Institut für
Elektrische Energiewandlung
Prof. Dr.-Ing. Nejila Parspour
Jun.-Prof. Dr.-Ing. Stefan Mönch

Universität Stuttgart

**Research Project**

# AI-Driven Efficiency Optimization of a Dual Active Bridge for Bidirectional EV Charging

vorgelegt an der Universität Stuttgart von

Diego San José Vega

| | |
|---|---|
| Matrikelnummer: | 3816356 |
| IEW-Nummer: | IEW 4039 |
| Ausgabe der Arbeit: | 15.03.2025 |
| Abgabe der Arbeit: | 7.08.2025 |
| Betreuer: | M.Sc. Ole Bauer |
| Prüfer: | Jun.-Prof. Dr.-Ing. Stefan Mönch |

# I. Abstract

The rapid global transition toward electromobility and renewable energy systems has highlighted the need for efficient, scalable, and intelligent power conversion technologies. The Dual Active Bridge (DAB) converter, known for its high efficiency and bidirectional operation, has become a popular component in the field of electric vehicle (EV) chargers, where it is used as a DC-DC stage. This research focuses on optimizing the design of the DAB converter for bidirectional EV chargers by integrating advanced computational tools, such as artificial intelligence (AI), to enhance efficiency and reduce the engineering workload in converter design.

The primary objective of this project is to develop a practical and generalized workflow that leverages on AI to optimize the efficiency of a DAB converter under a real-time control scenario for a continuous range of load conditions. By utilizing *PLECS* simulations and AI supervised prediction models, this work aims to identify the most combination of control parameter to achieve peak efficiency operating points for the converter, thereby improving the overall performance of EV chargers. The proposed methodology combines both classification and regression models to predict power losses and efficiency, using a comprehensive dataset derived from simulated DAB converter behaviour.

Through the application of this AI-driven workflow, the study demonstrates significant improvements in the efficiency of the converter across a wide range of operating conditions typical of an EV charger with bidirectional nature. The results indicate that AI can effectively enhance the design of power electronics, making the optimization process faster, more accurate, and less reliant on manual engineering interventions. The findings contribute to the ongoing development of high-performance and cost-effective solutions for the next generation EV chargers and renewable energy systems.

# II. Table of content


**I. Abstract ..... I**

**II. Table of content ..... II**

**III. List of Tables ..... IV**

**IV. List of Figures ..... V**

**V. List of Equations ..... IX**

**VI. Listings ..... X**

**VII. List of Abbreviations ..... XI**

**VIII. List of Symbols ..... XIII**

**1 Introduction ..... 1**

*1.1 Background and Motivation ..... 1*

*1.2 Presentation of DAB topology and design ..... 3*

*1.3 Problem Statement ..... 5*

*1.4 Objectives and Research Questions ..... 6*

*1.5 Scope and Limitations ..... 7*

*1.6 Structure of thesis ..... 7*

**2 Literature Review ..... 9**

*2.1 Working principles of DAB converter ..... 9*

*2.2 Efficiency Related to Design Decisions ..... 11*

2.2.1 Resonant Tank ..... 11

2.2.2 Switches and Losses ..... 13

*2.3 Control Strategies for a DAB Converter ..... 15*

2.3.1 Single Phase Shift (SPS) ..... 16

2.3.2 Extended Phase Shift (EPS) ..... 16

2.3.3 Dual Phase Shift (DPS) ..... 17

2.3.4 Triple Phase Shift (TPS) ..... 17

*2.4 Artificial Intelligence ..... 18*
2.4.1 Introduction to AI ..... 18
2.4.2 Machine Learning ..... 18
2.4.3 Deep Learning and Neural Networks ..... 19
2.4.4 Data Augmentation for Tabular Data ..... 20

**3 Design of Solution ..... 22**

*3.1 Introduction and Workplan ..... 22*

*3.2 Dataset ..... 23*
3.2.1 Data Gathering through PLECS simulations ..... 23
3.2.2 Data Treatment ..... 29

*3.3 AI Training ..... 42*
3.3.1 AI Model Architectures ..... 42
3.3.2 Classification Predictive Model ..... 46
3.3.3 Regression Predictive Model ..... 48
3.3.4 Complete Model (Classification & Regression) ..... 50

*3.4 Optimization Solver Prediction Model ..... 53*
3.4.1 Prediction Generation ..... 53
3.4.2 Generation of single solution ..... 56

**4 Realisation of Solution ..... 60**

*4.1 Regions and Waveforms Analysis ..... 60*
4.1.1 Region Presentation ..... 60
4.1.2 Non-stable Region ..... 61
4.1.3 Stable Region ..... 64
4.1.4 Critical Regions ..... 66

**5 Conclusions and Future Work ..... 71**

**6 References ..... 73**

**7 List of Appendices ..... 81**

**IX. Erklärung ..... 84**

# III. List of Tables

**Table 2-1. Control Strategies comparison [25, 29]. 15**

**Table 3-1. Variables simulation ranges. 28**

**Table 3-2. Simulation constants. 28**

**Table 3-3. Recorded variables on dataset from simulations. 29**

**Table 3-4. Data augmentation for training vectors of regression model. 38**

**Table 3-5. Data augmentation for training vectors of regression model. 39**

**Table 3-6. PLECS and AI model data gathering comparison. 54**

# IV. List of Figures

**Figure 1-1. Sales tendency in millions of plug-in cars [3]. ..... 2**

**Figure 1-2. Technology tendency for the decades to come [1]. ..... 2**

**Figure 1-3. Scheme representation of a AC-DC & DC-DC converters for a battery charger [8]. ..... 3**

**Figure 1-4. DAB topology [9]. ..... 4**

**Figure 1-5. Primary Secundary side of a DAB Converter [9]. ..... 4**

**Figure 2-1. Current through L based on primary and secondary switches [28]. ..... 9**

**Figure 2-2. DAB topology [9]. ..... 10**

**Figure 2-3. Reduced design of a DAB [25]. ..... 10**

**Figure 2-4. LC Resonant tank [25]. ..... 11**

**Figure 2-5. LLC Resonant tank [25]. ..... 12**

**Figure 2-6. CLLC resonant tank [25]. ..... 12**

**Figure 2-7. Power losses classification [37]. ..... 13**

**Figure 2-8. SPS modulation [25]. ..... 16**

**Figure 2-9. EPS Modulation [25]. ..... 16**

**Figure 2-10. DPS Modulation [25]. ..... 17**

**Figure 2-11. TPS modulation [25]. ..... 17**

**Figure 2-12. AI, machine learning and deep learning [44]. ..... 18**

**Figure 2-13. Machine learning classification [46]. ..... 19**

**Figure 2-14. Neuron definition [47]. ..... 19**

**Figure 2-15. Neural Network [47]. ..... 20**

**Figure 3-1. Methodology diagram. ..... 22**

**Figure 3-2. Stage 1: Data Gathering. ..... 23**

**Figure 3-3. PLECS Demo Model of Dual Active Bridge [50]. ..... 24**

**Figure 3-4. PLECS complete simulation model. 24**

**Figure 3-5. Triple phase shift control model on PLECS. 25**

**Figure 3-6. Efficiency calculation and stopping block diagram 26**

**Figure 3-7. Stage 1.2: PLECS model control with python. 26**

**Figure 3-8. Stage 2: Data treatment. 29**

**Figure 3-9. Voltage threshold over sample number. 31**

**Figure 3-10. Efficiency histogram of not stable and stable samples. 31**

**Figure 3-11. Power losses over $D_1$ and $D_2$ for $V_{OUT}$ = 920 V & $P_{OUT}$ = 10 kW. 32**

**Figure 3-12. Efficiency over $D_1$ and $D_2$ for $V_{OUT}$ = 920 V & $P_{OUT}$ = 10 kW. 32**

**Figure 3-13. Failed results of first training strategy. 33**

**Figure 3-14. AI complete model proposal. 33**

**Figure 3-15. Efficiency over power losses and histogram for all samples. 34**

**Figure 3-16 Efficiency over power losses and histogram for best efficiency samples. 34**

**Figure 3-17. Regression model relative error grid. 35**

**Figure 3-18. Classification model false positive grid. 35**

**Figure 3-19. Original sweep grid. 35**

**Figure 3-20. Final data grid with complementary data. 36**

**Figure 3-21. Inductance current waveforms for region 2. 36**

**Figure 3-22. Training set augmented data over original data grid for classification model.. 39**

**Figure 3-23. Training set augmented over original data grid for regression model. 40**

**Figure 3-24. Augmented jitter data comparison for regression model. 40**

**Figure 3-25. Augmented data with jitter over original data comparison. 41**

**Figure 3-26. Stage 3: AI Classification model. 46**

**Figure 3-27. Confusion matrix results for augmented and original data trained architectures. 47**

**Figure 3-28. Best classification result for LightGBM architecture. 47**

**Figure 3-29. Stage 4: AI regression model. 48**

**Figure 3-30. Regression model results for original and augmented trained architectures... 48**

**Figure 3-31. Best regression model results for Pytorch Dense NN. ..... 49**

**Figure 3-32. Relative Error results for Pytorch Dense NN. ..... 49**

**Figure 3-33. Relative Error results for Pytorch Dense NN. ..... 49**

**Figure 3-34. Stage 5.1: Complete model selection. ..... 50**

**Figure 3-35. Stage 5.2: AI complete model. ..... 50**

**Figure 3-36. Validation set classification result for Complete Model. ..... 51**

**Figure 3-37. Validation set regression result for complete model. ..... 52**

**Figure 3-38. Relative error results for complete model for validation set. ..... 52**

**Figure 3-39. Stage 6: Solution Generation ($D_1$, $D_2$) ..... 53**

**Figure 3-40. Prediction for $P_{LOSS}$ and efficiency over $D_1$ $D_2$ for $V_{OUT}$ = 800 V $P_{OUT}$ =8. 5kW. 54**

**Figure 3-41. Best prediction results over $D_1$ & $D_2$ for $V_{OUT}$ = 800 V & $P_{OUT}$ =8.5 kW. ..... 55**

**Figure 3-42. Volume of solutions for $D_1$. Figure 3-43. Volume of solutions for $D_2$. ..... 55**

**Figure 3-44. Real Time control integration of the final solution. ..... 56**

**Figure 3-45. Stage 7: Single solution solver. ..... 57**

**Figure 3-46. Data clustering $D_1$ with DBSCAN. ..... 58**

**Figure 3-47 Data clustering for $D_2$ with DBSCAN. ..... 58**

**Figure 3-48. Smoothed single solution for $D_1$ (1). ..... 58**

**Figure 3-49. Final smoothed single solution for $D_2$ (1). ..... 58**

**Figure 3-50. Smoothed single solution for $D_1$ (2). ..... 59**

**Figure 3-51. Final smoothed single solution for $D_2$ (2). ..... 59**

**Figure 3-52. Heatmap top view of smooth mesh solution for $D_1$ & $D_2$. ..... 59**

**Figure 4-1. Region definition on the Heatmap. ..... 60**

**Figure 4-2. Voltage output not reaching the output reference. ..... 62**

**Figure 4-3. Inductance current for working point not reaching the reference. ..... 62**

**Figure 4-4. Low efficiency point region. ..... 63**

**Figure 4-5. Efficiency graph for $D_1$ and $D_2$. ..... 64**

**Figure 4-6. Compare of power loss for high and low power output. ..... 65**

**Figure 4-7. Inductance current waveforms for region 1. ..... 65**

**Figure 4-8. Inductance current waveforms for the border on high power conditions. ..... 67**

**Figure 4-9. Inductance current waveforms for low voltage conditions. ..... 68**

**Figure 4-10. Best efficiency reached for load conditions grid..... 69**

# V. List of Equations

**Equation 2-1. Current through the inductance. 10**

**Equation 2-2. Power transferred from $V_1$ to $V_2$. 11**

**Equation 2-3. Conduction losses of a MOSFET [38]. 13**

**Equation 2-4. Switching losses of a MOSFET [38]. 13**

**Equation 3-1. Total number of simulations. 29**

# VI. Listings

**Listing 3-1. Library presentation and model port access. ..... 27**

**Listing 3-2. Load variables and start simulation. ..... 27**

**Listing 3-3. Results label gathering. ..... 27**

**Listing 3-4. Weighted average calculation. ..... 30**

**Listing 3-5. Supervised learning input vector definition for classification model. ..... 37**

**Listing 3-6. Supervised learning input vector definition for classification model. ..... 37**

**Listing 3-7. SMOTEENN definition for classification model. ..... 38**

**Listing 3-8. Jitter definition for classification model. ..... 38**

**Listing 3-9. Jitter definition for classification model. ..... 39**

**Listing 3-10. Pipeline definition for xgboost ..... 42**

**Listing 3-11. Hyper-parameter coarse-to-fine method. ..... 43**

**Listing 3-12. Pipeline of lightGMB. ..... 43**

**Listing 3-13. Class definition of SkipConnectionNN. ..... 44**

**Listing 3-14. Class definition of LeakySwishNN. ..... 45**

**Listing 3-15. Class definition of DenseNN. ..... 46**

**Listing 3-16. Complete model prediction function. ..... 51**

**Listing 3-17. DBSCAN Clustering definition. ..... 58**

# VII. List of Abbreviations

| | |
|---|---|
| EV | Electrical Vehicule |
| V2G | Vehicle-to-Grid |
| V2H | Vehicle-to-Home |
| G2V | grid-to-vehicle |
| DAB | Dual Active Bridge |
| AI | Artificial Intelligence |
| ML | Machine Learning |
| DL | Deep Learning |
| ANN | Artificial Neural Network |
| NN | Neural Network |
| SPS | Single Phase Shift |
| EPS | Extended Phase Shift |
| DPS | Dual Phase Shift |
| TPS | Triple Phase Shift |
| ZVS | Zero Voltage Switching |
| ZCS | Zero Current Switching |
| SMOTEENN | Synthetic Minority Over-sampling Technique Edited Nearest Neighbor |

| | |
|---|---|
| XGB | eExtreme Gradient Boosting |
| LightGBM | Light Gradient Boosting Machine |
| RMSE | Root Mean Square Error |
| AUC | Aera Under the Curve |
| MOSFET | Metal-Oxide-Semiconductor Field-Effect Transistor |
| DBSCAN | Density-Based Spatial Clustering of Applications with Noise |

# VIII. List of Symbols

| *Symbol* | *Units* | *Description* |
|---|---|---|
| $D_1$ | - | Primary Side Duty Cycle |
| $D_2$ | - | Secondary Side Duty Cycle |
| $d$ | - | Phase Between Bridges |
| $P_{OUT}$ | W | Output Power |
| $V_{IN}$ | V | Input Voltage |
| $V_{OUT}$ | V | Output Voltage |
| $P_{OUT\ REF}$ | W | Output Reference Power |
| $V_{OUT\ REF}$ | V | Output Reference Voltage |
| $f_s$ | Hz | Switching Frequency |
| $P_{LOSS}$ | W | Power Losses |
| $I_L$ | A | Inductance Current |
| $V_1$ | V | Voltage Primary Bridge |
| $V_2$ | V | Voltage Secondary Bridge |
| $L$ | H | Inductance |
| $C$ | F | Capacitor |
| $R_{LOAD}$ | Ω | Load Resistance |

| | | |
|---|---|---|
| $P_{COND}$ | W | Power Conduction Losses |
| $P_{SW}$ | W | Power Switching Losses |
| $I_{RMS}$ | A | Current Root Mean Square |
| $R_{DS}$ | Ω | Resistance Drain-Source |
| $V_{DS}$ | V | Voltage Drain-Source |
| $I_{LOAD}$ | A | Current Load |
| $t_{ON}$ | s | Time switching on |
| $t_{OFF}$ | s | Time switching off |

# 1 Introduction

The present chapter covers the current situation and demand regarding the use of the Dual Active Bridge (DAB) converter within industry. The climate change situationship has had a huge impact on the markets, specifically on the automotive industry where the manufacturing of Electrical Vehicles (EV) has risen rapidly throughout recent decades.

Due to that, the analysis and study regarding the efficiency and design of the DAB topology has indirectly risen and currently holds a great amount of interest and investigations.

Through the next chapter, this tendency is better introduced following the presentation of the studied topology and the assumption of the situation, problems and proposed solution that the industry is turning to face for the decades to come.

## 1.1 Background and Motivation

The accelerated global transition toward electromobility and renewable energy systems has brought forth critical challenges and opportunities in the design of efficient, scalable, and intelligent power conversion systems. A pivotal component within this domain has impacted the automative industry where the electrical vehicles manufacturing has turned into the investigation of bidirectional AC-DC converters due to not only facilitating charging and discharging of electric vehicle batteries but also supports vehicle-to-grid (V2G) and vehicle-to-home (V2H) applications. Among the most promising topologies for isolated DC-DC conversion in such systems is the **Dual Active Bridge**, due to its flexibility, soft-switching capabilities, and suitability for wide voltage ranges.

The global automotive industry is undergoing a transformative shift towards electrification, driven by climate commitments, and the exponential adoption of electric vehicle [1, 2]. Over the past decade, plug-in vehicle sales have experienced significant growth, especially in markets such as China, Europe, and the United States. Global plug-in car sales grew from less than 0.1 million units in 2011 to over 1.1 million in 2018, with China leading the uptake. At the national level, countries like Norway, the UK, and Germany have shown marked increases in EV adoption (see Figure 1-1), with Norway surpassing 60,000 units in annual sales by 2018, largely due to **aggressive electrification policies and incentives** [3].

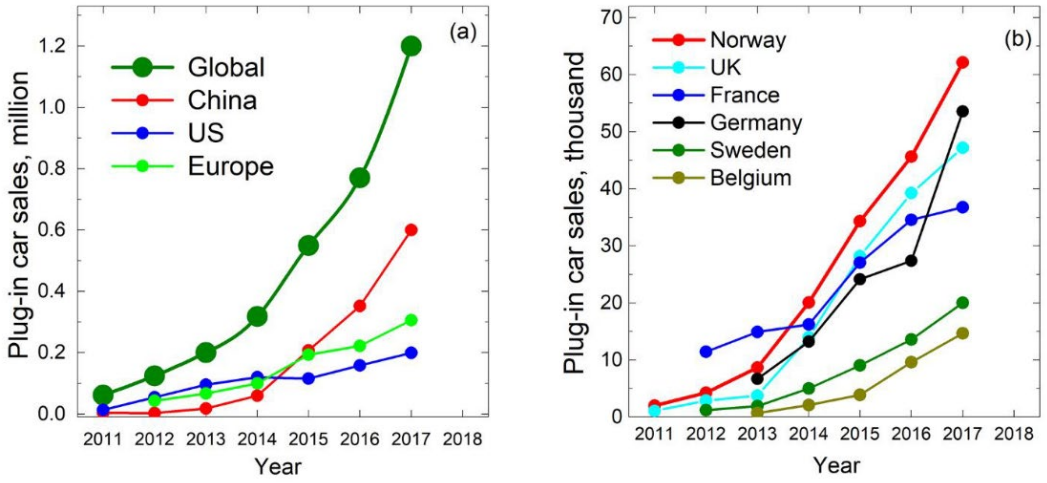


*Figure 1-1. Sales tendency in millions of plug-in cars [3].*

This growth reflects a broader systemic transition toward low-carbon and renewable energy systems. As shown in Figure 1-2, electrification of end-use sectors, such as transportation, is a key pillar in future energy systems alongside decentralization, digitization, and flexible infrastructure. Electric vehicles are increasingly seen not only as transportation assets but as **energy resources** capable of providing grid services through bi-directional power flow. Integrating EVs with the renewable grid calls for advanced power electronics that enable efficient, intelligent, and reversible energy transfer between batteries and the utility grid [1, 4].

