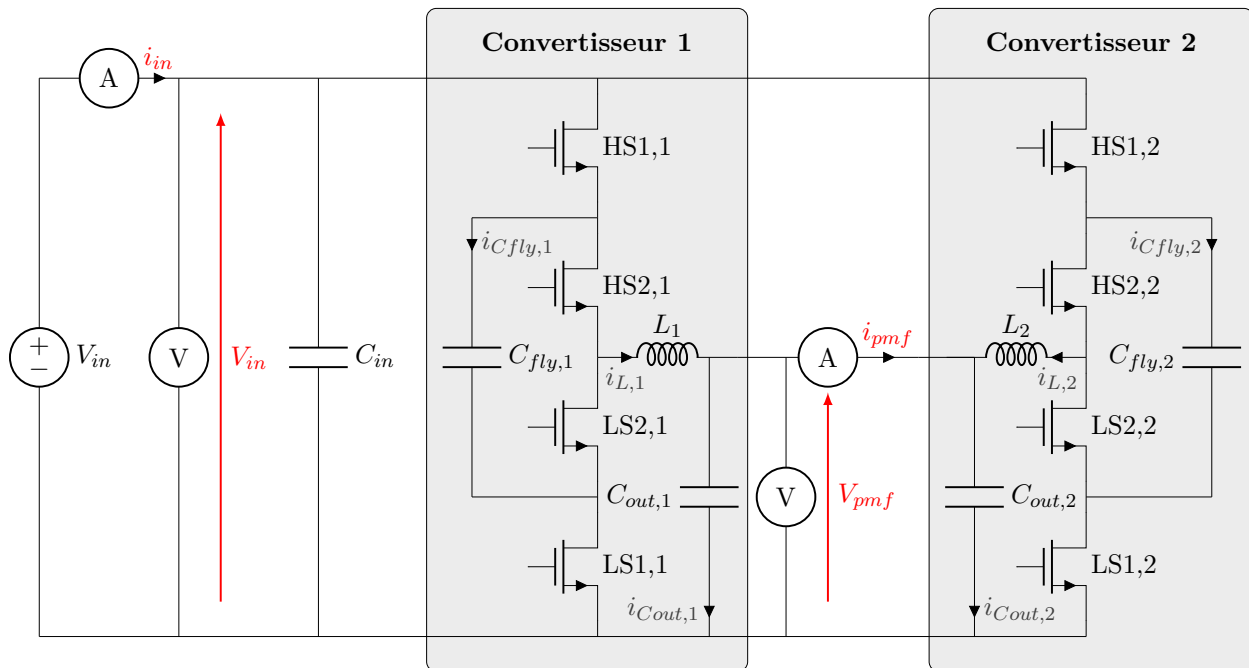


FIGURE 2.31 – Circuit électrique du montage expérimental de la méthode d’Opposition

Le montage expérimental de la méthode d’Opposition pour nos prototypes est schématisé en Figure 2.31. Des photographies du banc d’essai ont été prises en Figure 2.32. Les prototypes entiers ont été placés en Opposition, ce qui inclue leurs filtres de sortie L et C_{out} . Comme la mesure est prise “après” filtrage (entre les filtres des points milieu des deux convertisseurs), nous utiliserons l’indice pmf (point milieu filtré) plutôt que pm . Il n’a pas été nécessaire de rajouter une inductance externe supplémentaire pour assurer le bon fonctionnement du circuit en Opposition. Les pertes des inductances P_L faisant donc partie intégrante des convertisseurs, elles ne sont pas incluses dans les pertes additionnelles P_{add} . La présence des condensateurs de filtrage en sortie $C_{out,i}$ complique la mise en œuvre du mode de fonctionnement E_{off} , pour des raisons expliquées en section 2.4.3.3. De plus, les inductances utilisées, *IHLP8787MZ* de *Vishay* sont conçues pour filtrer un courant DC et produisent des pertes très importantes en AC. Le mode de fonctionnement E_{off} risque donc d’être à la fois limité par l’échauffement des inductances, et de causer des problèmes de précision

au moment d'isoler les pertes E_{off} des transistors face à une puissance P_L proportionnellement très importante. Pour toutes ces raisons, le mode de fonctionnement E_{off} ne sera pas mis en œuvre sur notre banc d'essai.

Les deux mesures de tension V_{in} et V_{pmf} sont référencées à la terre, ce qui permet de les mesurer directement à la pointe-ressort d'oscilloscope sur des points de mesure prévus à cet effet directement sur le PCB. La tension V_{in} est notamment mesurée sur le PCB et non sur les terminaux de la source de tension d'entrée, afin d'exclure de la mesure les possibles chutes de tension liées au câblage entre l'alimentation et le système en Opposition. Les mesures des courants au point milieu i_{pmf} et à l'entrée i_{in} sont prises par des capteurs de courant transducteurs à effet Hall fabriqués par *LEM* (vue oscilloscope en Annexe, Figure A.5). La mesure de i_{pmf} est réalisée par un capteur *LA55-p* dont le courant maximal de mesure est $100A$ et l'erreur relative est $\pm 0.65\%$. Les estimations des pertes du prototypes, calculées en section 1.9 sur la Table 1.5, indiquent des pertes théoriques maximales pour deux convertisseurs de $2 \cdot P_{loss} = 54.62W$, pour $V_{in} = 80V$, $V_{pmf,1} = 28V$ et $i_{pmf} = 71.4A$. Nous pouvons donc estimer le courant $i_{in} = 2 \cdot P_{loss}/V_{in} = 0.68A$, soit un courant très faible. Le capteur *LA25-NP/SP7*, avec son courant maximal admissible de $3.6A$ (le plus petit de la gamme proposée par *LEM*) présente une erreur relative $\pm 0.5\%$. Ces capteurs fonctionnent en boucle fermée, la mesure est transcrite sous la forme d'un courant proportionnel au courant mesuré. Afin de pouvoir lire la mesure sur notre oscilloscope, ces deux mesures sont converties sous forme de tension à l'aide de résistances dont la valeur est connue à $\pm 1\%$. L'oscilloscope *DPO5054* ajoute une imprécision de $\pm 1.5\%$ sur son Gain DC, nous l'utiliserons avec un mode d'acquisition moyenné sur 50 échantillons (pris en compte pour la précision indiquée). Finalement, si la mesure de la puissance revient à faire le produit de la mesure de tension et de courant $P = V \cdot I$, l'erreur relative $\frac{\delta P}{P}$ d'une puissance mesurée est :

$$\frac{\delta P}{P} = \frac{\delta V}{V} + \frac{\delta I}{I} \quad (2.37) \quad \frac{\delta \eta}{1 - \eta} = \frac{\delta P_{pmf}}{P_{pmf}} + \frac{\delta P_{pmf} + \delta P_{loss}}{P_{pmf} + P_{loss}} \quad (2.38)$$

Où δx est l'erreur absolue de la variable x . Il faut noter que l'erreur produite sur la mesure du rendement par la méthode par Opposition n'est pas relative au rendement η , mais à la proportion des pertes sur la puissance en entrée : $1 - \eta$. Il est possible de convertir cette erreur relative vers l'erreur relative au rendement η :

$$\frac{\delta \eta}{\eta} = \left(\frac{\delta \eta}{1 - \eta} \right) \cdot (1 - \eta) \div \eta \quad (2.39)$$

Un rendement proche de 1 résulte en une erreur relative $\frac{\delta \eta}{\eta} \rightarrow 0$. Cela implique que la méthode par Opposition est très intéressante pour les hauts rendements. Mais à mesure que le rendement diminue, l'erreur relative diverge $\frac{\delta \eta}{\eta} \rightarrow \pm \infty$, l'erreur absolue des pertes devient prépondérante devant la valeur absolue de la puissance d'entrée. La Table 2.8 présente l'erreur absolue du rendement $\delta \eta$, pour deux points de fonctionnement extrêmes. Le cas $P_{pmf} = 2kW$ est directement issu des estimations théoriques de rendement réalisées en section 1.9, Table 1.5, tandis que le cas $P_{pmf} = 100W$ est un point de fonctionnement faible tension d'entrée et faible puissance, dont nous supposons le rendement à 97.5% .

TABLE 2.8 – Calcul des erreurs relatives de la méthode d’opposition pour 2 points de fonctionnement extrêmes

P_{pmf}	V_{pmf}	I_{pmf}	P_{in}	V_{in}	I_{in}	η	$\frac{\delta P_{pmf}}{P_{pmf}}$	$\frac{\delta P_{in}}{P_{in}}$	$\frac{\delta \eta}{1 - \eta}$	$\frac{\delta \eta}{\eta}$	$\delta \eta$
2kW	28V	71.4A	55W	80V	0.68A	98.6%	4.65%	4.5%	9.3%	0.13%	$\pm 0.13\%$
100W	28V	3.57A	$\approx 5W$	35V	0.143A	97.5%	4.65%	4.5%	9.3%	0.24%	$\pm 0.23\%$

Les estimations donnent une erreur relative sur la quantité $1 - \eta$ de 9.3%, soit presque 10%, ce qui n’est pas aussi précis que ce qui aurait pu être réalisé avec un outillage mieux pensé. Pour le point de fonctionnement théorique 2kW, le rendement mesuré est compris entre 98.47% et 98.73%. Les deux sources majeures d’erreur proviennent de l’oscilloscope, à $\pm 1.5\% \times 4 = \pm 6\%$ car les 4 sondes sont directement lues sur l’oscilloscope, et des résistances utilisées, $\pm 1\% \times 2 = \pm 2\%$ pour convertir les deux mesures de courant en tensions lisibles. L’utilisation d’un wattmètre de précision adapté aurait permis de considérablement réduire cette imprécision. Il faut néanmoins noter qu’avec le même outillage employé dans une méthode de mesure conventionnelle *Entrée - Sortie*, l’erreur relative du rendement aurait été $\frac{\delta \eta}{\eta} = 9.3\%$, ce qui aurait donné au point de fonctionnement théorique 2kW un rendement illisible de $98.6\% \pm 9.17\%$.

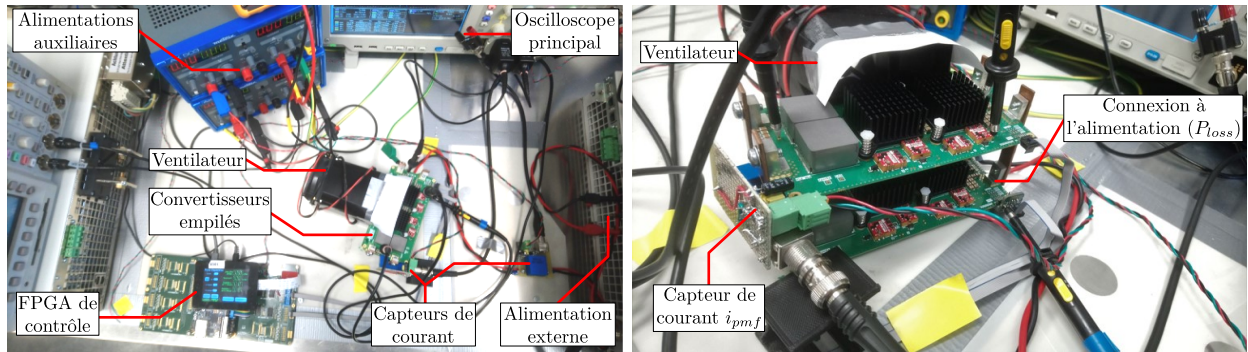


FIGURE 2.32 – Montage expérimental de la méthode d’Opposition

2.4.3.3 Intérêt des capacités C_{out} sur le circuit en Opposition

La méthode par Opposition a été décrite en section 2.4.3.1 pour un circuit dénué de capacité de filtrage C_{out} placée en sortie des convertisseurs. Le principe de la méthode par Opposition n’en est pas fondamentalement modifié, mais certaines formes d’ondes sont transformées, impliquant un impact sur l’équilibrage des tensions des capacités flottantes V_{Cfly} , et rendre très limitée la mise en œuvre du fonctionnement $E_{on} + E_{off}$ du circuit.

Mode de fonctionnement $E_{on} + E_{off}$ de la méthode par Opposition

La présence des capacités de filtrage C_{out} vient modifier les propriétés du *mode de fonctionnement* $E_{on} + E_{off}$. Les capacités de filtrage C_{out} viennent en effet lisser la tension au point milieu V_{pmf} . Pour des convertisseurs *Flying Capacitors* dotés de N -niveaux, les formes d’onde des tensions *point-milieu* des deux convertisseurs représentées en Figure 2.29a restent valables, mais la fréquence

de découpage apparente au point milieu devient $(N - 1) \cdot f_{sw}$. Le courant mesuré i_{pmf} se retrouve néanmoins entre les deux capacités $C_{out,i}$ et n'est plus directement corrélé aux courants circulant dans les transistors. Ainsi :

$$i_{pmf} = i_{L,1} - i_{Cout,1} = -i_{L,2} + i_{Cout,2} \quad (2.40)$$

Nous nous référons au sens des courants indiqués en Figure 2.31, le courant $i_{L,2}$ est notamment positif lorsque le courant s'écoule du nœud V_{pm} au nœud V_{pmf} . Pour la suite du raisonnement, nous ferons les hypothèses suivantes (en s'assurant qu'elles soient raisonnables) :

- Les deux capacités C_{out} sont parfaitement symétriques, les impédances ainsi que les différentes résistances entre $C_{out,i}$ et L_i , et entre $C_{out,i}$ et le capteur de i_{pmf} sont parfaitement égales.
- La capacitance C_{out} peut être considérée comme de valeur infinie à la fréquence de découpage apparente $(N - 1) \cdot f_{sw}$, tel que la tension v_{pmf} soit constante à $V_{pmf} \approx D_1 \cdot V_{in}$.

Dès lors, le courant $i_{L,i}$ traversant l'inductance prend sa forme triangulaire typique décrite en Figure 1.2a (section 1.2.1), et les courants $i_{Cout,i}$ sont toujours égaux.

La Figure 2.33 représente les tensions et courants traversant les inductances de filtrage, à la fréquence apparente $(N - 1) \cdot f_{sw}$. En effet, puisque v_{pmf} est désormais constant à la valeur $V_{pmf} \approx D_1 \cdot V_{in}$ (contrairement à v_{pm}), la tension aux bornes de l'inductance $v_{L,i}$ prend à nouveau la forme d'un signal carré alternant entre $\frac{V_{in}}{N-1} - V_{pmf}$ et $-V_{pmf}$.

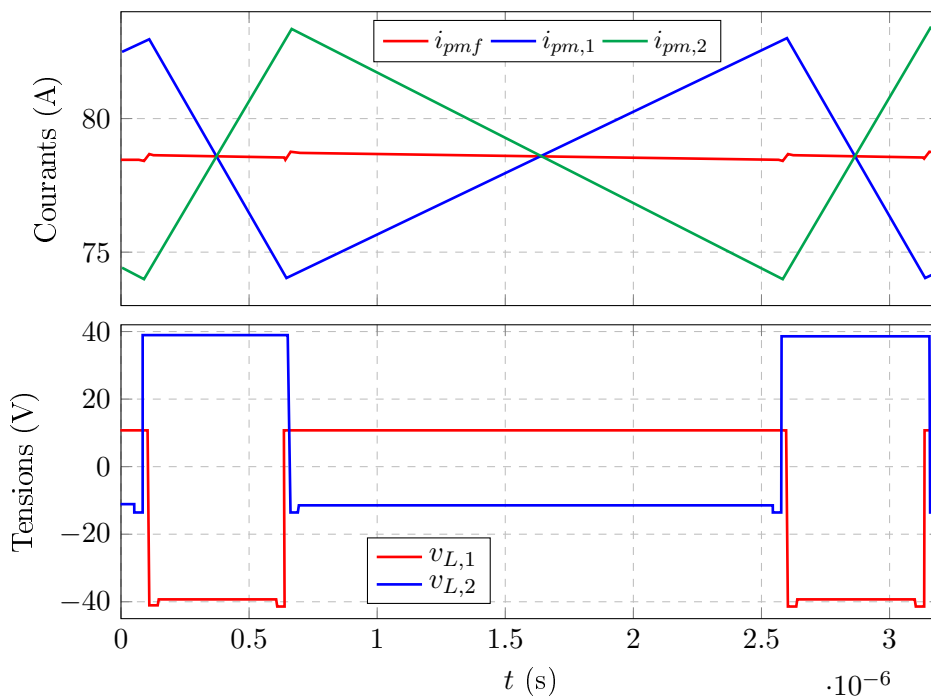


FIGURE 2.33 – Illustration des tensions $v_{L,i}$ et courants $i_{pm,i}$ et i_{pmf} pour le mode de fonctionnement $E_{on} + E_{off}$ en présence de $C_{out,i}$

Le différentiel de courant entre $i_{L,1}$ et $i_{L,2}$ est absorbé par les capacités $C_{out,i}$, ce qui permet aux courants $i_{L,i}$ de conserver leur forme triangulaire standard, avec l'ondulation de courant Δi_L attendue dans un montage normal *alimentation-charge*. Cela implique que le courant mesuré i_{pmf} va afficher la même allure que celle décrite en Figure 2.29a, avec une amplitude Δi_{pmf} donnée par l'équation 2.33. Néanmoins, cette forme d'onde ne correspond pas au courant $i_{L,i}$, et donc au courant traversant effectivement le transistor. Soit le taux d'ondulation du courant Δi_L , obtenu en reformulant l'équation 1.14, pour un convertisseur *FC3L* où $0 \leq D \leq 0.5$:

$$\Delta i_L = \frac{V_{in} \cdot D(\frac{1}{2} - D)}{f_{sw} \cdot I_{pmf} \cdot L} \quad (2.41)$$

Comme généralement $D(\frac{1}{2} - D) \gg \delta_{opp}$, nous en déduisons que $\Delta i_L \gg \Delta i_{pmf}$. La méthode par opposition sans capacités de filtrage est adaptée pour rendre quasi-constant le courant $i_{L,i}$ et maîtriser la valeur du courant i_{ds} lors des commutations des transistors, ce qui permet de caractériser de manière précise les pertes des transistors par rapport à un courant I_{ds} spécifique. L'implémentation de la capacité de filtrage C_{out} permet, au contraire, de correctement caractériser le convertisseur complet dans un fonctionnement réaliste. Le courant $i_{L,i}$ peut être mesuré directement si cela est possible, ou déduit à partir de V_{in} , f_{sw} et D , via l'équation 2.41. Sur le convertisseur 1 pour les transistors *High Side* et sur le convertisseur 2 pour les transistors *Low Side*, le *turn-on* se produit à un courant $I_{ds,min} = I_{pmf}(1 - \frac{\Delta i_L}{2})$, et le *turn-off* se produit à $I_{ds,max} = I_{pmf}(1 + \frac{\Delta i_L}{2})$.

Déduire $i_{L,i}$ via l'équation 2.41 n'est préférable que lorsqu'il est complexe ou non-souhaitable de placer un capteur de courant au plus près de l'inductance de filtrage du convertisseur, avant le filtrage capacitif de C_{out} . Idéalement, il est préférable de directement mesurer $i_{L,i}$ plutôt que de le déduire. Un tel capteur doit être implémenté dès la conception du PCB, pour les convertisseurs de faible puissance réalisés sur PCB. Il s'agit alors généralement de capteurs à effet Hall qui vont ajouter une résistance électrique, agrandir ou alourdir le PCB, et ne pourront pas être retirés du convertisseur une fois la mesure réalisée. Pour les convertisseurs de plus gros calibres réalisés directement sur busbars, les capteurs peuvent être assemblés et désassemblés bien plus simplement.

Mode de fonctionnement E_{off} de la méthode par Opposition

Il a été décidé, principalement à cause des pertes AC trop élevées des inductances *IHLP8787MZ*, que le *mode de fonctionnement E_{off}* de la méthode par Opposition ne serait pas réalisé pour notre banc d'essai. Néanmoins, dans le cadre de cette discussion, nous supposons les inductances de filtrage idéales et adaptées à l'essai afin d'étudier la faisabilité du *mode de fonctionnement E_{off}* pour des convertisseurs *Flying Capacitors* dotés de *N-niveaux*.

De la même manière que pour le mode de fonctionnement $E_{on} + E_{off}$, les capacités de filtrage C_{out} permettent de rendre indépendants les deux courants $i_{L,1}$ et $i_{L,2}$ l'un de l'autre. Contrairement au montage sans C_{out} , le déphasage entre la commande des deux convertisseurs n'a plus aucun effet sur les courants traversant les inductances. L'amplitude Δi_L est toujours régie par l'équation 2.41, et ne peut être modifiée que par une variation du rapport cyclique D ou de la fréquence de commutation f_{sw} , en prenant soin de fixer les rapports cycliques $D_1 = D_2 = D$ afin de conserver le courant moyen $I_{L,i} = 0$. Notamment, si f_{sw} est diminué, Δi_L augmente de manière inversement proportionnelle.

Comme dans le cas du mode de fonctionnement $E_{on} + E_{off}$, il est préférable de directement mesurer $i_{L,i}$ plutôt que de le déduire via l'équation 2.41. Mais, à défaut, il est possible de le mesurer indirectement via un capteur sur i_{pmf} . Pour ce faire, le rapport cyclique est fixé à $D = \frac{1}{2(N-1)}$ et le déphasage entre les deux convertisseurs $\varphi_{opp} = \frac{2\pi}{2(N-1)}$, tel que $i_{L,1} = -i_{L,2}$ à tout instant, ce qui implique que $i_{Cout} = 0A$, et donc que la mesure $i_{pmf} = \pm i_{L,i}$.

Néanmoins, cette méthode repose sur une diminution de la fréquence f_{sw} pour augmenter Δi_L , ce qui revient à diminuer les pertes mesurées $P_{sw,off}$ produites par le *turn-off* desquelles nous souhaitons déduire l'énergie E_{off} . La précision du résultat final s'en voit réduite, d'autant plus que les pertes AC et fer de l'inductance vont, elles, augmenter, et avec elles l'erreur absolue δP_{loss} . Alternativement, il est possible de faire varier Δi_L avec le rapport cyclique D , sans diminuer f_{sw} . Le taux d'ondulation Δi_L est alors limité par la valeur de l'inductance L sélectionnée, et il n'est plus possible de mesurer "directement" i_L par un capteur sur i_{pmf} . Une variation de f_{sw} et de D conjointement peut permettre de maximiser la proportion $P_{sw,off}/P_{loss}$ tout en atteignant des valeurs Δi_L plus élevées.

Équilibrage naturel des tensions des capacités flottantes

Les phénomènes d'équilibrage naturel des capacités flottantes des convertisseurs *FCML* sont complexes à décrire et à anticiper, [60] et [61] en font une description, respectivement, dans le domaine fréquentiel et temporel. Sur notre convertisseur *FC3L*, lorsque la capacité flottante est déséquilibrée tel que $V_{Cfly} = \frac{V_{in}}{2} + \Delta V_{Cfly}$, une composante alternative $v_{pm(1)}$ à la fréquence de découpage f_{sw} , apparaît sur la tension découpée au point milieu v_{pm} . Cette composante génère une harmonique $i_{L(1)}$ de fréquence f_{sw} sur le courant traversant l'inductance i_L . Le déphasage de $i_{L(1)}$ par rapport à $v_{pm(1)}$ dépend de la charge en sortie du convertisseur. Pour un déphasage compris entre $-\frac{\pi}{2}$ et $\frac{\pi}{2}$, l'équilibrage naturel a toujours lieu [62], mais plus la charge est réactive, moins l'effet est prononcé, sa constante de temps étant inversement proportionnelle à la partie réelle de l'impédance de la charge [60]. La Figure 2.34 représente le courant i_{Cfly} traversant la capacité flottante, à partir du courant i_L . Les zones colorées montrent les temps de conduction des interrupteurs HS1 et HS2, et mettent en valeur la différence de charge entre les deux circuits. Pour un $\Delta V_{Cfly} < 0$, le courant moyen de l'harmonique 1 (et de toutes les harmoniques impaires) traversant la capacité flottante, $I_{Cfly(1)}$, est positif et tend à charger, donc à rééquilibrer, la tension V_{Cfly} , tandis que l'harmonique 2 à la fréquence $2 \cdot f_{sw}$ (et les harmoniques paires) produit un courant moyen nul.

Pour deux convertisseurs *FCML* assemblés dans un circuit en opposition, cette mécanique est toujours présente sur les deux convertisseurs grâce à la présence des capacités de filtrage C_{out} qui lissent la tension au point milieu V_{pmf} . Si les hypothèses associées sont respectées, l'équation 2.40 est toujours valide dans ce *mode de fonctionnement* $E_{on} + E_{off}$, le différentiel de courant entre $i_{L,1}$ et $i_{L,2}$ est absorbé par les capacités $C_{out,i}$, ce qui permet aux courants $i_{L,i}$ de conserver leur forme triangulaire habituelle, composés de l'ondulation de courant Δi_L attendue dans un montage sur charge et des harmoniques liées aux déséquilibres de leurs capacités flottantes respectives.

Comme expliqué précédemment, il peut sembler souhaitable de retirer du circuit les capacités de filtrage C_{out} afin de simplifier la mesure au *mode de fonctionnement* E_{off} . Cela peut sembler judicieux pour tester le rendement de la *phase* de conversion individuelle pour un convertisseur entrelacé (cf. section 1.2.2), où la capacité de filtrage C_{out} n'est pas routée de manière fractionnelle sur chaque *phase*, mais en aval du convertisseur complet, après le nœud connectant l'ensemble des courants de chaque bras. Mais ce faisant, les mécanismes d'équilibrage naturel s'en trouvent extrêmement perturbés. Sans C_{out} , on obtient $i_{L,1} = -i_{L,2}$, ce qui signifie que l'inductance effective

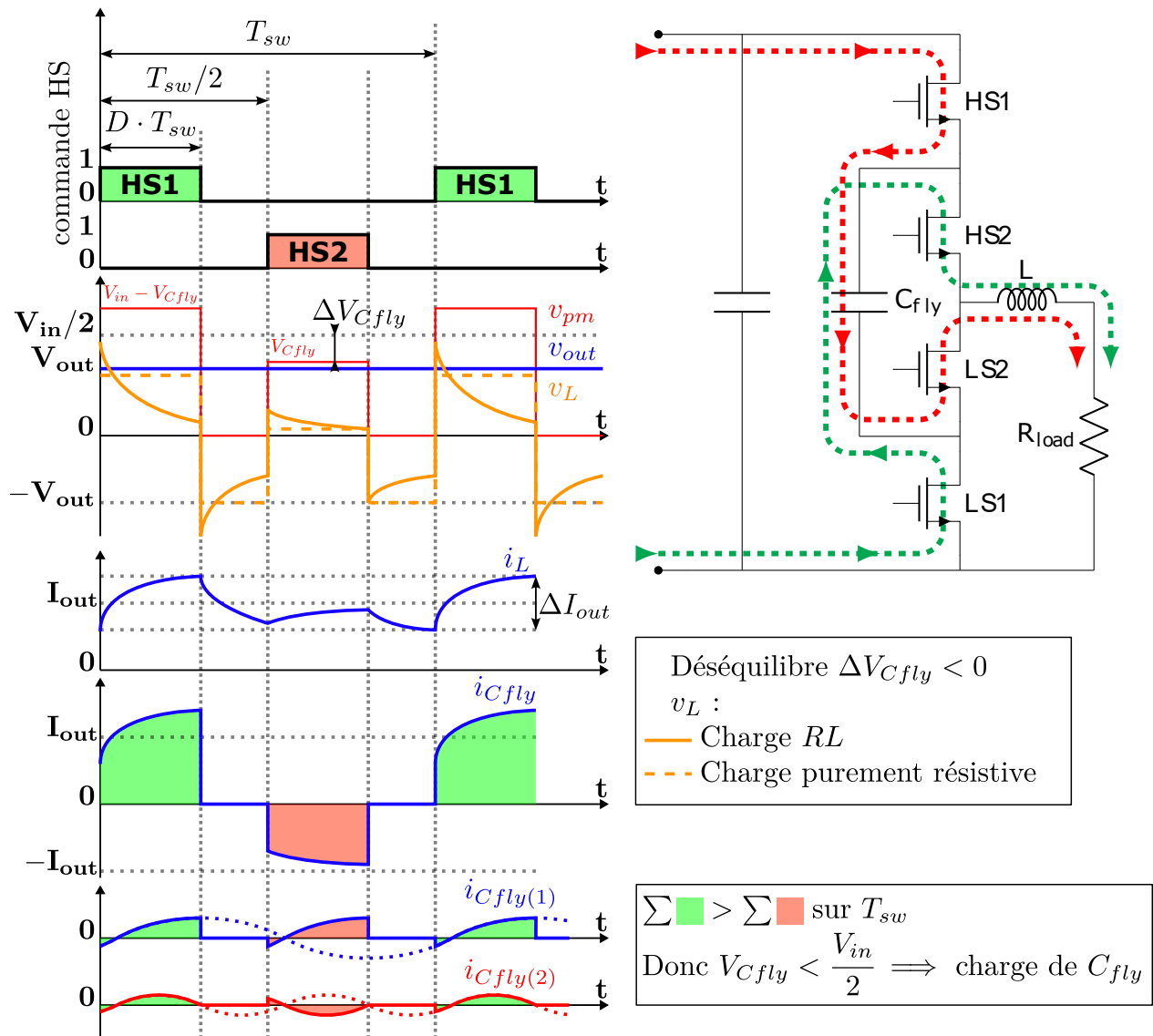


FIGURE 2.34 – Influence de ΔV_{Cfly} sur les harmoniques du courant i_{Cfly} (harmoniques i_L en pointillés) pour un convertisseur $FC3L$ alimentant une charge résistive

en sortie d'un convertisseur est doublée (L_1 et L_2 sont directement en série), et la partie réelle de l'impédance au point milieu est réduite. La dynamique d'équilibrage naturelle s'en voit donc a priori ralentie. Cela signifie aussi et surtout que les perturbations sur le courant $i_{L,1}$ générées par un déséquilibre $\Delta V_{Cfly,1}$ vont être entièrement transmises à $i_{L,2}$ et passer à travers $C_{fly,2}$ avec un déphasage de π .

Une simulation sous le logiciel *PLECS* nous permet d'observer cet effet de partage du déséquilibre entre les deux capacités flottantes opposées, en Figure 2.35. Le circuit simulé consiste en deux convertisseurs *FCML* 3-niveaux en opposition, sans capacité de filtrage C_{out} . La tension d'entrée est $V_{in} = 80V$, $V_{Cfly,1}$ est chargée à sa tension nominale de $\frac{V_{in}}{2}$ sans déséquilibre, tandis que $V_{Cfly,2}$ est complètement déchargé, soit un déséquilibre initial $\Delta V_{Cfly,2}|_{t=0} = -\frac{V_{in}}{2}$. Ce déséquilibre sur le convertisseur 2 va générer des harmoniques tendant à rééquilibrer $V_{Cfly,2}$. Mais ce faisant, elles

vont aussi venir déséquilibrer d'autant la tension $V_{Cfly,1}$. A mesure que le déséquilibre $\Delta V_{Cfly,1}$ augmente dans les négatifs, le convertisseur 1 génère à son tour des harmoniques de rééquilibrage naturel qui vont tendre à annuler les harmoniques du convertisseur 2, déphasées de π . Les tensions des deux capacités flottantes opposées vont converger non pas vers $\frac{V_{in}}{2}$, mais vers le point où les deux harmoniques déphasées s'annulent complètement, soit :

$$\Delta V_{Cfly,1} = \Delta V_{Cfly,2} = \frac{\Delta V_{Cfly,1}|_{t=0} + \Delta V_{Cfly,2}|_{t=0}}{2} \quad (2.42)$$

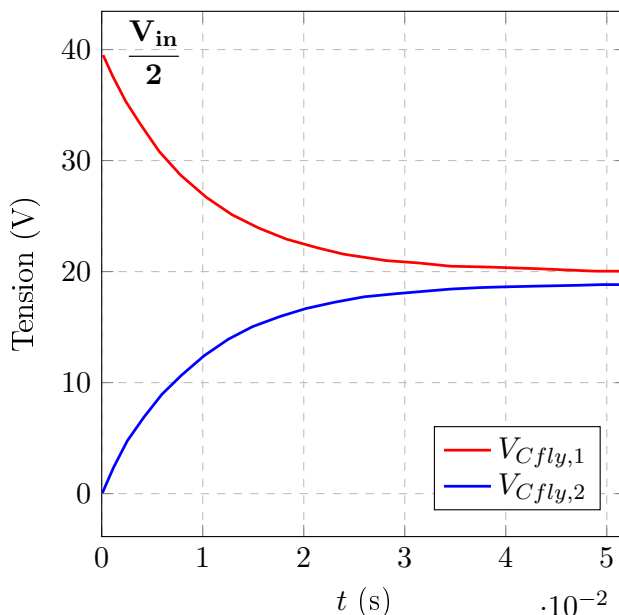


FIGURE 2.35 – Convergence de $V_{Cfly,1}$ et $V_{Cfly,2}$ à $\frac{V_{in}}{4}$ (simulation *PLECS*)

Aucun mécanisme d'équilibrage naturel ne peut donc avoir lieu. Sans circuit externe, à l'allumage des convertisseurs, les capacités flottantes restent bloquées à leur tension initiale de $0V$. Dans la même logique, l'implémentation d'un contrôle actif de la tension $V_{Cfly,i}$ par une loi de commande mènerait aux mêmes difficultés à cause du courant au point milieu entièrement partagé, $i_{L,1} = i_{L,2}$. Le courant supplémentaire alloué à la charge d'une capacité contribuerait mécaniquement à la décharge de l'autre. Il est impératif de disposer de capacités de filtrage pour assurer le bon fonctionnement d'un tel circuit en Opposition.

2.4.3.4 Programmation du FPGA de contrôle

Comme expliqué en section 2.4.3.1, à cause de la très faible résistance électrique du circuit en Opposition, la valeur du courant I_{pmf} est très sensible à la variation du décalage de rapport cyclique δ_{opp} , et donc à la finesse de la commande des convertisseurs. Cette commande est générée à l'aide d'une carte FPGA *DE10-Nano*, doté d'une clock de $50MHz$ qui nous sert à générer une seconde clock à $200MHz$ via une PLL. Les porteuses triangulaires des signaux PWM sont générées par un compteur *cpt* incrémenté de ± 1 à chaque montée de la clock $200MHz$, soit toutes les

$1/200MHz = 5ns$. Nos prototypes fonctionnent à une fréquence de commutation $f_{sw} = 200kHz$, soit une période $T_{sw} = 5\mu s$. La porteuse triangulaire doit donc passer de $cpt = 0$ à sa valeur maximale cpt_{max} en $2.5\mu s$, ce qui implique que $cpt_{max} = \frac{2.5\mu s}{5ns} = 500$.

Une modulante assignée à $mod = 0$ correspond à un rapport cyclique $D = 0$, et $mod = 500$ à $D = 1$. Le pas du rapport cyclique est donc de 0.2%. Nous pouvons estimer l'impact de ce pas sur la variation du courant à l'aide de l'équation 2.31. Nous estimerons $R_{opp} = 2 \cdot R_{dsON} + R_L = 6.49m\Omega$. Pour $V_{in} = 80V$ et $\delta_{opp} = 0.2\%$, le courant $I_{pmf} = 24A$, soit 33% de l'amplitude du courant à tester, rendant impossible de correctement employer le circuit en Opposition. Dans ce cas de figure, un asservissement du courant en boucle fermée est habituellement développé. La programmation d'une régulation a été faite en VHDL, mais elle n'a pas été testée en pratique par manque de temps et risque d'endommager les prototypes. Le développement d'une sous-modulation en boucle ouverte a été suffisant pour réduire la taille du pas effectif de la porteuse. En ajoutant un compteur 4 bits de périodes de la porteuse, l'utilisateur peut imposer que la modulante soit fixée à $mod + 1$ pendant $n < 16$ périodes, résultant en un décalage moyen $\bar{\delta}_{opp}$ compris entre δ_{opp} et $\delta_{opp} + 2 \cdot \frac{5ns}{T_{sw}}$. Cela permet de diviser par 16 le pas effectif de δ_{opp} , le pas du courant I_{pmf} devient approximativement 1.5A. Cette méthode fait néanmoins apparaître une oscillation supplémentaire sur le courant i_{pmf} due à la surmodulation, à la fréquence $f_{sw}/16 = 12.5kHz$. Une boucle fermée en courant aurait elle aussi ce défaut, mais sa fréquence de surmodulation serait bien plus élevée (théoriquement jusqu'à $f_{sw}/2 = 100kHz$), minimisant l'ondulation non souhaitée du courant i_{pmf} .

La méthode par Opposition est donc réalisée en boucle ouverte. Un limiteur de courant a toutefois été programmé, testé et implanté afin d'empêcher une erreur de l'utilisateur ou un emballement imprévu du courant i_{pmf} . En cas de dépassement d'une valeur limite de 80A programmée directement dans le FPGA, le rapport cyclique du convertisseur 2, D_2 est imposé à la valeur de D_1 afin de ramener $I_{pmf} \rightarrow 0A$ sans pour autant couper la commutation afin d'éviter une décharge des capacités flottantes qui pourrait résulter en un dépassement des tensions de claquage des transistors *EPC2022* ($BV_{ds} = 100V$) pour les tensions V_{in} les plus élevées (jusqu'à 115V).

La synchronicité des signaux de commande doit être la meilleure possible. Assurer la bonne logique du programme écrit en VHDL et implanté dans le FPGA est nécessaire mais pas suffisant. Lors de la configuration d'un FPGA, la programmation est reproduite sous forme de circuits logiques à l'intérieur du circuit intégré, c'est à dire des pistes conductrices avec leurs longueurs propres, les imperfections des éléments logiques, etc. Il en résulte l'apparition de délais dans les signaux de commande entre le moment où le FPGA est censé générer une impulsion, et le moment où la piste de cuivre associée au dit signal bascule d'un potentiel électrique à un autre. Cette notion de *jitter* et de délais a déjà été abordée en section 2.3.1, mais contrairement au cas des drivers qui sont des composants aux propriétés fixées et connues sur leurs datasheets, les imperfections temporelles des signaux créés par un FPGA peuvent varier d'une programmation à une autre. Le logiciel de programmation de FPGA *Quartus* fournit des outils pour anticiper et contrôler ces délais dans les signaux (*Time Quest*).

Pendant cette thèse, une négligence de cette synchronicité des signaux générés en sortie de FPGA a mené à la destruction d'un prototype. Des différences de quelques nanosecondes sur la durée et la synchronicité des signaux de commande envoyés aux transistors ont suffi à déséquilibrer la capacité flottante du prototype, ce mécanisme de déséquilibre a été décrit par [63] et vérifié en simulation sous *PLECS*. Après corrections, les signaux sont garantis synchrones et de durée

équivalente à environ $2ns$ près. Malgré cela, lors des expérimentations qui ont suivi, il est encore possible d'attribuer un léger déséquilibre $\Delta V_{Cfly} \approx 3V$ à ces défauts de commande ($V_{in} = 80V$, $P = 2kW$).

2.4.3.5 Mesures des pertes et du rendement & déduction d'un modèle empirique des pertes

Dû à l'erreur de routage du prototype *Traversant*, il n'a pas été possible de mener davantage de tests sur ces derniers que ceux présentés en section 2.4.1. Par conséquent, la campagne de mesure du rendement via la méthode par Opposition n'a été menée que sur le prototype *Adjacent*. Ce sont en tout près de 340 points de mesure qui auront été relevés et analysés. La tension V_{out} est fixée à $28V$. Pour V_{in} allant de $35V$ à $105V$ (soit un rapport cyclique $D = \frac{V_{out}}{V_{in}} \in [0.267; 0.8]$), le rendement du convertisseur est relevé pour $P_{in} \in [100W; 2kW]$. Étant donné que la pile à combustible ne peut pas excéder certaines tensions V_{in} pour les puissances les plus élevées, il n'a pas été nécessaire de mesurer certains points de fonctionnement pour des V_{in} et P_{in} élevés.