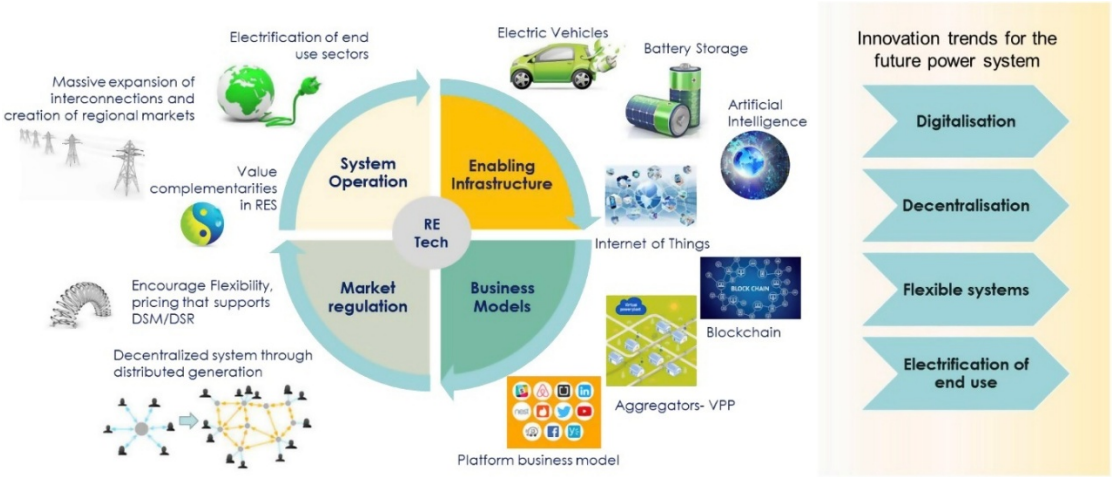


*Figure 1-2. Technology tendency for the decades to come [1].*

In this context, **bidirectional EV chargers** have emerged as a critical technological enabler. Indeed, these chargers allow not only for energy intake (G2V) but also for energy discharge (V2G or V2H), thus supporting energy arbitrage and backup supply [5–7].

A typical bidirectional charger consists of two main power conversion stages. First, an **AC-DC converter**—often implemented as an active front end or power factor corrector, and then an isolated **DC-DC converter** that connects directly to the battery, as depicted in Figure 1-4 [8].

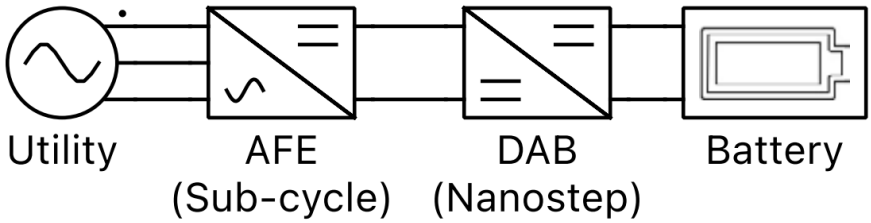


*Figure 1-3. Scheme representation of a AC-DC & DC-DC converters for a battery charger [8].*

Therefore, the optimization and achievement of a valid and efficient EV charger goes through the focus and optimization of two topologies (AC-DC and the DC-DC converter) which can be separately analysed.

So on, the present research paper focuses on the **DC-DC converter stage** of the bidirectional charger for an EV which relates to the DAB topology. From this point on, the thesis work will present and in-deep study the usage and operation of this topology.

## 1.2 Presentation of DAB topology and design

A conventional DAB topology, shown in Figure 5 [9], consists of two full-bridge converters on either side of a high-frequency transformer. When the power flows left to right, the left-side full bridge processes the input voltage $v_i$, while the right-side bridge delivers the regulated output voltage $v_o$. Both bridges manage the current flow on each side through the switching of four **MOSFET switches** that allow to introduce different control strategies over the topology by changing their duty cycles and phases (see chapter 2.3). The transformer can provide either a voltage step-up or step-down based on application requirements—which is frequently know as a Buck or Boost mode operation [10, 11].

This current outflow between the two bridges charge and discharge an inductance $L$ that led to the voltage gap. The current through this inductance produces the main influence over the topology efficiency where its waveforms study and optimization is key to achieve an efficient working point [12]. The topology can be effectively modelled using an equivalent circuit, as shown in Figure 1-5 [9], where $v_p$ and $v_s$ represent square-wave voltages generated by the primary and secondary bridges, respectively, and $L$ is the total leakage inductance prior introduced referred to the primary side.

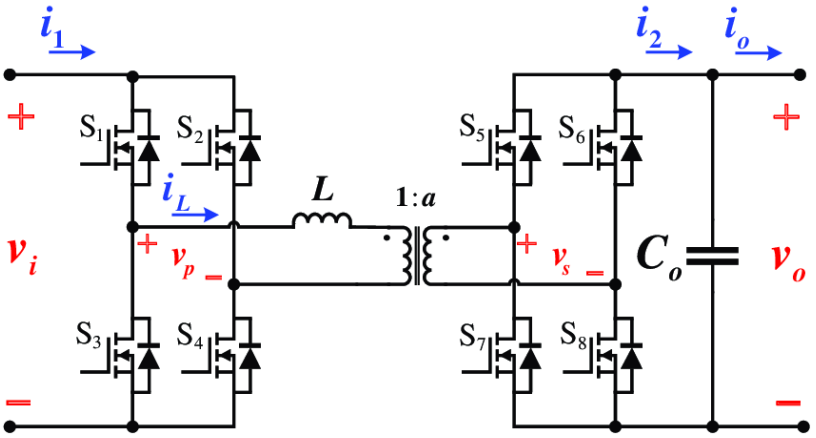


*Figure 1-4. DAB topology [9].*

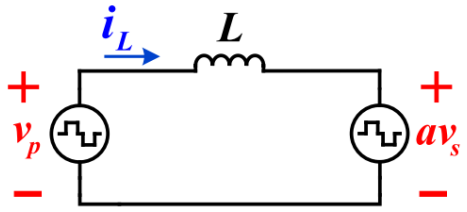


*Figure 1-5. Primary Secundary side of a DAB Converter [9].*

Key advantages of the DAB include [13, 14]:

- **Bidirectional operation** due to its symmetric hardware which is appealing for the bidirectional nature need of the EV chargers. This enables to produce and control a voltage step from either side of the topology.

- **Galvanic isolation**, improving safety and compliance in EV and grid-connected applications. Regarding the DAB topology, the galvanic isolation is provided by a high-frequency intermediary transformer that prevents any direct conductive path between the input and output DC stages—blocking both DC and unwanted AC currents—while enabling bidirectional energy transfer [15, 16].

- **Multiple control degrees of freedom** which results in several control strategies that may enhance conversion efficiency (see chapter 2.3).

- **High power density,** achieved by operating at high switching frequencies (typically above 50–100 kHz), which drastically reduces the volume and weight on magnetic components (transformers and inductors) [17]. Additionally, **soft-switching** techniques (Zero Voltage Switching or Zero Current Switching) minimize both switching and conduction losses, reduce thermal stress, and permit the use of smaller heat sinks and current paths, thereby further enhancing the converter's overall power density [18].

At any rate, design challenges include the accurate modelling of transformer leakage inductance, optimization of switching transitions to minimize losses, and selection of wide-bandgap semiconductors (e.g., SiC or GaN) to further increase efficiency and switching speed [12].

In summary, the DAB's versatility, combined with its efficiency and isolation properties, makes it an ideal candidate for next-generation EV charging systems and renewable energy converters, particularly in applications requiring bidirectional power flow and advanced control strategies. On the other hand, its accurate and efficient design establishes a challenging working environment due to the complex understanding of the converter.

## 1.3 Problem Statement

The inherent advantages of DAB converters are often tempered by the **depth of design complexity**, due to the difficulty of defining parameters for high efficiency, and the absence of a single, universally accepted design workflow, which leads to heavy human engineering workload that has not until recent years being leveraged with machine computation [19].

The DAB involves tightly coupled electrical, magnetic and control parameters that influence the behaviour of the converter, such as the switching frequency, leakage inductance, transformer turns ratio and design, semiconductor and modulation strategy This all interact non-linearly, creating a **high-dimensional optimization problem** without a single strategy solution.

Current efficiency-oriented workflows typically decompose the complex understanding of the DAB into **piecewise**-linear intervals or adopt **harmonic**-balance formulations, trading technical accuracy for complexity with engineer-driven calibration. Each new load condition demands manual boundary identification, exhaustive parameter sweeps and the pruning of Fourier terms—due to the nature of infinite harmonics [20–23]. As switching frequencies rise, the number of discrete regions and harmonic orders grows combinatorially, enlarging simulation times and forcing empirical correction factors that hamper engineering work [24]. In high-power, high-density applications this manual tuning becomes the critical path in the development schedule.

For that reason, for every operational point of the DAB converter, it is needed to thoroughly analyse and correlate the behaviour of design. Thus, the workload upon engineering manual designing tasks is harder and not usually generalized for a large operating ranged and therefore it has not been possible to stablish a generalize optimization model for the topology [25]. That is the main reason why last year's studies are focused on finding a generalized design methodology for the topology [23–25].

Recognizing this bottleneck, the problem regarding the efficient design of a dual active bridge is the **high workload upon manual engineering task and analysis** due to the absence of a generalized comprehensive model for the converter. This leads to current heavy manual mathematical analyses methods to look for optimization points from an efficiency-oriented perspective.

## 1.4 Objectives and Research Questions

Based on the statement of the previous chapter, the objective of this job is to develop a methodical, practical and generalized **workflow** to achieve an efficiency-optimized design for a given operating condition of the dual active bridge. By doing so, the aim of the proposed method approach is to **assist the design process with computational power,** so the manual workload upon human engineers is lighter and therefore the optimization process is more accurate, interactive and much quicker. The results aim to predict the most efficient values for the control parameters on a real-time control implementation of the converter for the typical load range of a bidirectional EV charger.

To achieve this goal, the research may be aware of the current methodologies and study approach upon the studied topology and intervene the process with practical computer tools such as artificial intelligence algorithms (see chapter 2.4), as well as understanding the working nature and principles of the converter (see chapters 2.1 to 2.3).

Therefore, the problems are shifted into achieving **a successful integration** of forefront technologies to suit and resolve an optimization problem. The sum of all those elements will give outcome to **a methodical workplan** that allows engineers to support the design task with state-of-art computer calculation that overall enhance the resolution of efficient calculation points for a given DAB operating condition.

In sum, the questions that will be answered in the present research are as follows:

- **Understand the nature of the complexity** regarding efficiency-oriented design of a DAB by undergoing the current methods, control strategies, design decisions and analysing studies to find an efficient working point for the topology.

- Find and develop **suitable technologies and algorithms** powered by computer engines to lighten the prior manual working method.

- Trace a deep, complete and functional **working path to empower engineers** with the ability to find accurate efficiency results on given operating requirements.

With that being said, the following job tries to find a solution to each point exposed with the aim of assembling all together and solving the main objective of the thesis.

## 1.5 Scope and Limitations

At this point of the introduction, it is worth being aware of the constraints faced by the proposed problem solution, so the research scope is better defined. In addition, a better working landscape is proposed in this chapter regarding the broader interpretation given at the first point of the introduction.

As previously defined, a bidirectional battery charger for an electrical vehicle is composed of an AC-DC and DC-DC converter. This thesis only focusses on the developing and studying of an efficient solution for the DC-DC stage driven by a dual active bridge topology. **This limits the investigation and focus on only the DAB topology**, which will be thoroughly analysed and explained in future chapters.

On the other hand, while optimizing the DAB converter, it will be needed to stipulate a boundary for the operating condition of the topology, so the aiming of the work will be to suit **operating necessities of the DAB mounted on an EVs bidirectional charger** (see chapter 3.2). This makes the proposed study more practical and accurate, meanwhile it is worth mentioning that the followed approach can be replicated and suited for many others desired applications for the DAB topology.

Furthermore, while working with **artificial intelligence**, the demand upon computer power can be compromising, so the work of the thesis is aware of the trade-off involved on data gathering and management and the performance and sustainability of the algorithm. So on, the working scope aims to achieve solutions with a feasible and reasonable machine power deployment.

Overall, this delimitation ensures a focused investigation on optimizing the DAB-based DC–DC stage under realistic EV charging conditions while explicitly acknowledging the computational trade-offs of AI-driven methods. Moreover, by clearly defining operating boundaries and algorithmic constraints, the proposed framework remains both practically feasible and readily adaptable to other DAB applications.

## 1.6 Structure of thesis

The remainder of this thesis is organized as follows:

- **Chapter 2: Literature Review:** Presents foundational principles of the DAB converter (Section 2.1), examines how key design decisions influence efficiency (Section 2.2), explain most frequent modulation strategies (Section 2.3), and introduces relevant artificial intelligence background and techniques suitable to power-electronics (Section 2.4). This chapter aim to gain a proper and thoughtful understanding of the studied problem and environment.

- **Chapter 3: Design of Solution:** It outlines the overall workplan (Section 3.1), details dataset preparation through *PLECS* simulation and data treatment (Section 3.2), develops and compares multiple predictive models (Section 3.3), and implements optimization solving strategies to resolve design selection (Section 3.4). It composes the main innovative and technical explanation regarding the proposal of the thesis.

- **Chapter 4: Realization of Solution:** It reviews the results from the AI-driven methodology to representative operating scenarios, presenting efficiency results from simulation (Section 4.1) and provides a final overview on the quality and usage of the results (Section 4.2). With those sections, the goal is to reflect and analyse the results of the proposed resolution.

- **Chapter 5: Conclusions and Future Work:** Where a summary of the main findings, reflects on the contributions to DAB converter design, and proposes avenues for extending this research.

# 2 Literature Review

## 2.1 Working principles of DAB converter

As previously introduced, the dual active bridge converter has been gaining recognition and usage as the developments over the power converters have enhanced. It was first introduced by Prof. Rik W. De Doncker in 1988 as a DC-DC converter solution to achieve low switching losses at high switching frequency [26, 27]. However, it was not until the popularization of the **silicon carbide and gallium-nitride** power devices that DAB really stated to be appealing for research [25].

The DAB is a high-power density converter for high frequency switching that enables to operate at **soft-switching conditions**, which makes the topology able to minimize switching losses and electromagnetic interferences (EMI). It is composed of two bridges that have two pairs of MOSFET switches each, which act alternatively. Those switches are connected by a transformer that charges and discharges based on the effect of the primary and secondary squared forms of the voltage produced by the switches [13]. Therefore, this topology enables three degrees of freedom regarding the composition and relation between the primary and secondary voltage functions (figure 2-1).

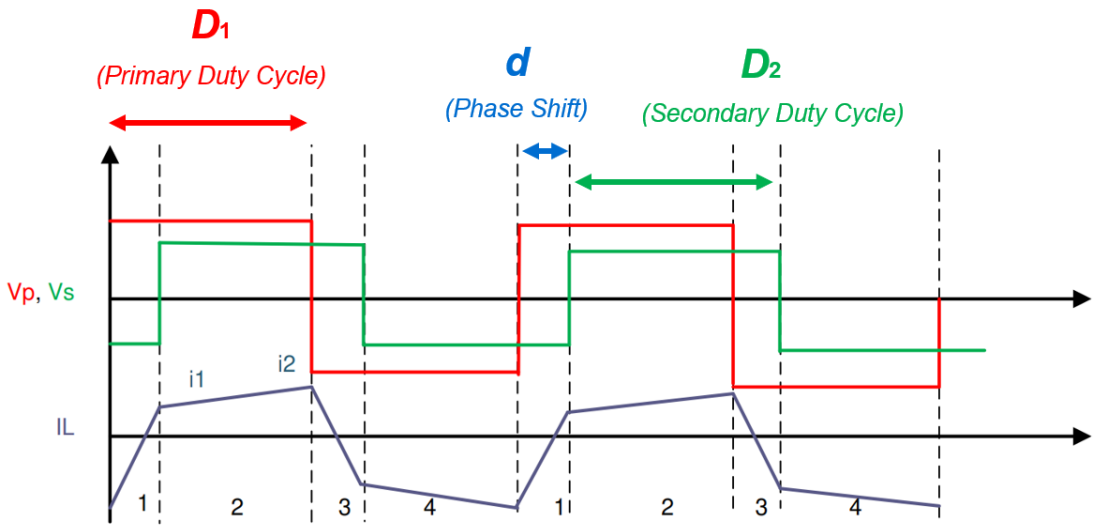


*Figure 2-1. Current through L based on primary and secondary switches [28].*

As shown on the previous chart (see figure 2-1), the difference of phase between the squared signal produced by each pair of switches outcome what we call the duty cycle of the voltage square signal connected to the terminal of the inductance *L*—on both the primary and secondary respectively. Therefore, we define the **duty cycle** for the **primary side** (***D*₁**) and for the **secondary side** (***D*₂**) voltage signals. Those values depend only on the switches on each side. Furthermore, as those signals are independent, it exists a third degree of freedom which is the **phase shift (*d*)** between the primary and secondary voltage squared signal. The sum of those factors influences the inductance *L* behaviour whose current will have a certain waveform and ripple. This directly affects the current output on the secondary side which produces the final voltage output $v_o$—on the capacitator $c_o$—with the same waveform as the current through the inductance *L* [11, 29–32].

The expression that defines the value of the current through the inductance is straightforward calculated form the equation that defines the behaviour of an inductance (see equation 2-1).

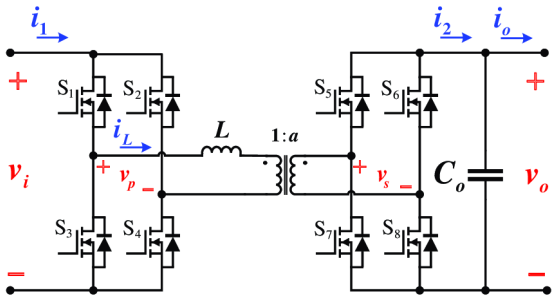


*Figure 2-2. DAB topology [9].*

$$\frac{di_L}{dt} = \frac{v_p - v_s}{L}$$

*Equation 2-1. Current through the inductance.*

Overall, the behaviour of the DAB (presented on figure 2-2) can be reduced to two bridges connected by an inductance where each bridge produces a squared voltage signal that influences the current through the inductance (see figure 2-3).

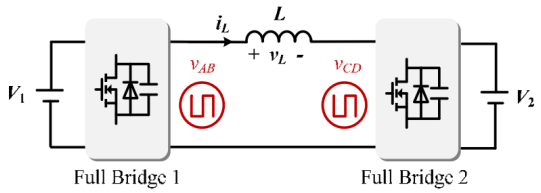


*Figure 2-3. Reduced design of a DAB [25].*

As a result, we can define **the power transferred** through the primary bridge to the secondary bridge as equation 2-1 [9, 13, 25]. This power equation describes the square form transferred from $V_1$ to $V_2$. The parameter of the phase shift (*d*) describes the

difference of phases between the primary and secondary side, which ended up influencing the amplitude and direction of the square signal of the power. This equation does consider that the power losses are minimal to simplify the calculation [25].

$$P = \frac{nV_1V_2}{2\pi^2 f_s L} d(\pi - d)$$

*Equation 2-2. Power transferred from $V_1$ to $V_2$.*

## 2.2 Efficiency Related to Design Decisions

The DAB topology can offer high efficiency points, but that efficiency relies on many different aspects of the design. Two of the main aspects regarding efficiency are: the resonant tank, and the efficiency of the switches.

Regarding the first condition, a DAB converter can work with different passive components to connect the two bridges, which is known as the resonant tank of the topology. For the switches, each bridge is composed by four MOSFET that due to the high frequency demand, they induce in conduction and switching losses that affect the efficiency of the converter.

### 2.2.1 Resonant Tank

The thoughtful design of a resonant tank for a dual active bridge converter is mainly important to expand the soft-switching range of the converter (see chapter 2.2.2) and reduce core losses, which can lead to higher efficiency [33, 34].

#### A. LC Tank

This type of resonant tank includes a capacitator connected in series with the inductance. It is the simplest construction of the resonant tank which highly reduces the design load and aims to mitigate power losses under wide range variation of voltage and current. That also enables fast transition speeds but the main downsides are the reduction of the soft-switching range and the increase in cost and size [25].

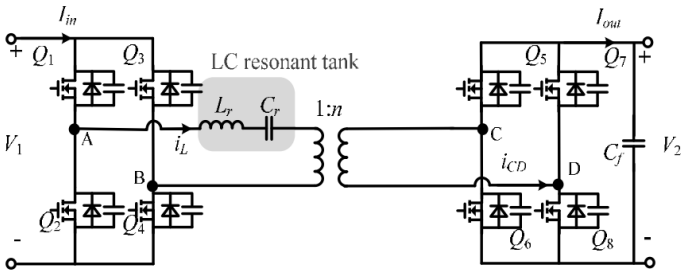


*Figure 2-4. LC Resonant tank [25].*

## B. LLC Tank

This design includes a magnetizing inductance called $L_m$ that enables to produce zero voltage switching (ZVS) for the primary bridge and zero current switching (ZCS) for the secondary bridge. This gives higher efficiency that the previous resonant tank due to the much slim switching frequency variation. In contrast, it is highly complicated to design accurately and increase cost and size as well as achieving a lower transition speed than the previous one [25].

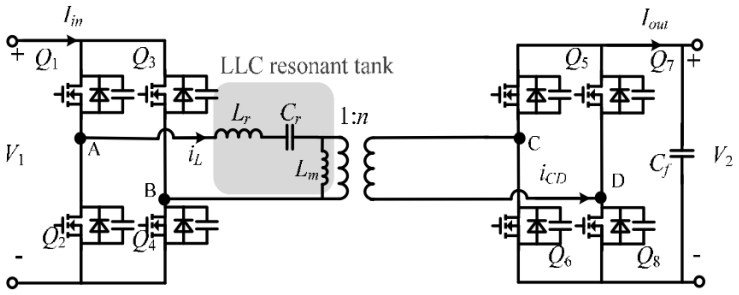


*Figure 2-5. LLC Resonant tank [25].*

## C. CLLC Tank

This construction enables to reduce voltage stresses and allows identical operation for opposite direction of the power. This bestows the topology with wider soft-switching range and narrower switching frequency variation which overall lead to better performance and higher efficiency. On the contrary, it is heavy complicated to design, has larger size than the previous alternatives and a lower transition speed [25, 35, 36].

This tank can be constructed symmetrically or asymmetrically. If constructed symmetrically, it easier to design and allows symmetric operation, and if designed asymmetrically stands for the generalized and more versatile version but it also outstands as the more complicated to design efficiently [25].

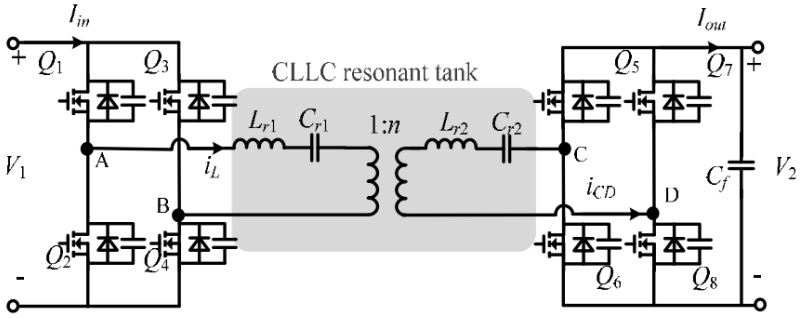


*Figure 2-6. CLLC resonant tank [25].*

For all the resonant tanks here mentioned, there are many techniques that are currently under research to achieve an efficient design (see references [25, 33–36]). For the present work, this scope falls out of reach due to time constraints. Anyway, the presented methodology (see chapter 3) is independent of the resonant tank and can be applied to any of the previous construction. In sum, the design and thoughtful selection of a suitable resonant tank is highly important to hugely upgrade the efficiency performance of the converter.

### 2.2.2 Switches and Losses

The power losses can be classified as on figure 2-7, which describes the losses possibly encounter at each element of the converter [37].

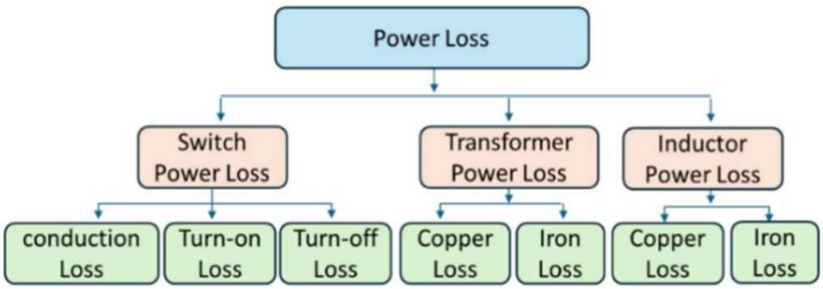


*Figure 2-7. Power losses classification [37].*

Regarding the losses that appear on the MOSFET, they can be described as the **switching and conduction losses** which behaviour is represented with equation 2-3 and 2-4 [38]. To avoid or minimize switching losses, commutation with ZVS or ZCS must be achieved.

$$P_{COND} = I_{RMS}^2 R_{DS(on)} n$$

*Equation 2-3. Conduction losses of a MOSFET [38].*

$$P_{SW} = \frac{1}{2} V_{DS} I_{LOAD} (t_{ON} + t_{OFF}) f_s$$

*Equation 2-4. Switching losses of a MOSFET [38].*

#### A. Zero Voltage Switching (ZVS)

Zero Voltage Switching is a technique that aims to minimize switching losses and electromagnetic interference by ensuring that a transistor turns on when the voltage across it is nearly zero or zero. In a DAB converter, ZVS is naturally achieved during the **dead time** between the turn-off of one MOSFET and the turn-on of another. During this interval, the energy stored in the transformer's leakage inductance discharges the output capacitances of the MOSFETs, bringing their voltages close to zero before they are turned on.

Achieving ZVS in a DAB converter depends on several factors such as the stored inductive energy or MOSFET output capacitance. The first concerns the energy available to discharge the MOSFET capacitances which is proportional to the inductance and the square of the current. The other one regards the energy required to charge the MOSFET's output capacitance in accordance with the capacitance and the square of the voltage [18].

For ZVS to occur, the stored inductive energy must be sufficient to discharge the MOSFET capacitances. This condition is **influenced by the load of the converter** and the input-to-output voltage ratio. Under light load conditions, achieving ZVS becomes challenging due to the reduced inductive energy available. In such scenarios, precise control of the dead time and consideration of the non-linearity of the MOSFET's parasitic capacitance are crucial for ensuring ZVS.

However, achieving ZVS requires careful design of the **dead time** and **modulation strategy.** The dead time must be optimized to allow sufficient energy transfer without causing shoot-through conditions. Advanced modulation techniques, such as the Extended Phase Shift (EPS) or Triple Phase Shift (TPS) (see chapter 2.3), can be employed to extend the ZVS operating range and improve efficiency across varying load conditions.

## B. Zero Current Switching (ZCS)

Similarly to the previously explained concept, Zero Current Switching is a technique in power electronics where a switch turns off when the current through it is zero or near zero. This aim to minimize switching losses and reduce electromagnetic interference (EMI), similar to ZVS, but **focuses on the waveform of the current** rather than the voltage waveform.

Achieving ZCS requires careful design of the converter's resonant circuit and modulation strategy. The resonant inductor and capacitor must be selected to ensure that the current through the switching device reaches zero before it is turned off. This often involves using modulation techniques that **control the timing of the switch-off to coincide with the zero-crossing of the current.**

This soft-switching technique also reduces EMI by eliminating the abrupt voltage and current transitions typical of hard-switching methods. Additionally, this optimization translates into lower heat generation, simplifying thermal management and potentially allowing for smaller heat sinks [39].

In summary, Zero Current Switching is a valuable technique in power electronics that complements Zero Voltage Switching. While ZVS focuses on minimizing voltage-related losses during turn-on, ZCS addresses current-related losses during turn-off.

### C. Soft Switching Range

The soft switching range of a power converter refers to **the operational window within which the converter can achieve ZVS or ZCS**, thereby minimizing switching losses and electromagnetic interference. This range is influenced by factors such as load conditions, input and output voltage, the design of the resonant circuit and dead time.

In converters employing ZVS, the soft switching range is primarily determined by the input voltage and the transformer leakage inductance. At higher input voltages, the energy required to charge the MOSFET's output capacitance increases, potentially extending the ZVS range. Conversely, at lower input voltages, achieving ZVS becomes challenging due to insufficient energy for charging the capacitance, thereby narrowing the soft switching range. Similarly, in ZCS converters, the soft switching range is influenced by the load current and the resonant tank parameters. Under light load conditions, the current through the switch may not reach zero before turn-off, limiting the ZCS range.

To extend the soft switching range, advanced modulation techniques have been developed to optimize the timing of switch transitions, thereby enhancing the ZVS range across varying load conditions [40–42].

## 2.3 Control Strategies for a DAB Converter

As previously introduced, the DAB converter can be controlled with a total of three degrees of freedom—***$D_1$, $D_2$* and *d*.** The next table compares the most frequently used controls strategies for a DAB converter.

| | ***CONTROL VARIABLES ($D_1$, $D_2$, d)*** | ***POWER TRANSFER EQUATION*** | ***ADVANTAGES & DISADVANTAGES*** | |
|---|---|---|---|---|
| ***SPS*** | *0.5, 0.5, d* | $P = \frac{nV_1V_2}{f_sL}d(1-d)$ | *Easy implementation* | *High RMS* |
| ***EPS*** | *D, 0.5, d* | $P = \frac{nV_1V_2}{f_sL}\left[(1-d)d + \frac{1}{2}D(1-D-2D)\right]$ | *Wider ZVS* | *Not full ZVS* |
| ***DPS*** | *D, D, d* | $P = \frac{nV_1V_2}{f_sL}\left[(1-d)d - \frac{1}{2}D^2\right]$ ; $\lvert d\rvert \geq D$<br>$P = \frac{nV_1V_2}{f_sL}d\left[1-D-\frac{1}{2}d\right]$ ; $\lvert d\rvert < D$ | *Wider ZVS* | *Not full ZVS* |
| ***TPS*** | *$D_1$, $D_2$, d* | $P = \frac{nV_1V_2}{f_sL}\left[(1-d)d - \frac{D_1^2+D_2^2}{4}\right]$ | *Full ZVS* | *complex* |

*Table 2-1. Control Strategies comparison [25, 29].*

### 2.3.1 Single Phase Shift (SPS)

This method is considered as the simplest way to control the DAB converter. It stands for **only using the phase** difference between both sides of the bridge to control the topology as the duty cycles of both bridges stays at a 50%. The next figure illustrates the switching behaviour under the SPS modulation strategy.

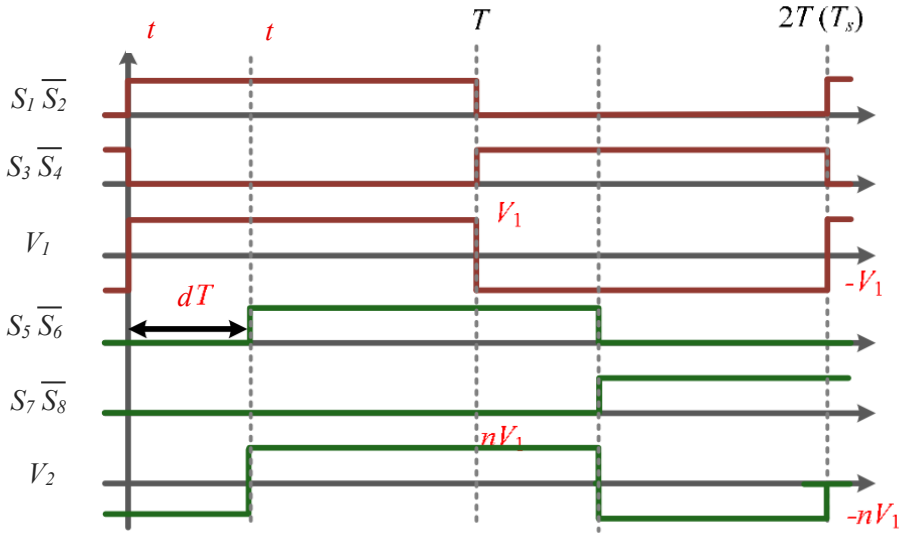


*Figure 2-8. SPS modulation [25].*

### 2.3.2 Extended Phase Shift (EPS)

This strategy continues with the previous one but enabling also the control over the duty cycle of one of the bridges—for the example given, the secondary bridge. Therefore, the strategy has **two degrees of freedom.**

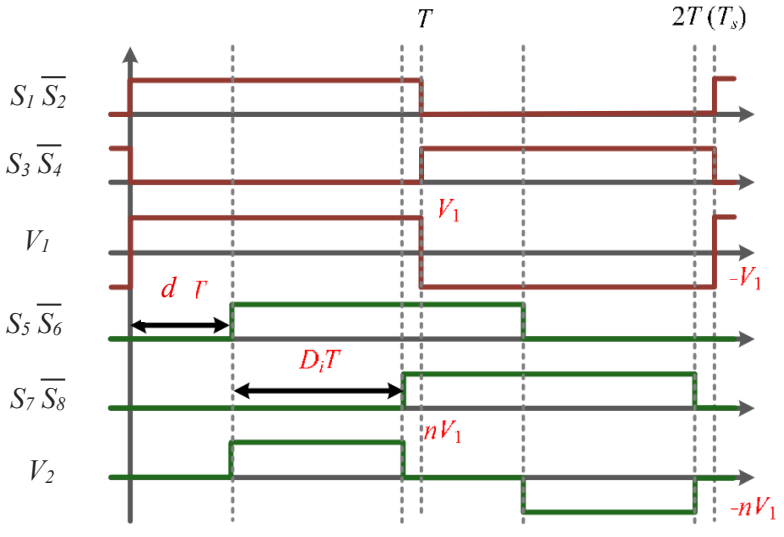


*Figure 2-9. EPS Modulation [25].*

### 2.3.3 Dual Phase Shift (DPS)

The DPS strategy also enable **two degrees of freedom** but uses the same duty cycle on both the primary and secondary bridge. This strategy can be understood as a combination of the two previous ones.

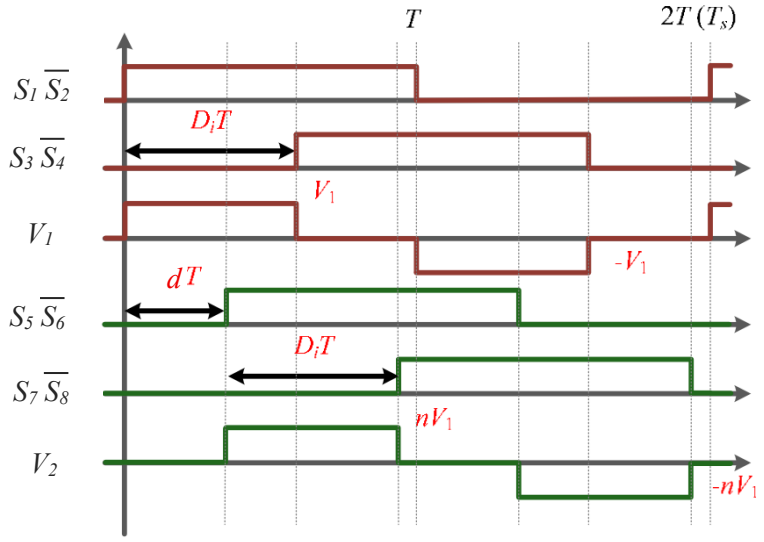


*Figure 2-10. DPS Modulation [25].*

### 2.3.4 Triple Phase Shift (TPS)

Finally, the TPS control strategy enables a **three-degrees control** on the DAB. Therefore, a greater flexibility and full range is cover on the converter, enabling higher efficiency operation control. Nevertheless, this strategy is highly complex to optimize and can have a slow dynamic response [25].