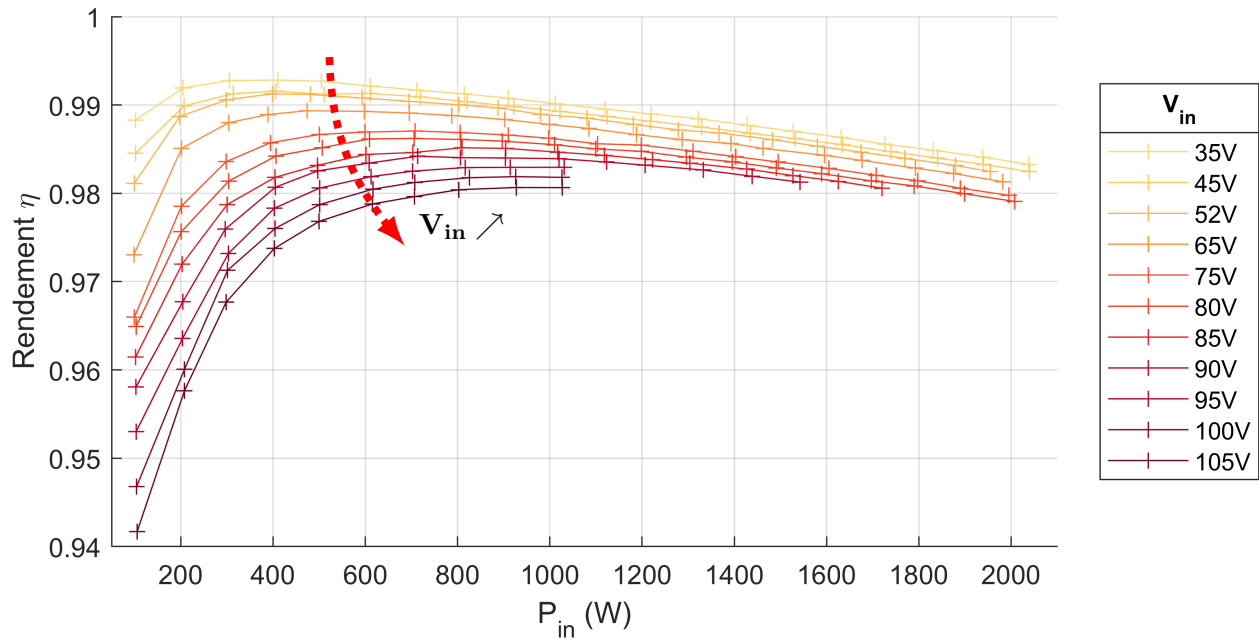


FIGURE 2.36 – Rendement du prototype v2.1V (pertes driver et ventilation non incluses)

La Figure 2.36 présente l'ensemble de ces mesures de rendement. La consommation du ventilateur et des circuits de drivers n'a pas été incluse dans ces rendements, seules les pertes directement liées à la conversion de puissance sont comptabilisées. Le rendement moyen de l'ensemble de ces points de mesure est de 98.3% , pour un écart-type de 0.788% , avec un rendement maximal de 99.27% atteint pour $V_{in} = 35V$, $P_{in} = 300W$. La tension d'entrée V_{in} a un impact majeur sur le rendement aux plus faibles puissances P_{in} , le rendement est en revanche toujours supérieur à l'objectif de 97% fixé par le cahier des charges du projet, lorsque la puissance absorbée est supérieure à $350W$.

Le rendement global du convertisseur est donc plus que satisfaisant. Bien que le circuit d'alimentation des drivers et que le ventilateur n'aient pas fait l'objet d'une étude pour minimiser leurs pertes, il est intéressant d'en observer l'impact sur les performances du convertisseur. Le ventilateur consomme $4.8W$ mais il permet de refroidir les deux convertisseurs en opposition, soit $P_{fan} = 2.4W$.

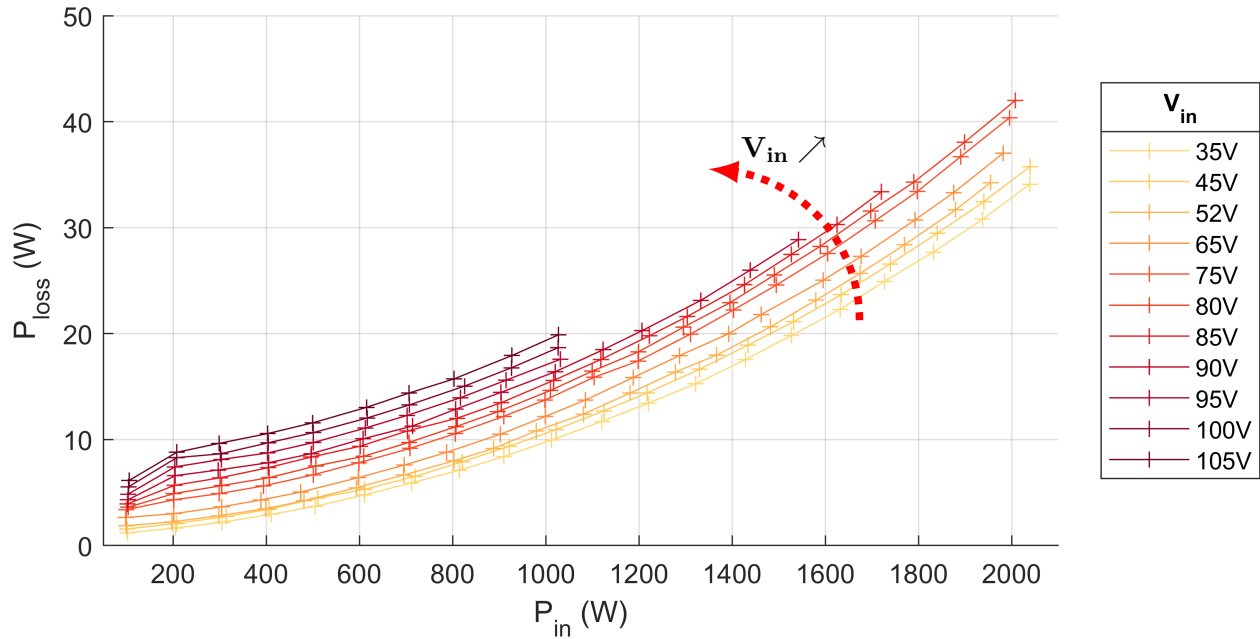


FIGURE 2.37 – Pertes P_{loss} du prototype v2.1V (pertes driver et ventilation non incluses)

L'alimentation des drivers est, elle, de $P_{driver} = 1.55W$. Il faut noter que P_{driver} serait abaissé à $0.21W$ si les alimentations isolées de *Wurth Elektronik* avaient été montées à la place des alimentations *Murata* (section 2.2.4), soit une division par 11 de la consommation. Néanmoins, même avec ces consommations auxiliaires excessives, la moyenne des rendements sur l'ensemble des points de mesure est encore supérieure à 97%, située à 97.53%, pour un rendement maximal de 98.64%.

Comme cela a été présenté en section 1.2.2, il est possible d'optimiser le rendement du convertisseur entrelacé de $20kW$ par la mise en œuvre d'une stratégie de *phase shedding*. En connaissant le rendement d'une phase du convertisseur $20kW$ sur l'ensemble de sa plage de fonctionnement, il est possible d'optimiser le nombre de phases actives à un point de fonctionnement donné pour maximiser le rendement du convertisseur. Pour ce faire, une approche consiste à dresser une formule empirique des pertes $P_{loss,i}$ d'une phase, en fonction de V_{in} et I_{out} (les autres paramètres V_{pmf} , f_{sw} , etc., étant fixés).

Il a, dans un premier temps, été envisagé d'isoler les différentes sources de pertes au sein du modèle, c'est à dire de construire le modèle $P_{loss,i}$ comme la somme de modèles empiriques pour les pertes par conduction P_{cond} , P_{diode} et P_{pcb} , les pertes de l'inductance P_L et les pertes par commutation des transistors P_{sw} . Les pertes par conduction sont très simples à estimer, si les différentes résistances sont bien connues et l'influence de la température négligeable sur ces dernières. En revanche, les pertes P_L et P_{sw} sont plus problématiques. Les formules empiriques développées par *Vishay* et présentées en section 1.7 pour calculer P_L semblent présenter une erreur importante dans leur estimation des pertes fer. En effet, en reprenant les points de mesure de la Figure 2.36, nous pouvons alors tenter d'isoler les énergies de commutation des transistors en fonction de V_{ds} et I_{ds} :

$$E_{sw} = \frac{P_{loss} - P_L - P_{cond} - P_{diode} - P_{pcb}}{f_{sw}} \quad (2.43)$$

La fréquence f_{sw} est fixée à $200kHz$. Pour calculer les pertes P_{diode} et P_{cond} , nous utiliserons les valeurs de V_f et R_{dsON} du modèle *SPICE* produit par *EPC* et étudié en section 1.5 (environ $2.3V$) pour un *deadtime* $t_{dt} = 35ns$. La Figure 2.38 présente les valeurs de E_{sw} obtenues de cette manière en fonction de la tension V_{ds} , chaque série correspondant à un courant I_{ds} différent. On remarque immédiatement que, pour les courants I_{ds} plus faibles, E_{sw} apparaît décroissante passé une certaine tension V_{ds} . Ce n'est évidemment pas cohérent avec le comportement attendu d'un transistor. A faibles courants, ce sont les pertes fer de l'inductance qui prédominent. La décroissance observée sur l'estimation de E_{sw} semble donc indiquer que le modèle empirique de *Vishey* surestime ces pertes par rapport au comportement réel de l'inductance *IHLP-8787MZ-51*. Chercher à décomposer les différentes sources de pertes n'est donc en l'état pas pertinent.

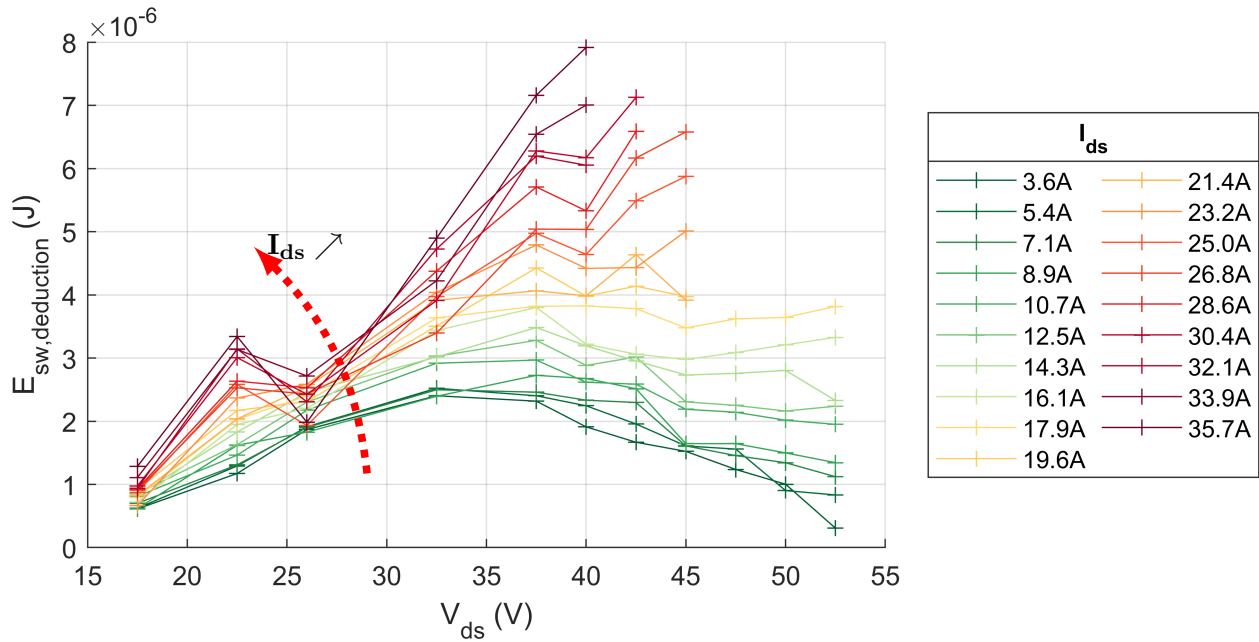


FIGURE 2.38 – Energies de commutation E_{sw} de l'*EPC2022* déduites de P_{loss} en fonction de V_{ds} et I_{ds}

Il semble donc plus judicieux de formuler un modèle empirique “boîte noire” directement à partir de la mesure P_{loss} . Nous procédons à un *fitting* de P_{loss} en fonction de P_{in} et V_{in} via le logiciel *MATLAB*. La fonction *fit*tée obtenue est :

$$P_{loss} = 4.439 \cdot 10^{-5} \cdot P_{in} \cdot V_{in} + 7.327 \cdot 10^{-6} \cdot P_{in}^2 + 6.282 \cdot 10^{-4} \cdot V_{in}^2 \quad (2.44)$$

La Figure 2.39a présente les valeurs de P_{loss} directement issues de la mesure et leurs approximations par la fonction *fit*tée, et l'erreur relative du *fit* pour chaque point est présenté en Figure 2.39b. L'erreur relative moyenne sur l'ensemble des points de mesure est de 4.93%, avec une erreur maximale de 29%. L'erreur relative est relativement faible, inférieure à $\pm 5\%$ sauf pour les valeurs

de P_{in} les plus faibles, où la valeur absolue de P_{loss} est faible. L'écart-type de la fonction *fittée* est de $0.61W$, soit une très bonne corrélation avec les valeurs de P_{loss} dont 80% d'entre elles sont supérieures à $0.61W \times 10 = 6.1W$.

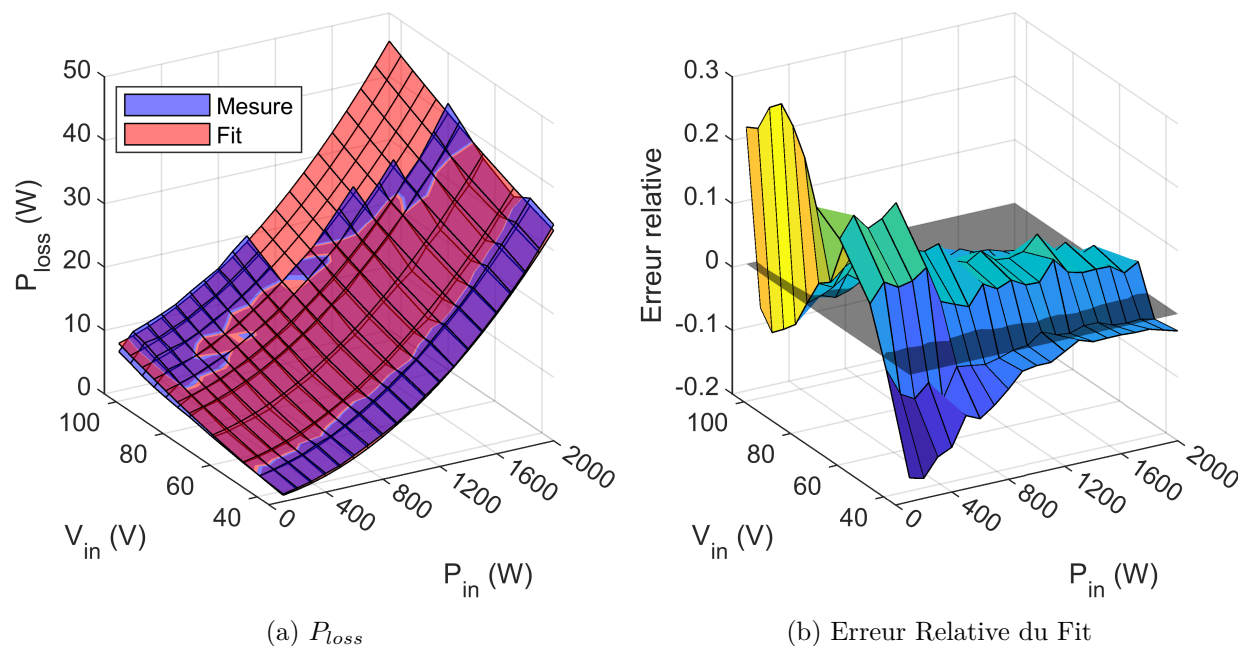


FIGURE 2.39 – Comparaison de la mesure P_{loss} et de la fonction *fittée*

A partir de cette modélisation des pertes pour une phase du convertisseur entrelacé $20kW$, nous en déduisons le rendement pour plusieurs phases actives en parallèle. Le vieillissement de la Pile à Combustible va avoir un impact important sur ce rendement, puisqu'elle impacte la relation tension-courant de la Pile (cf. Figure 2). La Figure 2.40 présente l'évolution du rendement en fonction du nombre de phases actives, pour une Pile à Combustible en début et en fin de vie. Quel que soit l'état de la Pile, le convertisseur entrelacé conserve en (presque) tout point un rendement supérieur à 98% si le nombre de phases actives est contrôlé de manière optimale, avec un rendement moyen glissant de 98.25% à 98.75% à mesure que la Pile à Combustible vieillit. Le rendement du convertisseur entrelacé est certes fluctuant à basses puissances, mais tend à croître jusqu'à son maximum atteint pour $n_{phase,max} = 10$ où toutes les phases sont actives. Cela est dû à la relation tension-courant de la Pile à Combustible, qui tend à diminuer la tension V_{in} lorsque le courant augmente, ce qui réduit les pertes fer des inductances et les pertes par commutation des transistors par rapport à un scénario où V_{in} serait constant. Le vieillissement de la Pile favorise lui aussi le rendement car V_{in} diminue globalement.

2.5 Conclusion du chapitre

Le développement de plusieurs prototypes a été détaillé, du choix des différents composants auxiliaires à la chaîne de conversion jusqu'au routage des PCB sous *Altium*. Une attention particulière a été portée à la qualité du routage, à la fois pour permettre une bonne parallélisation des transistors GaN et pour minimiser les perturbations électriques sur les cellules de commutation. Ces dernières ont fait l'objet d'une étude des différentes impédances parasites, d'abord via des logiciels

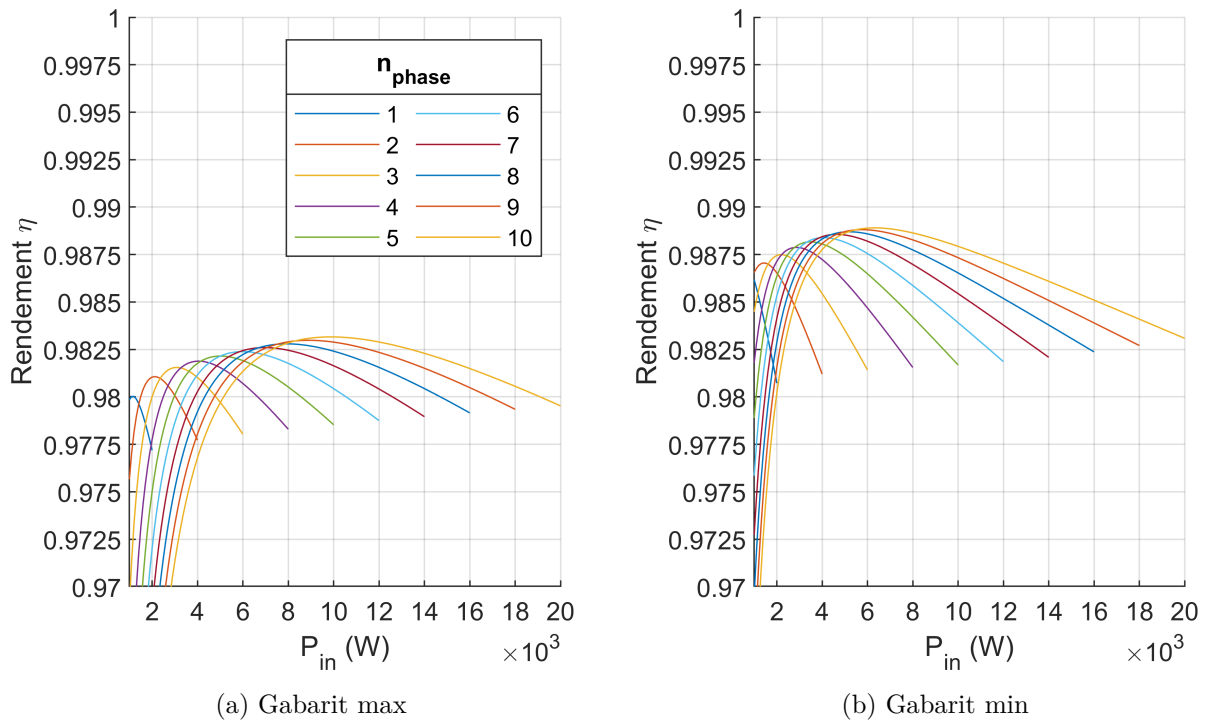


FIGURE 2.40 – Rendement du convertisseur entrelacé, en fonction du vieillissement de la Pile à Combustible

de simulation (*Ansys Q3D*, *LTSPICE*), puis ensuite par une vérification expérimentale. Sur les deux prototypes développés, seulement un a pu être testé à pleine puissance. Son rendement excède les 98% sur une grande partie de sa plage d'utilisation, et il a pu être montré qu'un convertisseur entrelacé $20kW$ composé de 10 phases basé sur ce prototype pourrait maintenir un rendement stable autour de 98.25% pour une Pile à Combustible en début de vie, rendement qui tend à s'améliorer jusqu'à 98.75% en fin de vie. Sa masse totale serait de $2.576kg$ pour une puissance massique de $7.77kW/kg$, largement supérieure à l'exigence de $4kW/kg$ issue du cahier des charges.

Les travaux menés lors de ce chapitre ont été très formateurs, et soulignent de nombreux aspects sur l'entièreté du processus de conception d'un convertisseur basse tension forts courants compact. Néanmoins, malgré deux chapitres consacrés à la question, il reste encore difficile d'estimer si le convertisseur développé ici correspond à la configuration optimale pour obtenir à la fois le meilleur rendement et la masse et le volume les plus faibles possibles. En parallèle des travaux présentés lors de ce chapitre, l'exploration des différentes solutions de conversion abordé au chapitre 1 a été prolongée à travers un nouvel axe : le développement d'un algorithme d'optimisation permettant d'évaluer de très nombreuses solutions automatiquement.

Chapitre 3

Approche de conception par optimisation algorithmique Geometric Programming

3.1 Problématiques de la phase de design

Dans le Chapitre 1 a été présentée une comparaison de différents designs de convertisseurs réalisée “manuellement”, visant à sélectionner pour prototypage la configuration produisant les pertes et la masse totale les plus faibles. Une telle démarche présente rapidement ses limites. Le nombre k d'options possibles pour chacun des n choix de conception vient très rapidement démultiplier le nombre de configurations totales à évaluer, si l'on veut être exhaustif, suivant une complexité exponentielle $O(k^n)$. Les variables “continues”, pouvant prendre une valeur comprise dans un ensemble indénombrable, nécessitent d'être discrétisées. Typiquement, la fréquence de commutation du convertisseur, f_{sw} , aurait pu prendre dans notre problème une valeur comprise entre $10kHz$ et $1MHz$, mais il a été décidé de n'évaluer que trois cas, $f_{sw} = \{200kHz; 400kHz; 600kHz\}$, afin de conserver un nombre de solutions à évaluer humainement réalisable. Ce type de décision repose entièrement sur quelques estimations qualitatives et quantitatives préalables, et sur l'expérience du concepteur, qui doit intuitivement quelles solutions sont les plus probablement intéressantes, et écarter les solutions probablement inintéressantes. Le nombre de solutions évaluées est finalement restreint, ce qui permet de limiter le temps passé à cette tâche, mais le risque de manquer une meilleure solution est important.

Il est d'usage que les concepteurs s'appuient sur des outils informatiques pour automatiser la recherche du design le plus adapté à leurs besoins. La discipline mathématique et informatique dédiée à ce type de programmes est l'*Optimisation*. La recherche d'un design de convertisseur électrique à la fois efficace et léger est un cas typique de *Problème d'Optimisation*. Dans le but de créer un algorithme capable de résoudre un tel problème d'optimisation, nous présenterons dans ce Chapitre les différentes grandes familles d'algorithmes d'optimisation, leurs atouts et inconvénients. Un algorithme de type *Geometric Programming*, dérivé de l'optimisation *Convexe*, sera retenu pour sa capacité à converger très rapidement vers la meilleure solution possible pour un problème donné. Afin d'implémenter dans le problème des variables discrètes, pertinentes pour représenter avec le plus de fidélité possible le problème physique réel, l'intégration d'un solveur *Geometric Programming* dans un algorithme *Branch & Bound* sera développée. Ce Chapitre se concentrera sur les principes généraux associés à ces différents algorithmes, tandis que le Chapitre 4 présentera l'implémentation de notre problème d'optimisation d'un convertisseur électrique dans notre nouvel algorithme.

3.2 Formuler et résoudre un Problème d'Optimisation de design d'un convertisseur

Un problème d'optimisation est avant tout défini par sa *fonction objectif*, $f_0(x) : \mathbf{dom} f_0 \rightarrow \mathbb{R}$, où $\mathbf{dom} f_0$ est le domaine de la fonction f_0 , et $x \in \mathbf{dom} f_0$ est une variable d'optimisation. La fonction f_0 prend l'ensemble des variables d'optimisation que souhaite évaluer l'utilisateur, et retourne un nombre réel $f_0(x)$ représentatif de la performance de la solution x . Le problème d'optimisation revient à trouver la valeur de x pour laquelle la valeur de $f_0(x)$ est minimale (ou maximale, pour les problèmes de maximisation, non abordés ici). À cette fonction objectif f_0 sont associées des fonctions contraintes $f_{constr,i}(x)$ qui doivent respecter des égalités ou, plus couramment, des inégalités venant contraindre les valeurs acceptables de x . L'association des domaines d'application de la fonction objectif f_0 et des fonctions contrainte $f_{constr,i}$ forme le *Domaine du problème d'optimisation*, \mathcal{D} , tel que :

$$\mathcal{D} = \mathbf{dom} f_0 \cap \bigcap_{i=1}^{i_{max}} \mathbf{dom} f_{constr,i} \quad (3.1)$$

On notera l'ensemble image de f_0 correspondant : $f_0(\mathcal{D}) \subset \mathbb{R}$. Certaines valeurs de x ne permettent pas de respecter les contraintes du problème et ne peuvent donc pas être une solution du problème. Si $x \in \mathcal{D}$ et x est solution, alors x appartient à l'*Espace des Solutions* \mathcal{S} , soit $\mathcal{S} \subset \mathcal{D}$. S'il n'existe aucune valeur de x capable de respecter les contraintes, soit $\mathcal{S} = \emptyset$, le problème est infaisable. Autrement, il existe nécessairement une ou plusieurs valeurs de $x \in \mathcal{S}$, que l'on nommera *solution optimale* ou *optimum*, et noté x^* , pour lesquelles la fonction objectif f_0 produit sa valeur minimale dans $f_0(\mathcal{S})$:

$$\mathcal{S} \neq \emptyset \implies \forall x \in \mathcal{S}, \quad \exists x^* \in \mathcal{S} | f_0(x^*) \leq f_0(x) \quad (3.2)$$

On appelle $f_0(x^*)$ la *valeur optimale* du problème. Nous ferons la distinction entre un *optimum global*, c'est à dire une solution x^* pour laquelle il n'existe aucun $f_0(x)$ qui soit (strictement) inférieur à $f_0(x^*)$ (tel que décrit par l'équation 3.2), et un *optimum local* x^\dagger , pour lequel $f_0(x^\dagger)$ n'est une valeur minimale de f_0 que pour les valeurs x adjacentes à x^\dagger . S'il existe un optimum local x^\dagger , alors :

$$\exists R > 0, \quad R \in \mathbb{R}, \quad (x, x^\dagger) \in \mathcal{S}^2, \quad \|x^\dagger - x\|_2 \leq R, \quad f_0(x^\dagger) \leq f_0(x) \quad (3.3)$$

Nécessairement, $f_0(x^\dagger) \geq f_0(x^*)$. L'existence d'optimums locaux est liée à la forme du problème, c'est à dire à la nature de la fonction objectif f_0 .

Prenons pour exemple de problème d'optimisation le design d'un convertisseur électrique dont on voudrait minimiser les pertes :

$$\left\{ \begin{array}{l} \text{minimiser } f_0(x) = P_{sw}(x) + P_{cond}(x) + P_L(x) \\ \text{sujet à :} \\ f_{constr,1}(x) = P_{MOSFET}(x) \cdot R_{\theta,j,amb}(x) \leq \Delta T_{j,max} \\ f_{constr,2}(x) = \Delta i_L(x) \leq 20\% \\ \dots \end{array} \right. \quad (3.4)$$

La solution x doit à la fois minimiser les pertes, sans pour autant que l'élévation de température du transistor causée par ses pertes n'excède l'élévation maximale définie par $\Delta T_{j,max}$, et en gardant l'ondulation du courant de sortie Δi_L inférieure à 20%. Une solution x est la combinaison entre le choix d'un transistor, d'une inductance de filtrage, d'une topologie de convertisseur, d'une fréquence de commutation, etc. Lorsqu'il formule le problème, le concepteur ne connaît généralement pas quelles configurations respectent l'ensemble des contraintes, c'est à dire qu'il ne connaît pas l'espace \mathcal{S} , seulement l'espace de design \mathcal{X} de toutes les combinaisons entre les n variables du problème qu'il souhaite tester. Le processus de résolution du problème doit à la fois être capable d'écartier les valeurs de x infaisables ($x \notin \mathcal{S}$) et minimiser $f_0(x)$. Le choix parmi les différents algorithmes

existants capables de résoudre des problèmes d’optimisation se fait en fonction de la nature de la fonction objectif, du temps de calcul et de la garantie que l’algorithme trouvera, s’il existe, l’optimum global.

L’algorithme le plus simple à mettre en œuvre est une méthode de recherche exhaustive, dite *Brute Force*, où l’ensemble des solutions possibles sont testées et comparées jusqu’à trouver l’optimum global [64–69]. C’est la méthode “manuelle” utilisée en section 3.2, tirant cette fois partie de la vitesse de calcul d’un ordinateur pour augmenter la vitesse et le nombre de solutions évaluées. Si l’ensemble des variables d’entrée du problème ne peuvent prendre que des valeurs discrètes, la méthode *Brute Force* garantit de trouver l’optimum global. Elle suit en revanche une complexité exponentielle $O(k^n)$ (n le nombre de variables, k le nombre de valeurs discrètes possibles pour chaque variable), ce qui la rend lente voire incalculable dans un temps réaliste même pour un ordinateur performant, dès que le nombre de variables du problème devient conséquent. Si le problème comporte des variables continues, cette méthode nécessite de les discrétiser. Non seulement cette discrétisation va contribuer à l’inflation de la complexité (proportionnellement, pour chaque variable discrétisée, au pas de discrétisation), elle réduit l’espace de design \mathcal{X} des combinaisons que le concepteur souhaite tester à un sous-ensemble $\mathring{\mathcal{X}}$ dont sont exclues toutes les configurations pour lesquelles les variables concernées ne prennent pas l’une des valeurs discrètes retenues. L’ensemble des solutions \mathcal{S} n’est donc pas exploré, mais seulement l’intersection $\mathring{\mathcal{X}} \cap \mathcal{S}$. L’optimum obtenu sera global, mais sur cette intersection $\mathring{\mathcal{X}} \cap \mathcal{S}$ et non sur \mathcal{S} . En fonction de la finesse du pas de discrétisation et de la forme de f_0 , il peut même être très éloigné de l’optimum global sur \mathcal{S} . Autrement dit, la méthode *Brute Force* est excessivement longue et ne garantit pas de trouver l’optimum global sur \mathcal{S} .

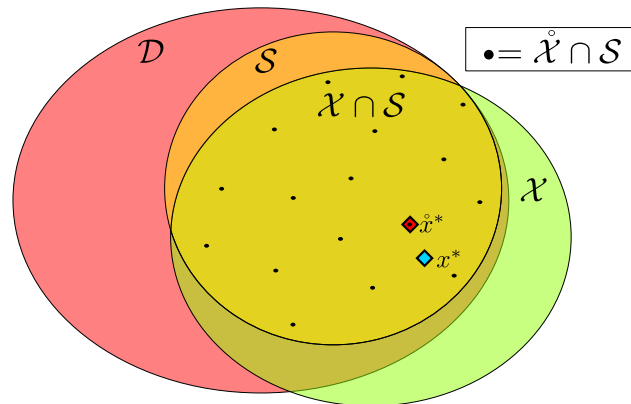


FIGURE 3.1 – Représentation des différents espaces d’un problème d’optimisation

Pour limiter le temps de calcul et la difficulté à traiter les variables continues, nous pourrions préférer à la recherche exhaustive l’emploi d’*algorithmes d’optimisation Heuristiques*, c’est à dire basées sur des approximations simplificatrices, ou des choix aléatoires (on parle de méthodes *Stochastiques* dans ce cas), pour rechercher rapidement des solutions à un problème d’optimisation. La méthode stochastique la plus simple dans son principe est la méthode de *Monte Carlo* [14, 70, 71]. Des points $x \in \mathcal{X}$ sont générés de manière purement aléatoires, leur faisabilité est ensuite testée pour chaque contrainte, pour déterminer s’il s’agit de solutions, et leurs performances $f_0(x)$ sont comparées pour pouvoir sélectionner les meilleures solutions. Un très grand nombre de configurations doivent être testées pour avoir une image statistiquement représentative de l’espace des solutions \mathcal{S} , soit un nombre très important de calculs à réaliser. Les *algorithmes évolutionnistes* forment une approche heuristique plus économe en temps de calcul [72–76], un état de l’art des usages de ce type

d’algorithmes pour les convertisseurs électriques a été réalisé par [77] en 2015. Des configurations randomisées de x sont là encore générées au lancement du programme. Puis, à partir de ces configurations initiales, sont créées des configurations de x dérivées où les variables subissent de petites modifications aléatoires, que l’on nommera “mutations”. Les “mutations” produisant des solutions plus performantes sont exploitées (de manières différentes en fonction de l’algorithme choisi) pour chercher à produire des solutions “filles” plus performantes. Les algorithmes évolutionnistes ont pour avantage d’être capables de traiter des problèmes dotés de variables continues et discrètes. Contrairement aux algorithmes d’optimisation présentés ci-après, les algorithmes évolutionnistes présentent une grande flexibilité dans le type de fonctions objectif et contraintes qu’ils peuvent traiter. Néanmoins, ils ne garantissent pas de trouver un optimum, ni global, ni même local. Un algorithme évolutionniste ne peut déterminer avec certitude si sa meilleure solution tend vers un minimum local et non le minimum global, et ne peut pas non plus déterminer avec certitude si cette meilleure solution est bien égale au dit optimum local tel que défini par l’équation 3.3, ou si elle pourrait encore être améliorée. Le choix du paramétrage de l’algorithme par l’utilisateur à son lancement influence son fonctionnement, il est donc courant de relancer plusieurs fois ce type d’algorithmes pour comparer les résultats obtenus et vérifier leur cohérence, avant de sélectionner une solution performante. C’est une approche flexible, mais non-déterministe et sans garantie de convergence vers un optimum.

Certains problèmes d’optimisation sont dotés de fonctions objectif et contraintes d’une nature particulière, compatible avec des algorithmes d’optimisation particulièrement rapides et capables de garantir une convergence sur l’optimum global. C’est la famille des *Problèmes Convexes* [78–82], présentés extensivement par [83] (résumé par [84]). Un problème est convexe si les fonctions f_0 et $f_{constr,i}$ qui le composent sont toutes convexes (ou linéaires, dans le cas des contraintes d’égalité), c’est à dire si elles respectent la propriété suivante :

$$\forall (x_1, x_2) \in (\text{dom } f)^2, \quad x_1 \neq x_2, \quad \forall \theta \in [0; 1], \quad f(\theta x_1 + (1 - \theta)x_2) \leq \theta f(x_1) + (1 - \theta)f(x_2) \quad (3.5)$$

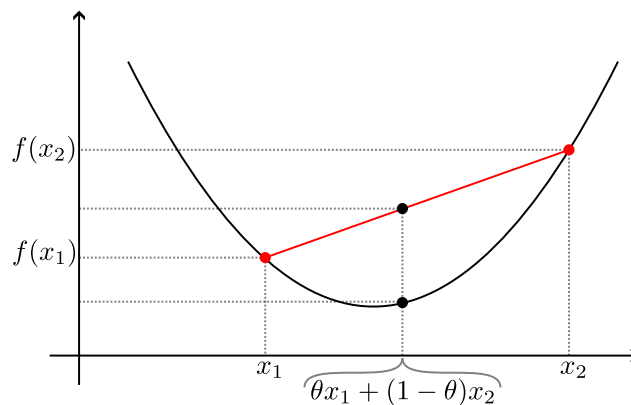


FIGURE 3.2 – Représentation d’une fonction convexe $f : \mathbb{R} \rightarrow \mathbb{R}$ et du segment entre les deux points $f(x_1)$ et $f(x_2)$

Nous pouvons représenter cette propriété graphiquement pour une fonction à une dimension $f : \mathbb{R} \rightarrow \mathbb{R}$ par la Figure 3.2. En traçant un segment entre deux points quelconques de la courbe f , il ne coupe la courbe qu’en ces deux points et est entièrement inclus dans son épigraphe, c’est

à dire “au-dessus” de la courbe ou se confondant avec la courbe. Les *Problèmes de Moindres Carrés* et les *Problèmes Linéaires* sont des cas particuliers des *Problèmes Convexes*. L’intérêt majeur des problèmes convexes est qu’il est possible de rigoureusement prouver qu’un point $x \in \mathcal{D}$ est l’*optimum global* du problème d’optimisation, si ce point x vérifie les *Conditions de Karush-Kuhn-Tucker* (KKT). Ces conditions ne seront pas détaillées ici, mais elles sont à la base de la résolution des problèmes d’optimisation convexes. La *Méthode des Points Intérieurs* est typiquement utilisée pour résoudre des problèmes convexes. Il s’agit d’une famille d’algorithmes basée sur une succession de résolutions des conditions KKT “simplifiées”, jusqu’à converger vers une solution x^* dite *optimale*, c’est à dire respectant les conditions KKT complètes. Etant donné que les conditions KKT et la méthode des points intérieurs reposent sur l’étude des dérivées des fonctions objectif et contraintes, ces fonctions doivent aussi être différentiables. En plus de garantir, s’il existe, de trouver l’optimum du problème, ces algorithmes sont généralement très rapides y compris pour des problèmes dotés d’un grand nombre de paramètres, avec une complexité polynomiale. Mais ils requièrent que le problème ait une forme très spécifique, convexe différentiable, et ils ne peuvent pas traiter les variables discrètes sans être combinés à d’autres algorithmes. Ce prérequis oblige les designers à chercher des modélisations approximatives pour les phénomènes qu’ils ne peuvent décrire rigoureusement sous forme de fonctions convexes différentiables. Il existe aussi d’autres méthodes de résolution des problèmes convexes présentant les mêmes difficultés. Un état de l’art plus détaillé sur l’usage de chacune de ces méthodes dans le domaine des véhicules électriques a été réalisé par [78].

D’une manière plus générale, il est possible d’appliquer des méthodes similaires à des problèmes différentiables mais non-convexes. On parle d’*Algorithmes de Descente de Gradient* [85–87], car ils calculent le gradient entre deux solutions x afin de trouver la “pente” de la fonction et de la descendre jusqu’à converger vers une solution respectant les conditions KKT. Seulement, comme la fonction n’est pas convexe, ces conditions KKT garantissent que la solution trouvée est un optimum local, mais ne peuvent pas garantir qu’il s’agit aussi de l’optimum global. Il est possible de relancer plusieurs fois l’algorithme en partant de solutions initiales différentes, ou de solutions “adjacentes” à l’optimum local, et de vérifier si elles convergent toutes sur la même solution finale ou bien, dans le cas contraire, de retenir la meilleure solution proposée.

Les différentes méthodes de résolution de problèmes d’optimisation communément employées dans la conception de convertisseurs électriques sont résumées en Table 3.1. Chacune de ces méthodes présente des avantages et inconvénients, aucune d’entre elles ne semble capable de correctement traiter des problèmes hybrides, c’est à dire dotés à la fois des variables continues et des variables discrètes, dans un temps de calcul raisonnables et avec une garantie de convergence vers un optimum global. Les problèmes convexes présentent deux de ces trois caractéristiques attractives. Il est possible de développer un programme hybride pour gérer les variables discrètes séparément des variables continues. Mais ils souffrent de la trop grande difficulté à formuler certains phénomènes physiques sous forme de fonctions convexes différentiables. Typiquement, la fonction $x \mapsto \sqrt{x}$ n’est pas convexe, comme toutes les fonctions du type $x \mapsto x^\alpha$ pour $0 < \alpha < 1$. Il existe néanmoins un type de problème qui présente les avantages des problèmes convexes mais permettent d’employer certaines fonctions non-convexes qui se prêtent bien mieux à la modélisation mathématique des convertisseurs électriques : la *Programmation Géométrique* (ou GP pour *Geometric Programming*).

TABLE 3.1 – Comparaison des différentes méthodes de résolution de problèmes d’optimisation

Problème	Méthodes	Avantages	Inconvénients
Discret	Brute Force	<ul style="list-style-type: none"> <input type="checkbox"/> Simple à mettre en œuvre <input type="checkbox"/> Exhaustif et garantie d’optimum global pour des problèmes discrets 	<ul style="list-style-type: none"> <input type="checkbox"/> Complexité exponentielle, temps de calcul très excessifs <input type="checkbox"/> Discrétisation des variables continues → Pas de garantie d’optimum global pour les problèmes continus ou hybrides <input type="checkbox"/> Utilisable seulement pour des problèmes avec un nombre de variables très limité
Continus & Discrets	Heuristique & Stochastique	<ul style="list-style-type: none"> <input type="checkbox"/> Grande flexibilité dans le type de problèmes traitables <input type="checkbox"/> Meilleures performances que le Brute Force 	<ul style="list-style-type: none"> <input type="checkbox"/> Ne garantie pas que la meilleure solution trouvée soit un optimum du problème <input type="checkbox"/> Son paramétrage peut changer le résultat, résultat confirmé par plusieurs itérations
Convexe Différentiable	Points Intérieurs	<ul style="list-style-type: none"> <input type="checkbox"/> Garantie d’optimum global sur des problèmes continus <input type="checkbox"/> Complexité polynomiale, très bonnes performances même pour un nombre de variables élevé 	<ul style="list-style-type: none"> <input type="checkbox"/> Requièrè un problème convexe différentiable <input type="checkbox"/> La modélisation sous forme convexe de phénomènes non-convexes est trop souvent réductrice ou impossible <input type="checkbox"/> Ne traite pas les problèmes discrets et hybrides <input type="checkbox"/> Utilisable principalement pour des études de tendance
Non-Convexe Différentiable	Descente de Gradient	<ul style="list-style-type: none"> <input type="checkbox"/> Garantie d’optimum local sur des problèmes continus 	<ul style="list-style-type: none"> <input type="checkbox"/> Requièrè un problème différentiable <input type="checkbox"/> Ne garantie que l’optimalité locale, pas l’optimalité globale <input type="checkbox"/> Ne traite pas les problèmes discrets et hybrides <input type="checkbox"/> Utilisable principalement pour des études de tendance

3.3 Présentation du Geometric Programming

3.3.1 Principes généraux

Les explications qui suivent se basent sur celles délivrées dans l'ouvrage [88]. Les *Problèmes GP* sont composés de fonctions objectif et contraintes qui ne sont pas forcément convexes en tant que tel, mais peuvent être rendues convexes après une transformation logarithmique. Les fonctions du problème sont de la forme :

$$f(x) = c_0 + \sum_{j=1}^m c_j \prod_{i=1}^n x_i^{\alpha_{i,j}}, \quad x = \{x_1; \dots; x_n\} \succeq 0, \quad c_0, c_j \in \mathbb{R}_+, \quad \alpha_{i,j} \in \mathbb{R} \quad (3.6)$$

exemple : $f(x) = 4.53 \cdot x_1^{0.52} \cdot x_2^{1.63} \cdot x_3^4 + 8.7 \cdot 10^{-6} \cdot x_1^{-1} \cdot x_2^3 \cdot x_3^{-0.015}$

Il s'agit de polynômes dont le coefficient c_j de chaque monôme et chaque terme de x sont positifs ou nuls, mais dont les puissances $\alpha_{i,j}$ peuvent prendre une valeur réelle quelconque. Par convention, on nomme ces monômes particuliers des *GP-monômes*, et les polynômes qu'ils composent des *posynômes* (pour "positive polynomials"). Si l'on pose $y = \{y_1; \dots; y_n\}$ tel que :

$$x_i = e^{y_i} \quad \Leftrightarrow \quad y_i = \ln(x_i) \quad (3.7)$$

Alors en passant au logarithme une fonction posynôme, on obtient :

$$\ln(f(x)) = \ln(f(e^y)) = \ln \left(c_0 + \sum_{j=1}^m \exp \left(\sum_{i=1}^n \alpha_{i,j} y_i + \ln(c_j) \right) \right) \quad (3.8)$$

Soit une fonction *Log-Sum-Exp* (LSE), qui est elle-même une fonction convexe. Ces fonctions faites de *GP-monômes* et de *posynômes* sont donc dites "*log-log-convexes*". Le problème GP original peut être reformulé pour obtenir un problème convexe classique, solvable par une méthode des Points Intérieurs, de la manière suivante :

$$\left\{ \begin{array}{l} \text{minimiser } f_0(x) \\ \text{sujet à :} \\ f_{constr,1}(x) \leq 1 \\ f_{constr,2}(x) = 1 \\ \dots \end{array} \right. \quad \Leftrightarrow \quad \left\{ \begin{array}{l} \text{minimiser } \ln(f_0(e^y)) \\ \text{sujet à :} \\ \ln(f_{constr,1}(e^y)) \leq 0 \\ \ln(f_{constr,2}(e^y)) = 0 \\ \dots \end{array} \right. \quad (3.9)$$

Le concepteur n'a pas besoin de connaître le fonctionnement sous-jacent d'un solveur de problèmes GP, il lui faut seulement formuler son problème à l'aide de posynômes, en respectant leurs règles. Les posynômes permettent de représenter une grande variété de phénomènes physiques, notamment des fonctions *fittées* dont les coefficients et les puissances peuvent prendre de multiples valeurs. Les fonctions GP ont pour propriété intéressante leur additivité (propriété que possèdent aussi les fonctions convexes) : si deux fonctions sont GP, alors la somme de ces fonctions est aussi GP. Il est aussi possible de "sommer" deux problèmes GP : les deux fonctions objectifs sont sommées pour former la nouvelle fonction objectif, et leurs contraintes sont concaténées. Une fonctions

GP reste GP si elle est multipliée par un coefficient positif, ou par une autre fonction GP. En revanche, une fonction GP ne peut être divisée que par un GP-monôme, une division par un posynôme n'est pas GP. L'ensemble de ces propriétés permet de créer des modélisations de différents phénomènes indépendamment les uns des autres, et de facilement les imbriquer ensemble pour former des problèmes plus complexes.

3.3.2 Problèmes GP étendus : Generalized Geometric Programming

Dans la littérature, le domaine des fonctions GP compatibles est généralement étendu aux fonctions max et $f(x)^\beta$, en considérant certaines reformulations et l'ajout de contraintes appropriées au problème. On parle de *Generalized Geometric Programming* (GGP).

Soit n fonctions GP, on considère l'implémentation de la fonction $g : x \mapsto max(f_1(x), \dots, f_n(x))$. Cette fonction ne peut être qu'au numérateur, jamais au dénominateur, et jamais multipliée par une constante négative. Au sein du problème GP, que ce soit dans sa fonction objectif ou dans une fonction contrainte, $g(x)$ est remplacé par une variable t , et n nouvelles contraintes sont créées telles que :

$$f_i(x) \leq t \quad 1 \leq i \leq n \quad (3.10)$$

Le solveur employé pour résoudre le problème GP va chercher à réduire t à sa plus petite valeur possible, c'est à dire jusqu'à saturer l'une des nouvelles contraintes. La variable t va donc converger sur $max(f_1(x), \dots, f_n(x))$.

Soit $h : x \mapsto f(x)^\beta$ une fonction de passage à la puissance β d'une fonction GP $f(x)$. Nécessairement $\beta > 0$ (afin de ne pas diviser par un posynôme). La fonction $h(x)$ est là encore remplacée par une variable t , et est ajoutée au problème GP la contrainte :

$$f(x) \leq t^{1/\beta} \quad (\Leftrightarrow f(x)^\beta \leq t) \quad (3.11)$$

De la même manière que pour la fonction max , la variable t va converger vers la valeur $f(x)^\beta$. Ces deux fonctions ajoutées par le GGP sont intégrées et prises en charge par le solveur de la manière décrite ci-dessus, sans que l'utilisateur n'ait à créer manuellement une variable t et les contraintes associées. Pour cette raison, il est commun, quoique un peu abusif, d'englober le GGP dans la simple appellation GP. Nous suivrons nous-même cette "convention" tout au long de ce manuscrit

3.3.3 Intérêts et Usages

Tout comme les problèmes Convexes dont ils sont une sous-famille, les problèmes GP ne peuvent pas traiter de variables discrètes. Dans le domaine de la conception de convertisseurs électriques où ces dernières sont très nombreuses (nombre de spires d'une inductance, nombre de niveaux d'un convertisseur multiniveaux, etc.), cela signifie que le GP ne peut être employé que pour des problèmes très fermés où toutes ces variables discrètes ont préalablement été fixées, ou bien pour des études de tendances où ces variables discrètes sont supposées continues. Une autre limitation majeure est le formalisme requis pour qu'un problème soit GP. Certains phénomènes physiques

fondamentalement non *log-log-convexes* ne peuvent être que grossièrement approximés, voire impossibles à raisonnablement représenter. Par exemple, le comportement des inductances et notamment les phénomènes de saturation, posent des difficultés très importantes.

Il est possible que ces difficultés soient les raisons principales expliquant pourquoi le Geometric Programming n'a que peu été mis en œuvre pour modéliser des problèmes d'optimisation de design de convertisseurs électriques, et ce bien que le GP soit déjà une méthode de dimensionnement déjà populaire dans d'autres domaines d'ingénierie. Par exemple, pour les circuits analogiques et numériques, une rapide recherche sur *Google Scholar* sur l'optimisation GP des *CMOS* retourne plus d'une centaine d'articles pertinents. Pour le design des convertisseurs statiques, la première publication où ont été formulés sous forme de problèmes GP des convertisseurs DC-DC BUCK et BOOST a été faite en 2011 [89]. Elle donnera suite à deux autres articles sur le sujet par la même équipe [90,91]. Indépendamment de cette équipe, [92,93] présente une modélisation GP pour la topologie *Multilevel Flying Capacitor Converters* (FCML) en 2016. Seulement quelques articles similaires seront produits par la suite [94–97].

Pourtant, le Geometric Programming se prête particulièrement bien au *fitting* de données, bien mieux que les fonctions convexes standards, ce qui permet de créer et d'utiliser des modélisations empiriques plutôt que de dépendre de formulations théoriques [98–100]. De plus, les propriétés d'additivité des problèmes GP permettent d'assembler relativement aisément les modélisations GP de différents composants, au sein d'un problème plus englobant d'optimisation du design d'un convertisseur. A partir des travaux déjà publiés, il serait aujourd'hui possible d'optimiser simultanément le circuit de conversion de puissance, le design de l'inductance de filtrage, la géométrie des dissipateurs thermiques des transistors, etc., avec une garantie de convergence à l'optimum global dans un temps de calcul très court.

Les travaux présentés ci-après visent à développer un algorithme d'optimisation permettant de contourner l'impossibilité de traitement des variables discrètes par le *Geometric Programming*, tout en conservant sa garantie de convergence vers l'optimum global, et en tirant parti de son faible temps de calcul autant que possible.

3.4 Résoudre un problème GP contenant des variables discrètes

Soit P un problème GP. Nous distinguerons deux types de variables de design. Soit $\tilde{x} \in \tilde{\mathcal{X}}$ les variables devant prendre des valeurs dans un ensemble continu, et $\mathring{y} \in \mathring{\mathcal{Y}}$ les variables ne devant prendre que des valeurs discrètes. Nous emploierons les symboles \sim pour dénoter les variables continues, et \circ pour les variables discrètes. Les algorithmes standards de résolution des problèmes GP ne peuvent pas résoudre ces problèmes. Pour ce type de problème, on parle communément de *Mixed-Integer Geometric Programming* (MIGP), bien que cette appellation n'est pertinente que pour le cas particulier où \mathring{y} ne peut prendre que des valeurs entières ($\mathring{\mathcal{Y}} \subseteq \mathbb{Z}$). Il serait plutôt préférable de parler de *Mixed-Discrete Geometric Programming* (MDGP) dans le cas général où $\mathring{\mathcal{Y}} \not\subseteq \mathbb{Z}$, nous emploierons cette dénomination dans le cadre de ces travaux.

La nature hybride du problème ("*Mixed*") doit donc être traitée par un algorithme lui-même hybride. Les variables continues \tilde{x} peuvent toujours être traitées par un solveur GP (*Points Intérieurs...*), tandis que les variables discrètes \mathring{y} doivent être traitées par des méthodes adaptées aux problèmes discrets. L'approche exhaustive (*Brute Force*) est là encore la plus simple à programmer.

A partir du problème original P , dont les variables sont $X = (\tilde{x}, \hat{y})$, $X \in \tilde{\mathcal{X}} \times \hat{\mathcal{Y}}$, est créé pour chaque combinaison possible de \hat{y} un problème p_i dont les variables \hat{y} ont été fixées à leurs valeurs discrètes $\hat{y}(i)$ et n'ayant pour seules variables que \tilde{x} les variables continues. Ces sous-problèmes sont donc chacun des problèmes GP solvables. Il est donc possible de trouver pour chaque sous-problème p_i , s'il existe, l'optimum global des variables continues \tilde{x}_i^* , et en déduire leur optimum global $X_i^* = (\tilde{x}_i^*, \hat{y}(i))$. Soit l'espace des solutions optimales de tous les sous-problèmes :

$$\mathcal{S}^* = \bigcup_{i=1}^{i_{max}} \{X_i^*\} \quad (3.12) \quad \mathcal{S}^* \subset \mathcal{S} \quad (3.13)$$

La solution optimale globale de P , X^* , est donc la solution optimale X_i^* pour laquelle l'image par la fonction objectif $f_0(X_i^*)$ est minimale. Soit :

$$\forall X_i^* \in \mathcal{S}^*, \quad \exists X^* \in \mathcal{S}^*, f_0(X^*) \leq f_0(X_i^*) \quad (3.14)$$

Cette méthode exhaustive hybride présente la même faiblesse que la méthode *Brute Force* conventionnelle, appliquée à un problème discret. Le nombre de combinaisons possibles de \hat{y} , en fonction du nombre m de variables discrètes et du nombre k de valeurs admissibles pour chacune de ces variables, est $i_{max} = k^m$. Pour un nombre de variables discrètes faibles, le nombre de sous-problèmes p_i à résoudre reste relativement raisonnable. Autrement, la complexité exponentielle héritée de la méthode *Brute Force* éclipsé rapidement la complexité polynomiale de chaque sous-problème GP. Le temps de résolution d'un unique sous-problème par une méthode des *Points Intérieurs* devient marginal devant le temps total, rapidement prohibitif, consacré à la résolution du problème original P .

3.4.1 Algorithme Branch & Bound (BB)

Afin de réduire le nombre de combinaisons à traiter, nous utiliserons un algorithme *Branch & Bound* (BB). La stratégie BB est un algorithme de parcours en profondeur. Il consiste en la résolution successive de sous-problèmes *relaxés*, c'est à dire dont certaines contraintes du problème complet P ont été simplifiées ou retirées, afin de déterminer s'il est possible d'éliminer des solutions optimales potentielles toutes les combinaisons de \hat{y} appartenant à leur espace des solutions. Autrement dit, d'éliminer plusieurs combinaisons en un seul calcul. Notamment, si à un instant t dans sa recherche en profondeur, l'algorithme a calculé l'optimum global, il est capable d'éliminer toutes les autres solutions envisageables en seulement quelques calculs pour finalement reconnaître le résultat précédent comme optimum global. Le principe de cette méthode est le suivant :

Rang : Variables discrétisées :

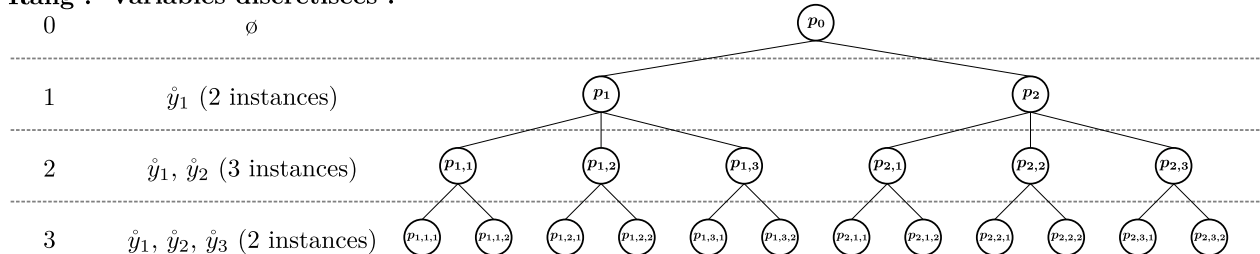


FIGURE 3.3 – Exemple d'un arbre de décision *Branch & Bound* montrant l'ensemble des combinaisons possibles

Un arbre de décision est généré, représenté en Figure 3.3. A sa base se trouve le nœud ν_0 . À ce nœud est associé le problème relaxé p_0 . Soit p_0 un sous-problème de P , pour lequel l'ensemble des variables sont considérées comme continues. Soit $\tilde{y}_i \in \tilde{\mathcal{Y}}_i$ la variable continue remplaçant la variable $\dot{y}_i \in \dot{\mathcal{Y}}_i$ discrète. Pour ce faire, pour chaque variable discrète \dot{y}_i (composantes de \dot{y}), la contrainte $\dot{y}_i \in \dot{\mathcal{Y}}_i$ est remplacée par deux contraintes d'encadrement sur la nouvelle variable continue \tilde{y}_i :

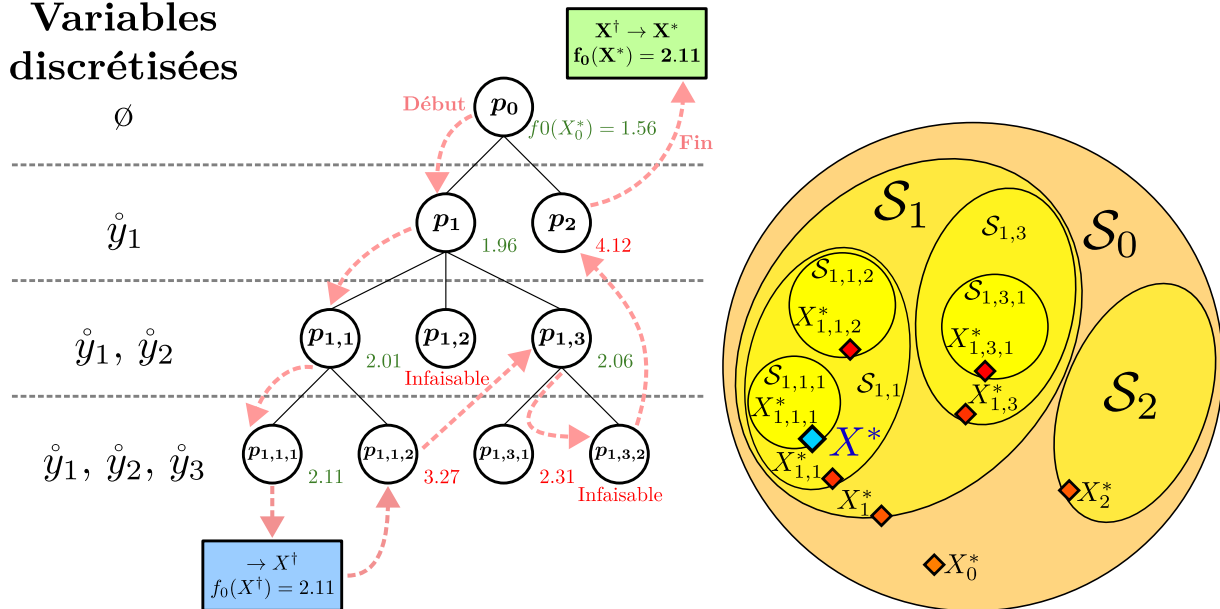
$$\dot{y}_i \in \dot{\mathcal{Y}}_i \implies \begin{cases} \min(\dot{\mathcal{Y}}_i) \leq \tilde{y}_i \\ \tilde{y}_i \leq \max(\dot{\mathcal{Y}}_i) \end{cases} \quad (3.15)$$

Le problème p_0 est dit *relaxé* car il est moins contraint que le problème complet P . Donc l'espace des solutions de P , \mathcal{S} , est un sous-ensemble de l'espace des solutions du problème relaxé \mathcal{S}_0 , soit $\mathcal{S} \subset \mathcal{S}_0$. Le sous-problème p_0 est GP. Si p_0 est infaisable, alors P est lui aussi infaisable, démontré en un seul calcul, et l'algorithme peut être arrêté.

Il faut maintenant "rediscrétiser" la première variable \tilde{y}_1 . A partir du nœud ν_0 , sont créés des nœuds *enfants* ν_{k_1} , $k_1 \in \{1, \dots, |\dot{\mathcal{Y}}_1|\}$ pour chaque valeur admissible de \dot{y}_1 (où $|\dot{\mathcal{Y}}_1|$ est le nombre d'éléments dans l'ensemble $\dot{\mathcal{Y}}_1$). Est imposé à chaque sous-problèmes p_{k_1} que $\dot{y}_1 = \dot{y}_1(k_1)$. Soit l'espace des solutions $\mathcal{S}_{k_1} \subset \mathcal{S}_0$ tel que :

$$\mathcal{S} \not\subset \mathcal{S}_{k_1}, \quad \text{mais } \mathcal{S} \subseteq \bigcup_{k_1=1}^{|\dot{\mathcal{Y}}_1|} \mathcal{S}_{k_1} \quad (3.16)$$

**Variables
discrétisées**



(a) Arbre de décision *Branch & Bound*

(b) Espaces des solutions explorés

FIGURE 3.4 – Exemple d'exploration d'un algorithme *Branch & Bound*

L'espace des solutions relaxées S_0 a donc été subdivisé en des ensembles plus resserrés dont ont été exclues les valeurs de \tilde{y}_1 n'appartenant pas à l'ensemble discret \mathcal{Y}_1 . On dira de l'ensemble de ces nœuds qu'ils sont de *rang* 1, car issus de la première opération de "rediscrétisation", et qu'ils ont pour *parent* le nœud ν_0 . Comme, pour chaque problème p_{k_1} , la valeur de \tilde{y}_1 a été fixée mais que le reste des variables \tilde{y}_i , $i \neq 1$ restent sous leur forme continue, ces problèmes sont tous GP. Ils sont résolus pour trouver leur optimum $X_{k_1}^*$ et leur valeur optimale $f_0(X_{k_1}^*)$ respectifs. On appelle *borne inférieure* la valeur optimale $f_0(X_{k_1}^*)$ obtenue pour ce nœud, car elle est la valeur optimale minimale que pourront produire les nœuds *enfants* de ce nœud ci. Nous utiliserons cette propriété ultérieurement. Nous pouvons néanmoins déjà utiliser $f_0(X_{k_1}^*)$ pour déterminer l'embranchement le plus intéressant à poursuivre pour l'algorithme. Nous supposons que, parmi tous les nœuds *frères*, c'est à dire ayant le même *parent*, celui présentant la plus petite borne inférieure $f_0(X_{k_1}^*)$ a le plus de chance de contenir l'optimum global du problème complet P dans son espace des solutions.

Pour l'étape suivante, l'algorithme sélectionne le nœud le plus prometteur, et génère ses nœuds enfants ν_{k_1, k_2} , pour lesquels la variable \tilde{y}_2 a été discrétisée sachant que, pour tous ces nouveaux nœuds, la première variable a déjà été fixée à $\tilde{y}_1 = \tilde{y}_1(k_1)$ à l'étape précédente. Le problème est à nouveau resserré, et les optimums calculés. Ce processus est répété successivement, le sous-problème de chaque nœud *enfant* est nécessairement contenu dans le sous-problème du nœud *parent* :

$$\forall i \in \{3, \dots, m\}, \quad \bigcup_{k_i=1}^{|\mathcal{Y}_i|} S_{k_1, \dots, k_i} \subseteq S_{k_1, \dots, k_{i-1}} \quad (3.17)$$

Ce qui signifie que les optimums des *parents* sont toujours aussi performants ou meilleurs que les optimums de leurs *enfants* :

$$\forall i \in \{3, \dots, m\}, \quad f_0(X_{k_1, \dots, k_{i-1}}^*) \leq f_0(X_{k_1, \dots, k_i}^*) \quad (3.18)$$

Lorsque la génération et l'exploration de l'arbre atteint le rang maximal $m = \mathbf{dim} \tilde{y}$, toutes les variables y à ce rang ont été discrétisées ($\tilde{y} = \emptyset$ et $y = \tilde{y}$). Les optimums calculés pour ces nœuds sont des solutions complètes recevables au problème P .

Si nous répétons ce processus de génération tel que l'arbre de décision soit entièrement généré, l'ensemble de tous les nœuds de rang m obtenus formerait l'ensemble de toutes les combinaisons possibles, tel que présenté pour l'algorithme *Brute Force*. Et donc :

$$\bigcup_{k_1=1}^{|\mathcal{Y}_1|} \dots \bigcup_{k_m=1}^{|\mathcal{Y}_m|} S_{k_1, \dots, k_m} = \mathcal{S} \quad (3.19)$$

$$\bigcup_{k_1=1}^{|\mathcal{Y}_1|} \dots \bigcup_{k_m=1}^{|\mathcal{Y}_m|} \{X_{k_1, \dots, k_m}^*\} = \mathcal{S}^* \quad (3.20)$$

L'équation 3.17 indique que l'optimum global X^* du problème complet P est contenu dans un seul et unique nœud de rang m (si f_0 est *strictement* convexe), noté ν^* , et dans tous ses nœuds *ancêtres* :

$$\forall i \in \{2, \dots, m\}, \quad \exists!(k_1, \dots, k_m) \in \{1, \dots, |\mathcal{Y}_1|\} \times \dots \times \{1, \dots, |\mathcal{Y}_m|\}, \quad X^* \in \mathcal{S}_{k_1, \dots, k_i} \quad (3.21)$$

Tandis que l'équation 3.18 implique que les nœuds *parents* du nœud ν^* possèdent nécessairement des optimums (aux problèmes sous-contraints!) aussi performants ou meilleurs que X^* :

$$\forall i \in \{1, \dots, m\}, \quad \exists!(k_1, \dots, k_m), \quad X_{k_1, \dots, k_m}^* = X^* \implies f_0(X_{k_1, \dots, k_i}^*) \leq f_0(X^*) \quad (3.22)$$

En reprenant notre déroulé, l'algorithme n'a exploré qu'une seule *branche* de l'arbre de décision. Il calcule les optimums des nœuds de rang m de cette branche (c'est à dire les nœuds en bout de branche), et mémorise l'optimum X_{k_1, \dots, k_m}^* découvert pour lequel $f_0(X_{k_1, \dots, k_m}^*)$ est minimal. Soit le meilleur optimum connu X^\dagger tel que :

$$X^\dagger = X_{k_1, \dots, k_m}^* \quad \text{si } f_0(X_{k_1, \dots, k_m}^*) < f_0(X^\dagger) \quad (3.23)$$

$$f_0(X^*) \leq f_0(X^\dagger) \quad (3.24)$$

Une fois cette valeur enregistrée, l'algorithme remonte d'un rang sur la *branche* du nœud *parent*. La valeur $f_0(X^\dagger)$ est comparée aux bornes inférieures $f_0(X_\kappa^*)$ des nœuds frères, soit κ une combinaison quelconque (k_1, \dots, k_i) , $i \leq m$. Si pour un nœud quelconque ν_κ , $f_0(X_\kappa^*) < f_0(X^\dagger)$, alors il est possible qu'une meilleure solution complète du problème P soit découverte parmi les nœuds *enfants* de ν_κ , l'algorithme devra donc explorer cette branche. Il commencera par le nœud présentant la valeur $f_0(X_\kappa^*)$ la plus faible et qui n'a pas encore été exploré. Sinon, les équations 3.22 et 3.24 nous apprennent que X^* , l'optimum global du problème complet P , ne peut pas être découvert parmi les *enfants* de ν_κ et, dans ce cas, l'algorithme élimine cette branche, et s'épargne donc de calculer l'ensemble de ses nœuds *enfants*. Une fois tous les embranchements évalués ou éliminés, l'algorithme remonte à son nœud *parent* et réitère l'opération jusqu'à avoir exploré ou éliminé l'entièreté des nœuds de l'arbre de décision. L'algorithme BB est résumé en Figure 3.5 :

En théorie, le nombre de nœuds à évaluer peut excéder le nombre de combinaisons à tester dans un algorithme *Brute Force*, soit $k^m + k^{m-1} + \dots + k + 1 = \frac{k^{m+1}-1}{k-1}$ pour un problème à m variables discrètes pouvant chacune admettant k valeurs différentes. L'algorithme progresse de manière aveugle. Bien qu'improbable, ce scénario peut se produire si X^* est découvert pour le dernier nœud évalué. Etant donné que l'algorithme favorise toujours l'exploration des nœuds présentant les bornes inférieures $f_0(X_\kappa^*)$ les plus prometteuses, cela impliquerait que tous les nœuds intermédiaires de l'arbre de décision présentaient de meilleures valeurs $f_0(X_\kappa^*)$ que les nœuds de la branche optimale, à rang égal, alors que les nœuds de rang m seraient tous moins performants que le nœud optimal ν^* .

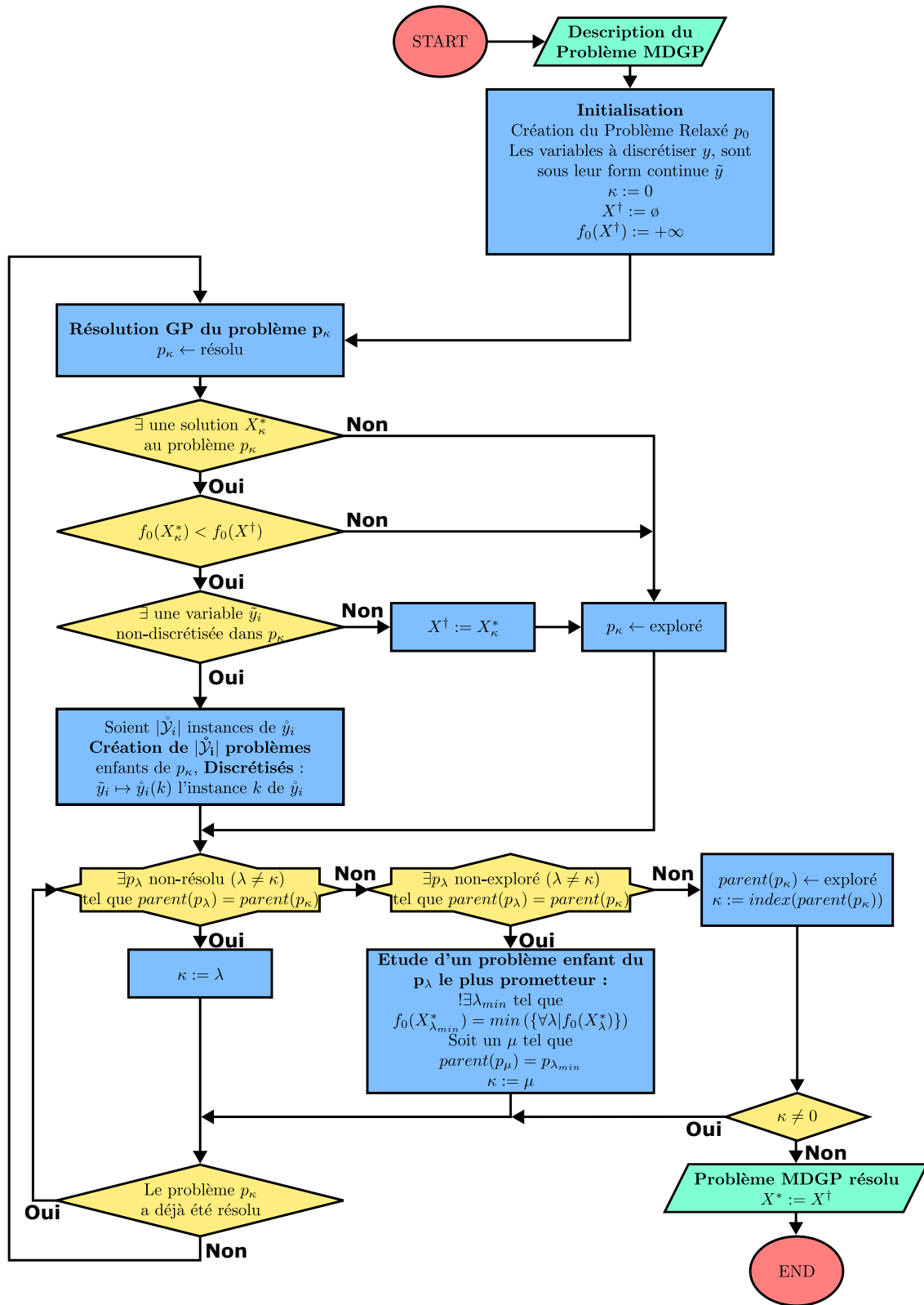


FIGURE 3.5 – Organigramme de programmation d'un algorithme *Branch & Bound*

En pratique, l'algorithme trouve généralement des solutions complètes X^\dagger suffisamment performantes pour éliminer des nœuds intermédiaires et avec eux toute leur descendance, et réduire ainsi considérablement le temps de calcul. Ce cas de figure extrême permet néanmoins de comprendre que la capacité de l'algorithme à éliminer des embranchements, et donc à réduire le temps de calcul, dépend de la "qualité" des sous-problèmes des nœuds intermédiaires. Plus les ensembles images de f_0 par les solutions aux sous-problèmes intermédiaires, $f_0(\mathcal{S}_\kappa)$, sont disjoints les uns des autres, plus il sera aisé pour l'algorithme *Branch & Bound* d'éliminer des embranchements. Pour favoriser de meilleures performances de l'algorithme BB, le concepteur peut placer en premières variables à rediscrétiser, les variables pour lesquelles la fonction objectif f_0 est le plus sensible. Les solveurs GP actuellement disponibles réalisent une analyse de sensibilité et la résolution du problème GP simultanément. Il serait donc possible de réaliser l'analyse de sensibilité des variables discrètes au nœud ν_0 , où toutes les variables y sont sous leurs formes continues \tilde{y} , et de décider automatiquement de l'ordre des rediscrétisations. Cette idée n'a toutefois pas été explorée dans ces travaux. Le nombre de valeurs admissibles $|\mathcal{J}_i^\circ|$ de chaque variable discrète \hat{y}_i joue aussi un rôle important car, chaque fois que l'algorithme BB atteint un nœud de rang $i - 1$, il calcule nécessairement un nombre $|\mathcal{J}_i^\circ|$ de nœuds de rang i et génère (si nécessaire) les nœuds enfants pour chacun. Placer aux rangs les plus bas (calculer en premier) les variables discrètes pour lesquelles $|\mathcal{J}_i^\circ|$ est faible et f_0 sensible favorise donc l'élimination d'un grand nombre de nœuds.

3.4.2 Algorithme Branch & Cut (BC)

L'algorithme *Branch & Bound* décrit ci-dessus présente l'avantage de réduire le nombre de calculs nécessaire par rapport à une méthode *Brute Force*, pour traiter un problème contenant à la fois des variables continues et discrètes. Mais dans les faits, il ne s'agit que d'un dérivé plus élaboré du *Brute Force* et, si les variables discrètes et leurs valeurs admissibles sont nombreuses, le nombre de calculs à réaliser reste prohibitif. La source de cette profusion de calculs vient du fait que chaque valeur admissible d'une variable discrète \hat{y}_i génère un nœud supplémentaire qui doit nécessairement être évalué si son parent n'a pas été éliminé. Il s'agit en fait d'une application très générale du principe de l'algorithme *Branch & Bound*, qui a l'avantage d'être une approche adaptable à tous types de problèmes combinatoires (ce qui nous servira par la suite), mais ce n'est pas sa forme la plus courante dans la littérature.

En général, la mention d'un algorithme *Branch & Bound* dans les travaux d'optimisation sur des problèmes "*Mixed-Integer*" désigne en fait un cas particulier parfois appelé *Branch & Cut* (BC). L'algorithme BC repose sur une hybridation entre la méthode *Branch & Bound* et la méthode des *Plans Sécants*. Plutôt que de tester toutes les combinaisons partielles (problèmes relaxés, nœuds intermédiaires) et complètes (nœuds de rang m) en forçant les variables à discrétiser y à prendre leurs valeurs discrètes admissibles assignées, l'espace des solutions est découpé successivement à l'aide de contraintes jusqu'à ce que la résolution d'un sous-problème ait naturellement convergé vers une solution admissible. Le principe, notamment décrit par [88], est le suivant :

L'algorithme crée le nœud ν_0 , pour lequel le sous-problème p_0 ne possède aucune contrainte de valeur discrète pour les variables y , soit $y = \tilde{y}$. Comme pour l'algorithme BB, la résolution de ce sous-problème relaxé permet dans un premier temps de déterminer en un calcul si le problème est infaisable. Il permet aussi de déterminer la solution optimale relaxée X_0^* qui va servir de point de départ à la recherche d'un optimum au problème complet P . Si la solution optimale relaxée d'une variable \hat{y}_i^* n'est pas admissible, alors il existe deux valeurs discrètes \hat{y}_i^- et \hat{y}_i^+ encadrant la solution optimale relaxée \tilde{y}_i^* :

$$\tilde{y}_i^* \notin \mathcal{Y}_i \implies \begin{cases} \hat{y}_i^- < \tilde{y}_i^* < \hat{y}_i^+ \\ \hat{y}_i^- = \max(\{\hat{y}_i \in \mathcal{Y}_i \mid \hat{y}_i < \tilde{y}_i^*\}) \\ \hat{y}_i^+ = \min(\{\hat{y}_i \in \mathcal{Y}_i \mid \hat{y}_i > \tilde{y}_i^*\}) \end{cases} \quad (3.25)$$

Lors de la vérification d'admissibilité de \tilde{y}_i^* , une tolérance ε est généralement implémentée pour éviter les faux positifs dus aux approximations numériques sur les valeurs de type *floats*. L'inégalité décrite en équation 3.25 forme une "bande interdite". Le sous-ensemble de S_0 , l'espace des solutions du sous-problème p_0 , ne peut pas appartenir à S l'espace des solutions du problème complet P . Il est donc possible de découper S_0 en deux nouveaux ensembles en créant deux sous-problèmes, et donc deux nœuds *enfants*. Au sous-problème du nœud "inférieur" ν_1^- est ajoutée la contrainte $\tilde{y}_i \leq \hat{y}_i^-$, et au sous-problème du nœud "supérieur" la contrainte $\tilde{y}_i \geq \hat{y}_i^+$.

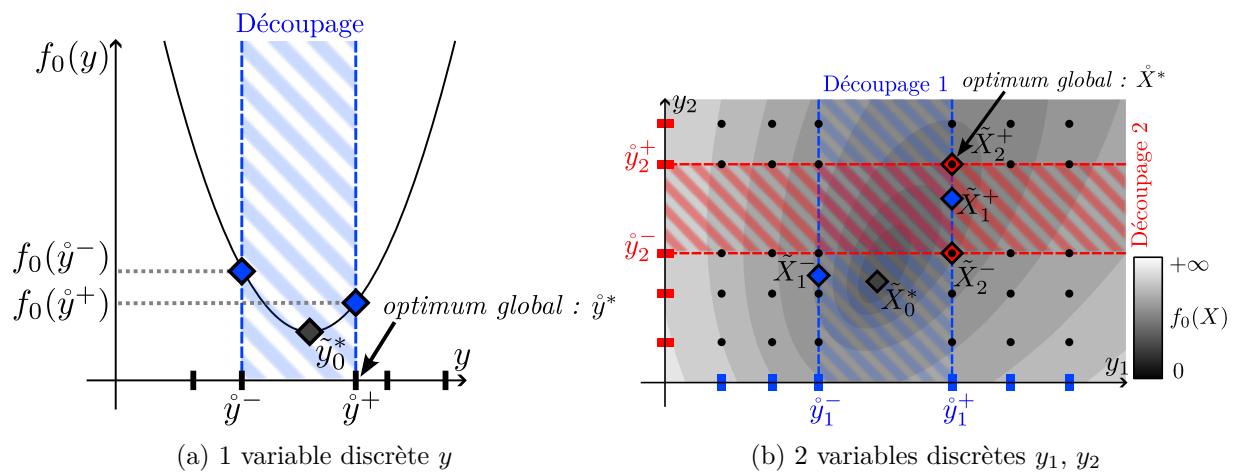


FIGURE 3.6 – Exemples de découpages de l'espace des solutions relaxé S_0 par l'algorithme *Branch & Cut* jusqu'à convergence vers l'optimum global discret

Après résolution de ces deux nœuds, le nœud présentant la solution la plus performante est sélectionné pour poursuivre l'exploration. La valeur optimisée \tilde{y}_i^* précédemment balisée aura probablement convergé vers sa borne discrète (\hat{y}_i^- ou \hat{y}_i^+ en fonction du nœud). Si ce n'est pas le cas, l'opération est réitérée jusqu'à ce que $\tilde{y}_i^* \in \mathcal{Y}_i$. Une fois ce résultat atteint, le reste des variables à discrétiser est parcouru et une autre variable \tilde{y}_j pour laquelle sa solution \tilde{y}_j^* n'est pas discrète est sélectionnée. Un nouveau découpage sur les deux valeurs discrètes d'encadrement \hat{y}_j^- et \hat{y}_j^+ est effectué, etc. jusqu'à finalement trouver une solution admissible au problème complet P , c'est à dire pour laquelle $\tilde{y}^* \in \mathcal{Y}$. Cette solution X_κ^* est mémorisée comme le meilleur optimum découvert X^\dagger si $f_0(X_\kappa^*) < f_0(X^\dagger)$ (ce qui est forcément le cas pour la première solution complète découverte). L'algorithme remonte ensuite l'arbre de décisions et explore les autres embranchements comme décrit ci-dessus en section 3.4.1, jusqu'à ce que tous les nœuds aient été explorés ou éliminés.