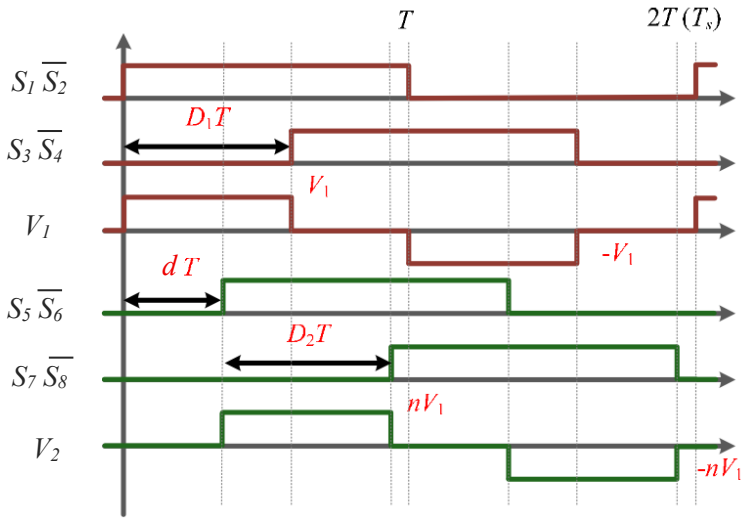


*Figure 2-11. TPS modulation [25].*

## 2.4 Artificial Intelligence

Artificial Intelligence (AI) is a multidisciplinary branch of computer science that seeks to build **machines capable of carrying out tasks** traditionally requiring human intellect, such as reasoning, perception, decision-making and learning. It was first introduced by John McCarthy in 1955, and it has gained high interest during the last decade due to its quick integration in the industry based on the development made in the field of deep learning.

### 2.4.1 Introduction to AI

Artificial intelligence, as previously explained, was firstly developed as creating a non-human system that could take decision or produce task. This field promptly progressed with the advent of **Machine Learning (ML)**, which is a branch of AI focused on algorithms that automatically discover patterns in data and use them to make predictions or decisions. These algorithms refine their performance as they encounter more data, reducing or eliminating the need for hand-crafted rules [43].

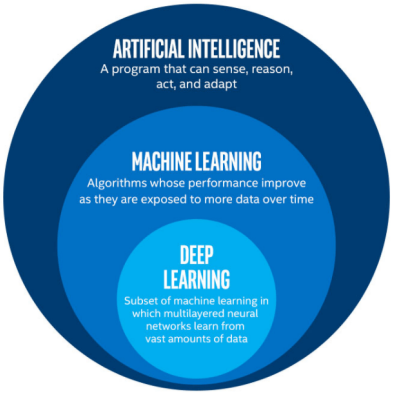


*Figure 2-12. AI, machine learning and deep learning [44].*

**Deep learning (DL)** is a sub-field of machine learning that employs multilayered artificial neural networks to automatically learn hierarchical representations of data, from raw inputs up to abstract concepts. While all deep learning models are machine learning models, they differ in that deep learning relies on deep network architectures to discover features end-to-end, whereas traditional machine learning approaches often models with manual or engineered feature extraction [45].

### 2.4.2 Machine Learning

From a practical standpoint, machine learning can be used and therefore classified in three different groups—**supervised, unsupervised, and reinforcement learning.** In supervised learning, an algorithm is shown input–output pairs, consequently it makes use of labelled data where the solution for each input is defined. Unsupervised learning drops the labels and instead looks for structure hidden inside raw data, by clustering, grouping or analysing data behaviour. Lastly, reinforcement learning approaches the problem as sequential decision-making, where an agent interacts with an environment, receives feedback in the form of rewards or penalties, and refines its policy to maximize long-term cumulative reward, enabling applications such as game-playing AIs or autonomous control systems.

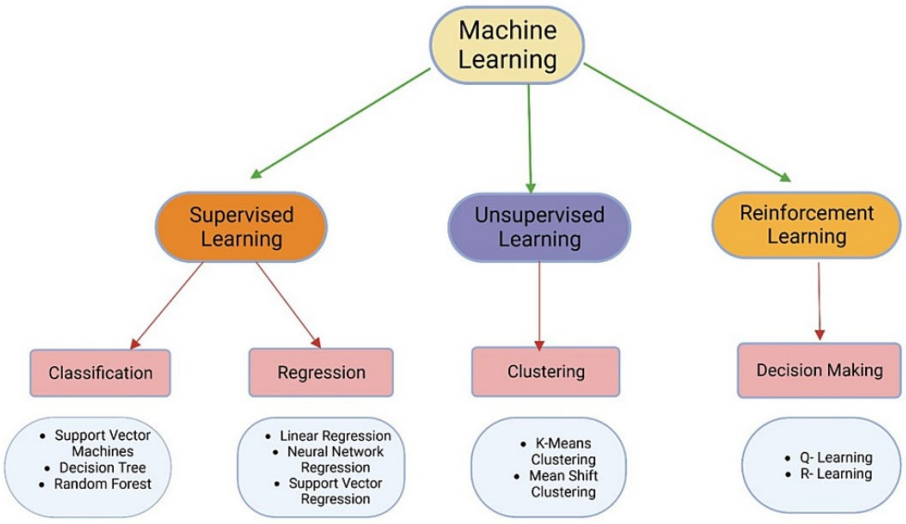


*Figure 2-13. Machine learning classification [46].*

For each category, different orientation for the aim of training can be applied as illustrated on figure 2-13—where also practical examples of technologies are given.

### 2.4.3 Deep Learning and Neural Networks

As previously introduced DL is a sub-field of machine learning that **trains artificial neural networks** with many **hidden layers** to perform task-specific representations directly from data. Mathematically, these deep models treat learning as large-scale, nonlinear function approximation, adjusting millions of parameters to minimize a loss through gradient-based optimization—an ability grounded in **the principle of universal approximation property of neural networks** and back-propagation related to gradient-descent techniques [47].

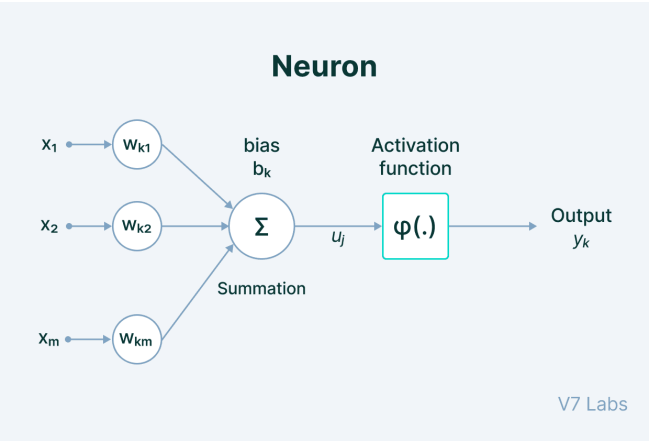


*Figure 2-14. Neuron definition [47].*

In order to achieve that, artificial neural networks (ANN) are used. An ANN is a directed graph of connected nodes called **neurons** (see figure 2-14); each neuron computes a

weighted sum of its inputs, passes the result through a nonlinear activation function, and forwards the signal to the next layer, with the weights adapting during training to capture statistical regularities in data (see figure 2-15). The power of this architecture is formalized by the universal-approximation theorem, which guarantees that all networks can approximate any continuous function given sufficient capacity [48].

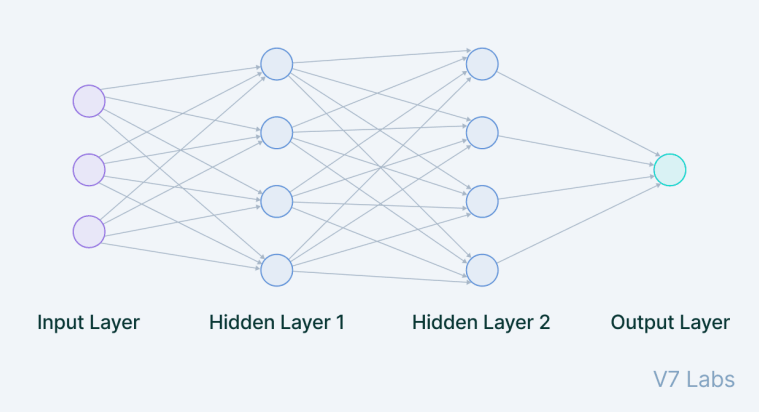


*Figure 2-15. Neural Network [47].*

A network is labelled as deep once **multiple hidden layers** are stacked, enabling successive layers transmit raw inputs into progressively more abstract features, so the model performs its own feature learning rather than relying on handcrafted descriptors of the data. In short, deep learning represents the computationally multi-layer evolution of neural-network research built on ANNs that enables to artificially produce decision and lead tasks with computational learning.

### 2.4.4 Data Augmentation for Tabular Data

Data augmentation is the practice of **artificially expanding a dataset** by creating new samples from the originals, a tactic that helps models generalize better when the real data is scarce, expensive or privacy-sensitive.

For classical machine learning pipelines on tables, augmentation often starts with row-level resampling. Techniques such as **SMOTE** and its variants generate new minority class by interpolating between existing examples, helping imbalanced dataset to reduce mismatching number of samples between classes. Other lightweight options such as jitter techniques add **Gaussian noise** to numerical features, shuffle categories within logical bounds, or drop/merge columns to test feature robustness. Nonetheless, although they are easy to implement and computationally light, such heuristics can fail to respect feasible data as they cannot utterly represent the natural behaviour of the dataset [49].

A more recent wave of **synthetic-data generators** tackles the downsides of the previously introduced techniques by using learning to generate data out of the entire

table—for example GAN, CTGAN, VAE, or diffusion-based synthesizers [49]. Compared with the previous techniques, these models can create balanced data, and more accurately represent the nature of the dataset, and even simulate rare scenarios. Although they demand greater computational power and higher complexity on development and deploy. Furthermore, it still a field of research and cannot be yet suitable to every case of use, where for tabular data still being questionable.

In short, tabular augmentation aim to generate new rows of valid data to feed the model, many techniques can be used where at each a trade-off between realism, resource cost and implementation complexity should be found.

# 3 Design of Solution

In this chapter, the totality of the proposed methodology is covered from a technical perspective. The objective is to explain the sequence of stages followed to achieve the results and explain the technical approach and adjustments needed at each step.

## 3.1 Introduction and Workplan

As introduction, the methodology for the whole process is presented. It consists of three main chapters where at each one other minor stages can be found. The following figures illustrate the overview of the workflow. Each stage will be cover deeper at its respective chapter.

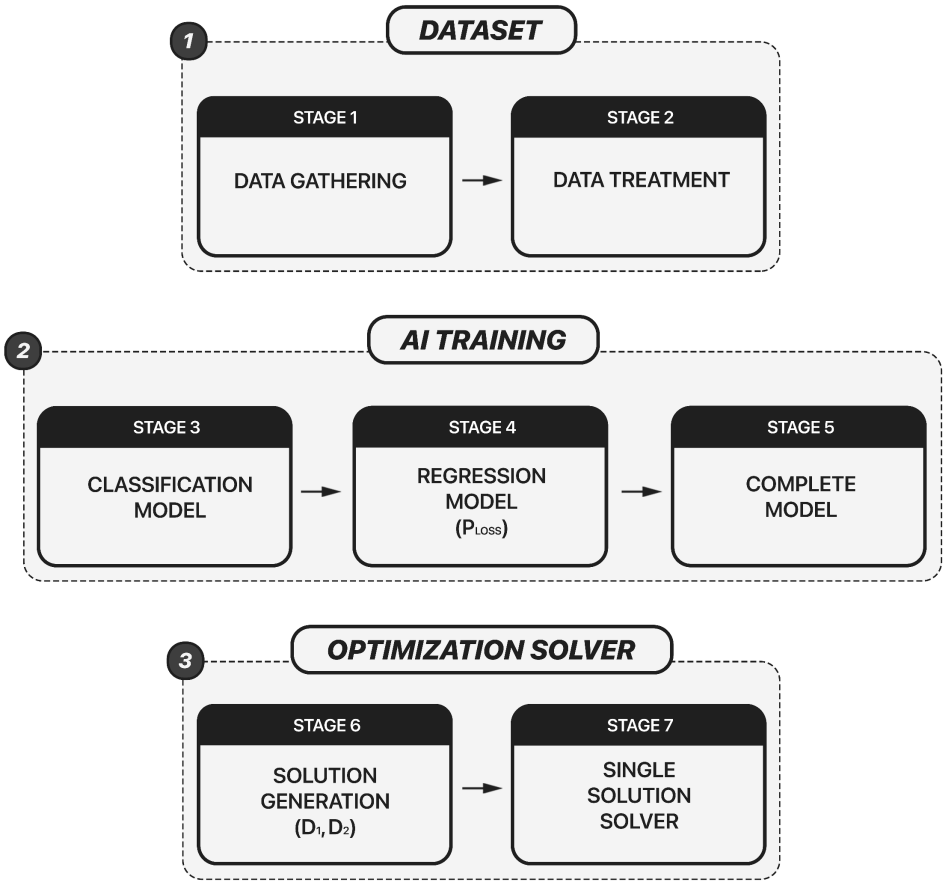


*Figure 3-1. Methodology diagram.*

As observed on figure 3-1, the chapters addressed are the generation of the dataset, the training of the artificial intelligence model and finally the optimization solver using process.

## 3.2 Dataset

In order to train the AI models, **supervised learning** will be used, therefore, it will require to build a dataset with a feature vector for the input, and a properly labelled output for the solution. At this stage, it is necessary to conceptualise and embody the desired functioning of the algorithm to build a well-oriented dataset for the purpose of the training.

The objective of the research is to find high efficiency points, which would be aimed to do by **predicting the power losses ($P_{LOSS}$)** of a DAB under different load conditions. Thus, we need to gather simulations results of the behaviour of the model for different input ranges. In order to solve this need, simulations in *PLECS* would be run.

Consequently, the next steps would need to design a functional model for a DAB in *PLECS* that enables to deploy a suitable control strategy (see chapter 2.3). Hence, the process of simulation will need to be automated with a python script to which a meaningful range of input and a thoughtful data gathering strategy should be addressed. Once completed the dataset, a process of data treatment must be following completed.

### 3.2.1 Data Gathering through PLECS simulations

To successfully gather simulations results that illustrate the behaviour of a dual active bridge, the following data gathering strategy is presented in figure 3-2.

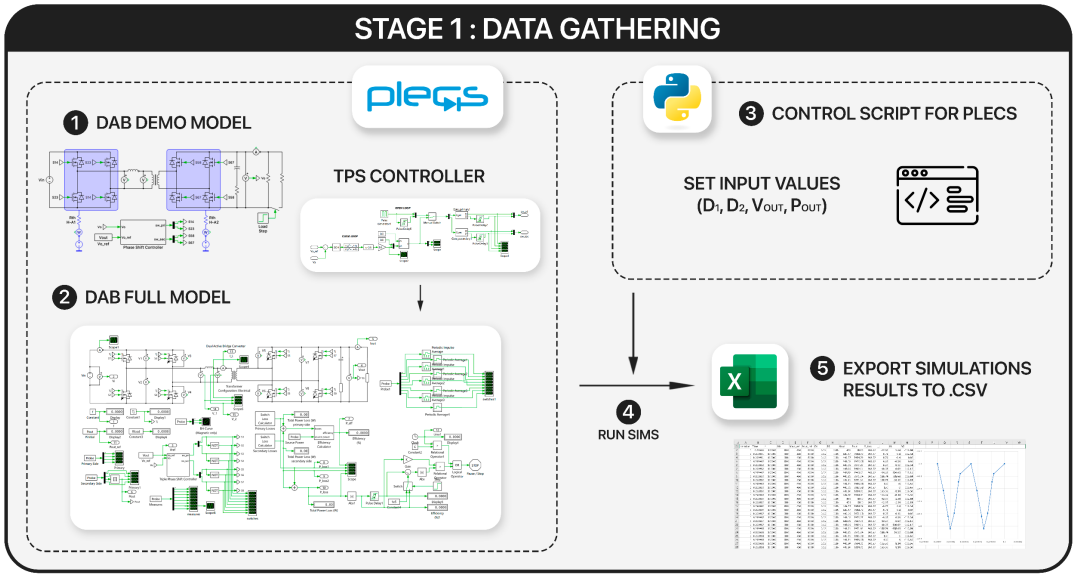


*Figure 3-2. Stage 1: Data Gathering.*

The first step is to build the *PLECS* model that describes a dual active bridge and its control strategy. To start the process, the **DAB demo model published by *PLECS*** available on its website was collected [50]. This model includes a close loop single phase shift strategy and describes the thermal behaviour of the transistor of each bridge. Furthermore, it includes the calculation of the efficiency and power losses—which are the conduction and switching losses of the primary and secondary bridge.

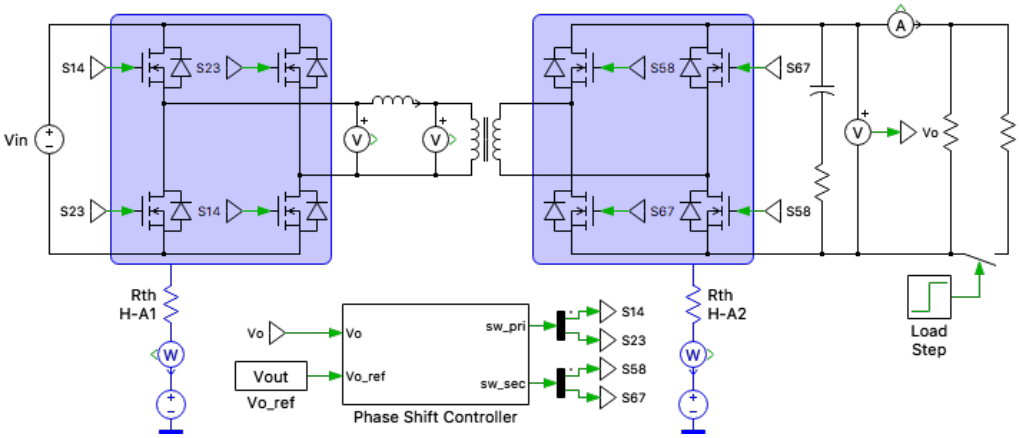


*Figure 3-3. PLECS Demo Model of Dual Active Bridge [50].*

The additions and changes required on this demo model are explained in the next section.

## A. PLECS DAB model

Starting from the demo model of figure 3-3, the following DAB model was designed to run the simulation of the topology.

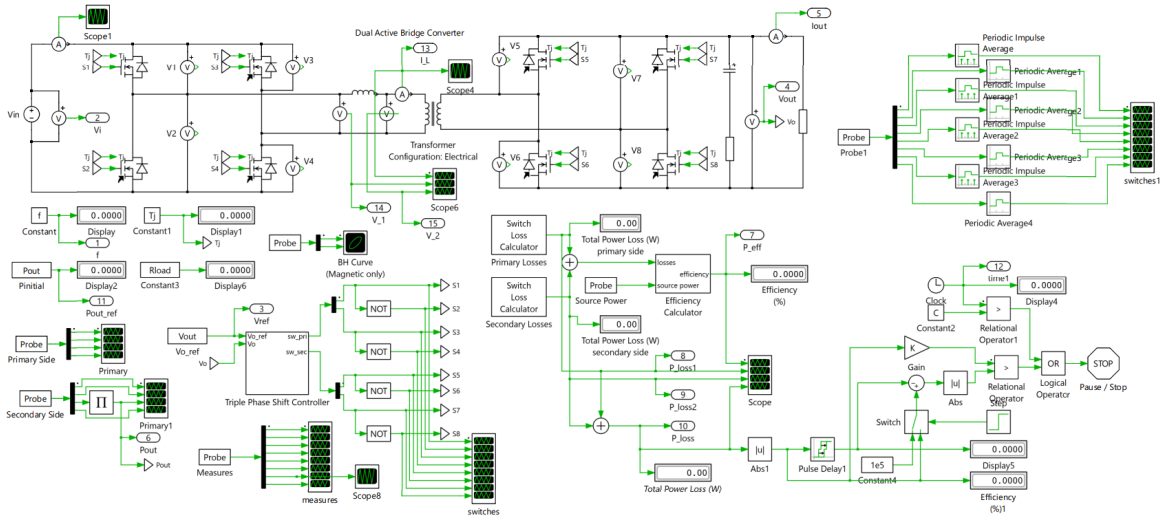


*Figure 3-4. PLECS complete simulation model.*

As the aim of the research is to achieve high efficiency points on the converted, it was decided to use a **triple phase shift control strategy.** Hence, an open loop TPS controller was designed defining $D_1$, $D_2$ and $d$ as model variables (see figure 3-5). At this point, and to be able to operate de converter at a referenced output load, a close loop strategy was proposed **controlling the parameter *d* with a PI controller** using the reference of the voltage on the load side. This usage was thought to be the more reasonable proposal as reviewed on references [25, 51, 52].

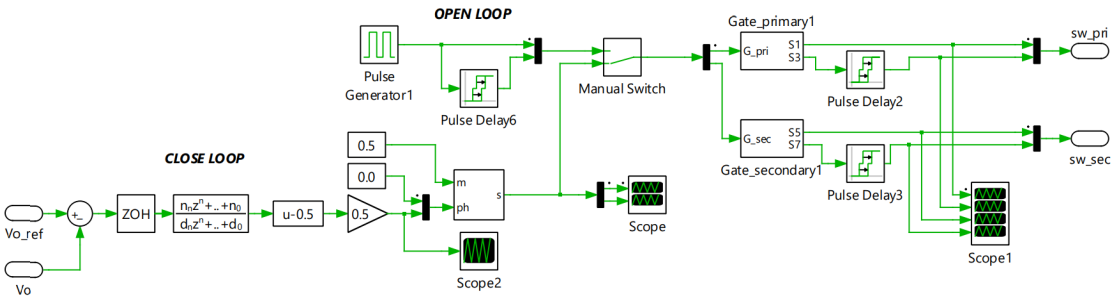


*Figure 3-5. Triple phase shift control model on PLECS.*

With this implementation, we reduce one degree of freedom on the converter, and we are also able to test the converter on repeatable load situation—which will facilitate the data gathering process. The PI controller was adjusted to be ideal, as its sample time is equal to the switching frequency, so it can adjust the output to the reference in real time. Furthermore, a small value for both the proportional ($K_p$) and integral gain ($K_i$) parameter were assigned ($K_p \approx 0.01$; $K_i \approx 0.11$) in order to get an overdamped behaviour that, despite having a slower response, reduces oscillations to ensure stability. Therefore, the **influence of the PI controller** on the behaviour of the converter **is minimized.**

Two more major changes had to be made on the demo model. Firstly, the temperature calculation on the joint of the diodes was designed on the demo models with look-up tables defining the evolution over time of the temperature. As this parameter varies with the initial conditions of the converter and its evolution over time, it was decided to simplify the behaviour of the joint temperature to normalize its performance under any load and time conditions. This change was made as we wanted to measure the results of the simulation data points when the reference value was reached, and if previous temperature changes to that point were considered, the losses on the steady state point would have been affected. The biggest downside to this implementation is that some point that may not be reachable due to high self-heating on the joints now can be reached, so false solutions could be founds. Despite knowing this assumption, the previous strategy still stay as the best solution for the study case as it normalizes data and reduces simulation time, at any rate, the mentioned downside must be bear in mind.

Secondary, as the converter would have to be simulated under different load and control conditions, stablishing a fixed simulation time was not evaluated as an efficient decision. Therefore, a reactive stopping condition was designed. This consists of comparing the current value of the total power losses with the one 2,000 periods before. If the change was smaller than a 0.1%, the stoppage of the simulation was triggered as the variation on the losses of the model are minimal. This addition enables to **reduce the simulation time** from an average of about 60-90 seconds per simulation to 10-15 seconds. When considering the number of simulations run (see chapter 3.2.1.2) this addition was a huge improvement in terms of time efficiency. The efficiency calculation and stopping block diagram are presented on the next figure.

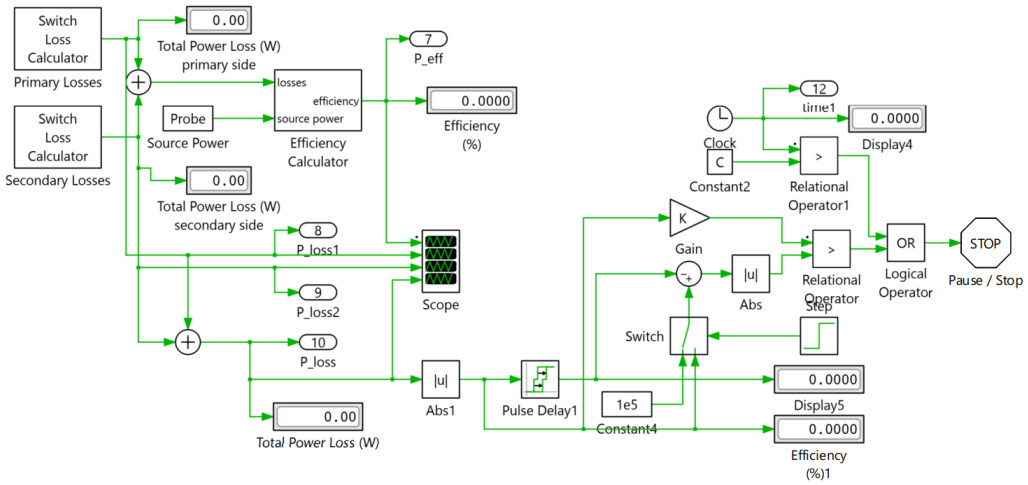


*Figure 3-6. Efficiency calculation and stopping block diagram*

## B. Python Sim Automatization

The figure 3-7 describes the simulation strategy that will now be covered.

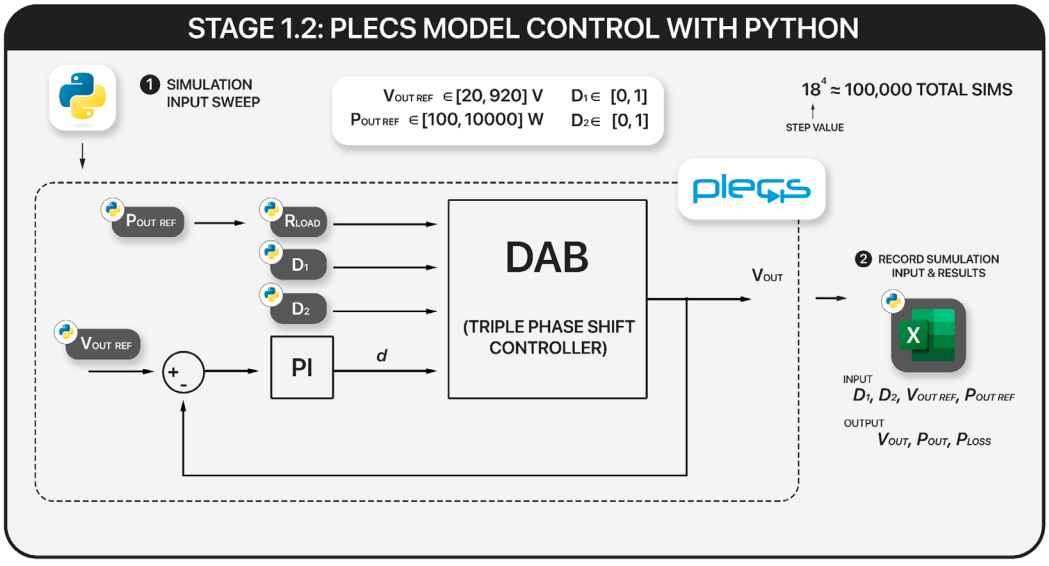


*Figure 3-7. Stage 1.2: PLECS model control with python.*

Once the *PLECS* model for the DAB was properly designed, tested and validated, it was time to **automate the running of simulation** with code. In order to achieve this, the ***xmlrpc*** client library available on python was implemented, following the manual available *plexim* website [53]. The following is a brief overview and presentation of the code available on the respective appendix.

```
import xmlrpc.client
[…]
proxy = xmlrpc.client.ServerProxy("http://localhost:1080")

proxy.plecs.load("Q:/user/SanJoseVegaDiego/RUNING
SIMULATIONS/dual_active_bridge_converter_TPS.plecs")
[…]
proxy.plecs.scope('dual_active_bridge_converter_TPS/Scope', 'ClearTraces')
```

*Listing 3-1. Library presentation and model port access.*

This library enables to access a *PLCES* model through a local hosted port. This port must be previously opened on the *PLCES* settings feature. Once open, it enables to load a model. It is worth mentioning that the *PLECS* model must be launched to use this feature, so no headless usage is available.

```
opts = {'ModelVars': {'D1': D1, 'D2': D2, 'f': f, 'Vin': Vin, 'Vout':
Vout_ref, 'Pout': Pout_ref, 'Rload': Rload}}

result = proxy.plecs.simulate("dual_active_bridge_converter_TPS", opts)
```

*Listing 3-2. Load variables and start simulation.*

With the call on listing 3-2, it is possible to stablish a value for the variables defined on the model and then simulate the model with those values loaded. To gather the results, that simulation call must be stored in an empty variable and then can be accessed to each vector row. The value of each recorded output refers to **the label of a wire in *PLECS*,** so each simulation step result can be recorded.

```
time_data = result['Values'][11]
[…]
V_out_data = result['Values'][3][i]
P_out_data = result['Values'][5][i]
P_loss_data = result['Values'][9][i]
```

*Listing 3-3. Results label gathering.*

It is worth mentioning that on the variable storing the results from the simulation call, they are recorded the totality of simulation steps taken by *PLECS*, which is an excessively high amount of data. Therefore, only the data results for the last cycle were

stored to reduce computational size. All the data was written in a **.csv file** as each simulation was finished, generating several rows of information for each simulation, which was properly label with an iteration index for it later use.

At this point, the previous code enables to run automatically simulations on *PLECS* by inputting the desired variables value and then record the output of the **last period waveform** for the labelled wires on *PLCES*. Therefore, the simulation strategy must be defined.

As explained in previous chapters, the dual active bridge topology is most frequently used as the DC/DC converter of EV chargers. Thus, the simulation points must be meaningful for this application. Therefore, the strategy for **stablishing the input values** will consist of deciding which variables to sweep—and consequently which ones to stay constant; then deciding **the range for the sweep**, and lastly the **sweep step**. Those decisions are obviously a trade-off between gaining model insights and a wider range of use but also sacrificing high data size and computation time and cost.

Based on references [54–57]—where frequently used value ranges were presented for a DAB converter mounted on an EV charger—the following **parameters ranges** where selected for running the simulations:

| $V_{OUT\ REF} \in [20, 920]$ V | $P_{OUT\ REF} \in [0.1, 10]$ kW |
|---|---|
| $D_1 \in [0, 1]$ | $D_2 \in [0, 1]$ |

*Table 3-1. Variables simulation ranges.*

And the following variables of the DAB were treated as **constant** with their respective values:

| $V_{IN} = 800$ V | $f_s = 100$ kHz |
|---|---|
| $n = 4$ | $L_{lk} = 2.052 \cdot 10^{-6}$ H |

*Table 3-2. Simulation constants.*

As it exists a limitation on the number of simulations possibly being able to run due to computational and time constrains, it is utterly important to optimize this stage in terms of computer efficiency without sacrificing dataset quality. Consequently, the establishment of a fixed value for the switching frequency and input voltage drastically reduce the total number of simulations by reducing the number of variables to be swept.

Then, once the range and number of variables was decided, a **step value** for the sweep must be assigned. The number of total simulations is the step size to the power of the

number of variables, which can be describe with equation four—where $N$ is the step size and $n$ is the number of variables.

$$Total\ simulations = N^n \quad ; \quad n = 4$$

*Equation 3-1. Total number of simulations.*

Therefore, it is needed to select a desired number of simulations to then know the step size. Based on references [25, 51, 52]—where up to 160,000 samples of data where generated for similar training purposes—the aim was to get **over 100,000 simulations**, so a **step of 18** was selected.

$$18^4 = 104.976\ simulations$$

Overall, the data gathering strategy with python is summarized as inputting the model variables with the values of $V_{OUT\ REF}$, $P_{OUT\ REF}$, $D_1$ **and** $D_2$ through the described ranges and steps and then storing the results for the last period results obtained in *PLECS* for $V_{OUT}$, $P_{OUT}$, $P_{LOSS}$ on a .csv file. The power on the output is adjusted by changing the value of the resistance on the load by doing fundamental calculation. This information is resumed on the next table.

| ***INPUT*** | | | | ***OUTPUT*** | | |
|---|---|---|---|---|---|---|
| $V_{OUT\ REF}$ | $P_{OUT\ REF}$ | $D_1$ | $D_2$ | $V_{OUT}$ | $P_{OUT}$ | $P_{LOSS}$ |

*Table 3-3. Recorded variables on dataset from simulations.*

### 3.2.2 Data Treatment

On the previous chapter, the methodology to generate and gather the data to build the dataset was accurately presented. With the available computer power, the total

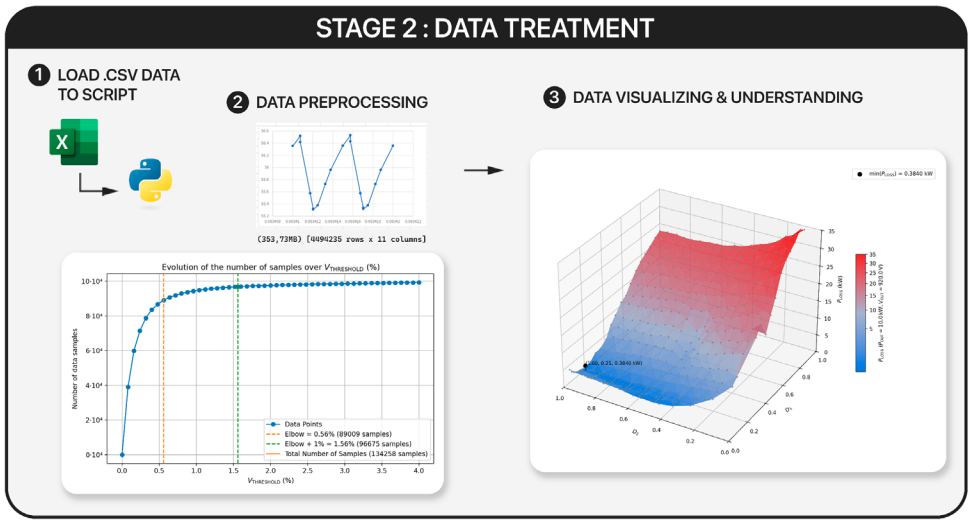


*Figure 3-8. Stage 2: Data treatment.*

simulation time for the complete dataset was around 20,000 minutes in total. Once completed, it is utterly important to thoughtfully treat the data before training the AI algorithm to understand and process the information in a better way. This chapter's overview is described in figure 3-8.

Once loaded the .csv content to a data frame of the library *pandas,* the shape of variable was [4494235 rows x 11 columns] where multiple points for each simulation were recorded to build the waveforms. To encapsulate that, the following **average time-domain calculation** were used to group each simulations results in a single row.

```
value_columns = data.columns.drop(['Iteration', 'Time'])

def weighted_avg(group, columns, weight_col='Time'):
    weights = group[weight_col]
    if weights.sum() == 0:
        return pd.Series({col: np.nan for col in columns})
    else:
        return pd.Series({col: np.average(group[col], weights=weights) for
col in columns})

data_ss = data.groupby('Iteration').apply(lambda g: weighted_avg(g,
value_columns, weight_col='Time')).reset_index()
```

*Listing 3-4. Weighted average calculation.*

As previously mentioned, the simulations were run using a close loop PI controller adjusting the value of the phase between bridges (*d*), thus not every simulation was able to reach the reference output. Therefore, several columns were added to the data frame as it was the case for the **voltage error**—which stands for the difference between the voltage reference and actual output value. With this variable, it should be stablished a hard limitation to consider a simulation as successful on reaching the reference output load (from now, this condition would be named as reaching the steady state for its usage in the field of control and automation).

To embody a **voltage threshold** to consider the steady state as true or false, the chart of the number of samples for different values of threshold is plotted. This figure describes a curve that tends to stabilise as the threshold value gest greater. Consequently, the elbow point was calculated by joining the first and last point with a straight line and drawing the normal to that line for each plotted point, thus the longest perpendicular will determine the elbow point. The voltage threshold was located by adding a +1% to the elbow point to ensure a margin, resulting in a value of **1.56%.** At this threshold value, the number of new samples for a greater threshold is minimal, so the method to calculate this value is validated.

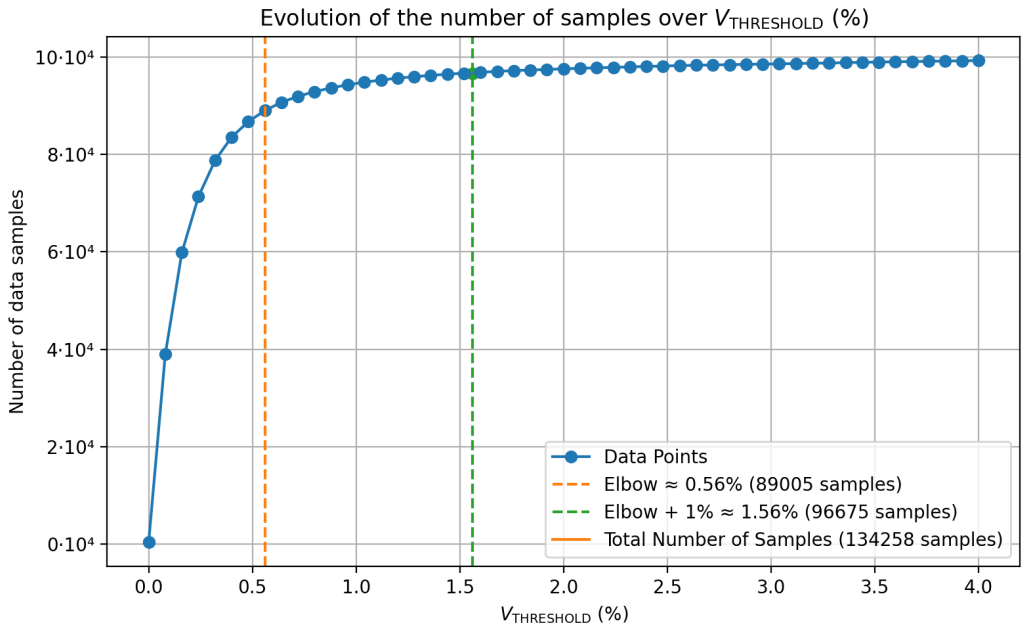


*Figure 3-9. Voltage threshold over sample number.*

At this stage, a binary classification for clinching this relative threshold on the voltage output was allocated in its proper column for each row. In addition, the next figure represents the histogram for stable output points to visualize how many high efficiency points where obtained. It is worth noticing that the majority of points (75.8 %) have a negative efficiency. In addition, it was spotted a great number of points not having steady state output that were reaching high efficiency, for the converter still have losses on the not stable point. Therefore, the data must be properly treated to avoid false inputting to the model.