L'avantage majeur de cette approche *Branch & Cut* est qu'elle permet, en quelque sorte, de n'évaluer que les valeurs discrètes admissibles des y_i adjacentes à l'optimum global du problème relaxé, et non de tester l'ensemble des valeurs admissibles, contrairement à la méthode *Branch & Bound* combinatoire présentée en section 3.4.1. L'approche BC s'impose pour les variables discrètes possédant un très grand nombre de valeurs admissibles (ou une quantité indénombrable, par exemple $y_i \in \mathbb{N}$ l'ensemble des entiers).

3.4.3 Le cas des Tuples de variables discrètes

Jusqu'ici, les variables discrètes y_i ont été supposées toutes indépendantes les unes des autres. Or, l'objectif de ces travaux est de réaliser un algorithme d'optimisation qui soit le plus complet et proche d'un design réaliste possible. Si les variables discrètes indépendantes peuvent représenter un certain nombre de phénomènes, comme le nombre de spires d'une inductance [101], elles sont en revanche insuffisantes pour correctement décrire des phénomènes multifactoriels. Par exemple, si nous souhaitons non pas concevoir une inductance sur mesure pour le convertisseur à optimiser, mais au contraire demander à l'algorithme d'optimisation de sélectionner une inductance parmi une liste de composants *sur étagère*, les variables discrètes associées au choix du composant ne pourront prendre que les valeurs spécifiquement décrites dans leurs datasheets. Le choix d'un composant nous imposera, dans le cas d'une bobine, une valeur unique d'inductance L , associée à une valeur unique de résistance R_{DC} , à une valeur unique de masse m_L , etc., et choisir un autre composant modifie l'ensemble de ces valeurs.

Nous qualifierons dès lors ce type de variables de *variables discrètes couplées*, et les variables discrètes indépendantes de *variables discrètes simples*. Les variables discrètes couplées sont liées les unes aux autres au sein d'un *Tuple* Θ . Chaque valeur différente d'un *Tuple* est appelée une *instance* du *Tuple*. Un problème d'optimisation MDGP peut être composé de plusieurs *Tuples* différents (exemple eq. 3.27), ainsi que de variables discrètes simples et de variables continues. Soit $\dot{\Theta}_j \in \dot{\mathcal{T}}_j$ le j -ème *Tuple* du problème P , et $\dot{\mathcal{T}}_j$ l'ensemble de ses *instances*. $\dot{\Theta}_j(k_j)$, $k_j \in \{1, \dots, |\dot{\mathcal{T}}_j|\}$ est l'*instance* k_j du *Tuple* $\dot{\Theta}_j$. Le *Tuple* $\dot{\Theta}_j$ contient $|\dot{\Theta}_j|$ variables discrètes couplées, et $|\dot{\mathcal{T}}_j|$ instances. Nécessairement, les variables discrètes y_i couplées dans Θ_j ont autant de valeurs admissibles $|\dot{\mathcal{Y}}_i|$ qu'il existe d'instances de $\dot{\Theta}_j$:

$$\begin{aligned} \forall y_i \in \Theta_j, \quad |\dot{\mathcal{Y}}_i| &= |\dot{\mathcal{T}}_j| \\ \forall y_a \in \Theta_j, \quad \forall y_b \in \Theta_j, \quad |\dot{\mathcal{Y}}_a| &= |\dot{\mathcal{Y}}_b| \end{aligned} \quad (3.26)$$

Exemples de Tuples :

$$\begin{aligned} \Theta_{Inductance} &= (L, R_{DC}, m_L, \dots) \in \{(4.7\mu H, 1.69m\Omega, 36g); (15\mu H, 5.62m\Omega, 36g); \dots\} \\ \Theta_{Busbar} &= (r_{busbar}, \rho_{busbar}) \in \{(2.8 \cdot 10^{-8}\Omega \cdot m, 2700kg \cdot m^{-3}); (1.7 \cdot 10^{-8}\Omega \cdot m, 8920kg \cdot m^{-3})\} \\ &\dots \end{aligned} \quad (3.27)$$

Le traitement de ces *Tuples* par une méthode *Branch & Cut* n'étant pas évident, dans le cadre de ces travaux, les *Tuples* sont traités séparément des variables discrètes simples via un algorithme *Branch & Bound* combinatoire. L'algorithme complet de résolution du problème MDGP est donc une hybridation du BB, du BC, et d'un solveur GP pour calculer les Bornes Inférieures de chaque nœud. Au lancement de l'algorithme, le sous-problème relaxé p_0 est créé et résolu. Les variables y_i contenues dans les *Tuples* Θ_j , au lancement du programme, sont traitées de la même manière que les variables discrètes simples. Sous leur forme continue \tilde{y}_i , ces variables couplées sont balisées par deux contraintes d'encadrement telles que décrites par l'équation 3.15, l'algorithme à ce stade ignore non seulement que y_i ne peut prendre que des valeurs discrètes dans le problème complet P , mais aussi qu'elle est couplée à d'autres variables. On note $\tilde{\Theta}_j$ le *Tuple* sous sa forme continue. Lors

de l'embranchement du nœud ν_0 , au lieu de rediscrétiser une unique variable \tilde{y}_i , on rediscrétise un *Tuple* Θ_j . Les nœuds *enfants* de ν_0 sont toutes les *instances* du *Tuple*. Dans chacun de ces nœuds, l'ensemble des variables couplées du *Tuple* ont été rediscrétisées (exemple eq. 3.28).

Exemple :

$$\begin{aligned} & \text{variables de } \nu_0 : \tilde{x}_1, \tilde{x}_2, \tilde{y}_1, \tilde{y}_3, \left\{ \tilde{\Theta}_1 = (\tilde{y}_2, \tilde{y}_4) \right\} \\ \implies & \text{variables de } \nu_1 : \tilde{x}_1, \tilde{x}_2, \tilde{y}_1, \tilde{y}_3, \left\{ \dot{\Theta}_1(1) = (\dot{y}_2(1) = 2.451, \dot{y}_4(1) = 104.12) \right\} \end{aligned} \quad (3.28)$$

On procède ainsi pour chaque *Tuple* jusqu'à ce que tous les *Tuples* soient rediscrétisés, pour les nœuds de rang m_Θ le nombre de tuples du problème P . La résolution d'un nœud de rang m_Θ se fait ensuite en lançant un algorithme *Branch & Cut*. Le sous-problème du nœud en question sert de sous-problème relaxé initial à l'algorithme BC, qui se charge de discrétiser les variables discrètes simples et de trouver une solution complète au problème P .

3.5 Résoudre un problème à Fonctions Substituables

La dernière brique élémentaire manquante à notre algorithme pour pouvoir représenter notre problème d'optimisation est la capacité à tester plusieurs fonctions concurrentes et choisir celle produisant la meilleure solution au problème. Les *Tuples* de variables discrètes permettent de représenter des types d'objets ayant une influence sur plusieurs variables simultanément. Mais le choix d'une instance du *Tuple* plutôt qu'une autre ne résulte qu'en la modification de la valeur prise par chacune de ses variables discrètes, il ne change pas fondamentalement la description des phénomènes physiques couverts par le problème. Le choix entre deux matériaux, l'Aluminium ou le Cuivre, pour les busbars du convertisseur n'a pas d'influence sur les fonctions de la masse ou de la résistance électrique des busbars, seulement sur leur application. En revanche, si l'on compare deux circuits électriques très différents, ayant par exemple chacun leur fonction de transfert $H(x, y)$ et $G(x, y)$, il devient apparent que les deux seules manières de traiter le problème avec les outils présentés jusqu'ici est soit de lancer l'algorithme d'optimisation pour H , puis pour G et de prendre l'optimum le plus performant entre ces deux résolutions, ou bien de créer un *Tuple* $\dot{\Theta}_j = (a_1, a_2) \in \{(1, 0); (0, 1)\}$ qui sert de sélecteur dans une nouvelle fonction $h(x, y, a) = a_1 \cdot H(x, y) + a_2 \cdot G(x, y)$.

La méthode exhaustive pose un problème de temps de calcul si le nombre de fonctions concurrentes à tester est important. La méthode du sélecteur est compatible avec l'algorithme *Branch & Bound*, mais son implémentation n'est pas très gracieuse et surtout, rapidement encombrante car elle allonge artificiellement la taille des expressions par le nombre de fonctions concurrentes à tester. Typiquement, les transistors sont souvent modélisés par des formules empiriques (pour leurs pertes, leurs temps de commutation, etc.) et nous souhaitons comparer un nombre important de transistors. Nous proposons donc au contraire de tirer au maximum parti de la flexibilité des algorithmes *Branch & Bound*, et de traiter ces fonctions comme s'il s'agissait de variables discrètes, tel que décrit en section 3.4.1. Cela nécessite néanmoins l'ajout de nouveaux objets spécifiques à cette tâche.

Soit une nouvelle variable continue \tilde{F} appelée *variable de substitution*, et F une *Fonction Substituable* du problème P . La fonction F doit prendre une expression parmi l'ensemble des fonctions concurrentes $\dot{\mathcal{F}} = \{f_1; f_2; \dots; f_{|\dot{\mathcal{F}}|}\}$. Comme pour les *Tuples*, nous nommerons ces fonctions concu-

rentes les *instances* de F . Chaque instance de F peut posséder des variables différentes les une des autres, un nombre de variable lui aussi différent, et donc être définies pour des domaines différents, mais l'ensemble d'arrivée de chaque fonction instance f est impérativement inclus dans \mathbb{R} . On nomme cet ensemble $\tilde{\mathcal{F}}$ l'ensemble image de l'union des instances de F :

$$\bigcup_{i=1}^{|\mathring{\mathcal{F}}|} \mathbf{Im}(f_i) = \mathbf{Im}(F) = \tilde{\mathcal{F}} \quad (3.29)$$

Au lancement de l'algorithme, la fonction substituable F est traitée comme une variable continue \tilde{F} , sa *variable de substitution*, libre de prendre n'importe quelle valeur image de l'une des fonctions *instances* de F , soit $\tilde{F} \in \tilde{\mathcal{F}}$. Deux contraintes d'encadrement sont donc ajoutées au problème relaxé. Elles garantissent que \tilde{F} ne peut pas être inférieure à la valeur minimale que peuvent produire toutes les instances f_k sur leurs domaines respectifs, ou supérieur à leurs valeurs maximales :

$$\mathring{F} \in \mathring{\mathcal{F}} \implies \begin{cases} \min(\mathring{\mathcal{F}}) \leq \tilde{F} \\ \tilde{F} \leq \max(\mathring{\mathcal{F}}) \end{cases} = \begin{cases} \min(\{k \in \{1, \dots, |\mathring{\mathcal{F}}|\} | \min(\mathbf{Im}(f_k))\}) \leq \tilde{F} \\ \tilde{F} \leq \max(\{k \in \{1, \dots, |\mathring{\mathcal{F}}|\} | \max(\mathbf{Im}(f_k))\}) \end{cases} \quad (3.30)$$

La “rediscrétisation” de \tilde{F} se fait ensuite comme s'il s'agissait d'un *Tuple*, lors d'un embranchement de l'algorithme *Branch & Bound* : un nœud est créé pour chaque *instance*, c'est à dire que dans les $|\mathring{\mathcal{F}}|$ sous-problèmes créés, \tilde{F} aura été remplacé par l'expression de f_k la k -ème *instance* de F , aussi notée $\mathring{F}(k)$.

La variable substitutive \tilde{F} , dans un sous-problème relaxé, permet au solveur GP de ne pas avoir à connaître exactement la nature, le comportement ou les variables des fonctions *instances* de F . La valeur attribuée à \tilde{F} par le solveur correspond donc à la meilleure valeur que l'une des fonctions *instances* de F peut supposément produire pour optimiser la fonction objectif f_0 en respectant les contraintes du problème. L'encadrement peut être plus large (par exemple $\tilde{F} \in \mathbb{R} \supseteq \tilde{\mathcal{F}}$), mais la qualité de la Borne Inférieure du nœud s'en voit diminuée : elle risque d'atteindre une valeur bien trop basse et irréaliste, et rendre ainsi moins probable l'élimination du nœud et de toute sa descendance par l'algorithme BB. L'encadrement décrit en équation 3.30 n'est pas la meilleure solution, mais elle est la plus simple à mettre en œuvre. Néanmoins, cette plage de valeurs acceptables pour \tilde{F} peut être extrêmement large et “imprécise” car elle permet au solveur GP d'attribuer à \tilde{F} des valeurs qui ne peuvent être produites par les fonctions *instances* que pour des valeurs extrêmes de leurs variables (pour l'instant “masquées” par \tilde{F}). Par exemple, soit \tilde{E}_{on} une variable continue recouvrant les fonctions décrivant les énergies perdues au *turn-on* du MOSFET. Nous souhaitons choisir le meilleur MOSFET parmi une liste pour notre circuit. Malgré le fait que nous sachions que nos MOSFETs devront tenir une tension drain-source V_{ds} supposée comprise entre 15V et 40V, la contrainte formulée par l'équation 3.30 n'empêche pas \tilde{E}_{on} de prendre une valeur qui ne pourrait être atteinte que pour $V_{ds} = 0.5V$, car les fonctions *instances* ont été définies pour fonctionner sur la plage entière de 0.5V à la tension de claquage BV_{ds} de leurs transistors respectifs, et la variable V_{ds} n'est pas encore implémentée, elle est “cachée” derrière la variable substitutive \tilde{E}_{on} .

Il apparaît donc intéressant de chercher à encadrer plus finement la variable substitutive \tilde{F} , mais il faut impérativement respecter la condition sur les sous-problèmes du *Branch & Bound* décrite en équation 3.17 qui veut que \mathcal{S} l'espace des solutions au problème complet P soit toujours inclus dans l'union des solutions aux nœuds d'un même rang. Pour resserrer l'encadrement de \tilde{F} , il faut être

certain qu'en aucune manière l'une des fonctions *instances* ne pourra produire une valeur optimale en dehors du nouvel encadrement, au risque de rendre fausse la Borne Inférieure des nœuds avant rediscrétisation de \tilde{F} , et de permettre à l'algorithme BB de les éliminer abusivement.

L'implémentation d'une Fonctions Substituables requière non seulement de mémoriser les expressions de chaque fonction *instance* f_k , mais aussi leurs domaines de définition et d'application respectifs à chacune. Il est intéressant, pour faciliter leur manipulation par le concepteur, de mémoriser aussi un nom pour chaque instance. Enfin, les objets que nous cherchons à décrire recouvrent généralement plusieurs phénomènes physiques à la fois, et nécessitent donc plusieurs Fonctions Substituables "couplées". Le choix d'une instance plutôt que qu'une autre change plusieurs fonctions à la fois. C'est typiquement le cas des transistors qui nécessitent une fonction pour leurs pertes par commutation, pour leur résistance à l'état passant (qui varie avec la température), etc.. Il nous faut donc un nouvel objet similaire au *Tuple*, nommée *Structure* (ou *funcStruct* pour *function structure*), et notée Φ .

Une *Structure* Φ contient une ou plusieurs Fonctions Substituables F_i . Pour chaque F_i , la *Structure* Φ contient les contraintes d'encadrement associées aux variables de substitution \tilde{F}_i , mais aussi des contraintes d'encadrement des variables de chaque fonction *instance*. A ces contraintes d'encadrement peuvent être ajoutées des contraintes supplémentaires que l'on souhaiterait voir associées à une *instance* en particulier. Soit $\Phi(k)$ la k -ième *instance* de Φ , c'est à dire la collection des $\tilde{F}_i(k)$ les k -ièmes *instances* de toutes les Fonctions Substituables couplées au sein de Φ , et des k -ièmes *instances* de leurs contraintes associées. L'Algorithme *Branch & Bound* crée son arborescence pour chaque *instance* $\Phi(k)$ lorsqu'il atteint le rang où doit avoir lieu l'assignation de la *Structure*, de la même manière que pour les *Tuples*. Les contraintes d'encadrement des variables substitutives \tilde{F}_i sont ajoutées aux sous-problèmes avant rediscrétisation. Les autres contraintes sont ajoutées après la rediscrétisation.

3.6 Conclusion du chapitre

Afin d'améliorer le processus de conception d'un convertisseur électrique, un nouvel algorithme d'optimisation a été développé. Reposant sur une hybridation entre des méthodes *Geometric Programming* et *Branch & Bound*, cet algorithme dit *Mixed-Discrete Geometric Programming* (MDGP) permet de résoudre des problèmes contenant à la fois des variables continues, des variables discrètes simples ou couplées, et des fonctions substituables. La Figure 3.7 présente un exemple de l'exploration de l'espace des solutions pour un problème MDGP quelconque par notre algorithme, suivant la méthodologie décrite dans ce chapitre. Notre méthode MDGP présente l'avantage d'une grande flexibilité dans le type de problèmes qu'elle peut résoudre, par rapport aux méthodes GP et BB prises séparément, et conserve une garantie de convergence vers l'optimum global (s'il existe) au prix d'une forte augmentation du nombre de calcul par rapport au GP. Son fonctionnement a été présenté dans ce Chapitre. Le Chapitre 4 est dédié à la formulation de notre problème d'optimisation d'un convertisseur électrique en vue d'une résolution par l'algorithme MDGP, et à l'exploitation des résultats obtenus.

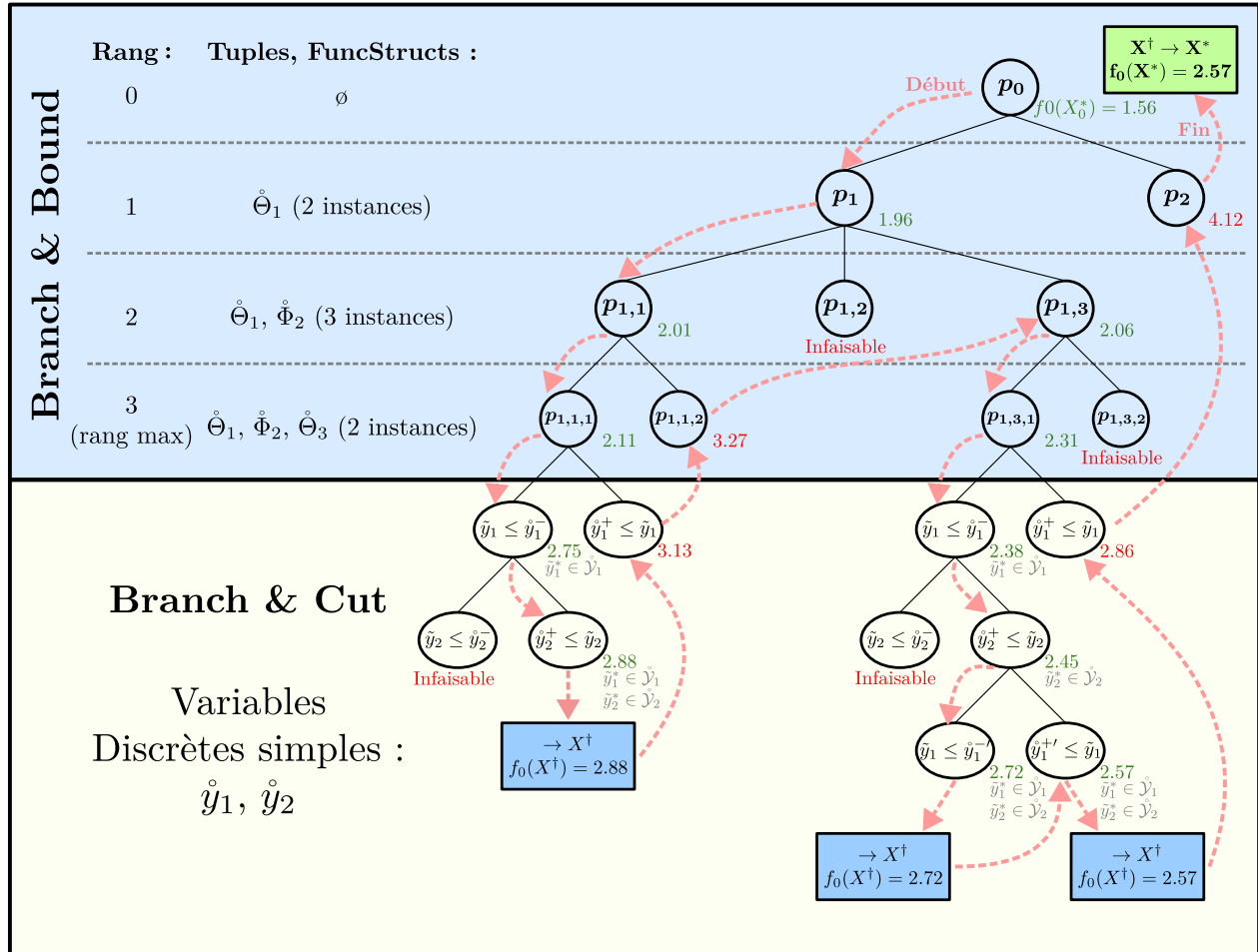


FIGURE 3.7 – Exemple de l’exploration des solutions pour un problème MDGP quelconque par l’algorithme MDGP

Chapitre 4

Optimisation du design d'un convertisseur FCML multiphase par un algorithme MDGP

4.1 Avant-Propos

L'objet de cette thèse est le développement d'un convertisseur électrique DC-DC présentant le meilleur rendement pour la plus forte densité de puissance possible. Un prototype a été conçu et ses performances ont pu être évaluées expérimentalement (Chapitres 1 et 2). En parallèle, les travaux menés sur les algorithmes d'optimisation ont abouti au développement d'un algorithme d'optimisation convexe combinatoire, basé sur une hybridation de méthodes *Branch & Bound* et *Geometric Programming* (Chapitre 3). Le programme a été développé sous *Matlab* et représente plus de 4000 nouvelles lignes de code, incluant l'algorithme en lui-même, les outils de mise en forme préalable de données, et de lecture des résultats. Il est basé sur la librairie *ggplab*, dédiée à l'optimisation GP sous *Matlab*.

La dernière version de la librairie *ggplab* a été publiée en 2006. Il est plutôt d'usage d'employer la librairie *cvx* pour des problèmes GP complexes, comme recommandé par la notice de *ggplab* elle-même [102]. En effet, l'outil *ggplab* a été développé à des fins d'apprentissage et n'est pas spécialement bien optimisé. Il a d'ailleurs fallu corriger un certain nombre d'erreurs de programmation et optimiser davantage son fonctionnement, lors de ces travaux. Néanmoins, la décision d'utiliser *ggplab* s'est faite naturellement. Ces travaux se sont appuyés sur un programme codé pour [101], capable de résoudre un problème MIGNP pour le design d'une inductance, à l'aide de la méthode *Branch & Cut* décrite en section 3.4.2, et ne devaient originellement être qu'une application de ce programme pour un design de convertisseur, soit une simple transcription des travaux de A. Stupar [93], pour notre cahier des charges avec une implémentation de quelques variables discrètes simples. La formulation du problème s'est progressivement complexifiée au-delà des ambitions originales, nous verrons en section 4.10 qu'il aurait été plus souhaitable de basculer sur un outil mieux optimisé que *ggplab*, la problématique des temps de calcul n'est apparue que bien trop tardivement lors de ces travaux pour reprendre la programmation de zéro.

Cela étant dit, un programme fonctionnel a été réalisé. La programmation de l'algorithme et la formulation du problème d'optimisation ont été faites indépendamment l'une de l'autre, de telle sorte que l'algorithme soit tout à fait adaptable à d'autres problèmes de type MDGP. Le Chapitre 3 a déjà couvert le fonctionnement général de l'algorithme, nous décrirons ici la formulation du problème de conception sous sa forme GP, et les processus mis en œuvre pour l'implémenter dans notre algorithme, puis nous analyserons les résultats.

4.2 Problématique

Le cahier des charges pour le développement de notre convertisseur électrique DC-DC basse tension et fort courant, décrit en Introduction de cette thèse, est retranscrit sous une forme exploitable par l'algorithme MDGP. Nous cherchons à maximiser le rendement η du convertisseur et sa puissance massique p_{conv} (kW/kg). Rappelons que η et p_{conv} sont donnés par :

$$\eta = 1 - \frac{P_{loss}}{P_{in}} \quad (4.1)$$

$$p_{conv} = \frac{P_{in,max}}{m_{conv}} \quad (4.2)$$

Où P_{loss} est l'ensemble des pertes du convertisseur, P_{in} la puissance entrante dans le convertisseur pour le point de fonctionnement étudié, $P_{in,max}$ est la puissance entrante nominale du convertisseur, et m_{conv} la masse totale du convertisseur (incluant aussi ses busbars, ses ventilateurs, etc.).

Ces deux grandeurs ne peuvent pas être employées comme objectif, car notre programme peut seulement minimiser des *posynômes*. Or, il s'agit de deux grandeurs à maximiser, et η n'est pas un *posynôme* (il est composé d'une soustraction). Nous emploierons donc les deux objectifs suivants :

$$\eta_{loss} = \frac{P_{loss}}{P_{in}} = 1 - \eta \quad (4.3) \quad M_{conv} = \frac{m_{conv}}{P_{in,max}} = \frac{1}{p_{conv}} \quad (4.4)$$

Nous cherchons donc à minimiser les pertes P_{loss} , ramenées à une proportion η_{loss} par rapport à la puissance du point de fonctionnement P_{in} , à des fins de normalisation (ce qui sera utile quand nous optimiserons plusieurs points de fonctionnement). Et en minimisant M_{conv} , la masse par kilowatt du convertisseur à son point de fonctionnement nominal, cela revient à maximiser son inverse la puissance massique p_{conv} . Les contraintes du cahier des charges sont directement transcrites sous forme de fonctions contraintes :

- $m_{conv} \leq 5kg$ La masse maximale admissible
- $V_{conv} \leq 15L$ Le volume maximal admissible
- $\Delta v_{in,max} \leq \pm 1\%$ Le taux d'ondulation relatif de la tension d'entrée V_{in}
- $\Delta v_{out,max} \leq \pm 10\%$ Le taux d'ondulation relatif de la tension de sortie V_{out}
- $\Delta i_L \leq \pm 10\%$ Le taux d'ondulation relatif du courant d'inductance I_L
(Garantit le fonctionnement en Mode Continu)
- $10kHz \leq f_{sw}$ La fréquence minimale de l'ondulation du courant I_{in}

Les critères de masse m_{conv} et d'ondulation $\Delta v_{in,max}$ ont été durcis par rapport au cahier des charges. Les choix du problème, dans leur expression la plus générique possible, sont :

- La topologie du circuit de conversion
- Le transistor
- Les condensateurs de filtrage (entrée, sortie, Flying Capacitor...)
- L'inductance de filtrage en sortie
- Le dissipateur thermique
- Les busbars

Quelques constantes du problème, liées au point de fonctionnement étudié, peuvent déjà être notées : P_{in} , V_{in} , V_{out} , dont sont déduits I_{in} , I_{out} et le rapport cyclique D . Nous détaillerons ci-dessous l'implémentation de ces choix, ainsi que les variables et les constantes à ajouter au problème pour pouvoir le décrire d'une manière interprétable par l'algorithme.

4.3 Implémentation de la Topologie et des Condensateurs

Bien que l'étude réalisée au Chapitre 1 aborde aussi les topologies *Double Buck* et *Double Flying Capacitor*, nous n'implémenterons dans notre problème GP que les topologies *Flying Capacitor* (FCML) que nous entrelacerons. Le choix de la topologie revient donc à choisir le niveau du convertisseur FCML, et le nombre de phases en parallèle. Les différentes grandeurs électriques pour cette topologie ont déjà été listées en Table 1.1, mais elles doivent être ajustées au formalisme imposé par le *Geometric Programming*. Nous nous inspirerons pour cela des travaux de [93]. Notons que la Table 1.1 n'a été écrite que pour une seule phase d'un convertisseur entrelacé.

L'équation 1.14 nous donne, à une opération près, Δi_L l'ondulation du courant traversant l'inductance de filtrage. Il s'agit néanmoins d'une équation incompatible avec le GP, à cause des nombreuses soustractions qu'elle contient. Dans l'ensemble des formules de la Table 1.1, le niveau du convertisseur N n'est jamais utilisé tel que. En posant comme variable \mathring{N}_{cell} le nombre de cellules de commutation, nous retirons une soustraction à ces équations. Soit :

$$\mathring{N}_{cell} = N - 1 \quad (4.5)$$

Notons que \mathring{N}_{cell} est une variable discrète, pouvant prendre une valeur entre 1 (topologie *BUCK*) à $+\infty$ mais, pour des raisons de complexité du problème, nous la limiterons à $\mathring{N}_{cell} \leq 6$. Il est proposé par [93] de définir une autre constante, soit \mathring{R}_i le coefficient d'ondulation de l'inductance :

$$\mathring{R}_i = \left(D - \frac{\mathring{i}_D - 1}{N - 1} \right) \cdot \left(\frac{\mathring{i}_D}{N - 1} - D \right) \quad (4.6)$$

Soit $D = V_{out}/V_{in}$ le rapport cyclique du convertisseur, et \mathring{i}_D l'indice de la région de fonctionnement du convertisseur déterminé par l'équation 1.8. L'équation 1.14 peut donc être réécrite d'une manière GP compatible, et adaptée à notre problème :

$$\Delta i_L = \frac{\mathring{R}_i \cdot V_{in}}{I_L \cdot \tilde{f}_{sw} \cdot L} \quad (4.7)$$

Où \tilde{f}_{sw} est la fréquence de commutation des interrupteurs, et I_L est le courant nominal dans l'élément inductif, c'est à dire le courant nominal en sortie d'une seule phase. Ce courant peut être exprimé en fonction de I_{out} le courant en sortie du convertisseur complet au point de fonctionnement étudié, et de la variable discrète \mathring{n}_{phase} le nombre de phases entrelacées dans le convertisseur :

$$I_L = \frac{I_{out}}{\mathring{n}_{phase}} \quad (4.8)$$

Il ne s'agit pas que d'une simple réécriture. Pour que l'équation 4.7 soit GP compatible, le solveur GP ne doit en aucun cas calculer lui-même \mathring{R}_i via l'équation 4.6. Cette difficulté est résolue en implantant la Topologie sous la forme d'un *Tuple*, qui contiendra entre autre les valeurs précalculées de \mathring{R}_i et \mathring{i}_D . Localement, au niveau du sous-problème d'un nœud du *Branch & Bound*, \mathring{R}_i et \mathring{i}_D seront traitées comme des constantes. Le passage par un *Tuple* est rendu possible car, pour un point de fonctionnement fixé, ces grandeurs sont constantes pour une topologie donnée. En effet, si V_{in} et V_{out} sont des constantes, alors D est constant, et donc \mathring{i}_D et \mathring{R}_i ne dépendent que de \mathring{N}_{cell} une variable discrète. Un *Tuple* $\mathring{\Theta}_{Topology}$ est donc créé, avec $|\mathring{N}_{cell}|$ instances. La partie *Branch & Bound* de notre algorithme MDGP permet donc d'éviter les calculs GP incompatibles en injectant des valeurs calculées avant le lancement du programme. Nous reviendrons sur ces équations lors de l'implémentation de l'inductance dans le problème.

La topologie FCML n'a pas d'influence sur Δv_{in} l'ondulation relative de la tension de bus, dont peut être déduite \tilde{C}_{in} la capacité de filtrage en entrée requise :

$$\Delta v_{in} = \frac{I_{out} \cdot D(1 - D)}{V_{in} \cdot \tilde{f}_{sw} \cdot \tilde{C}_{in}} \quad (4.9)$$

A noter ici que $D(1 - D)$, bien qu'étant une soustraction, n'est composé que de constantes et prend nécessairement une valeur positive. D est donc perçu comme une constante positive traitable par le solveur GP. Le nombre de cellules de commutations \dot{N}_{cell} a aussi directement un impact sur \tilde{C}_{out} la capacité de filtrage en sortie et $\tilde{C}_{fly,i}$ les $\dot{N}_{cell} - 1$ capacités flottantes de notre convertisseur FCML. Soit Δv_{out} le taux d'ondulation relatif de la tension de sortie, et $\Delta v_{ds,off}$ le taux d'ondulation relatif de la tension aux bornes des transistors lorsqu'ils sont bloquants, qui est directement lié à l'ondulation des tensions de charge des capacités flottantes $V_{Cfly,i}$ (exprimés pour le cas critique où ces ondulations s'additionnent toutes) :

$$\Delta v_{out} = \frac{I_L \cdot \Delta i_L}{8 \cdot \tilde{C}_{out} \cdot \dot{N}_{cell} \cdot \tilde{f}_{sw}} \quad (4.10)$$

$$\Delta v_{ds,off} = \frac{\dot{N}_{cell}}{V_{in}} \cdot \left(V_{in} \cdot \Delta v_{in} + \frac{I_L \cdot \dot{X}_i}{\tilde{f}_{sw}} \cdot \sum_{i=1}^{\dot{N}_{cell}-1} \frac{1}{\tilde{C}_{fly,i}} \right) \quad (4.11)$$

Où \dot{X}_i est le coefficient d'ondulation des capacités flottantes, défini par l'équation 1.9. Puisqu'il ne dépend que de D , \dot{i}_D et \dot{N}_{cell} , nous faisons de \dot{X}_i une variable discrète couplée à l'intérieur du *Tuple* $\dot{\Theta}_{Topology}$. De la même manière que dans l'étude réalisée au Chapitre 2, nous avons décidé de construire nos condensateurs en parallélisant les petits condensateurs céramique *GRM32EC72A106KE05*. Ce condensateur possède une capacitance unitaire $C_{unit} = 10\mu F$ pour une masse unitaire $m_{C,unit} = 110 \cdot 10^{-6} kg$ et une surface unitaire $S_{C,unit} = 3.2 \times 2.5 \cdot 10^{-6} m^2$, mais sa capacitance réelle dépend de la tension à ses bornes. La tension $V_{Cfly,i}$ n'est pas la même pour chaque capacités flottantes du circuit. Rappelons l'équation 1.12 :

$$V_{Cfly,i} = V_{in} \left(1 - \frac{i}{\dot{N}_{cell}} \right) \quad 1 \leq i < \dot{N}_{cell} \quad (4.12)$$

La tension $V_{Cfly,i}$ ne dépend que de \dot{N}_{cell} . En utilisant la Figure 2.1 extraite de la datasheet du condensateur, les coefficients de biais DC $\dot{c}_{Vfly,i}$, $\dot{c}_{V_{in}/2}$ et $\dot{c}_{V_{out}}$ peuvent être calculés pour chaque capacité flottante et pour les condensateurs d'entrée et de sortie. Notons que, pour le filtre d'entrée, nous appliquons la tension $V_{in}/2$ car il a été établi au Chapitre 2 que nous utiliserions des paires de 2 condensateurs en série pour diviser la tension V_{in} à tenir par condensateur. Les coefficients $\dot{c}_{Vfly,i}$ varient avec \dot{N}_{cell} . Ce sont donc eux-aussi des variables discrètes couplées dans le *Tuple* $\dot{\Theta}_{Topology}$, dont chaque instance représente leur valeur à la tension $V_{Cfly,i}$ correspondant à l'instance de \dot{N}_{cell} associée (eq. 4.12).

La manière dont nos *Tuples* sont construits nous oblige à ce que chaque variable couplée au sein d'un *Tuple* ait autant d'instances que le *Tuple* (et donc que ses autres variables couplées). Or, les capacités flottantes n'existent pas dans certaines topologies. Si $\dot{N}_{cell} = 1$, il s'agit d'un convertisseur *BUCK* sans capacité flottante, si $\dot{N}_{cell} = 3$ alors le convertisseur possède 2 capacités flottantes, mais pas 5 le nombre maximal possible dans notre problème. Il nous faut donc ajouter au *Tuple* $\dot{\Theta}_{Topology}$

une variable sélecteur binaire $\mathring{a}ct_{fly,i}$ pour chaque capacité flottante, qui prend la valeur 0 quand la topologie ne possède pas la i -ème capacité, ou 1 quand elle existe. Nous pouvons alors réécrire les équations 4.9, 4.10 et 4.11 :

$$\Delta v_{in} = \frac{I_L \cdot D(1 - D)}{V_{in} \cdot \tilde{f}_{sw} \cdot \tilde{C}_{in} \cdot \mathring{c}_{V_{in}/2}} \quad (4.13)$$

$$\Delta v_{out} = \frac{I_L \cdot \Delta i_L}{8 \cdot \tilde{C}_{out} \cdot \mathring{c}_{V_{out}} \cdot \mathring{N}_{cell} \cdot \tilde{f}_{sw}} \quad (4.14)$$

$$\Delta v_{ds,off} = \frac{\mathring{N}_{cell}}{V_{in}} \cdot \left(V_{in} \cdot \Delta v_{in} + \frac{I_L \cdot \mathring{X}_i}{\tilde{f}_{sw}} \cdot \sum_{i=1}^{|\mathring{N}_{cell}|-1} \frac{\mathring{a}ct_{fly,i}}{\tilde{C}_{fly,i} \cdot \mathring{c}_{V_{fly,i}}} \right) \quad (4.15)$$

Les variables continues \tilde{C}_{in} , \tilde{C}_{out} et les $\tilde{C}_{fly,i}$ sont les capacitances *sans biais*. Elles sont utilisées pour calculer le nombre de condensateurs *GRM32EC72A106KE05* par une simple division par C_{unit} . Le solveur optimise ces variables tel que, multipliées par leur coefficient de biais DC, les taux d'ondulation respectent les contraintes et minimisent les pertes. Les valeurs sans biais sont, quant à elles, utiles pour optimiser la masse du convertisseur en permettant le calcul de la masse et de la surface des condensateurs :

$$m_C = \frac{m_{C,unit}}{C_{unit}} \cdot \left(4 \cdot \tilde{C}_{in} + \tilde{C}_{out} + \sum_{i=1}^{|\mathring{N}_{cell}|-1} \mathring{a}ct_{fly,i} \cdot \tilde{C}_{fly,i} \right) \cdot \mathring{n}_{phase} \quad (4.16)$$

$$S_C = \frac{S_{C,unit}}{C_{unit}} \cdot \left(4 \cdot \tilde{C}_{in} + \tilde{C}_{out} + \sum_{i=1}^{|\mathring{N}_{cell}|-1} \mathring{a}ct_{fly,i} \cdot \tilde{C}_{fly,i} \right) \quad (4.17)$$

Où m_C est la masse de l'ensemble des condensateurs du convertisseur, c'est à dire en comptant toutes ses phases, qu'elles soient actives ou inactives (*phase shedding*), et S_C est la surface occupée par les condensateurs sur une seule phase. Cette surface n'est calculée que pour une phase, car elle servira à dimensionner le PCB pour une phase et à connaître sa masse de cuivre et de FR4, qui sera multipliée ultérieurement par \mathring{n}_{phase} .

Pour calculer les pertes associées aux condensateurs, il est nécessaire de connaître la résistance série équivalente du condensateur *GRM32EC72A106KE05*. A partir d'une courbe extraite de sa datasheet (Figure 4.1), une fonction GP compatible a été fittée pour une fréquence comprise entre 100Hz et 1MHz, avec un coefficient de détermination $R^2 = 0.9796$ satisfaisant. Nous obtenons les *ESR* pour les trois condensateurs :

$$ESR_{C_{in}} = 36.44 \cdot \tilde{f}_{sw}^{-0.797} \quad (4.18)$$

$$ESR_{C_{out}} = ESR_{C_{fly}} = 36.44 \cdot \tilde{f}_{sw}^{-0.797} \cdot \mathring{N}_{cell} \quad (4.19)$$

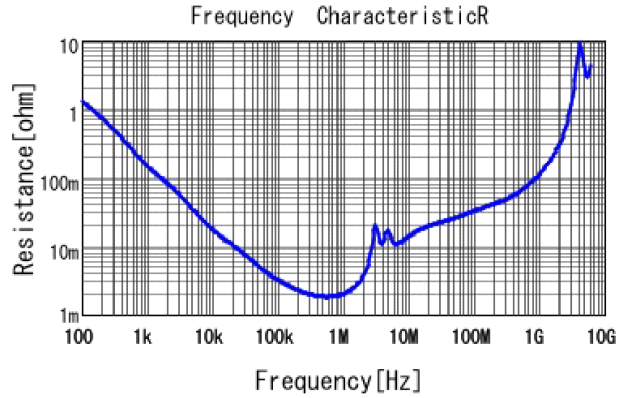


FIGURE 4.1 – Variation de l'ESR du condensateur *GRM32EC72A106KE05* en fonction de la fréquence de stimulation (datasheet)

Soit les courants efficaces à travers les capacités flottantes et de filtrage en sortie :

$$I_{Cin,RMS} = \sqrt{D(1-D) \cdot I_L^2 + \frac{D(1-D)^2}{12} (I_L \cdot \Delta i_L)^2} \quad (4.20)$$

$$I_{Cfly,RMS} = \sqrt{2\dot{X}_i \cdot I_L^2 + 2\dot{X}_i \cdot \frac{1}{12} (I_L \cdot \Delta i_L)^2} \quad (4.21)$$

$$I_{Cout,RMS} = I_L \cdot \Delta i_L \cdot \frac{1}{2\sqrt{3}} \quad (4.22)$$

Soit la formulation des pertes de chaque capacité :

$$P_{Cin} = \dot{n}_{phase} \cdot I_{Cin,RMS}^2 \cdot ESR_{Cin} \cdot \frac{C_{unit}}{\tilde{C}_{in}} \quad (4.23)$$

$$P_{Cfly} = \dot{n}_{phase} \cdot I_{Cfly,RMS}^2 \cdot \left(\sum_{i=1}^{|\dot{N}_{cell}|-1} \dot{a}_{ctfly,i} \cdot ESR_{Cfly,i} \cdot \frac{C_{unit}}{\tilde{C}_{fly,i}} \right) \quad (4.24)$$

$$P_{Cout} = \dot{n}_{phase} \cdot I_{Cout,RMS}^2 \cdot ESR_{Cout} \cdot \frac{C_{unit}}{\tilde{C}_{out}} \quad (4.25)$$

4.4 Modélisation des Inductances Vishay

Pour ces travaux, seules les inductances *Vishay* ont été étudiées. Elles présentent l'avantage d'être décrites par un modèle empirique fourni par le fabricant et homogène entre les différentes références, bien qu'il ne semble pas expérimentalement très fiable (section 2.4.3.5). Ce modèle a été décrit en section 1.7, mais nous expliciterons ici son implémentation sous une forme compatible au MDGP. Ces formules utilisent des paramètres fixés par le constructeur pour chacune de leurs

références, et listés dans [40, 103]. L'ensemble de ces paramètres sont donc implémentés sous forme de variables discrètes couplées au sein d'un *Tuple* $\mathring{\Theta}_L$, et l'algorithme MDGP sera en mesure de choisir la référence *Vishay* optimale (d'après son modèle empirique) pour notre convertisseur.

Le *Tuple* $\mathring{\Theta}_L$ contient les variables discrètes couplées suivantes :

— \mathring{L}	La valeur d'inductance
— $\mathring{DC}R_{typ}$	La résistance DC typique à 25°C
— \mathring{I}_{sat}	Le courant de saturation
— $\mathring{R}_{\theta,L}$	La résistance thermique entre le point chaud de l'inductance et l'air
— \mathring{ET}_{100}	Le produit $V \cdot \mu s$ pour une induction magnétique de 100G
— \mathring{K}_0	La constante des pertes fer
— \mathring{K}_1	La constante des pertes AC
— \mathring{K}_f	La constante de fréquence du matériau
— \mathring{K}_b	La constante de densité de flux du matériau
— \mathring{SRF}_{typ}	La fréquence de résonance
— $\mathring{m}_{L,unit}$	La masse du composant
— $\mathring{w}_L, \mathring{\ell}_L, \mathring{e}_L$	La largeur, la longueur et l'épaisseur du composant

Pour ce problème, comme nous avons pu le faire dans l'étude du Chapitre 1, nous nous limiterons aux 14 références de la série *IHLP-8787MZ-51* produite par *Vishay* afin de limiter l'explosion combinatoire et donc le temps de calcul de l'algorithme. En dehors du *Tuple*, nous ajouterons une variable discrète $\mathring{n}_{L,para}$, le nombre d'inductances en parallèle (égal à 1 ou 2). Les pertes dans l'inductance sont directement liées au courant I_L la traversant, à son ondulation Δi_L (eq. 4.7), à la tension découpée à ses bornes et à la fréquence de découpage \tilde{f}_{sw} . Rappelons les trois sources de pertes (DC, AC, et core) pour un unique composant :

$$P_{L,DC} = R_L \cdot \left(\frac{I_L}{\mathring{n}_{L,para}} \right)^2 \quad (4.26)$$

$$P_{L,AC} = \mathring{K}_1 \cdot (I_L \cdot \Delta i_L)^2 \cdot \sqrt{\mathring{N}_{cell} \cdot \tilde{f}_{sw}} \cdot R_L \quad (4.27)$$

$$P_{L,core} = \mathring{K}_0 \cdot f_e^{K_f-1} \cdot B_{pk}^{\mathring{K}_b} \cdot \mathring{N}_{cell} \cdot \tilde{f}_{sw} \cdot 10^{-14} \quad (4.28)$$

Les pertes $P_{L,DC}$ et $P_{L,AC}$ sont facilement implémentables. La seule difficulté réside dans le calcul de la résistance électrique R_L , qui dépend de la température du composant \tilde{T}_L . Or cette dernière dépend elle-même de la somme des pertes de l'inductance, ce qui forme une boucle de rétroaction, une fonction récursive. La méthode pour contourner ce problème a été décrite dans les travaux de A. Stupar [93]. Nous créons la variable continue \tilde{T}_L , que nous emploierons dans le calcul des pertes, et nous ajoutons au problème la contrainte suivante :

$$R_L = D\dot{C}R_{typ} \cdot \frac{234.5 + \tilde{T}_L}{259.5} \quad (4.29)$$

$$T_{L,calc} = T_{amb} + (P_{L,core} + P_{L,AC} + P_{L,DC}) \cdot \dot{R}_{\theta,L} \quad (4.30)$$

$$\text{contrainte : } T_{L,calc} \leq \tilde{T}_L \quad (4.31)$$

Où $T_{amb} = 25^\circ C$. Soit $T_{L,calc}$ l'expression de la température d'une inductance en fonction de ses pertes. La variable \tilde{T}_L est libre de prendre n'importe quelle valeur supérieure à $T_{L,calc}$. Comme une augmentation de \tilde{T}_L implique une augmentation de R_L et donc des pertes par conduction, ce qui dégraderait inutilement l'objectif de rendement du convertisseur, le solveur GP va nécessairement optimiser \tilde{T}_L en le minimisant. Ainsi, \tilde{T}_L va naturellement converger vers $T_{L,calc}$, le problème de récursivité est contourné.

Les pertes $P_{L,core}$ sont bien plus complexes à mettre en œuvre. Notons d'abord dans l'équation 4.28 la présence d'une soustraction $K_f - 1$. La constante K_f dans le *Tuple* $\dot{\Theta}_L$ est donc substituée par $\dot{k}_f = K_f - 1$ pour toutes ses instances. Bien plus problématique, les deux variables discrètes couplées \dot{k}_f et \dot{K}_b sont employées comme des puissances, ce qui n'est pas compatible avec les solveurs GP. Nous ne pouvons pas créer les variables continues associées \tilde{k}_f et \tilde{K}_b dans les nœuds du *Branch & Bound* précédant la sélection du *Tuple* $\dot{\Theta}_L$. Sans ces variables continues associées, il est impossible de créer des sous-problèmes relaxés dont la résolution permet de trouver des bornes inférieures à l'*optimum* du problème complet, bornes qui permettent potentiellement d'éliminer en un seul calcul un grand nombre de combinaisons possibles.

Nous avons vu en section 3.3.1 que les solveurs *Generalized Geometric Programming* adaptent l'expression $f(x)^\beta$, soit $f(x)$ un posynôme monté à une puissance β , en substituant $f(x)$ par une variable t et en ajoutant une contrainte $f(x) \leq t^{1/\beta}$ (équation 3.11). Cette méthode fonctionne pour $\beta > 0$, mais aussi et surtout pour β une constante. Or, le cas de figure rencontré dans l'équation 4.28 est une variable discrète $\dot{\beta}$, pouvant donc prendre plusieurs valeurs.

Plaçons-nous dans un sous-problème (relaxé) où la valeur de $\dot{\beta}$ n'a pas encore été fixée, nous souhaitons trouver la borne inférieure de $h(x, \dot{\beta}) = f(x)^{\dot{\beta}}$, où $\dot{\beta}$ est la forme continue de β tel que $\min(\dot{\beta}) \leq \dot{\beta} \leq \max(\dot{\beta})$. Si l'on peut connaître la valeur optimale $\tilde{\beta}^*$ sans faire appel au solveur GP, il est alors possible, à la création du sous-problème relaxé, de réaliser la substitution de $f(x)$ par t et imposer la contrainte :

$$f(x) \leq t^{1/\tilde{\beta}^*} \quad (4.32)$$

Où $\tilde{\beta}^*$ est traité comme une constante pour le solveur GP. Il existe deux cas particuliers qui permettent de déduire très simplement $\tilde{\beta}^*$:

$$\forall x \in \mathcal{X}, f(x) > 1 \implies \tilde{\beta}^* = \min(\dot{\beta}) \quad (4.33)$$

$$\forall x \in \mathcal{X}, 0 < f(x) < 1 \implies \tilde{\beta}^* = \max(\dot{\beta}) \quad (4.34)$$

Si $f(x)$ respecte la condition 4.33, il est possible d'imposer aux sous-problèmes relaxés précédant la sélection d'une instance de $\mathring{\beta}$ la contrainte :

$$f(x) \leq t^{1/\min(\mathring{\beta})} \quad (4.35)$$

Et il faudra ensuite impérativement remplacer dans cette contrainte la valeur $\min(\mathring{\beta})$ par la valeur $\mathring{\beta}(k)$ sur les sous-problèmes *enfants* auxquels a été assignée la k -ième instance de $\mathring{\beta}$, soit lorsque la k -ième instance de $\mathring{\beta}$ a été sélectionnée :

$$f(x) \leq t^{1/\min(\mathring{\beta})} \rightarrow f(x) \leq t^{1/\mathring{\beta}(k)} \quad (4.36)$$

Notre algorithme de résolution MDGP automatise ce processus de substitution. Rappelons que $\mathring{\beta}$ peut prendre des valeurs négatives, à la condition que $f(x)$ soit un GP-monôme. Sinon, si $f(x)$ est un posynôme, $\mathring{\beta} > 0$.

TABLE 4.1 – Coefficients K_f et \mathring{K}_b associés à chaque matériau de ferrite des inductances *Vishay*

Matériau	K_f	\mathring{K}_b
-01	1.188	2.118
-11	1.173	2.213
-51	1.044	2.497

Nous nous assurons de pouvoir utiliser la méthode substitutive décrite ci-dessus dans l'expression de $P_{L,core}$ (eq. 4.28). Les références d'inductance de *Vishay* utilisent trois matériaux différents pour leurs ferrites, dont les valeurs de K_f et \mathring{K}_b associées sont listées en Table 4.1. Notons que \mathring{K}_b et $\mathring{k}_f = K_f - 1$ sont positifs pour toutes leurs valeurs admissibles. Les expressions de la fréquence apparente f_e , de la densité de flux maximale B_{pk} , du rapport cyclique effectif D_{eff} et du produit $V \cdot \mu s$ effectif ET_{ckt} (eq. 1.29 - 1.32) sont réécrites :

$$f_e = \frac{\mathring{N}_{cell} \cdot \tilde{f}_{sw}}{2\pi \cdot (D_{eff} - D_{eff}^2)} \quad (4.37) \quad B_{pk} = \frac{ET_{ckt}}{ET_{100}} \cdot 100 \quad (4.38)$$

$$D_{eff} = \left(D - \frac{\mathring{i}_D - 1}{\mathring{N}_{cell}} \right) \cdot \mathring{N}_{cell} \quad (4.39) \quad ET_{ckt} = \frac{V_{in} \cdot \mathring{R}_i}{\tilde{f}_{sw}} \times 10^6 \quad (4.40)$$

Comme nous imposons $10kHz \leq \tilde{f}_{sw}$ et que par définition $0 < D_{eff} < 1$, nous pouvons garantir que la fréquence effective $f_e > 1$ et la condition 4.33 est donc respectée. En revanche, B_{pk} ne respecte pas cette condition sur \mathcal{X} . Notamment, l'équation 4.6 nous montre que $R_i = 0$ lorsque le rapport cyclique D est exactement à la jonction entre deux régions de fonctionnement i_D , c'est à dire qu'il n'y a aucune ondulation de courant Δi_L . Respecter cette condition revient à réduire l'espace des solutions \mathcal{S} d'une manière arbitraire. Non parce que les solutions conduisant à $B_{pk} < 1$ seraient physiquement infaisables, mais parce que notre algorithme ne fonctionne pas correctement si ces solutions sont incluses.

Cette restriction peut être implémentée dans le problème complet P , avant même de lancer l'algorithme. Il est possible de vérifier à la création du problème que, quelles que soient les valeurs de \tilde{f}_{sw} , \dot{N}_{cell} et \dot{ET}_{100} dans leurs plages respectives, $1 \leq B_{pk}$ pour le rapport cyclique $D = V_{out}/V_{in}$ du point de fonctionnement à optimiser. Il faut donc vérifier la condition suivante pour chaque instance du $Tuple \dot{\Theta}_{Topology}$:

$$\forall k \in \{1; \dots; |\dot{T}_{Topology}|\}, \quad 1 \leq \frac{V_{in} \cdot R_i(k)}{\max(\tilde{f}_{sw}) \cdot \max(\dot{ET}_{100})} \times 10^8 \quad (4.41)$$

Par simplicité, nous pouvons imposer au problème GP la contrainte $1 \leq B_{pk}$ mais, ce faisant, l'espace des solutions \mathcal{S} est dégradé en interdisant arbitrairement un certain nombre de points de fonctionnement du convertisseur. Par exemple, considérons le point de fonctionnement $V_{in} = 80V$, $V_{out} = 28V$, la valeur $\max(\dot{ET}_{100}) = 59.84V \cdot \mu s$ la plus élevée de la série d'inductances *IHLP-8787MZ-51*, et une fréquence maximale $\max(\tilde{f}_{sw}) = 1MHz$. Comme le rapport cyclique $D = 0.35$ est très proche de $1/3$, R_i devient très faible, et donc $\min(B_{pk}) < 1$ pour les topologies dont $\dot{N}_{cell} = 3, 6, \dots$ est un multiple de 3. Pour ces topologies problématiques spécifiquement, la contrainte $1 \leq B_{pk}$ va imposer au solveur de ne pas sélectionner les inductances aux valeurs \dot{ET}_{100} les plus élevées et/ou de diminuer la fréquence \tilde{f}_{sw} . Vérifier la condition 4.41 manuellement à la création du problème permet néanmoins au designer de repérer les topologies pénalisées par l'algorithme MDGP pour garantir la bonne implémentation du $Tuple \dot{\Theta}_L$.

Enfin, notons la présence d'une soustraction dans la fréquence effective f_e , et le rapport cyclique effectif D_{eff} est lui-même composé de soustractions. Néanmoins, comme D_{eff} ne dépend que de \dot{N}_{cell} et \dot{i}_D (deux variables discrètes couplées dans le $Tuple \dot{\Theta}_{Topology}$), cette soustraction peut être précalculée avant le lancement de l'algorithme. Nous créons donc la variable discrète \dot{D}_i et réécrivons f_e :

$$\dot{D}_i = D_{eff} - D_{eff}^2 \quad (4.42) \quad f_e = \frac{\dot{N}_{cell} \cdot \tilde{f}_{sw}}{2\pi \cdot \dot{D}_i} \quad (4.43)$$

La variable \dot{D}_i est couplée au sein de $\dot{\Theta}_{Topology}$, ses instances sont sa valeur pour chaque topologie. La fréquence effective f_e est à nouveau GP compatible. Les pertes $P_{L,core}$ (eq. 4.28) peuvent à nouveau être écrites sous une forme GP compatible en créant deux variables de substitution $\tilde{f}_{e,POW}$ et $\tilde{B}_{pk,POW}$ et en ajoutant au problème les contraintes associées :

$$P_{L,core} = \dot{K}_0 \cdot \tilde{f}_{e,POW} \cdot \tilde{B}_{pk,POW} \cdot \dot{N}_{cell} \tilde{f}_{sw} \cdot 10^{-14} \quad (4.44)$$

$$\text{contraintes :} \quad \begin{cases} f_e \leq \tilde{f}_{e,POW}^{1/\min(\dot{k}_f)} \\ B_{pk} \leq \tilde{B}_{pk,POW}^{1/\min(\dot{K}_b)} \end{cases} \quad (4.45)$$

Soit P_L les pertes et m_L la masse totales liées aux inductances :

$$P_L = \dot{n}_{phase} \cdot \dot{n}_{L,para} (P_{L,DC} + P_{L,AC} + P_{L,core}) \quad (4.46)$$

$$m_L = \dot{n}_{phase} \cdot \dot{n}_{L,para} \cdot \dot{m}_{L,unit} \quad (4.47)$$

4.5 Modélisation des Transistors

Le choix des transistors est déterminant pour les performances d'un convertisseur de puissance. Ils influent à la fois sur le rendement, via leurs pertes par conduction P_{cond} , en commutation P_{sw} et en conduction inverse P_{diode} , et sur la masse du convertisseur, via le dimensionnement du dissipateur thermique. Dans [93], A. Stupar crée une modélisation des différentes pertes du transistor en fonction de la surface de la puce (qui lui sert donc de variable continue). Nous avons souhaité dans ces travaux nous rapprocher au maximum de designs réalistes, où les caractéristiques physiques du transistor sont basées sur de véritables composants commercialisés. Les *Structures de Fonctions (FuncTruct)* décrits en section 3.5 ont été développés précisément pour cette raison. Ils nous permettent d'implémenter dans l'algorithme MDGP des modélisations différentes pour chaque transistor.

Soit la variable discrète simple $\mathring{n}_{sw,para}$. Elle nous permet d'explorer des designs où plusieurs transistors sont parallélisés pour former un interrupteur. Ses valeurs admissibles sont 1, 2 et 4. Une formulation des pertes totales liées aux transistors est établie :

$$P_{cond} = \mathring{N}_{cell} \cdot \mathring{n}_{sw,para} \cdot \mathring{n}_{phase} \cdot \left(\left(\frac{I_{out}}{\mathring{n}_{sw,para} \cdot \mathring{n}_{phase}} \right)^2 + \frac{(I_L \cdot \Delta i_L)^2}{12} \right) \cdot \tilde{R}_{dsON} \quad (4.48)$$

$$P_{sw} = (\tilde{E}_{on} + \tilde{E}_{off} + \tilde{E}_{rr}) \cdot \tilde{f}_{sw} \cdot \mathring{N}_{cell} \cdot \mathring{n}_{sw,para} \cdot \mathring{n}_{phase} \quad (4.49)$$

$$P_{diode} = (\tilde{E}_{diode,on} + \tilde{E}_{diode,off}) \cdot \tilde{f}_{sw} \cdot \mathring{N}_{cell} \cdot \mathring{n}_{sw,para} \cdot \mathring{n}_{phase} \quad (4.50)$$

$$P_{gate} = (\tilde{E}_{driver,LS} + \tilde{E}_{driver,HS}) \cdot \tilde{f}_{sw} \cdot \mathring{N}_{cell} \cdot \mathring{n}_{sw,para} \cdot \mathring{n}_{phase} \quad (4.51)$$

R_{dsON} est la résistance à l'état passant du transistor. E_{on} , E_{off} et E_{rr} sont les énergies dissipées à la commutation. $E_{diode,on}$ et $E_{diode,off}$ sont les énergies dissipées par la conduction inverse pendant les *deadtimes*. $E_{driver,LS}$ et $E_{driver,HS}$ sont les énergies consommées par le circuit d'alimentation de la grille d'un transistor au *Low Side* ou au *High Side* de la cellule de commutation sur une période de commutation. Ces termes sont toutes des propriétés intrinsèques au composant sélectionné, propriétés que nous modélisons par des fonctions substituables. Il nous faut donc créer le *FuncStruct* Φ_{MOSFET} , qui permettra à l'algorithme MDGP de sélectionner le transistor le plus performant pour notre application. Nous expliquerons ci-après la manière dont les fonctions substituables ont été établies pour chaque transistor envisagé. A cela, il faut ajouter un *Tuple* $\mathring{\Theta}_{MOSFET}$, couplé à Φ_{MOSFET} , qui contiendra toutes les grandeurs constantes associées à chaque transistor :

- $\mathring{B}V_{ds}$ La tension de claquage drain-source
- $\mathring{I}_{ds,max}$ Le courant maximal admissible
- $\mathring{R}_{\theta j,c}$ et $\mathring{R}_{\theta j,b}$ Les résistances thermiques jonction-boîtier et jonction-PCB
- $\mathring{V}_{gs,max}$ et $\mathring{V}_{gs,min}$ Les tensions de grille maximale et minimale
- \mathring{w}_{sw} , $\mathring{\ell}_{sw}$ et \mathring{e}_{sw} La largeur, la longueur et l'épaisseur du composant

4.5.1 Génération des modèles de transistors

Le comportement électrique des transistors est modélisé à partir de simulations *LTSPICE*. Nous utiliserons les modèles *SPICE* fournis par les fabricants, implémentés dans les circuits représentés en Figures 1.11 et 2.20. La méthodologie pour relever les pertes et autres propriétés électriques

du transistor simulé a été détaillée en section 1.5. Sont simulés à la fois le circuit sans inductance parasite, pour faciliter l'extraction des pertes par commutation, et le circuit doté d'inductances parasites. Ce dernier permet de simuler les surtensions de V_{ds} la tension drain-source lors de la commutation, pour tenter de les modéliser par une fonction GP. Il a été montré en section 2.4.1, Table 2.7, que la simulation *SPICE* des surtensions n'est pas extrêmement précise par rapport aux surtensions mesurées expérimentalement (erreur relative maximale : +36.4%), mais il reste pertinent de chercher à modéliser ces surtensions dans le problème d'optimisation MDGP. Il faut sinon imposer un coefficient de sécurité arbitraire entre le calibre en tension de claquage $\overset{\circ}{B}V_{ds}$ du transistor et la tension maximale imposée à ses bornes. Les transistors étudiés sont présentés en Table 4.2.

Les transistors sont simulés en faisant varier les paramètres suivants, qui deviendront des variables du problème MDGP :

- \tilde{V}_{ds} La tension drain-source
- \tilde{I}_{ds} Le courant drain-source
- $\tilde{R}_{g, SRC}$ et $\tilde{R}_{g, SNK}$ Les résistances de grille externes au *turn-on* et *turn-off*
- \tilde{t}_{dt} Le *deadtime*
- \tilde{T}_j La température de jonction

TABLE 4.2 – Transistors implémentés dans le *FuncStruct* Φ_{MOSFET}

Transistor	$\overset{\circ}{B}V_{ds}$	$\overset{\circ}{I}_{ds, max}$	R_{dsON}	Année
EPC2024	40V	90A	1.2m Ω	2016
EPC2066	40V	90A	1.1m Ω	2022
EPC2020	60V	90A	2.2m Ω	2016
EPC2021	80V	90A	2.2m Ω	2016
EPC2206	80V	90A	2.2m Ω	2019
EPC2065	80V	60A	3.6m Ω	2021
EPC2022	100V	90A	3.2m Ω	2016
GS61008P	100V	90A	7m Ω	2020
EPC2302	100V	101A	1.8m Ω	2022
EPC2034C	200V	48A	8m Ω	2019

Nous souhaitons modéliser en fonction de ces paramètres les propriétés électriques suivantes :

- \tilde{E}_{on} , \tilde{E}_{off} et \tilde{E}_{rr} Les énergies de commutation
- $\tilde{t}_{sw, off}$ Le temps pris par la commutation *turn-off* du transistor *High-Side*
- $\tilde{E}_{driver, HS}$ et $\tilde{E}_{driver, LS}$ Les énergies d'alimentation de la grille
- $\tilde{t}_{diode, on}$ et $\tilde{t}_{diode, off}$ Les temps de conduction inverse du transistor *High-Side*
- $\tilde{E}_{diode, on}$ et $\tilde{E}_{diode, off}$ Les énergies dissipées pendant la conduction inverse
- $\tilde{V}_{OSds, HS}$ et $\tilde{V}_{OSds, LS}$ Les surtensions (en volts) aux *turn-off* du *High-Side* et du *Low-Side*
- $\tilde{I}_{drv, on}$ et $\tilde{I}_{drv, off}$ Les courants maximaux en sortie de driver
- $\tilde{V}_{miller, on}$ et $\tilde{I}_{miller, on}$ Les courant et tension sur la grille *Low-Side* au *turn-on* du *High-Side*
- $\tilde{V}_{miller, off}$ et $\tilde{I}_{miller, off}$ Les courant et tension sur la grille *High-Side* au *turn-on* du *Low-Side*
- \tilde{R}_{dsON} La résistance à l'état passant

— \tilde{V}_f

La tension de seuil de la diode de corps

Chacune de ces propriétés est une fonctions substituables du *FuncStruct* Φ_{MOSFET} , ayant pour variables les paramètres de balayage listés ci-dessus. La température de jonction \tilde{T}_j n'a aucune influence sur la commutation dans les modèles *SPICE* des composants *EPC*. En revanche, la résistance à l'état passant R_{dsON} et la tension de diode de corps V_f sont très dépendantes de T_j et ne dépendent aucunement de t_{dt} ni des résistances de grilles et de la tension V_{ds} . Les tensions $V_{miller,on}$ et $V_{miller,off}$, et les courants $I_{miller,on}$ et $I_{miller,off}$ permettent de modéliser le comportement du transistor face au phénomène des pseudo-commutations de *Miller* décrit en section 2.2.3. Il est ainsi possible de contraindre l'algorithme MDGP pour protéger la grille du transistor ou le driver :

Contraintes :

$$\tilde{V}_{miller,on} \leq \mathring{V}_{gs,max} \quad (4.52)$$

$$\tilde{V}_{miller,off} \leq \mathring{V}_{gs,min} \quad (4.53)$$

$$\tilde{I}_{miller,on} \leq \mathring{I}_{drv,max} \quad (4.54)$$

$$\tilde{I}_{miller,off} \leq \mathring{I}_{drv,max} \quad (4.55)$$

$$\tilde{I}_{drv,on} \leq I_{SRC,peak} \quad (4.56)$$

$$\tilde{I}_{drv,off} \leq I_{SNK,peak} \quad (4.57)$$

Où $\mathring{V}_{gs,max}$ est la tension maximale admissible par la grille du transistor, et $\mathring{V}_{gs,min}$ la tension négative maximale admissible, dont on a permuté le signe pour rendre sa valeur positive et compatible avec le *Geometric Programming*. Ces deux tensions sont des variables discrètes couplées au sein du *Tuple* Θ_{MOSFET} . Les courants $I_{SRC,peak}$ et $I_{SNK,peak}$ sont les courants maximaux admissible par le driver. Il serait possible de créer un *Tuple* pour permettre à l'algorithme MDGP de sélectionner un driver parmi une liste, dans ce cas de figure ces deux courants seraient des variables discrètes couplées à ce *Tuple*. Dans le cadre de notre problème, nous supposerons que le driver employé est le *2EDF7275KXUMA1* d'*Infineon*, dont les caractéristiques sont résumées en Table 2.2 (section 2.2.4). Soit $I_{SRC,peak} = 4A$ et $I_{SNK,peak} = 8A$. Ces contraintes vont s'appliquer notamment à la sélection des résistances de grilles $\tilde{R}_{g,SRC}$ et $\tilde{R}_{g,SNK}$. Les résistances de grilles ont un effet prononcé sur les pertes par commutation, et l'algorithme MDGP va tendre à les minimiser tout en respectant les contraintes en courant des drivers. Parallèlement, l'algorithme cherchera à minimiser le *deadtime* \tilde{t}_{dt} pour réduire les pertes par conduction inverse, pouvant être décrites par :

$$E_{diode,off} = (\tilde{t}_{dt} - \tilde{t}_{sw,off}) \cdot \tilde{V}_f \cdot \tilde{I}_{ds} = \tilde{t}_{diode,off} \cdot \tilde{V}_f \cdot \tilde{I}_{ds} \quad (4.58)$$

Si \tilde{t}_{dt} la durée du *deadtime* est précisément égale à $\tilde{t}_{sw,off}$ le temps de *turn-off* du transistor, ce dernier ne produit aucune perte en conduction inverse, tout en réalisant une commutation ZVS qui minimise ses pertes E_{off} . Rappelons que $\tilde{t}_{sw,off}$ dépend principalement de $\tilde{R}_{g,SNK}$ la résistance de grille externe. Mais il devra pour ce faire trouver un compromis entre \tilde{t}_{dt} et $\tilde{R}_{g,SNK}$, car si le temps de *turn-off* $\tilde{t}_{sw,off}$ (proportionnel à $\tilde{R}_{g,SNK}$) est supérieur à \tilde{t}_{dt} , la commutation *turn-off* est réalisée en *hard-switching* et les pertes augmentent considérablement. Il a été tenté de modéliser les énergies E_{off} en incluant le phénomène de *hard-switching* dans les données *fittées*, sans succès. Nous imposerons donc la contrainte 4.59 ci-dessous, ce qui interdit à l'algorithme de rechercher

une solution optimale incluant du *hard-switching* au *turn-off* mais lui permet tout de même de potentiellement converger vers une commutation en ZVS où $t_{sw,off} = \tilde{t}_{dt}$. Ces résistances de grille $\tilde{R}_{g,src}$ et $\tilde{R}_{g,snk}$ ont aussi une forte influence sur les formes d'onde à la grille, au drain et à la source. Minimiser $\tilde{R}_{g,src}$ et $\tilde{R}_{g,snk}$ tend à augmenter V_{OSds} la surtension de V_{ds} , et I_{drv} le courant appelé par la grille à travers le driver, lors de la commutation. Inversement, augmenter $\tilde{R}_{g,src}$ et $\tilde{R}_{g,snk}$ pour protéger les composants augmente les pertes à la commutation et augmente V_{miller} le risque de commutations spontanées de *Miller*. L'algorithme doit être en mesure de trouver un bon compromis entre les pertes et la protection des différents composants. Les contraintes suivantes sont donc ajoutées :

Contraintes :

$$\tilde{t}_{sw,off} \leq \tilde{t}_{dt} \quad (4.59)$$

$$\tilde{V}_{OSds,HS} \leq \mathring{B}V_{ds} \quad (4.60)$$

$$\tilde{V}_{OSds,LS} \leq \mathring{B}V_{ds} \quad (4.61)$$

Chacun de ces paramètres est balayé sur une dizaine de valeurs, ce qui signifie qu'un très grand nombre de simulations *LTSPICE* doivent être exécutées pour pouvoir "cartographier" les propriétés électriques du composant. Par exemple, 25872 points de fonctionnement ont été simulés pour le seul transistor *EPC2022*. Comme évoqué en section 1.5, la simplicité de notre circuit *LTSPICE* permet de descendre le temps de simulation pour un point de fonctionnement à environ 2.5s, soit un temps de simulation de 18h pour l'*EPC2022*. Le logiciel *LTSPICE* intègre une commande pour réaliser ces balayages de paramètres mais, comme il n'évalue les commandes *.MEAS* (qui réalisent les relevés d'énergies, etc. à partir des courbes simulées) qu'après avoir exécuté le balayage entier, il remplit rapidement toute la mémoire RAM de l'ordinateur et ralentit considérablement la vitesse de calcul. La taille des fichiers de résultat complets (*.raw*) est elle aussi très excessive, de plusieurs dizaines de Gigaoctets, alors que seuls les résultats des commandes *.MEAS* sont intéressants et ne nécessitent qu'une très faible mémoire. Un script *Matlab* a donc été écrit pour répartir le balayage par paquets de 100 points de fonctionnements. Après la simulation de chaque paquet, les commandes *.MEAS* sont exécutés, leurs résultats sont sauvegardés (et compilés avec les précédents), et les données complètes *.raw* sont supprimées et la RAM purgée, avant de lancer le paquet suivant.

Les différentes grandeurs relevées pour chaque point sont ensuite ajustées puis modélisées sous forme de fonctions GP à l'aide du script de fitting *ggposy* fourni dans la librairie *ggplab* sous *Matlab*. La qualité du *fit* est estimée par l'erreur relative moyenne et l'erreur quadratique moyenne (RMSE) par rapport aux données *LTSPICE*, et peuvent être facilement examinées à l'aide d'un script dédié de représentation graphique (exemple en Figure 4.2 pour l'*EPC2022*). La précision des *fits* pour le composant *EPC2022* est donnée en Table 4.3.

Nous ne présenterons ici que le *fitting* du composant *EPC2022* car il est représentatif de la qualité des *fits* des autres transistors. Pour la plupart des grandeurs à modéliser, nous sommes en mesure de produire un *fit* de bonne qualité, avec des erreurs relatives moyennes inférieures à 10% et des RMSE très faibles. Certains de ces *fits* produisent des erreurs relatives maximales très importante, mais elles n'apparaissent qu'à des valeurs extrêmes des paramètres (\tilde{V}_{ds} et/ou \tilde{I}_{ds} à leurs

TABLE 4.3 – Erreurs des fonctions *fittées* pour le transistor *EPC2022*, par rapport aux simulations *LTSPICE*

Fonction	max(ε)	mean(ε)	RMSE(ε)
E_{On}	572.74%	4.27%	$2.8985 \cdot 10^{-7} J$
E_{off}	389.58%	5.345%	$3.4309 \cdot 10^{-8} J$
E_{rr}	649.39%	9.4991%	$1.2813 \cdot 10^{-7} J$
$E_{\text{driver,HS}}$	3.7982%	0.38707%	$3.1652 \cdot 10^{-10} J$
$E_{\text{driver,LS}}$	23.248%	2.7133%	$2.3562 \cdot 10^{-9} J$
$t_{\text{sw,off}}$	58.008%	1.8157%	$3.2053 \cdot 3.2053^{-10} s$
$t_{\text{diode,off}}$	$2.0918 \cdot 10^6\%$	170.8%	$6.7757 \cdot 10^{-9} s$
$t_{\text{diode,on}}$	113.64%	7.0587%	$2.2376 \cdot 10^{-9} s$
$E_{\text{diode,off}}$	$1.3066 \cdot 10^6\%$	140.44%	$1.0372 \cdot 10^{-6} J$
$E_{\text{diode,on}}$	1125.4%	10.859%	$4.5892 \cdot 10^{-7} J$
$V_{\text{OSds,HS}}$	25.774%	2.0165%	0.971V
$V_{\text{OSds,LS}}$	28.078%	5.3381%	4.1849V
$I_{\text{drv,on}}$	26.762%	7.2481%	0.36648A
$I_{\text{drv,off}}$	3.1192%	2.0286%	0.093853A
$V_{\text{miller,on}}$	16.823%	4.5503%	0.073838V
$V_{\text{miller,off}}$	3.3798%	2.2401%	0.073838V
$I_{\text{miller,on}}$	18.247%	4.5504%	0.082252A
$I_{\text{miller,off}}$	3.4234%	2.2685%	0.044891A
R_{dsON}	0.020468%	0.0054075%	$2.1593 \cdot 10^{-7} \Omega$
V_f	0.50465%	0.15998%	0.0046007V

valeurs minimales) où la valeur donnée par *LTSPICE* pour cette fonction est extrêmement faible comparée au reste des données, ce qui fait artificiellement enfler l'erreur relative avec la fonction *fittée*.

Remarquons néanmoins que deux de ces *fits* produisent des erreurs relatives très importantes. Il s'agit de $t_{\text{diode,off}}$ le temps de conduction inverse du transistor *High-Side* et de $E_{\text{diode,off}}$ les pertes associées à cette conduction. L'équation 4.58 ne peut décrire les pertes $E_{\text{diode,off}}$ d'un transistor, car elle n'est malheureusement pas GP-compatible à cause de la soustraction du temps de commutation au *deadtime*. Il a été tenté de contourner l'opération de soustraction en *fittant* directement $E_{\text{diode,off}}$ et $t_{\text{diode,off}}$ sur les données *LTSPICE*, pour en faire des fonctions GP "empiriques" dépendant de \tilde{V}_{ds} , \tilde{I}_{ds} , \tilde{t}_{dt} , $\tilde{R}_{g,SR}$ et $\tilde{R}_{g,SNK}$. Ces *fits* donnent malheureusement des erreurs relatives moyennes de 140% et 170%. Une comparaison graphique de ces deux *fits* avec les données *LTSPICE* est faite en Figure 4.3. Bien que les fonctions *fittées* soient de mauvaise qualité, elles décrivent tout de même correctement les tendances croissantes ou décroissantes de $E_{\text{diode,off}}$ pour chacun de ses paramètres. Nous utiliserons donc ces fonctions *fittées* dans notre modélisation des pertes des transistors, malgré leurs imperfections.

4.5.2 Implémentation des variables de Φ_{MOSFET}

Chacune des fonctions *fittées* listées en Table 4.3 est une fonctions substituable du *FuncStruct* Φ_{MOSFET} , ayant pour variables les paramètres de balayage listés ci-dessus. Notons néanmoins que certaines de ces variables ne seront pas injectées telle que dans le problème MDGP. Les variables $\tilde{R}_{g,SR}$, $\tilde{R}_{g,SNK}$ et \tilde{t}_{dt} seront des variables continues du problème. La température \tilde{T}_j sera elle-même

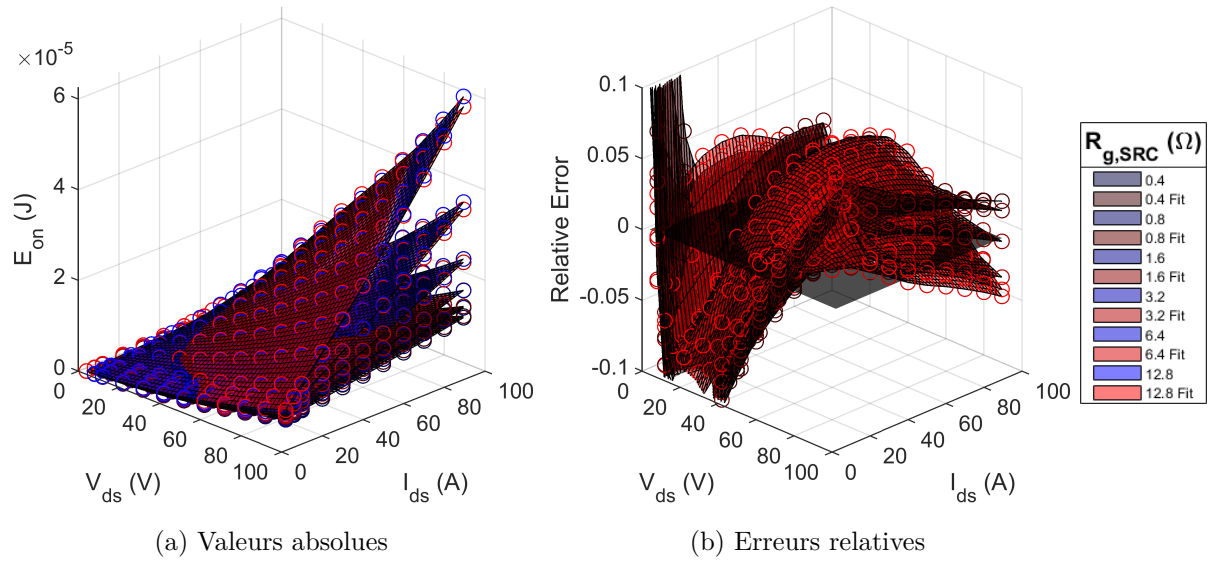


FIGURE 4.2 – *Fitting* de E_{on} l'énergie de commutation au *turn-on* du transistor *EPC2022* en fonction de \tilde{V}_{ds} , \tilde{I}_{ds} et $\tilde{R}_{g,SRC}$ (Ω), pour $\tilde{t}_{dt} = 30ns$ et $\tilde{R}_{g,SNK} = 0.4\Omega$. En bleu les résultats de simulation *LTSPICE*, en rouge le *fit*.