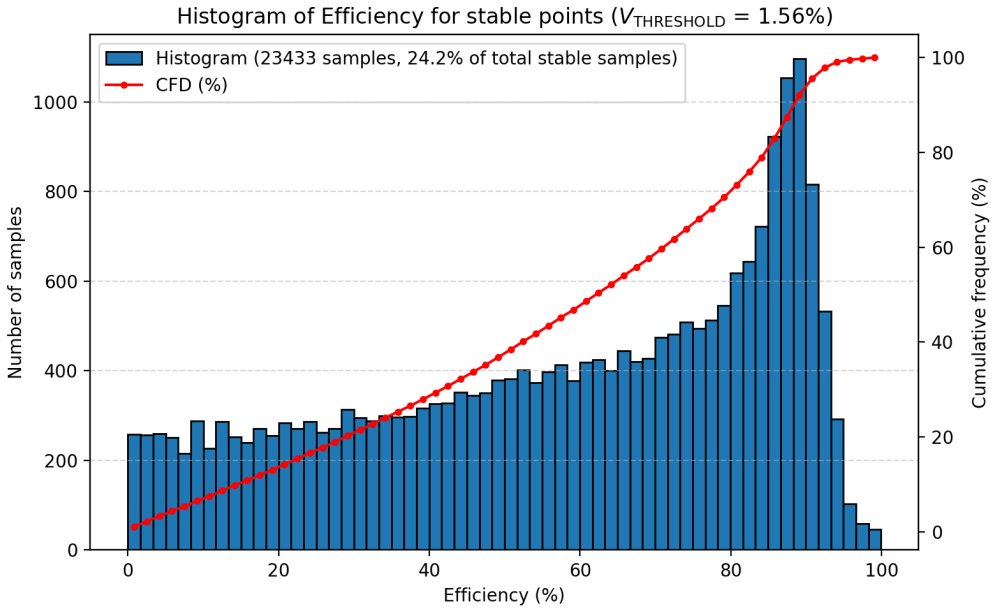


*Figure 3-10. Efficiency histogram of not stable and stable samples.*

## A. Data Exploration

Before defining the training, vectors and applying tabular data augmentation, it is worth taking a brief overview to meaningful representations of the data gathered. The AI models will aim to predict the best control combination for any power and voltage condition, therefore, for each load operation point, the power losses and therefore the efficiency for a grid of $D_1$ and $D_2$ can be plotted—only true steady state points were considered for the following plots.

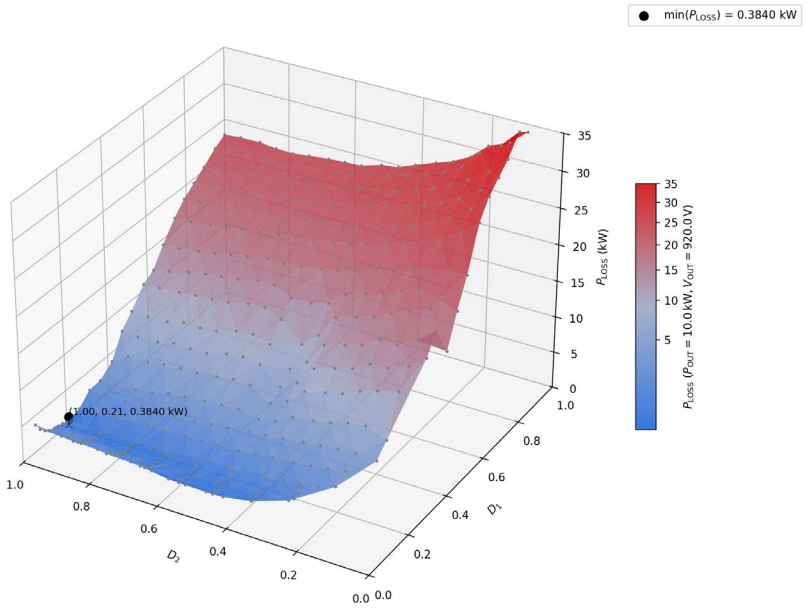


*Figure 3-11. Power losses over $D_1$ and $D_2$ for $V_{OUT}$ = 920 V & $P_{OUT}$ = 10 kW.*

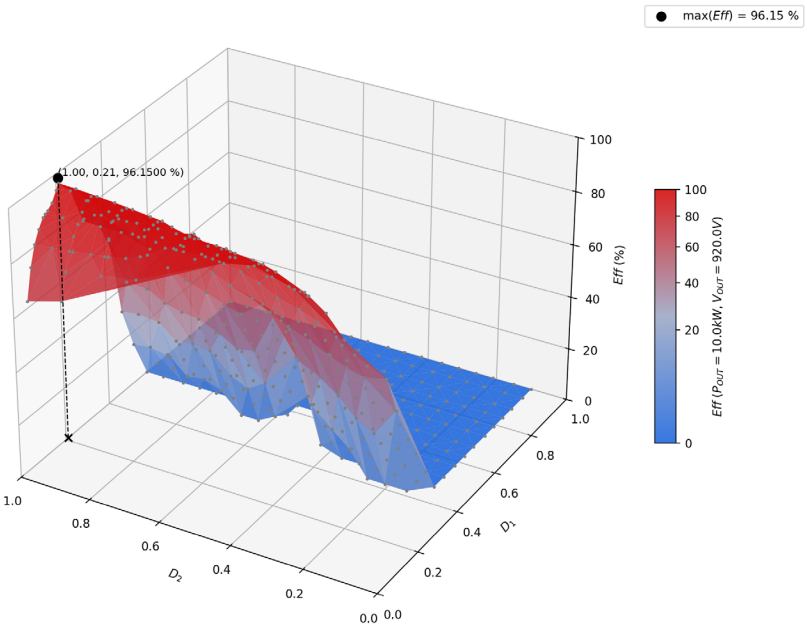


*Figure 3-12. Efficiency over $D_1$ and $D_2$ for $V_{OUT}$ = 920 V & $P_{OUT}$ = 10 kW.*

As shown on the previous graphs, the losses for some combination points are too high that the efficiency is below zero as noted on the histogram of figure 3-10. This means that a huge amount of datapoints will not be meaningful, so a threshold to consider a point as highly efficient should be addressed (see chapter 3.2.2).

## B. Model Description

At this point, and after reviewing the qualities of the available data, a redefined and suitable **strategy for the AI model training** must be considered. Therefore, the first proposal was to implement a simple regression predictive model that strives to predict the output of $P_{LOSS}$ for the whole range of possible solutions.

This strategy had to be quickly rethought as the range of possible power losses were extremely high, as shown on figure 3-11. The model was moderately able to find the linearity of the solutions, but due to the wide range, **the RMSE of the predictions were extremely high.** Furthermore, knowing the losses does not strictly lead to knowing if the converter was reaching the reference value or not.

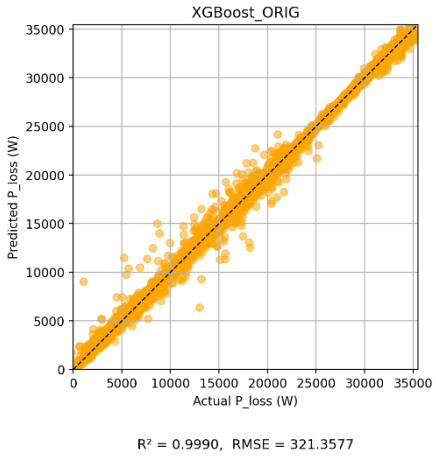


*Figure 3-13. Failed results of first training*

This finding helped on giving a much realistic perspective to solve the problem, and the following strategy was proposed and finally proved as better than the previous one.

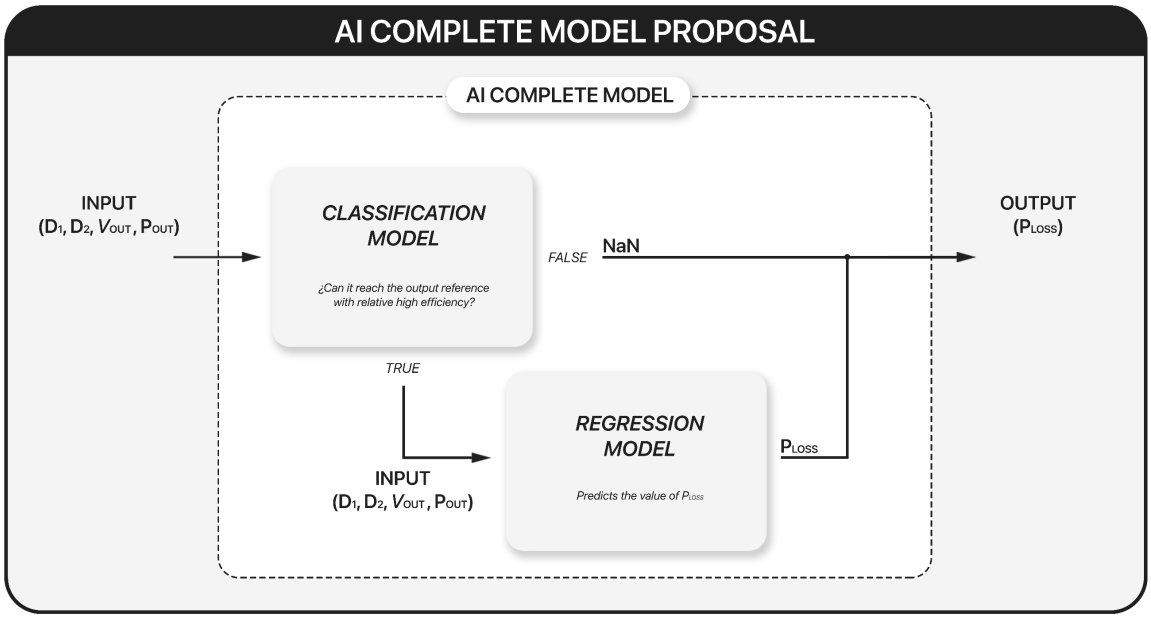


*Figure 3-14. AI complete model proposal.*

The **classification model** is introduced to validate if a control strategy for a load operation point could be reaching the reference output and doing that with considerably high efficiency. To set the **binary boundary** for the classification model, it is needed to describe what considerably high efficiency is. Therefore, both a hard limitation on the power losses or on the efficiency could be valid. To set this threshold, the relation between efficiency and power losses were plotted on the next two graphs. The first one describes the totality of the stable samples graph whereas the second one describes just the best efficiency result for each $V_{OUT}$ & $P_{OUT}$ combination.

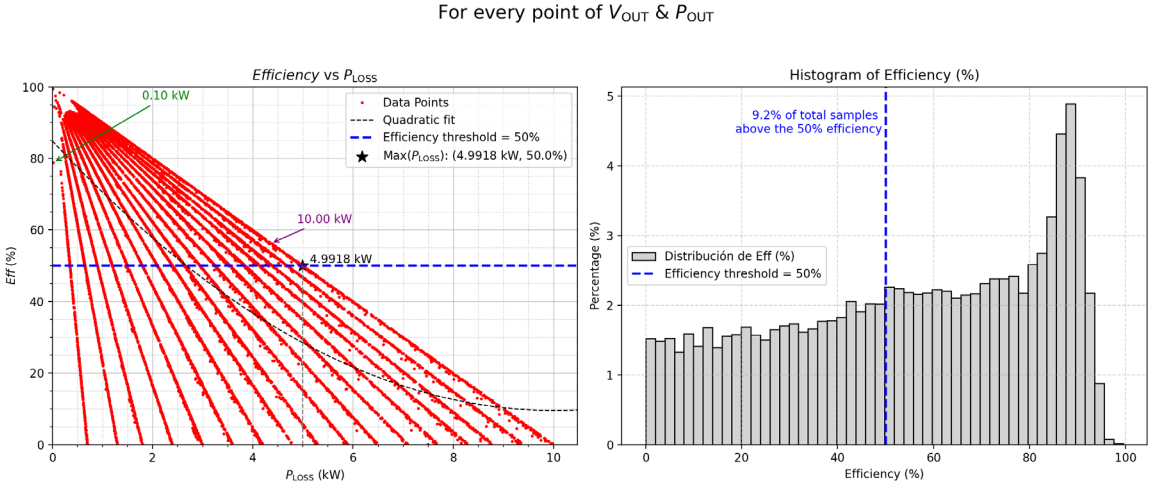


*Figure 3-15. Efficiency over power losses and histogram for all samples.*

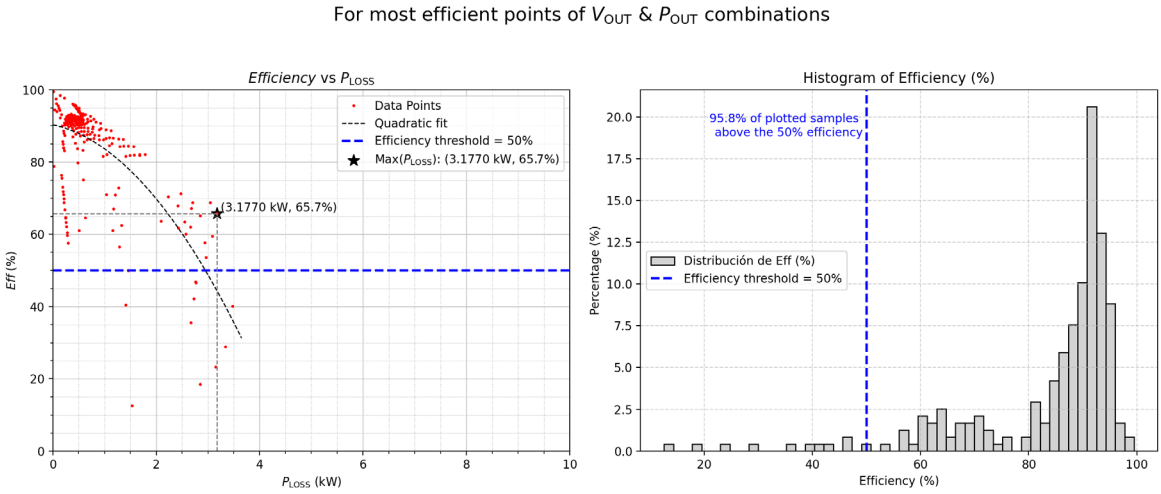


*Figure 3-16 Efficiency over power losses and histogram for best efficiency samples.*

With this information, it was discovered that a low value for the power losses did not strictly lead to get deliver high efficiency as previously suggested. Consequently, the hard limitation was decided to be stablish at the 50 % efficiency limit were most of the meaningful high efficiency results were found (the 95.8% of the top plotted results). Therefore, the classification model was defined with that classification objective.

## C. Dataset Adjustments & Input Feeding

At this point, and before explaining the data augmentation process, it is important to note two major adjustments made on the dataset after the training of the models. Firstly, two different regions with high density errors were found impoverishing the performance of the classification and regression model respectively. Those regions were found to be the range of values of $D_1$ from 0 to 0.2 approximately, and the region of $D_1$ and $D_2$ where both are close to 0.5.

New simulation where run on those sensible regions and added to the previous dataset. This oversampling utterly enhanced the behaviour of the model and highly reduced the severity of the errors.

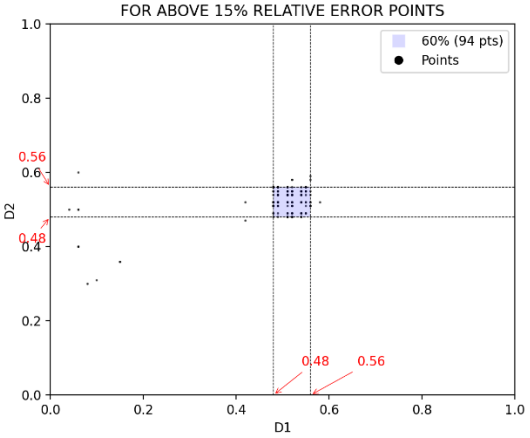


*Figure 3-17. Regression model relative error grid.*

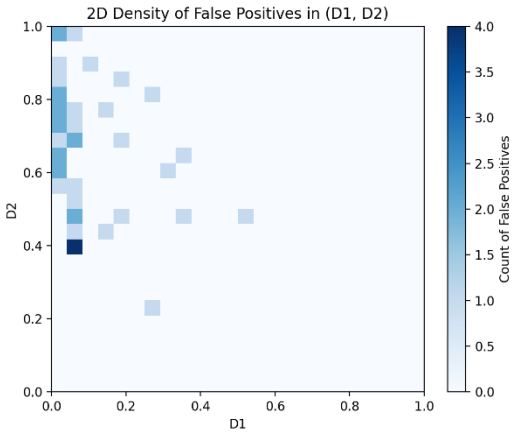


*Figure 3-18. Classification model false positive grid.*

Therefore, the previous grid for $D_1$ *vs* $D_2$ and $V_{OUT}$ *vs* $P_{OUT}$ is compared with the final one on figures 3-19 and 3-20.