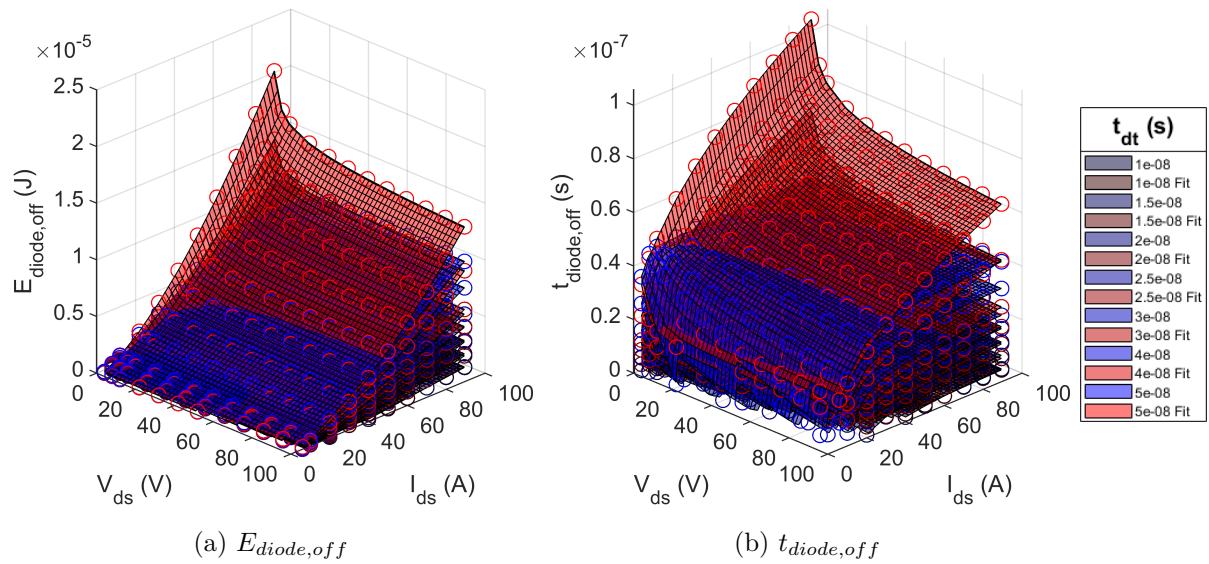


FIGURE 4.3 – *Fitting* de $E_{diode,off}$ et $t_{diode,off}$ le temps et l'énergie dissipée en conduction inverse du transistor *EPC2022* en fonction de \tilde{V}_{ds} , \tilde{I}_{ds} et \tilde{t}_{dt} , pour $\tilde{R}_{g,SRC} = 0.8\Omega$ et $\tilde{R}_{g,SNK} = 0.4\Omega$. En bleu les résultats de simulation *LTSPICE*, en rouge le *fit*.

une variable continue, mais avec une implémentation similaire à \tilde{T}_L que nous détaillerons plus bas. Enfin, \tilde{V}_{ds} et \tilde{I}_{ds} sont déterminées par la topologie et $\tilde{n}_{sw,para}$ le nombre de transistors en parallèle pour un interrupteur suivant les équations :

$$V_{ds} = \frac{V_{in}}{\mathring{N}_{cell}} \quad (4.62)$$

$$I_{ds} = \frac{I_{out}}{\mathring{n}_{sw,para} \cdot \mathring{n}_{phase}} \quad (4.63)$$

Nous créerons donc un *FuncStruct* $\Phi_{Topology}$ couplé au *Tuple* $\mathring{\Theta}_{Topology}$ déjà existant, dans lequel les équations 4.62 et 4.63 servent de fonctions substituables. Ce *FuncStruct* n'est pas fondamentalement nécessaire, il ne requière qu'une seule instance car les deux fonctions substituables ne changent pas d'une topologie à une autre (car nous n'étudions que les topologies FCML !), il suffirait de substituer les termes \tilde{V}_{ds} et \tilde{I}_{ds} directement à l'intérieur de Φ_{MOSFET} . Mais ce faisant, nous pouvons garder des expressions aussi génériques que possibles pour les fonctions substituables de Φ_{MOSFET} , ce qui permet de facilement réutiliser ce *FuncStruct* pour d'autres applications, comme une sorte de librairie GP de transistors.

4.5.3 Gestion de la température de jonction, dissipation thermique

Comme dans le cas de \tilde{T}_L la température de l'inductance, la température de jonction des transistors \tilde{T}_j est elle-même l'objet d'une boucle de rétroaction. Nous reprenons la méthode d'implémentation issue de [93] et décrite en section 4.4. Soit \tilde{T}_j une variable continue, la valeur $T_{j,calc}$ est calculée à partir des pertes du transistor, tel que :

$$T_{j,calc} = T_{amb} + (P_{sw} + P_{diode} + P_{cond}) \cdot R_{\theta_{sw,amb}} \quad (4.64)$$

Contraintes :

$$T_{j,calc} \leq \tilde{T}_j \quad (4.65)$$

$$T_{j,calc} \leq \mathring{T}_{j,max} \quad (4.66)$$

Où $\mathring{R}_{\theta_{sw,amb}}$ est la résistance thermique équivalente entre l'ensemble des pertes des transistors, comme décrite en section 2.4.2 où est présenté l'assemblage du dissipateur thermique sur notre prototype. Néanmoins, l'équation 2.27 n'est pas GP-compatible car elle comporte des divisions par des posynômes, ainsi qu'une soustraction dans l'expression de $R_{\theta_{b,hs}}$ la résistance thermique entre le PCB environnant le transistor et le dissipateur thermique (eq. 2.25). Malgré l'importance du chemin de conduction thermique à travers le PCB dans le refroidissement des transistors, démontrée en section 2.4.2, nous avons décidé de le négliger pour faciliter la formulation GP-compatible de $R_{\theta_{sw,amb}}$. Nous supposons donc que le flux thermique entre la jonction d'un transistor et l'air passe entièrement à travers le boîtier jusqu'au dissipateur thermique. Rappelons la résistance thermique de l'interface thermique entre le boîtier du transistor et le dissipateur thermique :

$$R_{\theta_{c,hs}} = \frac{1}{\kappa_{TIM}} \cdot \frac{e_{c,hs}}{\mathring{w}_{sw} \cdot \mathring{\ell}_{sw}} \quad (4.67)$$

Où κ_{TIM} est la conductivité thermique du matériau d'interface, $e_{c,hs}$ est l'épaisseur du matériau, et $\mathring{w}_{sw} \cdot \mathring{\ell}_{sw}$ est la surface du boîtier d'un transistor. L'équation 2.27 est réécrite avec une nouvelle formulation GP-compatible :

$$R_{\theta_{sw,amb}} = \frac{\mathring{n}_{sw/hs}}{2\mathring{N}_{cell} \cdot \mathring{n}_{sw,para} \cdot \mathring{n}_{phase}} \cdot \left(R_{\theta_{hs,amb}} + \frac{\mathring{R}_{\theta_{j,c}} + R_{\theta_{c,hs}}}{\mathring{n}_{sw/hs}} \right) \quad (4.68)$$

Où $R_{\theta_{hs,amb}}$ est la résistance thermique entre un unique dissipateur thermique et l'air. Pour le dissipateur *Wakefield-Vette 960-31-15-D-AB-0* sélectionné, $R_{\theta_{hs,amb}} = 2.25^\circ C/W$ pour un flux d'air de $4m/s$. $\dot{R}_{\theta_{j,c}}$ est la résistance thermique entre la jonction et le boîtier du transistor (variable discrète de $\dot{\Theta}_{MOSFET}$), $\dot{n}_{sw/hs}$ est le nombre de transistors couvert par un dissipateur, et $2\dot{N}_{cell} \cdot \dot{n}_{sw,para} \cdot \dot{n}_{phase}$ exprime le nombre de transistors sur l'ensemble du convertisseur. Il a été établi en section 1.8 qu'un dissipateur thermique peut couvrir jusqu'à 4 transistors. Il serait aussi possible de ne couvrir que 2 transistors par dissipateur pour réduire la résistance thermique, ou tout simplement car il n'y a que 2 transistors par PCB pour un convertisseur BUCK où le nombre de transistors en parallèle $\dot{n}_{sw,para} = 1$. Cette nouvelle variable discrète simple $\dot{n}_{sw/hs}$ peut donc prendre les valeurs 2 ou 4. L'algorithme MDGP sera en mesure de choisir la meilleure répartition de transistors par dissipateur (et donc le nombre de dissipateurs). Pour garantir la cohérence de la valeur de $\dot{n}_{sw/hs}$ dans le cas du BUCK à 2 transistors, la contrainte suivante est ajoutée :

Contrainte :

$$\dot{n}_{sw/hs} \leq 2\dot{N}_{cell} \cdot \dot{n}_{sw,para} \quad (4.69)$$

L'équation 4.64 permet donc de calculer la température de jonction d'un transistor. La variable continue \tilde{T}_j est, elle, employée comme variable dans R_{dsON} la fonction substituable de Φ_{MOSFET} donnant la résistance à l'état passant du transistor. Il faut impérativement que R_{dsON} soit strictement croissante suivant \tilde{T}_j . De cette manière, \tilde{T}_j est systématiquement minimisée (pour minimiser P_{cond} les pertes par conduction) et converge vers $T_{j,calc}$.

Sachant que la masse d'un unique dissipateur thermique *960-31-15-D-AB-0 Wakefield-Vette* est égale à $m_{hs,unit} = 19.5g$, soit m_{hs} la masse totale de l'ensemble des dissipateurs thermiques :

$$m_{hs} = m_{hs,unit} \cdot \frac{2\dot{N}_{cell} \cdot \dot{n}_{sw,para} \cdot \dot{n}_{phase}}{\dot{n}_{sw/hs}} \quad (4.70)$$

Sur notre prototype expérimental, le flux d'air de $4m/s$ traversant le dissipateur thermique est assuré par un ventilateur *San Ace 80 - 109R0824G4021* d'un volume $V_{fan,unit} = 80mm \times 80mm \times 25mm$ pour une masse $m_{fan,unit} = 100g$ et une consommation $P_{fan,unit} = 4.8W$. Le choix du ventilateur dans notre projet n'a pas fait l'objet d'une étude approfondie, nous proposons néanmoins de l'intégrer à notre problème MDGP sous une forme très simplifiée. Un ventilateur peut couvrir 4 dissipateurs thermiques. La masse m_{fan} , le volume V_{fan} et la consommation électrique P_{fan} totaux en fonction de n_{fan} le nombre de ventilateurs, sont exprimés sous une forme GP-compatible :

$$n_{fan} = \frac{2\dot{N}_{cell} \cdot \dot{n}_{sw,para} \cdot \dot{n}_{phase}}{4\dot{n}_{sw/hs}} \quad (4.71)$$

$$m_{fan} = n_{fan} \cdot m_{fan,unit} \quad (4.72)$$

$$V_{fan} = n_{fan} \cdot V_{fan,unit} \quad (4.73)$$

$$P_{fan} = n_{fan} \cdot P_{fan,unit} \quad (4.74)$$

4.5.4 Améliorer l'encadrement des variables de substitution

Comme nous avons pu l'expliquer en section 3.5, le comportement par défaut de l'algorithme MDGP dans sa gestion des fonctions substituables telles que celles du *FuncStruct* Φ_{MOSFET} est peu propice à faciliter l'élimination de solutions par l'algorithme *Branch & Bound*. Lors de l'exploration de l'arbre de décision du *Branch & Bound*, les fonctions substituables de Φ_{MOSFET} sont représentées par des variables de substitution \tilde{F}_i , et ces variables sont encadrées par les deux contraintes décrites en équation 3.30, Rappelées ci-après :

$$\begin{cases} \min \left(\left\{ k \in \{1, \dots, |\mathcal{F}_i|\} \mid \min(\mathbf{Im}(\mathring{F}_i(k))) \right\} \right) \leq \tilde{F}_i \\ \tilde{F}_i \leq \max \left(\left\{ k \in \{1, \dots, |\mathcal{F}_i|\} \mid \max(\mathbf{Im}(\mathring{F}_i(k))) \right\} \right) \end{cases}$$

\tilde{F}_i peut prendre n'importe quelle valeur atteignable par les instances $\mathring{F}_i(k)$ sur leurs domaines de définition respectifs. Cela signifie que, pour notre *FuncStruct* Φ_{MOSFET} , dont les domaines de définition sont les plages de fonctionnement entières des transistors, les variables continues de substitution \tilde{F}_i peuvent librement prendre des valeurs irréalistes, et produire par exemples des énergies de commutation E_{on} correspondant à une tension V_{ds} et un courant I_{ds} minimaux. Or, notre algorithme implémente une instance de Φ_{MOSFET} après avoir déjà implémenté une instance de $\Theta_{Topology}$. Cela signifie que, pour les sous-problèmes relaxés où $\Theta_{Topology}$ est implémenté mais Φ_{MOSFET} ne l'est pas encore, la Borne Inférieure calculée sera elle-même exagérément basse, ce qui diminue les chances pour l'algorithme *Branch & Bound* de pouvoir éliminer ce nœud et sa descendance en un seul calcul, par comparaison avec la meilleure valeur optimale connue $f_0(X^\dagger)$.

Il est toutefois possible, au prix de quelques calculs supplémentaires, d'améliorer la qualité de cet encadrement. Lors de la sélection d'une instance de $\Theta_{Topology}$, il est déjà possible de déduire les valeurs minimales et maximales possibles pour \tilde{I}_{ds} et \tilde{V}_{ds} , les deux variables les plus importantes des fonctions E_{on} , E_{off} et E_{rr} les pertes par commutation des transistors. Soit :

$$\frac{I_{out}}{\max(n_{sw,para}) \cdot \max(n_{phase})} \leq \tilde{I}_{ds} \quad (4.75)$$

$$\tilde{I}_{ds} \leq \min(\{I_{out}; \max(I_{ds,max})\}) \quad (4.76)$$

$$\tilde{V}_{ds} = \frac{V_{in}}{\mathring{N}_{cell}} \quad (4.77)$$

Pour chaque instance $\Phi_{MOSFET}(k)$, l'algorithme résout un problème d'optimisation prenant pour fonction objectif l'une des fonctions substituables qui nous intéressent, prenons par exemple $E_{on}(k)$, et pour contraintes les équations 4.75, 4.76 et 4.77. Si la tension V_{ds} de l'équation 4.77 excède la tension de claquage BV_{ds} d'un transistor, ce dernier n'est pas évalué. On note les résultats $E_{on}^\dagger(k)$. Il est alors possible d'en déduire les valeurs minimales et maximales admissibles pour la variable de substitution \tilde{E}_{on} , pour la topologie sélectionnée, quels que soient ultérieurement le transistor, le nombre de phases n_{phase} , ou les valeurs de résistance de grille retenus. Une nouvelle contrainte d'encadrement de la variable de substitution peut alors être ajoutée au sous-problème relaxé de notre algorithme MDGP :

$$\min(E_{on}^\dagger) \leq \tilde{E}_{on} \quad (4.78)$$

L'opération est répétée pour les autres fonctions substituables E_{off} , E_{rr} et R_{dsON} . Ce processus ajoute un certain nombre de calculs supplémentaires. Nous verrons en section 4.10 que ces calculs supplémentaires n'ajoutent que très peu de temps de calcul, car les problèmes GP d'encadrement sont continus et très simples, ils sont donc très rapidement résolus.

4.6 Modélisation de la structure du convertisseur : PCB, Busbars, assemblage

La géométrie du PCB d'une phase et de l'ensemble du convertisseur doit être prise en considération. Elle apparaît déjà dans les formulations GP exprimées ci-dessus à travers \dot{n}_{phase} le nombre de phases, c'est à dire de PCBs, mises en parallèle dans le convertisseur. Les dimensions envisagées pour un PCB ont été présentées en Figure 1.17. Soit la surface d'une phase :

$$S_{PCB} = S_{hs} \cdot \frac{2\dot{N}_{cell} \cdot \dot{n}_{sw,para}}{\dot{n}_{sw/hs}} + \frac{S_C}{2} + \dot{w}_L \cdot \dot{\ell}_L \cdot \dot{n}_{L,para} + S_{driver} \cdot \dot{N}_{cell} \quad (4.79)$$

Où $S_{driver} = 31mm \times 21mm$ comme établi en section 1.17. La surface des condensateurs S_C est divisée par 2 pour prendre en compte que les condensateurs sont équitablement répartis entre les faces *top* et *bottom* du PCB. Pour déterminer la masse totale des PCBs dans le convertisseur, la masse surfacique du PCB est établie :

$$\rho_{PCB,surf} = \dot{n}_{layer} \cdot \rho_{Cu} \cdot e_{layer} + \rho_{FRA}(e_{PCB} - \dot{n}_{layer} \cdot e_{layer}) \quad (4.80)$$

Où \dot{n}_{layer} est le nombre de couches de cuivre du PCB, e_{layer} l'épaisseur d'une couche, e_{PCB} l'épaisseur du PCB, $\rho_{Cu} = 8930kg/m^3$ et $\rho_{FRA} = 1850kg/m^3$ les masses volumiques du cuivre et du matériau FR4. Cette équation n'est pas GP-compatible, il serait sans doute possible de la reformuler à l'aide de variables discrètes couplées dans un *Tuple* $\dot{\Theta}_{PCB}$ dont chaque instance correspondrait à une structure interne de PCB standard sur le marché. Dans le cadre de ces travaux, nous supposons simplement que le PCB est construit de la même manière que notre prototype. Soit $n_{layer} = 6$, $e_{layer} = 70\mu m$ et $e_{PCB} = 1mm$. Ainsi, $\rho_{PCB,surf}$ devient une constante permettant d'exprimer la masse totale des PCBs sous une forme GP-compatible :

$$m_{PCB} = S_{PCB} \cdot \rho_{PCB,surf} \cdot \dot{n}_{phase} \quad (4.81)$$

Chacun de ces PCB sont connectés en parallèle à l'aide de 4 busbars, au potentiels V_{in} , V_{out} et $0V$, comme représenté en Figure 2.19. Ces busbars vont ajouter des pertes joules, car jusqu'à 715A de courant les traversent, et de la masse au convertisseur. Nous souhaitons optimiser le choix du matériau conducteur du busbar entre *cuivre* et *aluminium*. Un *Tuple* $\dot{\Theta}_{Busbar}$ est donc créé, ses deux variables discrètes couplées sont $\dot{\rho}_{busbar}$ la masse volumique, et \dot{r}_{busbar} la résistance électrique spécifique du matériau.

TABLE 4.4 – Caractéristiques des matériaux considérés pour les busbars, dans le *Tuple* Θ_{busbar}

Matériau	$\hat{\rho}_{busbar}$	\hat{r}_{busbar}
Cuivre	$8920kg/m^3$	$1.7 \cdot 10^{-8}\Omega \cdot m$
Aluminium	$2700kg/m^3$	$2.8 \cdot 10^{-8}\Omega \cdot m$

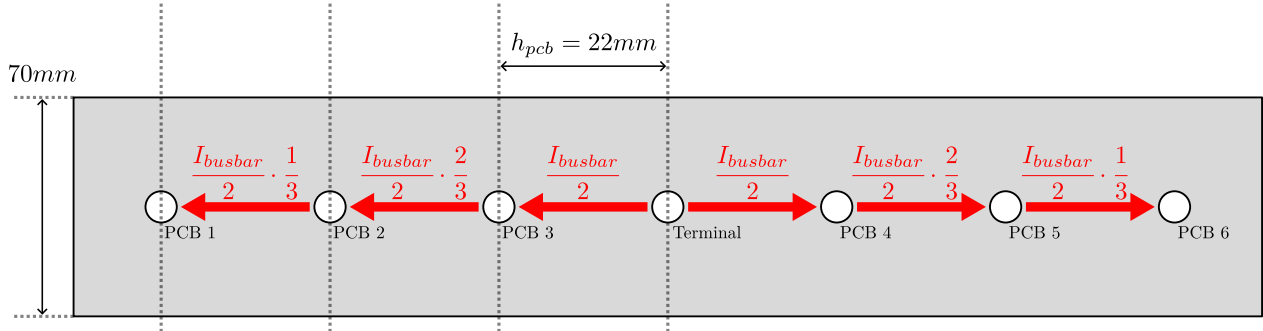


FIGURE 4.4 – Représentation schématique d'un busbar pour $n_{phase} = 6$

La géométrie d'un busbar ainsi que la répartition du courant de charge le traversant, sont représentées en Figure 4.4. La longueur du busbar varie avec le nombre de phases. Sa largeur est arbitrairement fixée à $70mm$, mais \tilde{e}_{busbar} l'épaisseur du busbar est implémentée comme une variable continue qui permet de trouver un compromis entre conductivité et masse du busbar. La section du busbar est donc :

$$s_{busbar} = \tilde{e}_{busbar} \times 70mm \quad (4.82)$$

Deux contraintes d'encadrement sont ajoutées, tel que $1mm \leq \tilde{e}_{busbar} \leq 5mm$ pour conserver des valeurs réalistes. Il serait là aussi possible d'implémenter un choix entre plusieurs standards d'épaisseurs de busbars via une variable discrète ou un *Tuple*, si nous avions souhaité développer cet aspect du problème d'optimisation. Soit la masse totale des busbars :

$$m_{busbar} = 4s_{busbar} \cdot h_{PCB} \cdot \hat{n}_{phase} \cdot \hat{\rho}_{busbar} \quad (4.83)$$

Où h_{PCB} est l'espace entre deux PCB dans l'assemblage du convertisseur. Nous avons fixé $h_{PCB} = 22mm$ à partir des dimensions de notre prototype expérimental, mais il serait possible de laisser l'algorithme MDGP calculer cet espacement à partir des épaisseurs de l'inductance ou du dissipateur thermique. Nous souhaitons calculer les pertes par conduction du busbar P_{busbar} , sachant que le courant de charge n'est pas réparti de manière homogène le long d'un busbar, ni de manière équitable entre les busbars en fonction du rapport cyclique. Soit :

$$P_{busbar} = R_{busbar} \cdot \left(I_{out}^2 + (D \cdot I_{out})^2 + \frac{((1-D) \cdot I_{out})^2}{2} \right) \quad (4.84)$$

Où R_{busbar} est la résistance équivalente d'un busbar. Celle-ci dépend de la répartition du courant à l'intérieur du conducteur, telle que décrite en Figure 4.4. Le terminal de connexion du busbar au réseau électrique embarqué de l'avion ou à la Pile à Combustible est situé à la moitié de la longueur du conducteur. Le courant I_{busbar} parcourant le busbar se divise en deux à cette intersection. Tous les h_{PCB} est connectée une phase du convertisseur qui tire un courant $I_{busbar}/\hat{n}_{phase}$, qui est soustrait à I_{busbar} . Soit $n = \hat{n}_{phase}/2$ le nombre de phases sur un demi-busbar. Les pertes joules pour un busbar i en fonction de I_{busbar} peuvent être exprimées de la manière suivante :

$$P_{busbar,i}(I_{busbar}) = \frac{2\dot{\rho}_{busbar} \cdot h_{PCB}}{s_{busbar}} \left(\left(\frac{I_{busbar}}{2} \right)^2 + \left(\frac{I_{busbar}}{2} \cdot \frac{n-1}{n} \right)^2 + \dots + \left(\frac{I_{busbar}}{2} \cdot \frac{1}{n} \right)^2 \right) \quad (4.85)$$

Ce qui n'est pas une formulation GP-compatible à cause des soustractions. Il est toutefois possible de la réécrire sous forme d'une série, et de la simplifier :

$$P_{busbar,i}(I_{busbar}) = \frac{2\dot{\rho}_{busbar} \cdot h_{PCB}}{s_{busbar}} \cdot \frac{I_{busbar}^2}{4n^2} \cdot \sum_{i=1}^n i^2 \quad (4.86)$$

$$P_{busbar,i}(I_{busbar}) = \frac{2\dot{\rho}_{busbar} \cdot h_{PCB}}{s_{busbar}} \cdot \frac{I_{busbar}^2}{4n^2} \cdot \frac{n(n+1)(2n+1)}{6} \quad (4.87)$$

Nous pouvons en déduire, pour \hat{n}_{phase} une expression GP-compatible de R_{busbar} injectable dans l'équation 4.84 :

$$R_{busbar} = \frac{\dot{\rho}_{busbar} \cdot h_{PCB}}{s_{busbar}} \cdot \frac{\left(\frac{\hat{n}_{phase}}{2} + 1 \right) (\hat{n}_{phase} + 1)}{6\hat{n}_{phase}} \quad (4.88)$$

Les deux objectifs de notre problème d'optimisation, P_{loss} et m_{conv} (équations 4.3 et 4.4), ainsi que V_{conv} le volume total du convertisseur (requis pour imposer la contrainte $V_{conv} \leq 15L$), peuvent désormais être explicités :

$$P_{loss} = P_{Cin} + P_{Cfly} + P_{Cout} + P_L + P_{sw} + P_{cond} + P_{diode} + P_{gate} + P_{busbar} + P_{fan} \quad (4.89)$$

$$m_{conv} = m_C + m_L + m_{hs} + m_{fan} + m_{PCB} + m_{busbar} \quad (4.90)$$

$$V_{conv} = (S_{PCB} + 4s_{busbar}) \cdot h_{PCB} \cdot \hat{n}_{phase} + V_{fan} \quad (4.91)$$

4.7 Optimisation pour plusieurs points de fonctionnement

Le problème MDGP ci-dessus n'a été formulé que pour un unique point de fonctionnement du convertisseur. En lançant le processus d'optimisation sous cette forme, la solution engendrée serait optimale pour cet unique point de fonctionnement mais risquerait de présenter de mauvaises

performances sur le reste de la plage de fonctionnement du convertisseur. Notre convertisseur doit être pensé pour une plage de fonctionnement très large, la tension d'entrée V_{in} peut varier de 35V à 110V et la puissance P_{in} de 1kW à 20kW.

Pour prendre en compte cette variabilité des points de fonctionnement, nous avons décidé de reformuler le problème MDGP pour qu'il puisse étudier plusieurs points de fonctionnement simultanément. Pour ce faire, nous avons simplement démultiplié toutes les variables continues et discrètes, les *Tuples* et *FuncStruct* du problème dont les valeurs et instances seraient spécifiques à un point de fonctionnement donné, listés ci-après :

- $\Theta_{Topology} \rightarrow \mathring{i}_D, \mathring{R}_i, \mathring{X}_i, \mathring{D}_e, \mathring{c}_{V_{in}/2}, \mathring{c}_{V_{fly},i}$
- \mathring{T}_L
- $\Phi_{MOSFET} \rightarrow$ Toutes les fonctions substituables et leurs variables $\mathring{V}_{ds}, \mathring{I}_{ds}$ et \mathring{T}_j
- \mathring{n}_{phase}

Chacune des équations de ce Chapitre comportant l'une des variables ci-dessus est donc elle aussi démultipliée pour chaque point de fonctionnement. La démultiplication de \mathring{n}_{phase} permet à l'algorithme de sélectionner un $\mathring{n}_{phase,i}$ différent pour chaque point de fonctionnement, ce qui revient à autoriser le contrôle du convertisseur en *phase-shedding*. Il devient possible pour l'algorithme de prendre en compte la possibilité d'allumer ou éteindre des phases du convertisseur pour en optimiser le rendement. Néanmoins, cette démultiplication n'est appliquée que pour les équations décrivant des propriétés électriques (pertes, ondulations de courant...). Pour les équations liées au volume et à la masse du convertisseur, comme le nombre total de phases reste invariable, nous remplacerons \mathring{n}_{phase} par l'expression GP-compatible $max(\mathring{n}_{phase})$. Les équations concernées sont : 4.16, 4.47, 4.70, 4.72, 4.73, 4.81, 4.83 et 4.91.

Pour n_{point} le nombre de points de fonctionnement étudiés, sont donc créées autant d'équations des pertes $P_{loss,i}$. Soit une démultiplication de la fonction objectif du rendement η_{loss} (équation 4.3) en $\eta_{loss,i}$ pour chaque point de fonctionnement. Afin de pondérer l'importance des différents points de fonctionnement, un coefficient γ_i est assigné à chacun, puis la somme de ces fonctions est optimisée :

$$\eta_{loss,i} = \frac{P_{loss,i}}{P_{in,i}} \quad i \in \{1; \dots; n_{point}\} \quad (4.92)$$

$$\eta_{loss} = \sum_{i=1}^{n_{point}} \gamma_i \cdot \eta_{loss,i} \quad (4.93)$$

La démultiplication des variables, et donc des équations associées, résulte en une complexification importante du problème. Pour des raisons de temps de calcul, nous n'avons retenu que 3 points de fonctionnement, listés avec leurs coefficients γ_i en Table 4.5.

TABLE 4.5 – Points de Fonctionnement étudiés dans notre problème MDGP

Point	$V_{in,i}$	$P_{in,i}$	γ_i
1	80V	15kW	0.625
2	80V	20kW	0.3125
3	110V	1kW	0.0625

Nous nous référerons aux caractéristiques puissance-tension de la Pile à Combustible décrites en Figure 2 (Introduction). Le point 1 est un point de fonctionnement à une puissance moyenne, pour une tension V_{in} de fin de rodage de la Pile à Combustible, le coefficient γ_i le plus élevé lui est assigné pour favoriser son optimisation. Le point 2 correspond au point critique de puissance maximale en gabarit max. Enfin, le point 3 est le point de tension V_{in} maximale à courant I_{out} minimal et présente un intérêt spécifiquement pour assurer le bon comportement en commutation des transistors. Notamment, comme le *deadtime* \tilde{t}_{dt} et les résistances de grille $\tilde{R}_{g, SRC}$ et $\tilde{R}_{g, SNK}$ sont partagés entre tous les points de fonctionnement, c'est le point 3 qui va particulièrement contraindre ces variables car il devrait présenter les temps de blocage $t_{sw, off}$ les plus longs. Son γ_i est néanmoins fixé à une faible valeur car le point ne présente que peu d'intérêt dans l'optimisation de η_{loss} .

4.8 Optimisation Multi-Objectifs

Un problème MDGP est doté de n_{obj} sous-fonctions objectifs $f_{obj,i}$ (dans nos travaux : η_{loss} et M_{conv}), contenues dans la fonction objectif f_0 . Ces sous-fonctions sont pondérées par des coefficients w_i qui servent à représenter l'importance à accorder à chacune de ces sous-fonctions dans le processus d'optimisation. Les sous-fonctions objectif n'ont pas le même Ensemble Image :

$$\sum_{i=1}^{n_{obj}} w_i = 1, \quad 0 \leq w_i \leq 1 \quad (4.94) \quad \mathbf{Im}(f_{obj,i}) \neq \mathbf{Im}(f_{obj,j}), \quad i \neq j \quad (4.95)$$

Nous souhaitons normaliser ces n_{obj} fonctions à l'aide de coefficients λ_i . Soit la fonction objectif f_0 :

$$f_0 = \sum_{i=1}^{n_{obj}} \frac{w_i}{\lambda_i} f_{obj,i} \quad (4.96)$$

Pour déterminer la valeur de ces coefficients de normalisation λ_i , nous emploierons la méthode de normalisation décrite par [104]. Pour chaque sous-fonction objectif $f_{obj,i}$, un problème MDGP idéalisé est créé à partir du problème MDGP complet, pour lequel est assigné un unique $f_{obj,i}$ comme fonction objectif. Cela signifie que seul $f_{obj,i}$ sera optimisé pour une solution optimale X_i^* , et les valeurs $f_{obj,j}(X_i^*)$, $j \neq i$ seront sous-optimales. Après avoir résolu les n_{obj} problèmes MDGP, une matrice contenant les valeurs de chaque sous-fonction objectif est produite :

$$\begin{matrix} & f_1 & \cdots & f_{n_{obj}} \\ X_1^* & \left(\begin{array}{ccc} f_1(X_1^*) & \cdots & f_{n_{obj}}(X_1^*) \\ \vdots & \ddots & \vdots \\ f_1(X_{n_{obj}}^*) & \cdots & f_{n_{obj}}(X_{n_{obj}}^*) \end{array} \right) \\ \vdots & & & \\ X_{n_{obj}}^* & & & \end{matrix} \quad (4.97)$$

Pour chaque sous-fonction objectif $f_{obj,i}$, sa borne inférieure L_i et sa borne supérieure U_i sont déterminées :

$$L_i = f_{obj,i}(X_i^*) \quad (4.98)$$

$$U_i = \max \left(\left\{ f_i(X_j^*) \mid i \neq j \right\} \right) \quad (4.99)$$

A l'aide des bornes inférieures et supérieures, il serait possible de normaliser les $f_{obj,i}$ tel que :

$$0 \leq \frac{f_{obj,i}(X) - L_i}{U_i - L_i} \leq 1 \quad (4.100)$$

$$f_0 = \sum_{i=1}^{n_{obj}} w_i \cdot \frac{f_{obj,i}(X) - L_i}{U_i - L_i} \quad (4.101)$$

$$= \sum_{i=1}^{n_{obj}} \left(w_i \cdot \frac{f_{obj,i}(X)}{U_i - L_i} \right) - \sum_{i=1}^{n_{obj}} \left(w_i \cdot \frac{L_i}{U_i - L_i} \right) \quad (4.102)$$

Cette formulation n'est pas GP-compatible à cause de la soustraction par $f_{obj,i}(X) - L_i$. Le terme $U_i - L_i$ peut quant à lui être considéré comme une constante à part entière, calculée avant injection dans le solveur GP, et ne pose donc pas de problème. Néanmoins, il faut noter que les termes $w_i \cdot \frac{L_i}{U_i - L_i}$ sont eux-mêmes des constantes. Ce qui implique que minimiser la fonction décrite par l'équation 4.101 revient à minimiser :

$$f_0 = \sum_{i=1}^{n_{obj}} w_i \cdot \frac{f_{obj,i}(X)}{\lambda_i} \quad (4.103) \quad \lambda_i = U_i - L_i \quad (4.104)$$

La solution optimale X^* est la même pour ces deux fonctions objectif. Nous emploierons donc la formulation de f_0 de l'équation 4.103 pour notre problème multi-objectif.

4.9 Présentation des Résultats

Afin de produire un front de Pareto des solutions optimales pour nos deux objectifs η_{loss} et M_{conv} , 13 résolutions du problème sont réalisées pour des valeurs des pondérations w_i différentes, en plus des 2 résolutions de normalisation des sous-fonctions objectif. Les pondérations sélectionnées, la valeur optimale $f_0(X^*)$, le rendement moyen pondéré, la masse, la topologie et le transistor de chacune des solutions optimales du front de Pareto sont listés en Table 4.6. Si nous avons employé les grandeurs M_{conv} et η_{loss} adaptées à notre formulation GP, nous préférons désormais, par lisibilité, présenter ces mêmes grandeurs sous leur forme m_{conv} et $\eta_{avg} = 1 - \eta_{loss}$ le rendement moyen pondéré. Le détail des pertes et de la masse de ces solutions optimales est présenté en Figures 4.5 (pour chaque point de fonctionnement) et 4.6. Les données "Prototype" correspondent aux performances calculées à partir de notre formulation MDGP pour les choix de design du Chapitre 1 qui ont mené à la réalisation du Prototype *Adjacent* testé expérimentalement en Chapitre 2. Il faut néanmoins prendre note que ce rendement théorique de notre prototype est bien supérieur à son rendement expérimental (réel). Son rendement moyen pondéré est $\eta_{avg,MDGP} = 98.5048\%$ contre $\eta_{avg,mesure} = 98.09\%$, soit $+0.41\%$ d'erreur absolue entre ces deux rendements, que nous

attribuons principalement à une mauvaise modélisation des pertes des inductances *Vishay* (cf. section 2.4.3.5). Enfin, la programmation de notre algorithme de résolution MDGP nous autorise à sauvegarder toutes les solutions non-optimales rencontrées par le processus de *Branch & Bound*, ce qui nous permet de pouvoir étudier des solutions quasi-optimales (comme alternatives aux solutions optimales), sans avoir pour autant parcouru l'entièreté des combinaisons possibles comme le ferait un algorithme *Brute Force*. Néanmoins, ces solutions sous-optimales ne sont pas exhaustives. Elles peuvent permettre de deviner des tendances, des influences de différentes variables sur leurs performances, sans pour autant pouvoir en tirer de conclusions définitives. Le rendement moyen pondéré η_{avg} et la masse m_{conv} de l'ensemble des solutions du Front de Pareto, ainsi que de toutes les solutions sous-optimales rencontrées pendant l'exploration de l'arborescence *Branch & Bound*, sont présentées en Figure 4.7. Afin de visualiser les meilleures solutions rencontrées employant uniquement les transistors disponibles en 2020, et les meilleures solutions n'employant que les transistors *EPC2022*, des pseudo-fronts de Pareto ont été tracés.

TABLE 4.6 – Rendement moyen pondéré η_{avg} , masse m_{conv} , niveau de la topologie N et transistors des solutions optimales sur le Front de Pareto

	Pondérations		$f_0(\mathbf{X}^*)$	η_{avg}	m_{conv}	N	Transistor	n_{phase}	n_{para}
	w_1	w_2							
1	0.99	0.01	1.192	99.0760%	5kg	4	EPC2302	20	1
2	$10/11$	$1/11$	1.2116	99.0707%	4.7156kg	4	EPC2302	20	1
3	$9/11$	$2/11$	1.2048	99.0332%	3.5468kg	4	EPC2302	12	2
4	$8/11$	$3/11$	1.18	99.0024%	3.1174kg	4	EPC2302	11	2
5	$7/11$	$4/11$	1.1413	98.9634%	2.7760kg	4	EPC2302	10	2
6	$6/11$	$5/11$	1.092	98.9566%	2.7324kg	4	EPC2302	10	2
7	0.5	0.5	1.0622	98.7420%	1.7450kg	3	EPC2302	15	1
8	$5/11$	$6/11$	1.0112	98.7356%	1.7188kg	3	EPC2302	15	1
9	$4/11$	$7/11$	0.90809	98.7325%	1.7093kg	3	EPC2302	15	1
10	$3/11$	$8/11$	0.80442	98.7294%	1.7028kg	3	EPC2302	15	1
11	$2/11$	$9/11$	0.70034	98.7267%	1.6993kg	3	EPC2302	15	1
12	$1/11$	$10/11$	0.5957	98.7210%	1.6955kg	3	EPC2302	15	1
13	0.01	0.99	0.46431	98.2994%	1.5443kg	4	EPC2302	6	1
	Modélisation Prototype			98.5048%	2.3884kg	3	EPC2022	10	2
	Expérimentation Prototype			98.09%	2.576kg	3	EPC2022	10	2

Nous constatons que l'ensemble des solutions optimales sur le front de Pareto utilisent comme transistor le composant *EPC2302*. Il s'agit du transistor ayant la résistance à l'état passant la plus faible ($R_{dsON} = 1.8m\Omega$) pour les composants aux tensions de claquages $BV_{ds} \approx 100V$. Les composants de qualibre $BV_{ds} = 40V$ (*EPC2024* et *EPC2066*) présentent un $R_{dsON} \approx 1.15m\Omega$ plus faible. Néanmoins, il apparaît en Figure 4.7 que les meilleures solutions rencontrées employant l'*EPC2066* présentent des rendements η_{avg} très sous-optimaux ($\approx -0.2\%$ par rapport au front de Pareto, à masse constante) pour les solutions ayant une masse totale $m_{conv} \leq 2.5kg$. Pour $m_{conv} \geq 3kg$, il apparaît que l'*EPC2066* est l'un des transistors les plus compétitifs, sans pour autant détrôner l'*EPC2302*, ni même l'*EPC2020*.

Globalement, pour les solutions proches du front de pareto, l'augmentation du rendement η_{avg} au détriment de la masse m_{conv} s'obtient par une augmentation du niveau N de la topologie du convertisseur. Cette augmentation de N est elle-même limitée par la masse maximale admissible du convertisseur, qui est fixée à $4kW/kg \rightarrow 5kg$. Cette tendance peut être observée sur le front

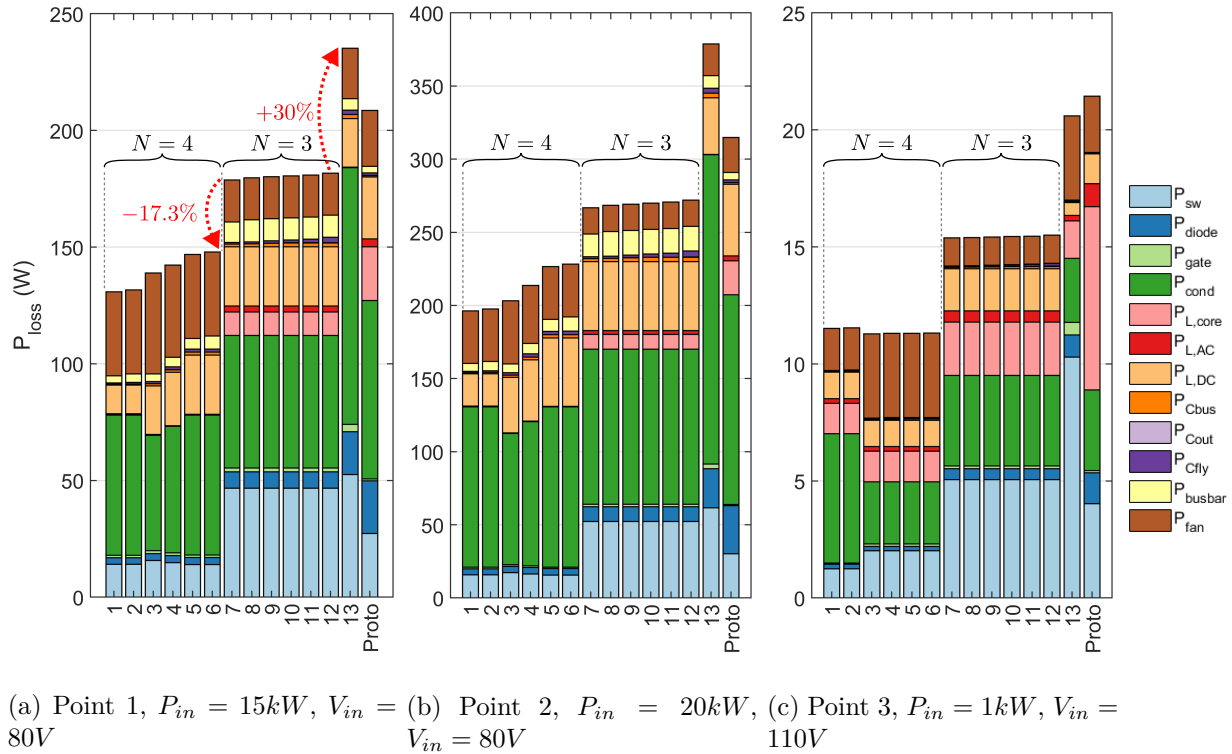


FIGURE 4.5 – Répartition des différentes sources de pertes pour chaque solution optimale et pour le Prototype, aux 3 points de fonctionnement évalués

de Pareto, où les solutions optimales pour les 13 pondérations w_i se divisent en deux groupes. Les solutions optimales où $N = 4$ (points $n^{\circ}1 - 6$) présentent de meilleurs rendements, et les solutions $N = 3$ (points $n^{\circ}7 - 12$) de meilleures masses (avec une solution $N = 4$ pour un coefficient de pondération très faible de l'optimisation du rendement). Le groupe de solutions optimales $N = 3$ est très resserré, et l'on note une augmentation de la masse m_{conv} de $+56.6\%$ entre la solution optimale $N = 3$ la plus lourde et la solution $N = 4$ la plus légère (hors point $n^{\circ}13$) pour une réduction de -17.3% des pertes totales, ce qui semble indiquer que m_{conv} est particulièrement sensible au niveau de la topologie N .

Une analyse des différentes sources des pertes et masses, représentées en Figures 4.5 et 4.6, permet de comprendre les “stratégies” adoptées par l'algorithme d'optimisation pour minimiser l'une ou l'autre de ces grandeurs. Rappelons que les deux groupes utilisent le même transistor *EPC2302*. Néanmoins, le groupe $N = 4$ présente une réduction d'environ -70% de pertes par commutation P_{sw} par rapport au groupe $N = 3$ pour des pertes par conduction relativement identiques à $P_{cond} \approx 60W$, au point de fonctionnement 1 (Fig. 4.5a). Les solutions du groupe $N = 4$ bénéficient d'une tension V_{ds} inférieure aux bornes de leurs transistors, ainsi que d'un courant drain-source I_{ds} moins élevé grâce à l'utilisation de 2 transistors en parallèle par interrupteur ($n_{sw,para} = 2$), ce qui permet de réduire drastiquement ces pertes P_{sw} au prix d'une élévation de la masse totale des PCB et du système de dissipation thermique. Étonnamment, le nombre de phases en parallèle n_{phase} est minimisé à ≈ 10 . Seules les deux solutions les plus biaisées en faveur du rendement (1 et 2), au contraire, maximisent n_{phase} à 20 afin de diminuer les pertes par conduction P_{cond} , tout en n'employant qu'un unique transistor par interrupteur pour limiter les pertes par commutation P_{sw} .

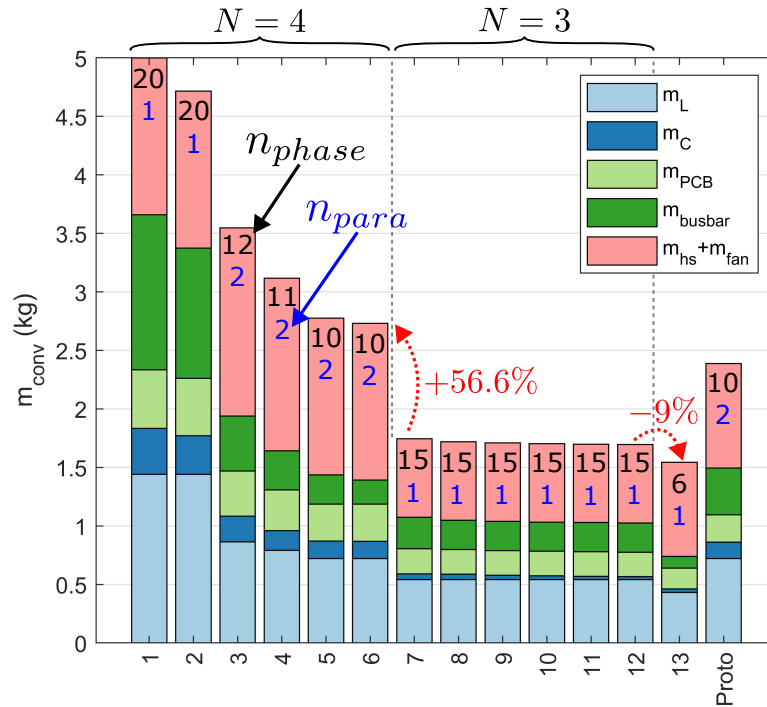


FIGURE 4.6 – Répartition des différentes sources de masse pour chaque solution optimale et pour le Prototype

Elles emploient toutes, par ailleurs, deux inductances $IHLP$ de $4.7\mu H$ en parallèle comme filtre de sortie. Le Front de Pareto passe brutalement de 20 phases à 12 sans valeurs intermédiaires sans doute car, pour $n_{sw,para} = 2$, des valeurs $n_{phase} > 12$ pénalisent trop la masse et augmentent davantage P_{sw} qu'elles ne diminuent P_{cond} .

Inversement, il apparaît que le groupe de solutions optimales $N = 3$ n'utilise qu'un seul transistor par interrupteur ($n_{w,para} = 1$) mais augmente n_{phase} à 15 pour maintenir le courant I_{ds} à une valeur raisonnable. Cette configuration permet de grandement réduire le nombre de dissipateurs thermiques et de ventilateurs requis, et donc leur masse. Le courant de charge est filtré par une unique inductance de $2.2\mu H$ par phase, expliquant ses plus grandes pertes joules $P_{L,DC}$ que pour le groupe $N = 4$. Les pertes fer $P_{L,core}$ sont plus importantes à cause d'une tension de point milieu V_{pm} plus grande pour $N = 3$, et d'un rapport cyclique effectif D_e plus élevé ($N = 3 \rightarrow D_e = 0.21$, contre $N = 4 \rightarrow D_e = 0.0475$, pour $D = 28/80$). L'ensemble des solutions de ce groupe, à l'exception de la solution 13 la plus biaisée en faveur de la puissance spécifique, présentent toutes la même masse $m_{conv} \approx 1.7kg$. Ce phénomène d'agglomération des solutions Pareto autour d'une unique configuration, particulièrement visible sur la Figure 4.7, montre l'impact des variables discrètes du problème MDGP. Dans un problème continu, les solutions optimales se répartissent de manière homogène le long du Front de Pareto. Mais dans le cadre d'un problème mixte discret, les variables discrètes ne peuvent pas être ajustées finement, et l'impact d'une variation de $\hat{y}_i(k)$ à $\hat{y}_i(k+1)$ sur les fonctions objectifs peut être très important. Ainsi, nous pouvons observer entre les solutions 12 et 13 qu'une réduction de $150g$ (-9%) de m_{conv} la masse du convertisseur, est obtenue au prix d'une augmentation des pertes totales de $55W$ ($+30\%$) au point de fonctionnement 1. L'étude du Front de Pareto illustre donc à quel point la résolution de problèmes Mixtes-Discrets peut être complexe, produisant des solutions parfois contre-intuitives, et s'agglomérant autour de valeurs discrètes spécifiques.

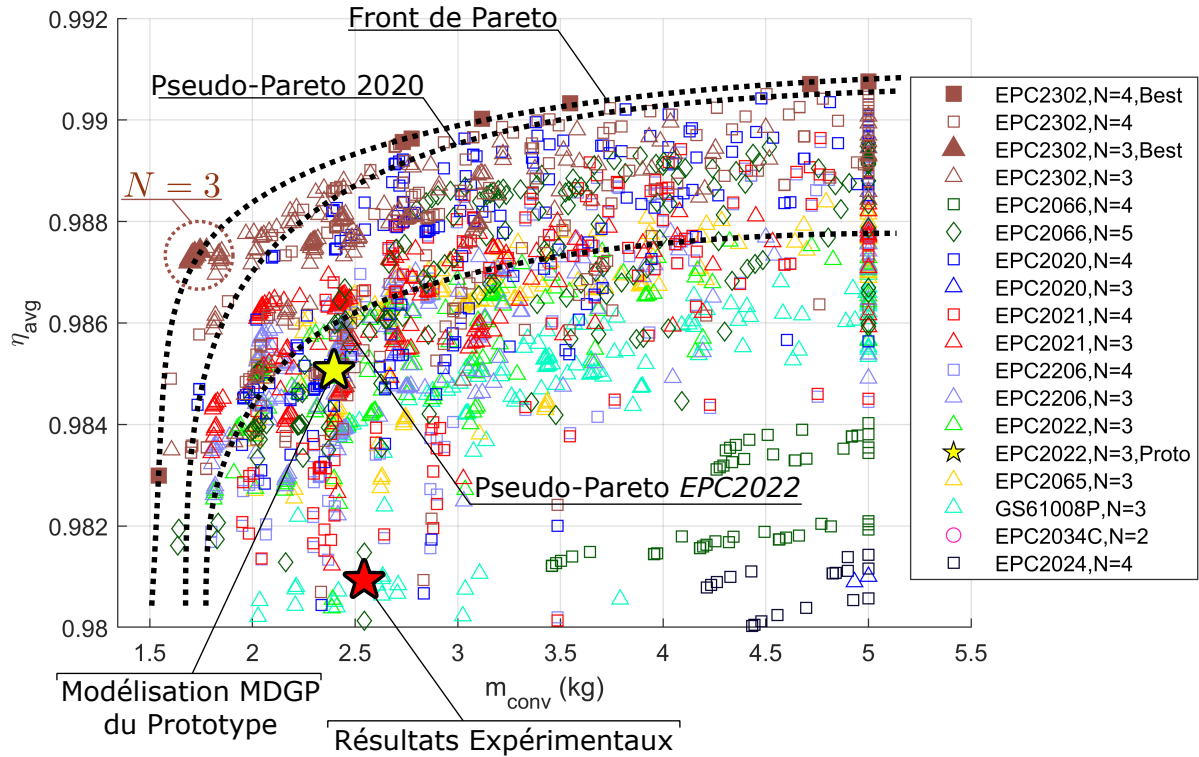


FIGURE 4.7 – Rendement moyen pondéré η_{avg} en fonction de la masse m_{conv} du Front de Pareto et des solutions sous-optimales rencontrées les plus performantes, groupées par transistor et niveau N de la topologie (zoomé pour $\eta_{avg} \geq 98\%$)

Le prototype développé pendant ces travaux de thèse n'est pas optimal, comme nous pouvons l'observer en Figure 4.7 (symbole \star pour la modélisation MDGP, et \star pour les résultats expérimentaux). Même en prenant en compte que ce convertisseur a été conçu en 2020, avant la commercialisation de transistors plus performants (notamment l'*EPC2302*), nous remarquons que le transistor *EPC2022* sélectionné n'était pas le plus adapté de son époque. Le transistor *EPC2020*, lui aussi commercialisé depuis 2016, était objectivement un bien meilleur choix. Il nous aurait permis, à masse m_{conv} égale, d'augmenter le rendement moyen pondéré η_{avg} d'une valeur absolue de 0.34%. Mais il n'a pas été évalué lors de l'étude réalisée au Chapitre 1, car un seul transistor par niveau de conversion N n'a été retenu afin de ne pas passer un temps excessif à cette tâche. Sa tension de claquage $BV_{ds} = 60V$ était trop basse pour un convertisseur FC3L ($N = 3$), et la plus faible résistance R_{dsON} du transistor *EPC2024* nous a laissé croire que ce dernier serait plus adapté pour un convertisseur $N = 4$. En comparant les performances de notre prototype à celles de toutes les solutions rencontrées employant l'*EPC2022* dans un convertisseur FC3L, il apparaît qu'il n'est pas non plus le meilleur de sa catégorie. Plutôt que d'utiliser deux inductances de $4.7\mu H$ en parallèle ($I_{sat} \times 2 = 74A$) pour un nombre de phases $n_{phase} = 10$ ($max(I_L) = 71A$), il aurait été préférable de sélectionner une seule inductance de $2.2\mu H$ ($I_{sat} = 48A$) et fixer $n_{phase} = 15$ ($max(I_L) = 47.3A$). Cela aurait permis une diminution de $\approx -50\%$ des pertes fer $P_{L,core}$ et de $\approx -25\%$ de m_L la masse totale des inductances du convertisseur.

4.10 Nombre d'opérations et Temps de calculs

L'algorithme MDGP a été développé pour conserver la garantie de trouver, s'il existe, l'optimum global du problème, tout en minimisant autant que possible le temps de calcul. Notre problème est doté de 24 variables continues et 6 variables discrètes simples, ainsi que de 3 *Tuples* et 2 *FuncStruct* regroupant 137 variables discrètes couplées et fonctions substituables, soit un total de 167 variables de tous genres. Les temps de calculs présentés ci-dessous ont été obtenus sur un ordinateur équipé d'un processeur *i7-1165G7* doté de 4 cœurs (8 threads) et d'une fréquence nominale de $2.4GHz$, de 16Go de RAM et d'un SSD. Le solveur GP est lui-même extrêmement rapide. Il a été évalué que, pour résoudre un sous-problème de l'arborescence *Branch & Bound*, le temps de calcul du solveur GP était de 0.486s. Néanmoins, une résolution du problème par une méthode *Brute Force + GP*, c'est à dire en résolvant les sous-problèmes GP pour chaque combinaison possible de variables discrètes, serait impossible à résoudre. En prenant en compte que $\Theta_{Topology}$ est couplé avec $\Phi_{Topology}$, à partir du nombre d'instances de chacune de ces variables discrètes, il existe 161.28 millions de combinaisons possibles, soit plus de 2 milliards de calculs pour générer le front de Pareto, en comptant les deux résolutions MDGP de normalisation des sous-fonctions objectif, et les 13 combinaisons de pondérations w_i . Ainsi, et en supposant qu'une machine puisse garder en mémoire 2 millions de résultats, le temps de calcul d'une résolution *Brute Force + GP* du front de Pareto serait de 32 ans.

Le temps de résolution de notre algorithme MDGP est, quant à lui, de $25h$, soit $8.83 \cdot 10^{-3}\%$ du temps du *Brute Force*. Au total, 128810 sous-problèmes GP ont été résolus. La comparaison des deux méthodes est résumée en Table 4.7. Bien que ce temps de calcul de $25h$ soit très excessif par rapport aux 0.486s de résolution GP, il s'agit néanmoins d'un temps réaliste, avec lequel un concepteur peut travailler. Sur ces $25h$ de calcul, le temps consacré à calculer les valeurs d'encadrement des variables de substitution \tilde{E}_{on} , \tilde{E}_{off} , \tilde{E}_{rr} et R_{dsON} , tel que décrit en section 4.5.4, est de seulement 16.7s. Ces encadrements ont permis d'améliorer la qualité des Bornes Inférieures aux nœuds d'implémentation des instances de $\Theta_{Topology}$. Ainsi, l'algorithme *Branch & Bound* a été capable d'éliminer, sans les calculer, 19536 nœuds sur les 26205 combinaisons complètes et partielles possibles des *Tuples* et *FuncStruct* du problème MDGP, pour la génération complète du front de Pareto (15 passes MDGP). Soit une élimination de 74.55% des nœuds des arbres de décision.

TABLE 4.7 – Performances de l'algorithme MDGP sur notre problème d'optimisation, par rapport à une estimation pour un algorithme *Brute Force + GP*

	Brute Force + GP	MDGP	Ratio
Nombre de résolutions GP	$2.097 \cdot 10^9$	$128.81 \cdot 10^3$	$6.144 \cdot 10^{-3}\%$
Temps de calcul	$282.761 \cdot 10^3h = 32 \text{ ans}$	$24.976h$	$8.833 \cdot 10^{-3}\%$

Il serait encore possible de grandement diminuer ce temps de calcul. En prenant quelques précautions supplémentaires, il semble possible d'étendre l'emploi de l'algorithme *Branch & Cut*, pour l'instant exclusivement employé pour le traitement des variables discrètes simples, aux *Tuples*. La programmation de cet algorithme MDGP a été effectuée sur Matlab, à l'aide de la librairie *gplab*, qui n'a pas été optimisée pour des problèmes aussi volumineux. Une partie importante du temps de calcul total de l'algorithme est attribuable à des fonctions de manipulation des objets *posynômes* en vue de créer les sous-problèmes GP à injecter dans le solveur, et non à la résolution GP desdits sous-problèmes. Ce temps de calculs auxiliaires est évalué à $7h18min$, soit 27% du temps de calcul. Des efforts significatifs ont été consacrés à la réduction de ce temps de traitement des *posynômes*. Notamment, des procédures permettent de ne pas générer plusieurs fois les mêmes objets d'un sous-problème à un autre. Ainsi, la quasi-totalité des sous-problèmes sont générés une

première fois pendant les 2 premières passes de normalisation des fonctions objectif, qui permettent de calculer les coefficients λ_i (section 4.8). Ces 2 passes, sur les 15 effectuées pour calculer le front de Pareto (2 normalisations + 13 points), représentent à elles seules $12h21min$, soit presque 50% du temps de calcul total pour seulement 13.3% du nombre total de résolutions de nœuds.

4.11 Conclusion du chapitre

En se basant sur les concepts et outils présentés au Chapitre 3, nous avons pu transcrire notre problème d'optimisation de conception d'un convertisseur DC-DC non-isolé sous une forme traitable par un nouvel algorithme d'optimisation. Cet algorithme, dit *Mixed-Discrete Geometric Programming* (MDGP), est capable de résoudre des problèmes GP dont certaines de ses variables ne peuvent prendre que des valeurs discrètes, parfois couplées entre elles, et dont certaines des sous-fonctions qui composent le problème sont elles-mêmes des variables. L'algorithme converge vers l'optimum global du problème dans un temps raisonnable de $25h$. Les solutions optimales obtenues nous permettent de remettre en perspective les travaux de conception réalisés "à la main" aux Chapitres précédents, et soulignent l'importance de pouvoir automatiser la recherche du design optimal.

Les travaux présentés peuvent encore faire l'objet d'améliorations significatives. Les temps de calcul peuvent encore être diminués, dans un premier temps en implémentant la méthode *Branch & Cut* pour traiter les *Tuples*, et en reprogrammant l'algorithme pour une librairie *Matlab* mieux optimisée, telle que *cvx*. Il est envisagé de basculer entièrement la programmation dans un langage plus adapté que *Matlab* à la manipulation d'objets polynômes, tel que *Julia* (il existe une librairie GP *JuGP.jl* en développement). Pour ce qui est de l'application de notre algorithme MDGP aux problèmes de design des convertisseurs statiques, nous ne nous sommes intéressés qu'aux topologies *Flying Capacitor* pour des applications DC-DC. Outre les imperfections de notre modélisation qui mériteraient davantage de travaux (problèmes des *deadtimes*, et du B_{pk} des inductances), il serait particulièrement intéressant de reproduire ces travaux pour d'autres types de topologies de convertisseurs, et pour des applications AC. Par ailleurs, la génération de modélisations empiriques des pertes (et autres propriétés) pour davantage de transistors et inductances, préférablement à partir de données expérimentales, permettrait de créer des librairies de *Tuples* et *FuncStruct* plus complètes. Enfin, il existe dans la littérature des algorithmes GP dits "*Modified Signomials*" [105], c'est à dire capables de gérer certains types d'objets polynômes dont les coefficients (mais toujours pas les variables!) peuvent porter des signes négatifs. Leur implémentation nécessite une adaptation des solveurs GP, mais permettrait de traiter les opérations de soustraction qui sont, pour l'instant, interdites dans notre algorithme MDGP.

Conclusion générale & Perspectives

Dans le cadre du Projet PIPAA, visant à l'étude de l'intégration d'une Pile à Combustible hydrogène à bord d'un avion pour alimenter son réseau électrique embarqué, un convertisseur électrique DC-DC non isolé basse tension ($110V - 28V$), moyenne puissance ($1kW - 20kW$) a été mis à l'étude. Nous nous sommes intéressés aux différents composants électriques et aux topologies de convertisseurs afin de déterminer les meilleurs choix de conception permettant d'optimiser la masse et le rendement du convertisseur électrique au Chapitre 1. Il est apparu que les transistors de type GaN associés à une topologie de convertisseur multiniveaux permettent de minimiser la masse et les pertes du convertisseur. Les transistors *EPC2022* et la topologie de Convertisseur à Capacités Flottantes 3-niveaux (FC3L), pour un convertisseur $20kW$ composé de 10 phases de $2kW$ entrelacées, ont été retenus. La vitesse de commutation très élevée des transistors GaN et les propriétés de la topologie FC3L permettent de réduire significativement la taille des éléments passifs et donc de minimiser la masse du convertisseur. Afin de tirer parti de la très faible résistance à l'état passant du transistor *EPC2022*, nous avons parallélisé deux transistors par interrupteur pour diminuer les pertes par conduction.

Plusieurs prototypes de phases $2kW$ ont été développés et fabriqués à partir de cette étude théorique. Le Chapitre 2 s'est intéressé aux enjeux et difficultés associées au routage des circuits imprimés, notamment pour paralléliser les transistors et minimiser les boucles d'inductances parasites causant des surtensions aux bornes des interrupteurs, lors des commutations. Des simulations par éléments finis sur le logiciel *Ansys Q3D* ont permis d'estimer ces caractéristiques parasites. La qualité de ces estimations a été vérifiée et (partiellement) validée expérimentalement par des mesures des surtensions et de leurs oscillations sur les prototypes. En comparant les prototypes, il a été possible de déterminer laquelle des deux approches de routage étudiées présente les meilleures performances. Son rendement a été évalué sur l'entièreté de sa plage de fonctionnement de $2kW$ par une Méthode par Opposition. En extrapolant ses performances pour un convertisseur entrelacé $20kW$ composé de 10 phases de $2kW$ et piloté en *phase-shedding*, le rendement moyen d'un tel convertisseur est évalué à 98.25% en début de vie de la Pile à Combustible, et devrait atteindre 98.75% en fin de vie, pour une puissance massique évaluée à $7.77kW/kg$ ($2.576kg$) presque deux fois supérieure à nos exigences initiales de $4kW/kg$. Il n'a pas été possible d'évaluer expérimentalement la mise en parallèle de plusieurs phases de conversion, et donc de tester le convertisseur $20kW$ complet. Autrement, nous aurions pu étudier l'implémentation du *phase-shedding*, de l'équilibrage des courants entre les phases, et mesurer le rendement réel du système complet. Lors du développement du prototype, le ventilateur n'a pas fait l'objet d'un travail d'optimisation de sa masse et de sa consommation électrique. L'implémentation d'une ventilation plus adaptée pourrait permettre d'augmenter le rendement et de réduire la masse du convertisseur complet.

En parallèle de ces travaux, un nouvel algorithme de résolution de problèmes d'optimisation *Mixed-Discrete Geometric Programming* (MDGP) a été développé au Chapitre 3. L'objectif était de créer un algorithme d'optimisation à la fois rapide, exhaustif, et capable de gérer l'entière des types d'objets qu'un designer doit prendre en compte dans son travail de conception. Parmi les algorithmes d'optimisation les plus communs, la méthode de recherche exhaustive, dite *Brute Force*, garantit de trouver l'optimum global du problème mais dont les temps de calculs sont prohibitifs. Alternativement, il existe des algorithmes rapides mais requérant une formulation simplifiée, souvent approximative, du problème. Ils présentent soit une incapacité à gérer des variables discrètes (problèmes Convexes), soit des méthodes heuristiques qui ne garantissent pas de trouver l'optimum global du problème. Notre algorithme MDGP est à mi-chemin entre ces deux alternatives. Il est capable de trouver une solution garantie optimale comportant des variables continues, des variables discrètes simples et couplées, ainsi que des fonctions substituables. La détermination des variables discrètes et des fonctions substituables est gérée par un algorithme de type *Branch & Bound*. La résolution des variables continues est effectuée par un solveur *Geometric Programming* (GP), adapté à de nombreuses modélisations empiriques de phénomènes physiques et présentant un temps de calcul extrêmement court. La combinaison de ces deux types d'algorithmes de résolution permet de bénéficier de leurs avantages respectifs, ainsi que d'une certaine flexibilité dans la formulation du problème.

Nous avons présenté notre modélisation MDGP du problème d'optimisation de notre convertisseur DC-DC au Chapitre 4. La formulation des différentes équations du modèle sous une forme GP-compatible a été explicitée. En plus des variables discrètes simples et couplées du problème, certains phénomènes physiques associés au choix de la topologie du convertisseur et fondamentalement incompatibles avec un solveur GP ont pu être incorporés dans le problème grâce à l'approche hybride de l'algorithme MDGP. L'algorithme supporte aussi les fonctions substituables, permettant de traiter un problème dont la formulation varie en fonction d'un choix de design. Les transistors sont typiquement décrits par ce type d'objet. Il a été montré la possibilité de générer des bibliothèques de composants (des transistors dans le cas présent) modélisés par des structures de fonctions *fittées* sur les données de simulation propre à chaque composant. L'algorithme MDGP est capable de calculer, aussi précisément que leurs modélisations le permettent, les performances de chaque composant. Ces bibliothèques de composants peuvent être réemployées aisément dans d'autres projets. Il serait par ailleurs possible d'augmenter la pertinence de l'algorithme d'optimisation en générant ces modélisations directement à partir de caractérisations expérimentales. Finalement, un Front de Pareto pour une optimisation multi-objectif de minimisation de la masse et des pertes du convertisseur a pu être calculé en 25h sur un ordinateur portable, pour un problème hybrides comportant 167 variables, soit 161.28 millions de combinaisons discrètes possibles. Un traitement *Brute Force* des variables discrètes aurait mené à un temps de calcul théorique de 32 ans. Notre algorithme MDGP permet donc de résoudre dans un temps raisonnable certains problèmes d'optimisation hybrides autrement insolubles par une approche exhaustive, tout en conservant la garantie de trouver l'optimum global du problème, et sans sacrifier la qualité des modélisations physiques. Il serait encore possible d'abaisser ces temps de calcul en améliorant certains aspects de l'algorithme et en effectuant un travail d'optimisation approfondi de sa programmation. Les résultats du Front de Pareto permettent quant à eux de mettre en évidence les angles morts de l'étude de conception réalisée au Chapitre 1. Il serait par ailleurs intéressant d'améliorer certains aspects de la modélisation MDGP du convertisseur DC-DC, et de développer des modélisations pour d'autres types de convertisseurs, notamment AC.

Annexe A

A.1 *.lib* SPICE component files for transistors

Listing A.1 – SiDR668DP SPICE component *.lib* file

```
1 .SUBCKT SiDR668DP D G S
2 M1 3 GX S S NMOS W= 12962500u L= 0.30u
3
4 M2 S GX S D PMOS W= 12962500u L= 0.20u
5 R1 D 3 3.485e-03 TC=8.280e-03,3.017e-05
6 CGS GX S 3.632e-09
7 CGD GX D 1.008e-13
8 RG G GY 1m
9 RTCV 100 S 1e6 TC=-4.473e-04,-1.064e-06
10 ETCV GX GY 100 200 1
11 ITCV S 100 1u
12 VTCV 200 S 1
13 DBD S D DBD 12962500u
14 *****
15 .MODEL NMOS NMOS ( LEVEL = 3 TOX = 7e-8
16 + RS = 0 KP = 7.305e-06 NSUB = 1.252e+17
17 + KAPPA = 7.713e-02 NFS = 3.050e+11
18 + LD = 0 IS = 0 TPG = 1 )
19 *****
20 .MODEL PMOS PMOS ( LEVEL = 3 TOX = 7e-8
21 +NSUB = 1.493e+15 IS = 0 TPG = -1 )
22 *****
23 .MODEL DBD D (
24 +FC = 0.1 TT = 2.000e-07 T_measured = 25 BV = 101
25 +RS = 9.729e-03 N = 1.114e+00 IS = 2.533e-12
26 +EG = 1.214e+00 XTI = 1.178e-01 TRS1 = 2.708e-03
27 +CJO = 4.911e-10 VJ = 1.753e+00 M = 9.362e-01 )
28 .ENDS
```

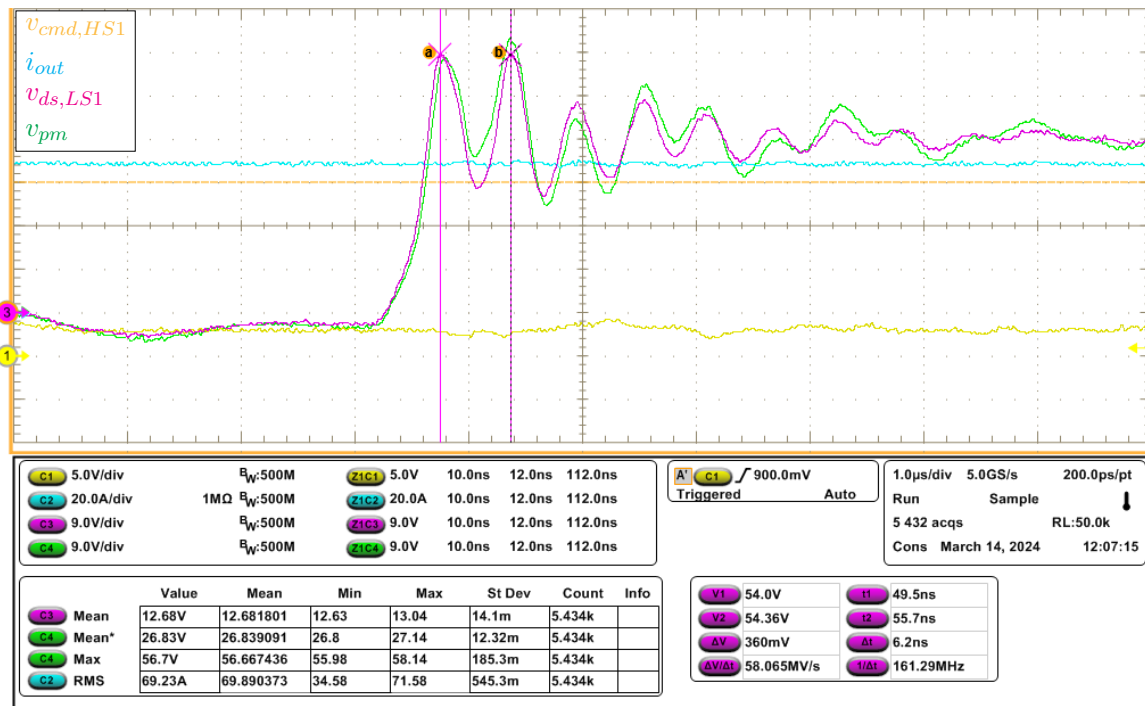
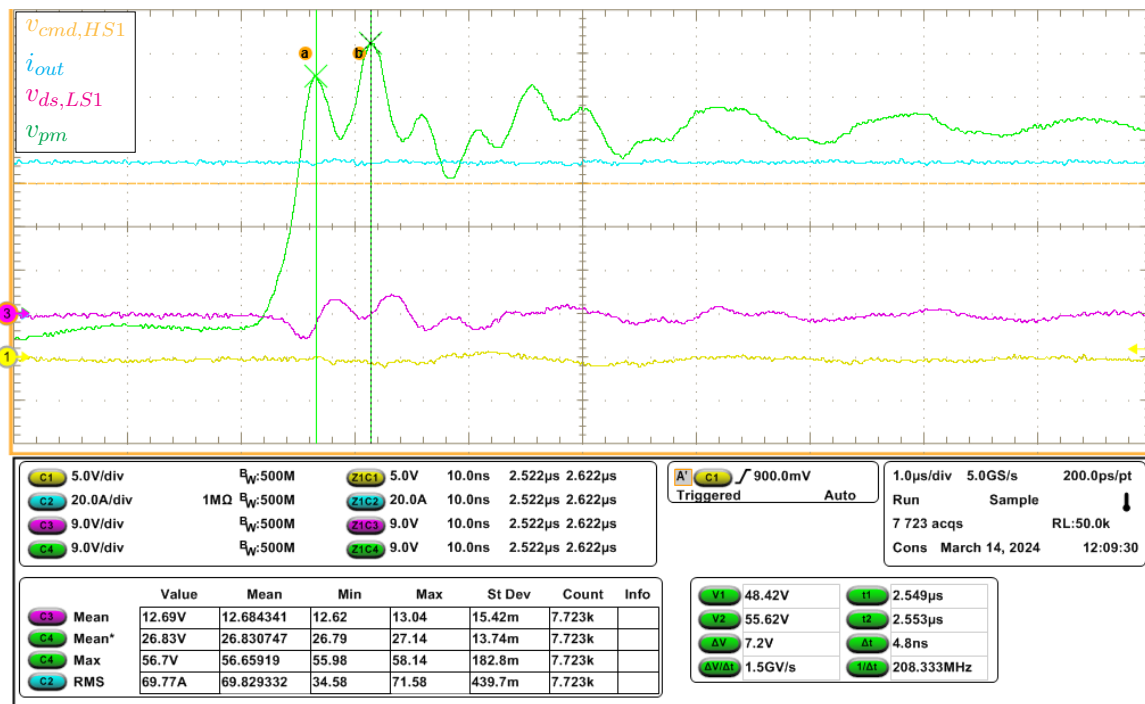
Listing A.2 – EPC2022 SPICE component *.lib* file

```
1 .subckt EPC2022x gatein drainin sourcein
2 .param aWg=2599 A1=83.991 k2=2.4169 k3=0.15 rpara=0.0016862 rpara_s_factor=0.24
3 + aITc=0.0039 arTc=-0.0078 k2Tc=0.001 x0_0=1.5866 x0_0_TC=0 x0_1=1.0e-6 x0_1_TC=0
4 + dgs1=4.3e-07 dgs2=2.6e-13 dgs3=0.8 dgs4=0.23
5 + ags1=1.3964e-09 ags2=6.2652e-10 ags3=1.9489 ags4=0.1161
6 + ags5=-3.1014e-14 ags6=-10.01 ags7=0.97936
7 + agd1=6.7965e-15 agd2=2.3905e-11 agd3=-0.082452 agd4=20.241
8 + agd5=5.1752e-10 agd6=-0.94707 agd7=4.5019
9 + agd8=4.6034e-12 agd9=-257.28 agd10=32.686
```

ANNEXE A.

```
10 + asd1=6.0339e-10 asd2=1.1355e-09 asd3=-17.177 asd4=3.8037
11 + asd5=1.0489e-09 asd6=-0.5384 asd7=38.828 rg_value=0.3
12
13 rd drainin drain {(1-rpara_s_factor)*rpara*(1-arTc*(Temp-25))}
14 rs sourcein source {(rpara_s_factor*rpara*(1-arTc*(Temp-25))}
15 rg gatein gate {(rg_value)}
16
17 Rcsdconv drain source {100000Meg/aWg}
18 Rcgsconv gate source {100000Meg/aWg}
19 Rcgdconv gate drain {100000Meg/aWg}
20
21 bswitch drain source I=if(v(drain,source)>0,
22 + (A1*(1-aITc*(Temp-25))*log(1.0+exp((v(gate,source)-(k2*(1-k2Tc*(Temp-25))))/k3))*
23 + v(drain,source)/(1 + (x0_0*(1-x0_0_TC*(Temp-25))+x0_1*(1-x0_1_TC*(Temp-25))*v(
24 + (-A1*(1-aITc*(Temp-25))*log(1.0+exp((v(gate,drain)-(k2*(1-k2Tc*(Temp-25))))/k3))*
25 + v(source,drain)/(1 + (x0_0*(1-x0_0_TC*(Temp-25))+x0_1*(1-x0_1_TC*(Temp-25))*v(
26 + gate,source))*v(drain,source))),
27 + (-A1*(1-aITc*(Temp-25))*log(1.0+exp((v(gate,drain)-(k2*(1-k2Tc*(Temp-25))))/k3))*
28 + v(source,drain)/(1 + (x0_0*(1-x0_0_TC*(Temp-25))+x0_1*(1-x0_1_TC*(Temp-25))*v(
29 + gate,drain))*v(source,drain))),
30
31 bgddiode gate drain I=if( v(gate,drain)>10,
32 + (0.125*aWg/1077*(dgs1*(exp((10.0)/dgs3)-1)+dgs2*(exp((10.0)/dgs4)-1))),
33 + (0.125*aWg/1077*(dgs1*(exp((v(gate,source))/dgs3)-1)+dgs2*(exp((v(gate,source))/
34 + dgs4)-1))))
35
36 C_GS gate source {ags1} TC=0,0
37 C_CGS1 gate source Q=(0.5*ags2*ags4*log(1+exp((v(gate,source)-ags3)/ags4))+
38 + ags5*ags7*log(1+exp((v(source,drain)-ags6)/ags7)))
39
40 C_GD gate drain {agd1} TC=0,0
41 C_CGD1 gate drain Q=(0.5*ags2*ags4*log(1+exp((v(gate,drain)-ags3)/ags4))+
42 + agd2*agd4*log(1+exp((v(gate,drain)-agd3)/agd4))+
43 + agd5*agd7*log(1+exp((v(gate,drain)-agd6)/agd7))+
44 + agd8*agd10*log(1+exp((v(gate,drain)-agd9)/agd10)))
45
46 C_SD source drain {asd1} TC=0,0
47 C_CSD1 source drain Q=(asd2*asd4*log(1+exp((v(source,drain)-asd3)/asd4))+
48 + asd5*asd7*log(1+exp((v(source,drain)-asd6)/asd7)))
49
50 .ends
```

A.2 Prises de vue à l'oscilloscope

FIGURE A.1 – Mesure de $v_{ds,LS1}$ au *turn-off* sur le Prototype *Adjacent*, pour $V_{in} = 75V$, $I_{out} = 70A$ FIGURE A.2 – Mesure de $v_{ds,LS2}$ (pris sur v_{pm}) au *turn-off* sur le Prototype *Adjacent*, pour $V_{in} = 75V$, $I_{out} = 70A$

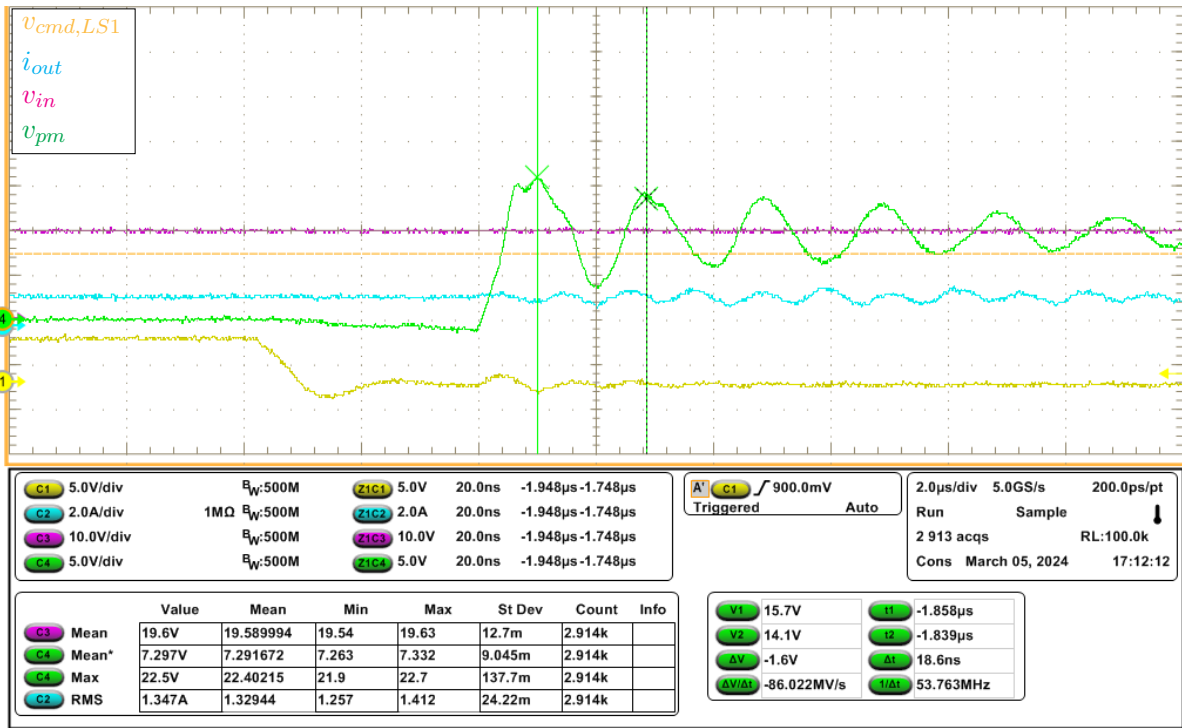


FIGURE A.3 – Mesure de $v_{ds,LS1}$ au *turn-off* sur le Prototype *Traversant*, pour $V_{in} = 20V$, $I_{out} = 1A$