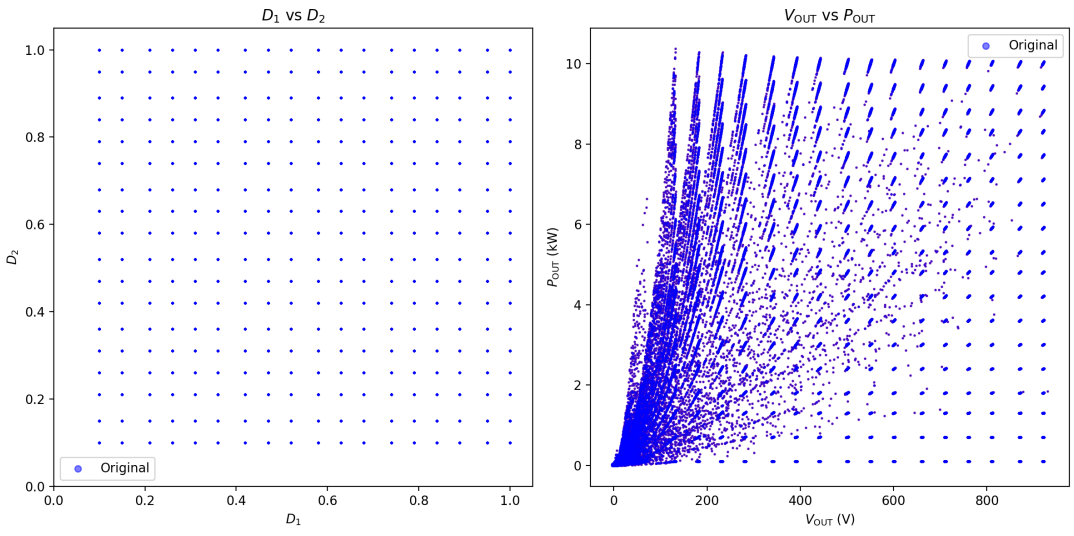


*Figure 3-19. Original sweep grid.*

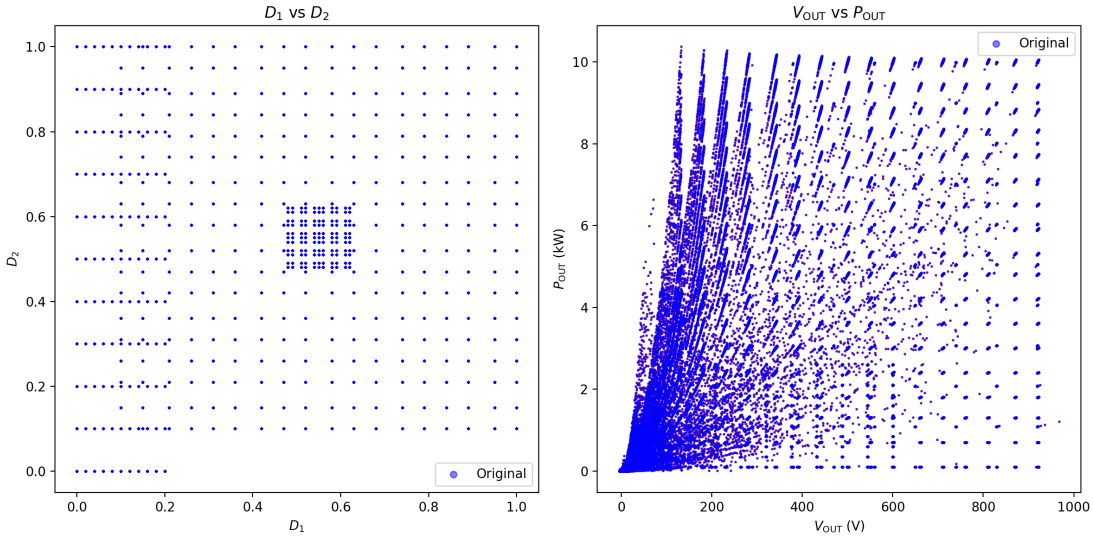


*Figure 3-20. Final data grid with complementary data.*

Furthermore, the second major change was found at the very end of the results, where an extremely high efficiency region was spotted. By analysing the waveforms on the solutions there was no apparent reason for achieving such great results. Simulation on *PLECS* were covered on those points and an error on the primary side losses calculation was spotted, where extremely low losses—even negative—were being calculated, enhancing inaccurately the efficiency simulated of the converter.

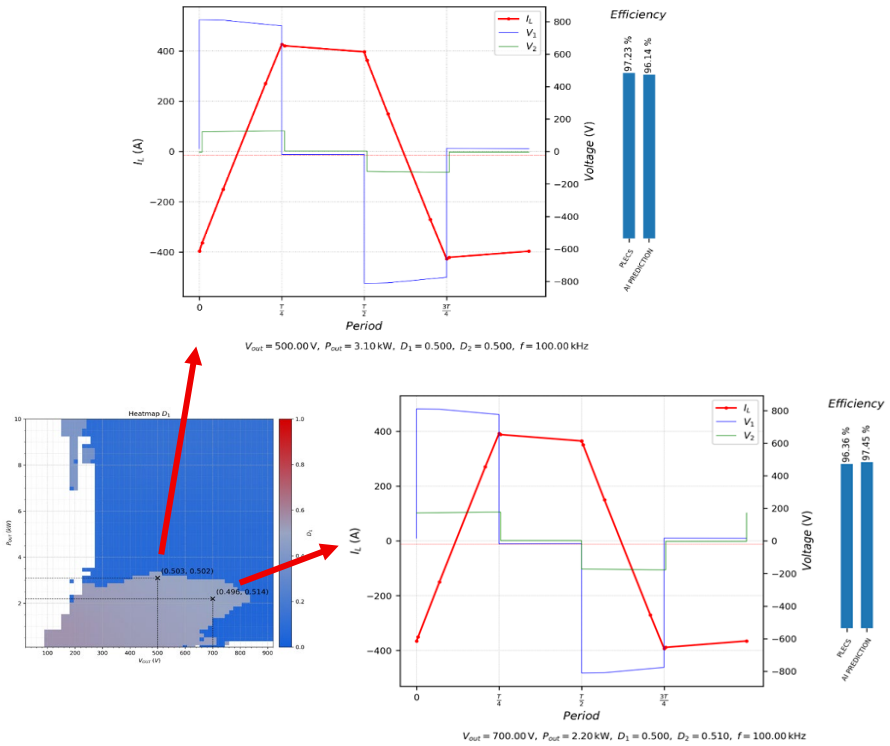


*Figure 3-21. Inductance current waveforms for region 2.*

Those points seemed to be operating on reactive power conditions and according to several reports of similar issues, it was decided to remove the affected samples. Other areas were checked to ensure the reliability of the rest of the dataset and no further issues were found. Therefore, this was treated as an isolated issue that could be caused by misleading interpolation on *PLECS* due to missing data on thermal or control condition.

Before this clarification, the **input vectors for *X* and *y*** were defined for the classification and regression model respectively. It is important to note the **train-test split** used (which was a 20% test size for the classification and 25% for the regression model) as well as the vector shapes. For the regression model training, only the datapoints that validate the defined conditions to be treated as a true result on the classification model were used.

```
X = data_ss[['D1', 'D2', 'Vout', 'Pout', 'Vin:Vout']]

y = data_ss[['P_loss', 'Steady State']]

y = np.where(y_train['Steady State'],
             (efficiency_train >= eff_threshold).astype(int), 0)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20,
random_state=30)
```

*Listing 3-5. Supervised learning input vector definition for classification model*

```
LX = Ldata_ss[['D1', 'D2', 'Vout', 'Pout', 'Vin:Vout']]

Ly = Ldata_ss[['P_loss']]

LX_train, LX_test, Ly_train, Ly_test = train_test_split(LX, Ly,
test_size=0.25, random_state=30)
```

*Listing 3-6. Supervised learning input vector definition for classification model.*

## D. Tabular Data Augmentation

Based on the previous statement, the designed AI model implementation make use of two different model, therefore data augmentation must be applied separately to each model to have a more accurate solution.

Firstly, for the **classification**, it is worth noticing the existence of an **imbalance** between the **classes**. Therefore, *SMOTEENN*, a technique use for minority classes augmentation that combines over- and under-sampling using *SMOTE* (Synthetic Minority Over-sampling Technique) and *ENN* (Edited Nearest Neighbours) [58]. Additionally, to this

technique and based on the fact of having a grid of points, a jitter augmentation algorithm also was implemented. Other techniques oriented into synthetic data augmentation were tested, such as *GAN* or *VAE*, but results were not successful based on references [49, 59, 60]. The next table describes the evolution of the shape of the training vectors after applying data augmentation techniques.

| | ***X_train*** | ***y_train*** | |
|---|---|---|---|
| | | *FALSE* | *TRUE* |
| ***ORIGINAL DATA*** | *[107 406 rows x 5 columns]* | *[107 406 rows x 1 columns]* | |
| | | *88.55 %* | *11.45 %* |
| ***AFTER SMOTEENN*** | *[105 212 rows x 5 columns]* | *[105 212 rows x 1 columns]* | |
| | | *78.32 %* | *21.68 %* |
| ***AFTER JITTER*** | *[210 424 rows x 5 columns]* | *[210 424 rows x 1 columns]* | |
| | | *78.32 %* | *21.68 %* |
| ***TOTAL AUGMENTATION RATIO*** | ***x 1.96*** | | |

*Table 3-4. Data augmentation for training vectors of regression model.*

The definition of each augmentation technique on the code is now attached to facilitate the meaningful parameters for the augmentation. For the jitter a, x2 augmentation ratio was assigned with a 1% jitter variable error.

```
from imblearn.combine import SMOTEENN
smoteenn = SMOTEENN(sampling_strategy=ratio, random_state=42)
X_res, y_res = smoteenn.fit_resample(X_train, y_train)
```

*Listing 3-7. SMOTEENN definition for classification model.*

```
X_aug, y_aug = augment_jitter_CLASSIFICATION(X_train, y_train, n_augments=1,
noise_level= 1) # noise_level (%)
```

*Listing 3-8. Jitter definition for classification model.*

The results of the augmentation can be observed on figure 3-22, where for each point, jitter was added to expand the dimension of the grid, and the *SMOTEENN* algorithm had augmented critical minority regions.

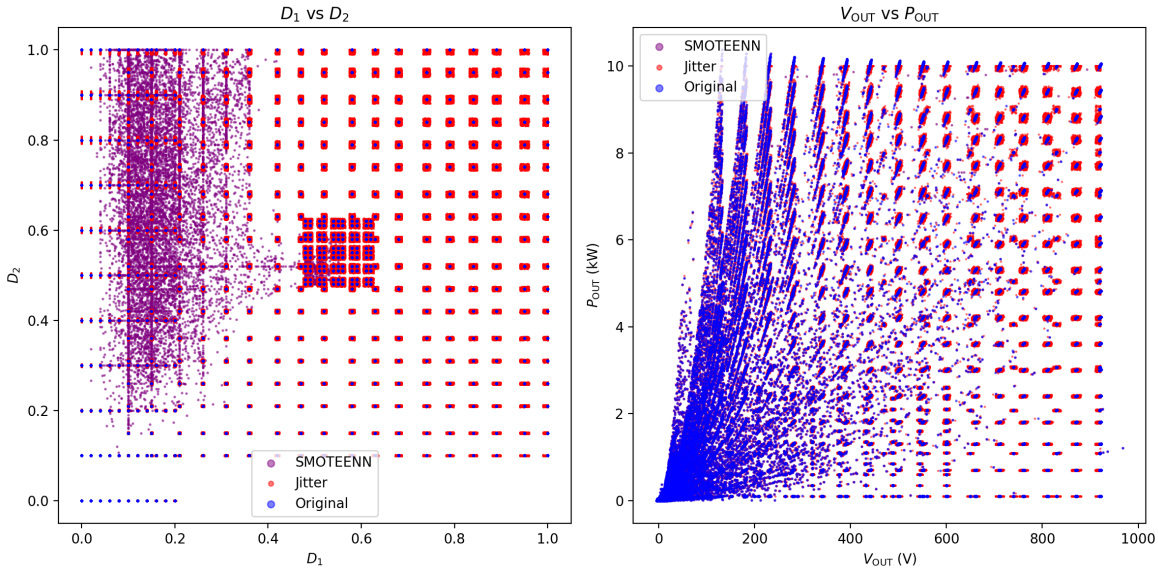


*Figure 3-22. Training set augmented data over original data grid for classification model.*

On the other hand, jitter augmentation technique was implemented for the **regression model.** The table and code definition are now presented.

| | ***X_train*** | ***y_train*** |
|---|---|---|
| ***ORIGINAL DATA*** | *[10 028 rows x 5 columns]* | *[10 028 rows x 1 columns]* |
| ***AFTER JITTER*** | *[20 056 rows x 5 columns]* | *[20 056 rows x 1 columns]* |
| ***TOTAL AUGMENTATION RATIO*** | ***x 2*** | |

*Table 3-5. Data augmentation for training vectors of regression model.*

```
LX_aug, Ly_aug = augment_jitter(LX_train, Ly_train, n_augments=1,
noise_level=0.5)
```

*Listing 3-9. Jitter definition for classification model.*

The results of the augmentation technique applied for the regression model can be visualized on the next figures.

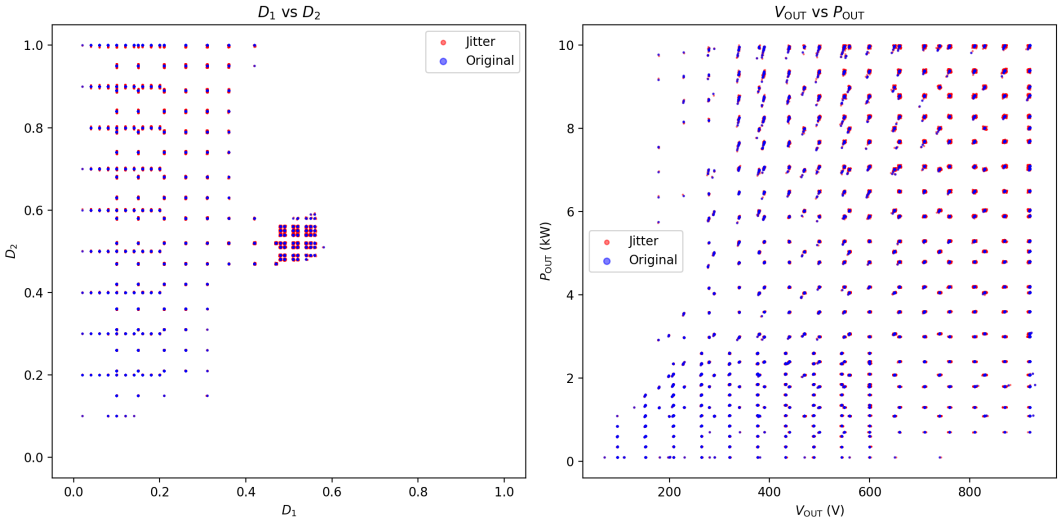


*Figure 3-23. Training set augmented over original data grid for regression model.*

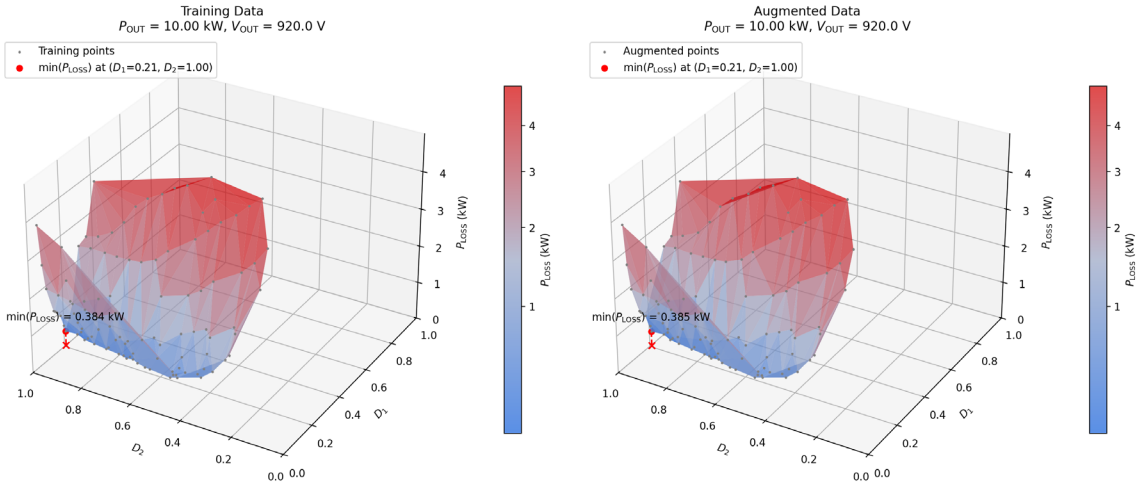


*Figure 3-24. Augmented jitter data comparison for regression model.*

As this technique generate data from an original row, the chart of the power loss for a combination of load condition can be plotted and compared to the one created with the augmentation jitter. While augmenting data, especially for tabular data, it is important to **avoid generating misleading or erroneous solutions** that could impoverish the behaviour of the model. Therefore, assigning a noise value too high, could help the model to cover a wider range of solution but sacrificing its accuracy, whereas a too small value will be more accurate but could lead to overfitting.

On figure 3-24, can be argued that both the original and jittered surface are similar. Indeed, on figure 3-25 both surfaces are plotted on the same axes. It is worth noticing that because of the sloping surface, the **mean relative error is greater than the noise assigned**—which is 0.5 %. Additionally, on high sloping regions, the difference can be even higher as for the plotted case goes to a 5.98 % maximum error.

Thus, this technique must carefully be applied with the awareness of knowing that the performance on the training of the model **could not always be enhanced with jittering technique.** Therefore, and to critically address this issue. For both the regression and classification model, each architecture will be trained with the augmented dataset and without it.

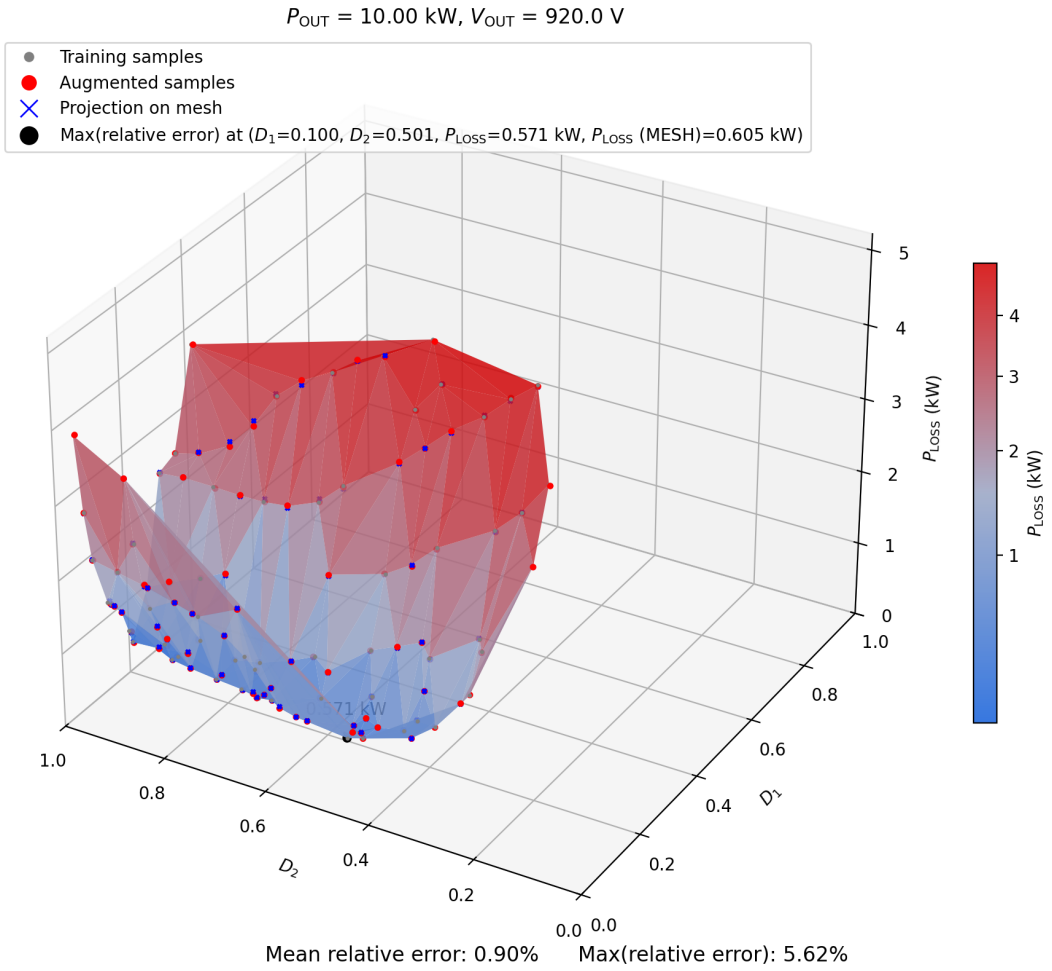


*Figure 3-25. Augmented data with jitter over original data comparison.*

Once the dataset is crafter, processed and adjusted, the training process for the AI models can be headed.