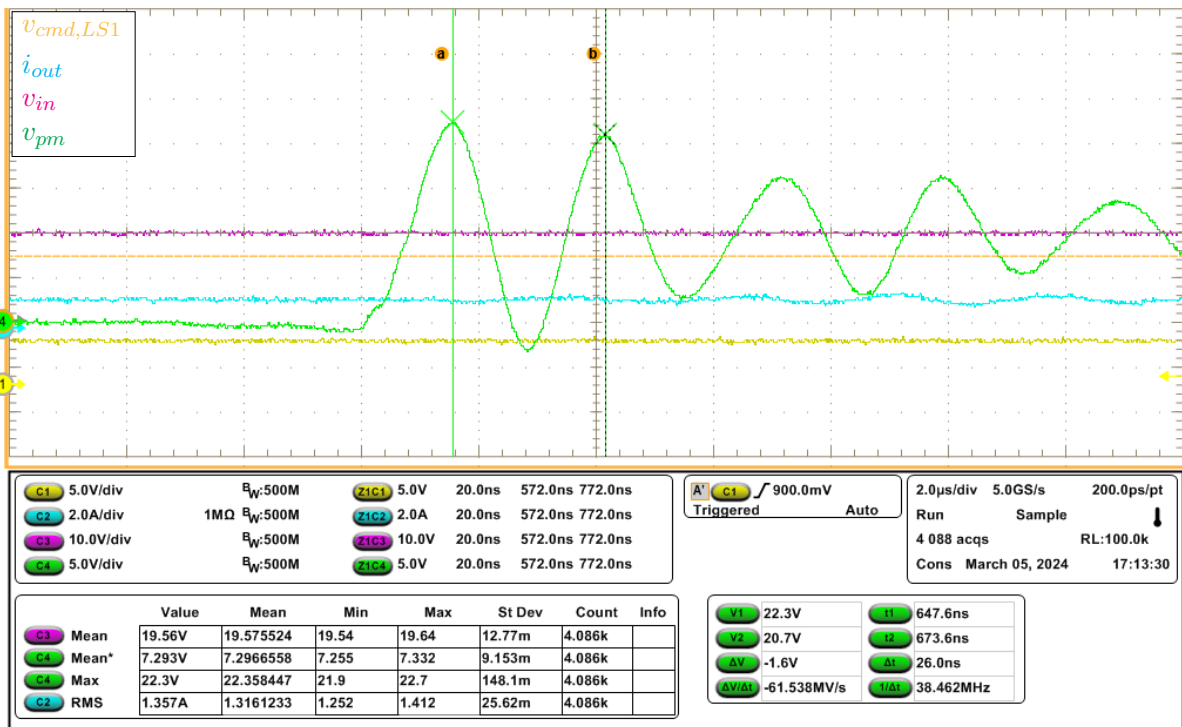


FIGURE A.4 – Mesure de $v_{ds,LS2}$ (pris sur v_{pm}) au *turn-off* sur le Prototype *Adjacent*, pour $V_{in} = 20V$, $I_{out} = 1A$

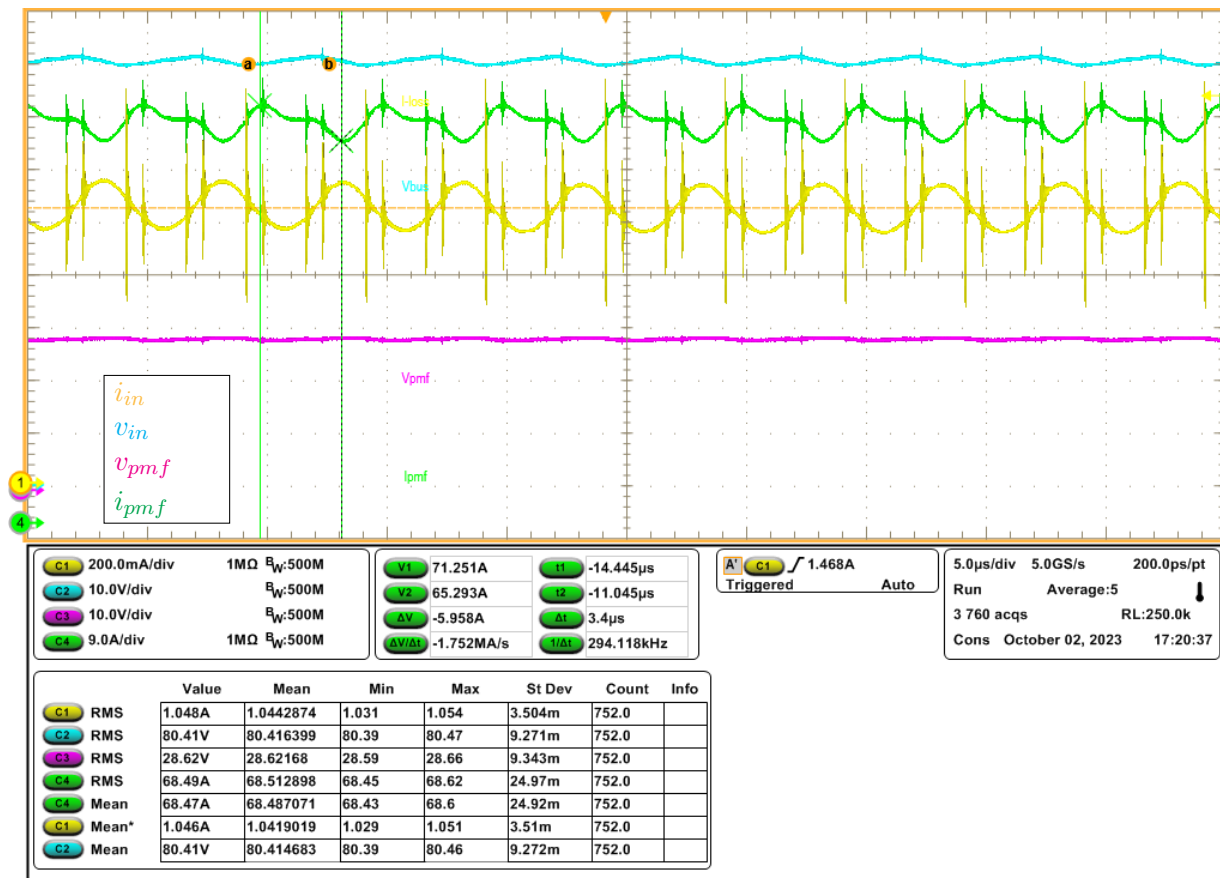


FIGURE A.5 – Mesure des tensions et courants du circuit d'Opposition du prototype *Adjacent* (sans le moyennage sur 50 échantillons) pour $V_{in} = 80V$, $I_{pm.f} = 69A$

Bibliographie

- [1] S. Sanchez, “Contribution à la conception de coupleurs magnétiques robustes pour convertisseurs multicellulaires parallèles,” Ph.D. dissertation, March 2015. [Online]. Available : <https://oatao.univ-toulouse.fr/14499/>
- [2] N. Kaminski and O. Hilt, “Sic and gan devices – wide bandgap is not all the same,” *IET Circuits, Devices & Systems*, vol. 8, no. 3, pp. 227–236, 2014. [Online]. Available : <https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/iet-cds.2013.0223>
- [3] S. Olson, M. Rodgers, S. P. Rao, and C. Borst. Wet clean challenges for various applications. [Online]. Available : https://www.linx-consulting.com/wp-content/uploads/2018/04/Steve_Olson_Business_of_cleans-SPCC2018-Presentation_20180409_v2-suggestions-included.pdf
- [4] J. Zou, N. C. Brooks, S. Coday, N. M. Ellis, and R. C. Pilawa-Podgurski, “On the size and weight of passive components : Scaling trends for high-density power converter designs,” *2022 IEEE 23rd Workshop on Control and Modeling for Power Electronics (COMPEL)*, pp. 1–7, 2022.
- [5] S. Biswas and D. Reusch, “Gan based switched capacitor three-level buck converter with cascaded synchronous bootstrap gate drive scheme,” *2018 IEEE Energy Conversion Congress and Exposition (ECCE)*, pp. 3490–3496, 2018.
- [6] Z. Chen, D. Boroyevich, and R. Burgos, “Experimental parametric study of the parasitic inductance influence on mosfet switching characteristics,” *The 2010 International Power Electronics Conference - ECCE ASIA -*, pp. 164–169, 2010.
- [7] D. Reusch and J. Strydom, “Understanding the effect of pcb layout on circuit performance in a high-frequency gallium-nitride-based point of load converter,” *IEEE Transactions on Power Electronics*, vol. 29, no. 4, pp. 2008–2015, 2014.
- [8] “Feuille de route de la décarbonation de l’aérien,” F NAM, France Hydrogène, GIFAS, UAF, UFE, UfipEM, Tech. Rep., Mar 2023. [Online]. Available : <https://www.ecologie.gouv.fr/sites/default/files/documents/Proposition%20de%20feuille%20de%20route%20d%C3%A9carbonation%20transport%20a%C3%A9rien.pdf>
- [9] F. Cells and H. . J. Undertaking, *Hydrogen-powered aviation – A fact-based study of hydrogen technology, economics, and climate impact by 2050*. Publications Office, 2020.
- [10] M. Baudy, “Etude de l’impact des caractérisations et des modes opératoires sur le vieillissement d’une pile à combustible à membrane échangeuse de protons haute température (pemht),” Ph.D. dissertation, Institut National Polytechnique de Toulouse-INPT, 2022.
- [11] M. Durand, “Etude et optimisation des phases de démarrage et d’arrêt des piles à combustible de type pem haute température en vue de maximiser leur durée de vie.” Ph.D. dissertation, Toulouse, INPT, 2022.

- [12] T. Jarry, “Dimensionnement optimal d’un système hybride direct pile à combustible pem haute température (pem-ht)-batterie et impact d’un tel système sur les interactions avec le convertisseur de puissance associé,” Ph.D. dissertation, Institut National Polytechnique de Toulouse-INPT, 2022.
- [13] H. Cao, X. Yang, C. Xue, L. He, Z. Tan, M. Zhao, Y. Ding, W. Li, and W. Qu, “A 12-level series-capacitor 48-lv dc–dc converter with on-chip switch and gan hybrid power conversion,” *IEEE Journal of Solid-State Circuits*, vol. 56, no. 12, pp. 3628–3638, 2021.
- [14] T. Modeer, N. Pallo, T. Foulkes, C. B. Barth, and R. C. N. Pilawa-Podgurski, “Design of a gan-based interleaved nine-level flying capacitor multilevel inverter for electric aircraft applications,” *IEEE Transactions on Power Electronics*, vol. 35, no. 11, pp. 12 153–12 165, 2020.
- [15] M. Cousineau and B. Cougo, “Interleaved converter with massive parallelization of high frequency gan switching-cells using decentralized modular analog controller,” *2015 IEEE Energy Conversion Congress and Exposition (ECCE)*, pp. 4343–4350, 2015.
- [16] M. Sinha, J. Poon, B. B. Johnson, M. Rodriguez, and S. V. Dhople, “Decentralized interleaving of parallel-connected buck converters,” *IEEE Transactions on Power Electronics*, vol. 34, no. 5, pp. 4993–5006, 2019.
- [17] Q. D. Phan, G. Gateau, P. C. Nguyen, M. Cousineau, H. P. To, B. A. Nguyen, L. Veit, R. De Milly, and M. Hillesheim, “A fast, decentralized, self-aligned carrier method for multicellular converters,” *Applied Sciences*, vol. 11, no. 1, 2021. [Online]. Available : <https://www.mdpi.com/2076-3417/11/1/137>
- [18] T. Meynard and H. Foch, “Multi-level conversion : high voltage choppers and voltage-source inverters,” *PESC '92 Record. 23rd Annual IEEE Power Electronics Specialists Conference*, pp. 397–403 vol.1, 1992.
- [19] Y. Lei, C. Barth, S. Qin, W.-c. Liu, I. Moon, A. Stillwell, D. Chou, T. Foulkes, Z. Ye, Z. Liao, and R. C. Pilawa-Podgurski, “A 2 kw, single-phase, 7-level, gan inverter with an active energy buffer achieving 216 w/in³ power density and 97.6% peak efficiency,” *2016 IEEE Applied Power Electronics Conference and Exposition (APEC)*, pp. 1512–1519, 2016.
- [20] M. Gleissner and M. M. Bakran, “Design and control of fault-tolerant nonisolated multiphase multilevel dc–dc converters for automotive power systems,” *IEEE Transactions on Industry Applications*, vol. 52, no. 2, pp. 1785–1795, 2016.
- [21] H. Mabboux and O. Parrot, “Alimentation sans coupure à neutre traversant,comportant un hacheur-élévateur double,” Patent FR2 713 030, May 02, 1995.
- [22] P. J. Grbović, P. Delarue, P. Le Moigne, and P. Bartholomeus, “A bidirectional three-level dc–dc converter for the ultracapacitor applications,” *IEEE Transactions on Industrial Electronics*, vol. 57, no. 10, pp. 3415–3430, 2010.
- [23] J. Kim, S. Kim, N. B. Sagnazar, and S. Choi, “A new zvs switched-capacitor dc-dc converter with low boost inductance,” *2019 10th International Conference on Power Electronics and ECCE Asia (ICPE 2019 - ECCE Asia)*, pp. 1939–1944, 2019.
- [24] P. Gammon, “Silicon and the wide bandgap semiconductors, shaping the future power electronic device market,” *2013 14th International Conference on Ultimate Integration on Silicon (ULIS)*, pp. 9–13, 2013.
- [25] A. Nakagawa, “Theoretical investigation of silicon limit characteristics of igbt,” *2006 IEEE International Symposium on Power Semiconductor Devices and IC's*, pp. 1–4, 2006.
- [26] N. Kaminski, “The ideal chip is not enough : Issues retarding the success of wide band-gap devices,” *Japanese Journal of Applied Physics*, vol. 56, no. 4S, p. 04CA03, feb 2017. [Online]. Available : <https://dx.doi.org/10.7567/JJAP.56.04CA03>

- [27] M. Rodríguez, A. Rodríguez, P. F. Miaja, D. G. Lamar, and J. S. Zúniga, “An insight into the switching process of power mosfets : An improved analytical losses model,” *IEEE Transactions on Power Electronics*, vol. 25, no. 6, pp. 1626–1640, 2010.
- [28] W. J. de Paula, G. H. M. Tavares, G. M. Soares, P. S. Almeida, and H. A. C. Braga, “Switching losses prediction methods oriented to power mosfets – a review,” *IET Power Electronics*, vol. 13, no. 14, pp. 2960–2970, 2020.
- [29] K. Peng, S. Eskandari, and E. Santi, “Characterization and modeling of a gallium nitride power hemt,” *IEEE Transactions on Industry Applications*, vol. 52, no. 6, pp. 4965–4975, 2016.
- [30] K. Shah and K. Shenai, “Simple and accurate circuit simulation model for gallium nitride power transistors,” *IEEE Transactions on Electron Devices*, vol. 59, no. 10, pp. 2735–2741, 2012.
- [31] J. Gareau, R. Hou, and A. Emadi, “Review of loss distribution, analysis, and measurement techniques for gan hemts,” *IEEE Transactions on Power Electronics*, vol. 35, no. 7, pp. 7405–7418, 2020.
- [32] *Circuit Simulation using EPC Device Models*, EPC, application Note : AN005.
- [33] J. Lautner and B. Piepenbreier, “Analysis of gan hemt switching behavior,” *2015 9th International Conference on Power Electronics and ECCE Asia (ICPE-ECCE Asia)*, pp. 567–574, 2015.
- [34] Y. Lei, W.-C. Liu, and R. C. N. Pilawa-Podgurski, “An analytical method to evaluate and design hybrid switched-capacitor and multilevel converters,” *IEEE Transactions on Power Electronics*, vol. 33, no. 3, pp. 2227–2240, 2018.
- [35] J. Azurza Anderson, G. Zulauf, P. Papamanolis, S. Hobi, S. Mirić, and J. W. Kolar, “Three levels are not enough : Scaling laws for multilevel converters in ac/dc applications,” *IEEE Transactions on Power Electronics*, vol. 36, no. 4, pp. 3967–3986, 2021.
- [36] P. C. Bolsi, H. C. Sartori, and J. R. Pinheiro, “Comparison of core technologies applied to power inductors,” *2018 13th IEEE International Conference on Industry Applications (INDUSCON)*, pp. 1100–1106, 2018.
- [37] H.-I. Hsiang, “Progress in materials and processes of multilayer power inductors,” *Journal of Materials Science : Materials in Electronics*, vol. 31, no. 19, pp. 16 089–16 110, Oct 2020. [Online]. Available : <https://doi.org/10.1007/s10854-020-04188-8>
- [38] “Selecting ihlp composite inductors for non-isolated converters utilizing vishay’s application sheet selecting ihlp composite inductors for non-isolated converters utilizing vishay’s application sheet,” Vishay, Tech. Rep., Nov 2011. [Online]. Available : <https://www.vishay.com/docs/34250/ihlpap.pdf>
- [39] “Ihlp selection example,” Vishay, Tech. Rep., Mar 2013. [Online]. Available : <https://www.vishay.com/docs/34252/ihlpse.pdf>
- [40] “Ihlp parameters,” Vishay, Tech. Rep., Nov 2023. [Online]. Available : <https://www.vishay.com/docs/34251/ihlpselectionexample.pdf>
- [41] J. Li, T. Abdallah, and C. Sullivan, “Improved calculation of core loss with nonsinusoidal waveforms,” *Conference Record of the 2001 IEEE Industry Applications Conference. 36th IAS Annual Meeting (Cat. No.01CH37248)*, vol. 4, pp. 2203–2210 vol.4, 2001.
- [42] “Power mosfet basics : Understanding gate charge and using it to assess switching performance,” Vishay, Tech. Rep., Feb 2016. [Online]. Available : <https://www.vishay.com/docs/73217/an608a.pdf>

- [43] J. Wang, H. S.-h. Chung, and R. T.-h. Li, "Characterization and experimental assessment of the effects of parasitic elements on the mosfet switching performance," *IEEE Transactions on Power Electronics*, vol. 28, no. 1, pp. 573–590, 2013.
- [44] Y. Xiao, H. Shah, T. Chow, and R. Gutmann, "Analytical modeling and experimental evaluation of interconnect parasitic inductance on mosfet switching characteristics," *Nineteenth Annual IEEE Applied Power Electronics Conference and Exposition, 2004. APEC '04.*, vol. 1, pp. 516–521 Vol.1, 2004.
- [45] J. Burkard and J. Biela, "Paralleling gan switches for low voltage high current half-bridges," *2019 IEEE Energy Conversion Congress and Exposition (ECCE)*, pp. 3245–3252, 2019.
- [46] Z. Ye, Y. Lei, W.-C. Liu, P. S. Shenoy, and R. Pilawa-Podgurski, "Improved bootstrap methods for powering floating gate drivers of flying capacitor multilevel converters and hybrid switched-capacitor converters," *IEEE Transactions on Power Electronics*, vol. 35, no. 6, pp. 5965–5977, 2020.
- [47] J. Lu, H. Bai, A. Brown, M. McAmmond, D. Chen, and J. Styles, "Design consideration of gate driver circuits and pcb parasitic parameters of paralleled e-mode gan hemts in zero-voltage-switching applications," *2016 IEEE Applied Power Electronics Conference and Exposition (APEC)*, pp. 529–535, 2016.
- [48] J. L. Lu and D. Chen, "Paralleling gan e-hemts in 10kw–100kw systems," *2017 IEEE Applied Power Electronics Conference and Exposition (APEC)*, pp. 3049–3056, 2017.
- [49] M. Robin Zäch, S. Í Garði, and S. Holt Olsen, "Design and optimisation of a half-bridge switching module with parallel gan hemts for high power applications using finite-element analysis," May 2021. [Online]. Available : https://kjdk-aub.primo.exlibrisgroup.com/permalink/45KBDK_AUB/a7me0f/alma9921567580305762
- [50] D. Reusch and J. Strydom, "Effectively paralleling gallium nitride transistors for high current and high frequency applications," *2015 IEEE Applied Power Electronics Conference and Exposition (APEC)*, pp. 745–751, 2015.
- [51] Y. Zhang, J. Li, and J. Wang, "Investigations on driver and layout for paralleled gan hemts in low voltage application," *IEEE Access*, vol. 7, pp. 179 134–179 142, 2019.
- [52] H. H. Sathler, T. Zhao, F. Costa, B. Cougo, G. Segond, R. Burgos, and D. Labrousse, "Design of three-level flying-capacitor commutation cells with four paralleled 650 v/60 a gan hemts," *2021 IEEE Applied Power Electronics Conference and Exposition (APEC)*, pp. 2277–2284, 2021.
- [53] F. Luo, Z. Chen, L. Xue, P. Mattavelli, D. Boroyevich, and B. Hughes, "Design considerations for gan hemt multichip halfbridge module for high-frequency power converters," *2014 IEEE Applied Power Electronics Conference and Exposition - APEC 2014*, pp. 537–544, 2014.
- [54] N. Sang, P.-O. Jeannin, and P. Lefranc, "Analyses of the unbalanced paralleled gan hemt transistors," *2018 20th European Conference on Power Electronics and Applications (EPE'18 ECCE Europe)*, pp. P.1–P.9, 2018.
- [55] A. Letellier, M. R. Dubois, J. P. F. Trovão, and H. Maher, "Calculation of printed circuit board power-loop stray inductance in gan or high di/dt applications," *IEEE Transactions on Power Electronics*, vol. 34, no. 1, pp. 612–623, 2019.
- [56] A. Lale, N. Videau, A. Bourennane, F. Richardeau, and S. Charlot, "Mixed monolithic-hybrid integration of multiphase power converter : Preliminary evaluation of the 3-chip integration concept," *2016 International Conference on Electrical Systems for Aircraft, Railway, Ship Propulsion and Road Vehicles & International Transportation Electrification Conference (ESARS-ITEC)*, pp. 1–7, 2016.

- [57] D. Reusch, D. Gilham, Y. Su, and F. C. Lee, "Gallium nitride based 3d integrated non-isolated point of load module," *2012 Twenty-Seventh Annual IEEE Applied Power Electronics Conference and Exposition (APEC)*, pp. 38–45, 2012.
- [58] F. Forest, J.-J. Huselstein, S. Faucher, M. Elghazouani, P. Ladoux, T. Meynard, F. Richardeau, and C. Turpin, "Use of opposition method in the test of high-power electronic converters," *IEEE Transactions on Industrial Electronics*, vol. 53, no. 2, pp. 530–541, 2006.
- [59] J. Brandelero, B. Cougo, T. Meynard, N. Videau, O. Goulard, X. Bonnin, and H. Schneider, "Evaluation des pertes par commutation pour la conception des convertisseurs et applications des composants grand gap," *Symposium de Génie Électrique 2014*, Jul. 2014. [Online]. Available : <https://hal.science/hal-01065310>
- [60] R. H. Wilkinson, T. A. Meynard, and H. du Toit Mouton, "Natural balance of multicell converters : The two-cell case," *IEEE Transactions on Power Electronics*, vol. 21, no. 6, pp. 1649–1657, 2006.
- [61] A. Penczek, A. Mondzik, R. Stala, and A. Ruderman, "Simple time-domain analysis of a multilevel dc-dc flying capacitor converter average aperiodic natural balancing dynamics," *IET Power Electronics*, vol. 12, no. 5, pp. 1179–1186, 2019. [Online]. Available : <https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/iet-pel.2018.5070>
- [62] T. Meynard, H. Foch, P. Thomas, J. Courault, R. Jakob, and M. Nahrstaedt, "Multicell converters : basic concepts and industry applications," *IEEE Transactions on Industrial Electronics*, vol. 49, no. 5, pp. 955–964, 2002.
- [63] Z. Ye, Y. Lei, Z. Liao, and R. C. N. Pilawa-Podgurski, "Investigation of capacitor voltage balancing in practical implementations of flying capacitor multilevel converters," *IEEE Transactions on Power Electronics*, vol. 37, no. 3, pp. 2921–2935, 2022.
- [64] R. Bosshard, J. W. Kolar, J. Mühlethaler, I. Stevanović, B. Wunsch, and F. Canales, "Modeling and $\eta - \alpha$ -pareto optimization of inductive power transfer coils for electric vehicles," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 3, no. 1, pp. 50–64, 2015.
- [65] R. Bosshard and J. W. Kolar, "Multi-objective optimization of 50 kw/85 khz ipt system for public transport," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 4, no. 4, pp. 1370–1382, 2016.
- [66] A. Fernandez-Hernandez, F. Gonzalez-Hernando, A. Garcia-Bediaga, I. Villar, and G. Abad, "Exploration of the $\eta - \rho$ pareto optimization of bidirectional isolated dc-dc power electronic converters for more electric aircraft," *2022 IEEE Energy Conversion Congress and Exposition (ECCE)*, pp. 1–8, 2022.
- [67] D. O. Boillat, F. Krismer, and J. W. Kolar, "Design space analysis and $\rho - \eta$ pareto optimization of lc output filters for switch-mode ac power sources," *IEEE Transactions on Power Electronics*, vol. 30, no. 12, pp. 6906–6923, 2015.
- [68] J. Stöttner, A. Rauscher, and C. Endisch, "Pareto optimization of multilevel inverter structures regarding the dc magnitude, switching frequency and switching angles," *International Journal of Electrical Power & Energy Systems*, vol. 142, p. 108259, 2022. [Online]. Available : <https://www.sciencedirect.com/science/article/pii/S0142061522002836>
- [69] H. Uemura, F. Krismer, Y. Okuma, and J. W. Kolar, " $\eta - \rho$ pareto optimization of 3-phase 3-level t-type ac-dc-ac converter comprising si and sic hybrid power stage," *2014 International Power Electronics Conference (IPEC-Hiroshima 2014 - ECCE ASIA)*, pp. 2834–2841, 2014.
- [70] L. Török, N. Christensen, S. Munk-Nielsen, and S. Bęczkowski, "Optimization of isolated dc-dc converter topologies for fuel cell applications," *2019 21st European Conference on Power Electronics and Applications (EPE '19 ECCE Europe)*, pp. P.1–P.7, 2019.

- [71] A. Stillwell and R. C. N. Pilawa-Podgurski, “Design optimization of a 1500 v gan-based solar inverter using flying capacitor multi-level converter stages,” *2019 IEEE Energy Conversion Congress and Exposition (ECCE)*, pp. 4605–4612, 2019.
- [72] D. Žarko, M. Kovačić, S. Stipetić, and D. Vuljaj, “Optimization of electric drives for traction applications,” *2017 19th International Conference on Electrical Drives and Power Electronics (EDPE)*, pp. 15–32, 2017.
- [73] N. Fritz, X. Pan, and R. W. De Doncker, “Speeding up converter design by analytical modeling and genetic optimization a more electric aircraft example,” *The 10th International Conference on Power Electronics, Machines and Drives (PEMD 2020)*, vol. 2020, pp. 908–913, 2020.
- [74] J. Águila León, C. D. Chiñas-Palacios, C. Vargas-Salgado, E. Hurtado-Perez, and E. X. García, “Optimal pid parameters tuning for a dc-dc boost converter : A performance comparative using grey wolf optimizer, particle swarm optimization and genetic algorithms,” *2020 IEEE Conference on Technologies for Sustainability (SusTech)*, pp. 1–6, 2020.
- [75] K. Stoyka, N. Femia, and G. Di Capua, “Optimizing power converters with partially saturated inductors by evolutionary algorithms,” *2017 14th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*, pp. 1–4, 2017.
- [76] G. I. Orfanoudakis, G. Foteinopoulos, E. Koutroulis, and W. Wu, “Design optimization of aalborg-type transformerless pv inverters with focus on power quality,” *2022 11th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, pp. 1–5, 2022.
- [77] S. E. De León-Aldaco, H. Calleja, and J. Aguayo Alquicira, “Metaheuristic optimization methods applied to power converters : A review,” *IEEE Transactions on Power Electronics*, vol. 30, no. 12, pp. 6791–6803, 2015.
- [78] Y. Li, X. Tang, X. Lin, L. Grzesiak, and X. Hu, “The role and application of convex modeling and optimization in electrified vehicles,” *Renewable and Sustainable Energy Reviews*, vol. 153, p. 111796, 2022. [Online]. Available : <https://www.sciencedirect.com/science/article/pii/S1364032121010650>
- [79] P. Zhou, W. H. Choi, B. Kim, C. H. Kim, and S. S. Sapatnekar, “Optimization of on-chip switched-capacitor dc-dc converters for high-performance applications,” *2012 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pp. 263–270, 2012.
- [80] A. De Keyser, M. Vandeputte, and G. Crevecoeur, “Convex mapping formulations enabling optimal power split and design of the electric drivetrain in all-electric vehicles,” *IEEE Transactions on Vehicular Technology*, vol. 66, no. 11, pp. 9702–9711, 2017.
- [81] F. Altaf, L. Johannesson, and B. Egardt, “Evaluating the potential for cell balancing using a cascaded multi-level converter using convex optimization*,” *IFAC Proceedings Volumes*, vol. 45, no. 30, pp. 100–107, 2012, 3rd IFAC Workshop on Engine and Powertrain Control, Simulation and Modeling. [Online]. Available : <https://www.sciencedirect.com/science/article/pii/S1474667015351454>
- [82] Y. Jiang, Y. Yang, S.-C. Tan, and S.-Y. R. Hui, “Distributed convex optimization-based control for distribution power loss minimization of islanded three-phase ac microgrids,” *2021 IEEE 12th Energy Conversion Congress & Exposition - Asia (ECCE-Asia)*, pp. 515–520, 2021.
- [83] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.
- [84] D. Rosenberg and S. Boyd. Extreme abridgement of boyd and vandenberghe’s convex optimization. [Online]. Available : <https://davidrosenberg.github.io/ml2015/docs/convex-optimization.pdf>

- [85] N. Kesbia, J.-L. Schanen, H. Alawieh, L. Garbuio, and Y. Avenas, "Design by optimization of multiphase inverter for electric vehicle drive," *2020 22nd European Conference on Power Electronics and Applications (EPE'20 ECCE Europe)*, pp. 1–8, 2020.
- [86] A. Voltaire, J.-L. Schanen, J.-P. Ferrieux, B. Sarrazin, and C. Gautier, "Predesign methodology of voltage inverters using a gradient-based optimization algorithm," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 9, no. 5, pp. 5895–5905, 2021.
- [87] S. Debnath and M. Saeedifard, "Simulation-based gradient-descent optimization of modular multilevel converter controller parameters," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 1, pp. 102–112, 2016.
- [88] S. Boyd, S.-J. Kim, L. Vandenberghe, and A. Hassibi, "A tutorial on geometric programming," *Optimization and engineering*, vol. 8, pp. 67–127, 2007.
- [89] U. Ribes-Mallada, R. Leyva, P. Garcés *et al.*, "Optimization of dc-dc converters via geometric programming," *Mathematical problems in Engineering*, vol. 2011, 2011.
- [90] R. Leyva, U. Ribes-Mallada, P. Garcés, and J. Reynaud, "Design and optimization of buck and double buck converters by means of geometric programming," *Mathematics and Computers in Simulation*, vol. 82, no. 8, pp. 1516–1530, 2012. [Online]. Available : <https://www.sciencedirect.com/science/article/pii/S0378475412000626>
- [91] R. Leyva, "Optimal sizing of cuk converters via geometric programming," *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, pp. 2480–2485, 2016.
- [92] A. Stupar, T. McRae, N. Vukadinovic, A. Prodic, and J. A. Taylor, "Multi-objective optimization and comparison of multi-level dc-dc converters using convex optimization methods," *2016 18th European Conference on Power Electronics and Applications (EPE'16 ECCE Europe)*, pp. 1–10, 2016.
- [93] A. Stupar, "Multi-objective optimization of unidirectional non-isolated dc/dc converters," Ph.D. dissertation, University of Toronto (Canada), 2017.
- [94] R. Rajamony, S. Wang, R. Navaratne, and W. Ming, "Multi-objective design of single-phase differential buck inverters with active power decoupling," *IEEE Open Journal of Power Electronics*, vol. 3, pp. 105–114, 2022.
- [95] A. Stupar, M. Halamicek, T. Moianou, A. Prodic, and J. A. Taylor, "Efficiency optimization of a 7-switch flying capacitor buck converter power stage ic using simulation and geometric programming," *2018 IEEE 19th Workshop on Control and Modeling for Power Electronics (COMPEL)*, pp. 1–8, 2018.
- [96] A. Stupar, T. McRae, N. Vukadinović, A. Prodić, and J. A. Taylor, "Multi-objective optimization of multi-level dc-dc converters using geometric programming," *IEEE Transactions on Power Electronics*, vol. 34, no. 12, pp. 11 912–11 939, 2019.
- [97] Y. Son, S. Mukherjee, R. Mallik, B. Majmunović, S. Dutta, B. Johnson, D. Maksimović, and G.-S. Seo, "Levelized cost of energy-oriented modular string inverter design optimization for pv generation system using geometric programming," *IEEE Access*, vol. 10, pp. 27 561–27 578, 2022.
- [98] A. Stupar, D. Flumian, B. Gouédard, and T. Meynard, "Generation and derivation of practical optimization-oriented models of inductors," *2019 20th Workshop on Control and Modeling for Power Electronics (COMPEL)*, pp. 1–8, 2019.
- [99] W. Hoburg and P. Abbeel, "Fitting geometric programming models to data," *Optimization and Engineering*, 2014.
- [100] K. Koh, S. Kim, A. Mutapcic, and S. Boyd, "gpposy : A matlab solver for geometric programs in posynomial form," *Technical report*, 2006.

- [101] A. Furlan, T. Meynard, A. Morentin, M. L. Heldwein, and G. Fontes, “Fast hybrid design approach for optimization of inductors,” *PCIM Europe digital days 2020; International Exhibition and Conference for Power Electronics, Intelligent Motion, Renewable Energy and Energy Management*, pp. 1–8, 2020.
- [102] A. Mutapcic, K. Koh, S. Kim, and S. Boyd, “Ggplab version 1.00 : a matlab toolbox for geometric programming,” 2006.
- [103] “Ihxx and power inductor weights,” Vishay, Tech. Rep., Mar 2024. [Online]. Available : <https://www.vishay.com/docs/34425/ihpowerinductorweights.pdf>
- [104] O. Grodzevich and O. Romanko, “Normalization and other topics in multi-objective optimization,” 2006.
- [105] G. Xu, “Global optimization of signomial geometric programming problems,” *European Journal of Operational Research*, vol. 233, no. 3, pp. 500–510, 2014. [Online]. Available : <https://www.sciencedirect.com/science/article/pii/S0377221713008394>

Titre : Étude d'un Convertisseur DC-DC 20kW pour application aéronautique : Prototypages et Développement d'un algorithme d'optimisation Mixed-Discrete Geometric Programming

Mots clés : Multiniveaux, Capacités flottantes, Routage, GaN, Optimisation, Programmation géométrique

Résumé : Dans un contexte industriel de réduction de la consommation d'énergies fossiles et d'émission de gaz à effet de serre, l'électrification des moyens de transports joue un rôle important dans la transition écologique. Les Piles à Combustible hydrogène sont le sujet de nombreux projets d'études dans le secteur de l'aéronautique. Les travaux ici présentés s'intéressent à l'implémentation d'une Pile à Combustible comme alimentation électrique pour un réseau embarqué d'aéronef 28V DC, 20kW nominal. Plus spécifiquement, l'objet de cette thèse est le développement d'un Convertisseur Électrique moyenne puissance, forts courants, adaptant la puissance en sortie d'une Pile à Combustible au réseau électrique embarqué. Les enjeux de l'aéronautique imposent des critères exigeants d'optimisation de la masse et du rendement du convertisseur. Une comparaison entre plusieurs topologies de convertisseurs a été réalisée, un convertisseur à capacités flottantes 3-niveaux a été retenu. Des transistors de type GaN ont été sélectionnés pour leur faible résistance à l'état passant et leurs vitesses de commutation élevées, permettant de minimiser les pertes du convertisseur, ainsi que la taille des composants passifs et de refroidissement du convertisseur. Un soin particulier a été apporté au routage des circuits imprimés pour réduire les impédances parasites et permettre une mise en parallèle de deux transistors pour chaque interrupteur. Un prototype présentant un rendement expérimental moyen de 98.25% a été produit.

Afin d'améliorer la rapidité et la pertinence des travaux d'optimisation des design des convertisseurs électriques, un nouvel algorithme d'optimisation a été développé à partir de travaux préexistants. Cet algorithme Mixed-Discrete Geometric Programming (MDGP) permet de résoudre des problèmes physiques, dotés d'un mélange de nombreuses variables d'optimisation continues et discrètes, dans un temps raisonnable tout en garantissant une convergence vers leur optimum global. Il est capable de gérer des variables discrètes multidimensionnelles, ainsi que des fonctions substituables. L'algorithme permet d'évaluer et de sélectionner des composants électriques spécifiques, plutôt que de représenter une gamme entière de composants par une modélisation continue imprécise. Les transistors ont notamment pu être modélisés sous forme de jeux de fonctions, différents pour chaque composant, fittés sur des résultats de simulation. Un front de Pareto suivant les critères de la masse et du rendement du convertisseur a été obtenu en 25h de calcul sur un ordinateur portable, pour un problème comportant 161.28 millions de combinaisons discrètes possibles.

Title: Study of a 20kW DC-DC Converter for aircraft applications : Prototyping and development of a Mixed-Discrete Geometric Programming optimization algorithm

Key words: Multilevel, Flying capacitors, Routing, GaN, Optimization, Geometric programming

Abstract: As the industry strives to reduce its consumption of fossil fuels and the associated greenhouse gas emissions, vehicle electrification plays a major role in the Ecological Transition. There is a growing number of studies aiming at evaluating and increasing Hydrogen Fuel Cells viability as a power supply for aircraft. This work aims to study the implementation of a hydrogen Fuel Cell as an electrical power supply for an aircraft 28V DC, 20kW on-board electrical network. More specifically, the subject of this thesis is the development of a low-voltage, high current Power Converter adapting the power flowing from the Fuel Cell, to the on-board electrical network. Converter mass, volume and efficiency are major optimization criteria in aircraft design. A comparison between several converter topologies has been realized, and a 3-level Flying Capacitor Converter has been selected.

GaN transistors were selected for their low on-state resistance and their high switching speed, allowing the reduction of the converter power losses and of the passive components and heat sink sizes. Particular attention was provided to printed circuit board routing in order to reduce parasitic impedance and to parallelize two transistors for each power switch. A prototype has been realized, measurements show an experimental average efficiency of 98.25%.

In order to enhance the speed and pertinence of optimization studies for electrical converters design, a new optimization algorithm has been developed based on pre-existing works. This Mixed-Discrete Geometric Programming algorithm (MDGP) can solve physics problems composed of a mix of many continuous and discrete optimization variables, in a reasonable calculation time while still guaranteeing convergence to the global optimum. It can handle multidimensional discrete variables, as well as substitutable functions. The algorithm allows to evaluate and select specific electrical components, instead of representing a range of components by a single continuous and imprecise modeling. Specifically, transistors are modeled as collections of functions, different for each component, fitted on simulation results. A Pareto Front for the converter mass and efficiency was generated in a 25h calculation time on a laptop, for an optimization problem involving 161.28 million possible discrete combinations.