## 3.3 AI Training

This chapter covers the process of supervised training of the AI models used. First, and overview of the different architectures is made and then the results for the classification and regression model are presented.

### 3.3.1 AI Model Architectures

Based on the references [25, 61], the first architecture proposed was ***XGBoost*** (XGB), which stands as an **eXtreme Gradient Boosting algorithm** with decision trees and multi objective training capacity [62]. As a comparison, ***LightGBM*** was also used as it is also a gradient boosting framework that uses tree-based learning algorithms similarly to previous architecture, and it has gained recent recognition as the main competitor to XGB [63].

Those algorithms are gaining more attention as they are a light and suitable AI prediction solver for different scenarios. Nonetheless, they do not enable to customize and adjust from a lower level the behaviour of the AI as they can only be modified with control predefined parameters. Consequently, three different classes of **neural networks** were proposed using the ***Pytorch*** library. Those classes use different architecture techniques for the design of NN that are frequently known as ***SkipConnectionNN***, ***LeakySwishNN*** *and* ***DenseFCNN***. Each architecture is known covered in more detail.

#### A. XGBoost

The script encapsulates the XGB extreme gradient boosting classifier inside a **Scikit-learn pipeline**, to later performs a **two-step grid search** to calibrate its most influential hyper-parameters by doing first wider research and a narrower second one, which is frequently known as a **coarse-to-fine search strategy.**

```
pipeline = Pipeline([
        ('scaler', StandardScaler()),
        ('xgb', XGBClassifier(
            objective='multi:softmax',
            num_class=2,
            use_label_encoder=False,
            eval_metric='mlogloss',
            random_state=42
        ))
    ])
```

*Listing 3-10. Pipeline definition for xgboost.*

Wrapping the standardiser and the classifier in a pipeline forces every cross-validation (CV) split to repeat the scaling inside the training folds, eliminating so-called training-test leakage and guaranteeing that the model never sees statistics from the hold-out data [64, 65]. The objective is set to ***softmax*** for a binary task, or to ***reg:squarederror*** for the regression model.

```
param_grid = {
        'xgb__learning_rate': [0.01, 0.05, 0.1],
        'xgb__max_depth': [3, 5, 7, 9],
        'xgb__n_estimators': [100, 250, 500],
        'xgb__colsample_bytree': [0.7, 0.9, 1.0],
        'xgb__subsample': [0.8, 1.0]
    }
```

*Listing 3-11. Hyper-parameter coarse-to-fine method.*

The first grid search sweeps these coarse ranges with three-fold CV, scoring each configuration. The script then narrows the search around the best triplet, launching a refined second grid; this coarse-to-fine strategy drastically reduces the combinatorial burden while still exploring the high-gain region of the space [66, 67].

## B. LightGBM

*LightGBM* was implemented with the same philosophy of *XGBoost* by applying the same **two-stage pipeline** described earlier, the definition of the pipeline, using a scaler, is the following,

```
pipeline_lgb = Pipeline([
        ('scaler', StandardScaler()),
        ('classifier', lgb.LGBMClassifier(
            objective='binary',
            random_state=42
        ))
    ])
```

*Listing 3-12. Pipeline of lightGMB.*

The same hyperparameters for a coarse-to-fine search were used, so the implementation is the same as the previous chapter

## C. Pytorch SkipConnectionNN

*SkipConnectionNN* is a small multilayer perceptron that **embeds a residual skip connection**—where the activation leaving the first hidden block is added back to the output of the second block. Such residual links were popularised by *ResNet* and are known to ease optimisation in deeper nets by keeping gradients flowing and reusing early layer representations [68, 69].

```
class SkipConnectionNN(nn.Module):
        def __init__(self, input_dim):
            super().__init__()
            self.fc1 = nn.Linear(input_dim, 128)
            self.bn1 = nn.BatchNorm1d(128)
            self.fc2 = nn.Linear(128, 128)
            self.bn2 = nn.BatchNorm1d(128)
            self.fc3 = nn.Linear(128, 64)
            self.out = nn.Linear(64, 1)  # Binary Output
            self.relu = nn.ReLU()
            self.dropout = nn.Dropout(0.2)
            self.sigmoid = nn.Sigmoid()  # Sigmoid for binary

        def forward(self, x):
            x1 = self.relu(self.bn1(self.fc1(x)))
            x2 = self.relu(self.bn2(self.fc2(x1))) + x1
            x3 = self.dropout(self.relu(self.fc3(x2)))
            return self.sigmoid(self.out(x3))
```

*Listing 3-13. Class definition of SkipConnectionNN.*

This architecture transplants the residual-learning problem from *ResNet* into a **fully connected setting** where the output of the second 128 units block is added element-wise to the activation of the first, creating a fast path for both activations and gradients. Batch normalisation layers placed before each *ReLU* stabilise the summed distributions, while a linear width schedule gradually compresses the representation to curb over-fitting. A value of 20 % dropout after the third block provides additional regularisation, and a final Sigmoid converts the single logit into a Bernoulli probability compatible with binary-cross-entropy loss for class classification purpose.

The presented code is for the classification model, to implement this on the **regression** model, the **sigmoid** distribution conversion is omitted. The same applies to the next two neural networks.

## D. Pytorch LeakySwishNN

*LeakySwishNN* is an *MLP* that combines *LeakyReLU* to **prevent dead neurons** in the first layer with the smoother *SiLU/Swish* activation in deeper layers. *SiLU* is a self-gating, non-monotonic function introduced by Google Brain and shown to outperform *ReLU* on many deep networks [70–72].

```
class LeakySwishNN(nn.Module):
        def __init__(self, input_dim):
            super().__init__()
            self.net = nn.Sequential(
                nn.Linear(input_dim, 128),
                nn.LeakyReLU(0.1),
                nn.Dropout(0.2),
                nn.Linear(128, 64),
                nn.SiLU(),
                nn.Linear(64, 32),
                nn.SiLU(),
                nn.Linear(32, 1)
            )
            self.sigmoid = nn.Sigmoid()
        def forward(self, x):
            return self.sigmoid(self.net(x))
```

*Listing 3-14. Class definition of LeakySwishNN.*

The network begins with a LeakyReLU(0.1) to avoid the dying *ReLU* problem, then switches to *SiLU* (also known as Swish) activations, whose smooth self-gating has been shown to accelerate optimisation in deeper *MLPs*. A single dropout layer follows the first dense block as *SiLU* already supplies implicit regularisation through its non-monotonic curvature. Layer widths taper from 128 to 64 and finally 32 units, enforcing progressive feature compression and reducing the model's capacity in line with typical tabular-data sizes.

## E. Pytorch DenseFCNN

*DenseFCNN* adopts the **dense-connectivity principle** from *DenseNet*: every hidden layer concatenates its output to all previous feature vectors, allowing unimpeded information flow and extensive feature reuse [73, 74].

```
class DenseFCNN(nn.Module):
        def __init__(self, input_dim):
            super().__init__()
            self.fc1 = nn.Linear(input_dim, 64)
            self.fc2 = nn.Linear(input_dim + 64, 64)
            self.fc3 = nn.Linear(input_dim + 128, 32)
            self.out = nn.Linear(input_dim + 160, 1)
            self.relu = nn.ReLU()
            self.sigmoid = nn.Sigmoid()
```

```python
def forward(self, x):
    x1 = self.relu(self.fc1(x))
    x2_input = torch.cat([x, x1], dim=1)
    x2 = self.relu(self.fc2(x2_input))
    x3_input = torch.cat([x2_input, x2], dim=1)
    x3 = self.relu(self.fc3(x3_input))
    out_input = torch.cat([x3_input, x3], dim=1)
    return self.sigmoid(self.out(out_input))
```

*Listing 3-15. Class definition of DenseNN.*

Inspired by *DenseNet*, every hidden layer **concatenates its output** with all previous features (including raw inputs) before feeding the next linear transformation. This dense skip scheme fosters feature reuse, shortens effective gradient paths, and allows later layers to synthesise low- and high-level information jointly. Because concatenation continually enlarges the feature vector, hidden dimensions are intentionally modest, and dropout is omitted—the network relies on structural regularisation from its dense connectivity to temper over parameterisation.

### 3.3.2 Classification Predictive Model

Once the architectures are defined, the models are properly fed with the input vectors and then trained. The following figure describes the workflow followed to train the classification model.

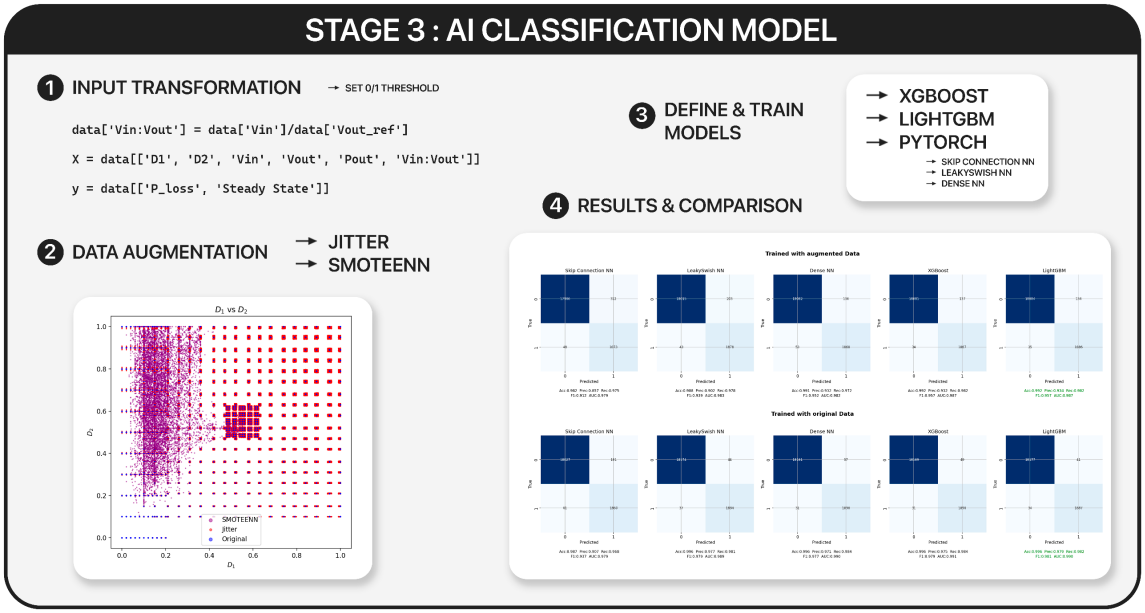


*Figure 3-26. Stage 3: AI Classification model.*

## A. Results

Now, the confusion matrix for the 10 models trained are presented. Each of the 5 AI models' proposal were both trained with **original and augmented data.** Important evaluation parameters such as the F1-score or the AUC are attached.

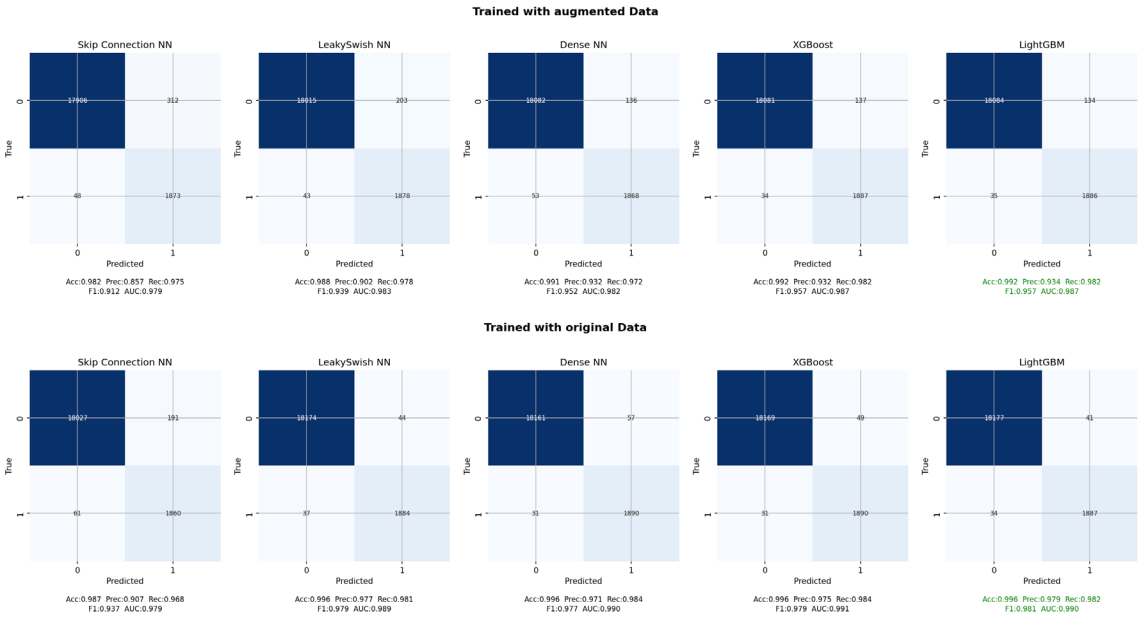


*Figure 3-27. Confusion matrix results for augmented and original data trained architectures.*

From the previous figure, it is needed to choose the one outperforming the others, and that was the case for the ***LightGBM*** architecture, which results are now highlighted.

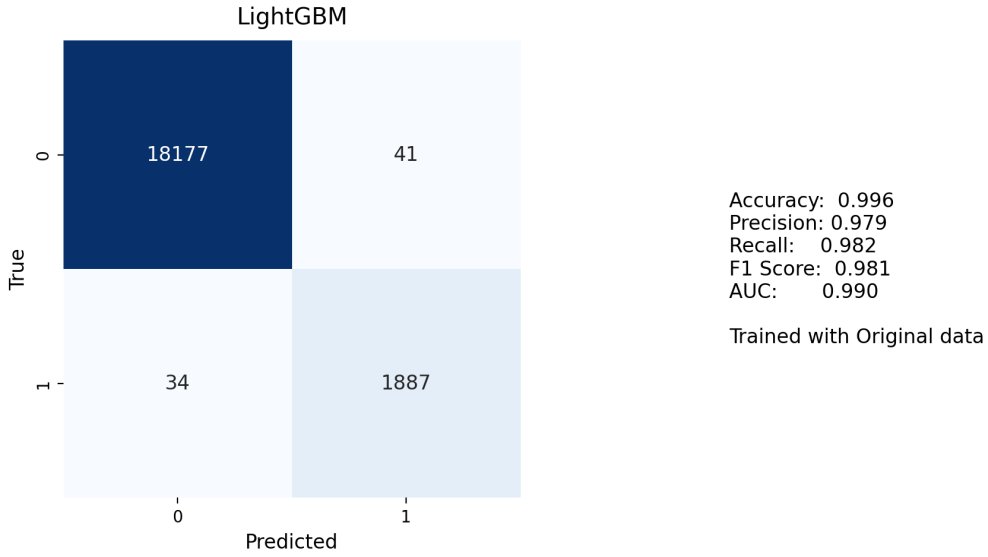


*Figure 3-28. Best classification result for LightGBM architecture.*

### 3.3.3 Regression Predictive Model

As with the classification model, the following diagram summarizes the steps on the training of the regression model.

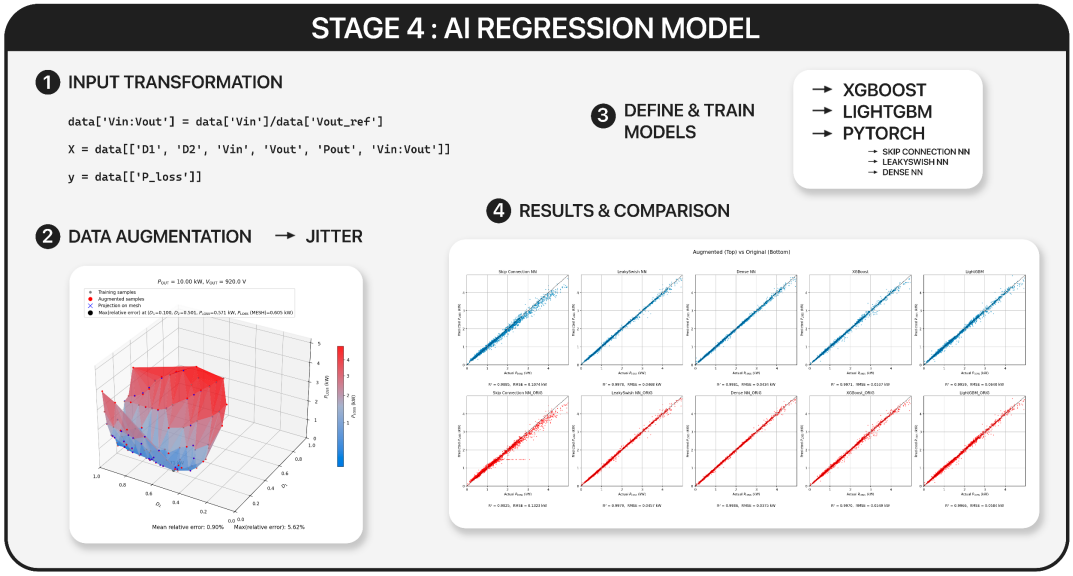


*Figure 3-29. Stage 4: AI regression model.*

Those models uses the same architecture previously explained and used on the classification model but with the respective configuration for regression training purpose.

## A. Results

The ground truth plot for all the trained architecture is presented on figure 3-29. In blue are represented the results for the models trained with augmented data. The results for the $R^2$ and the RMSE are attached.

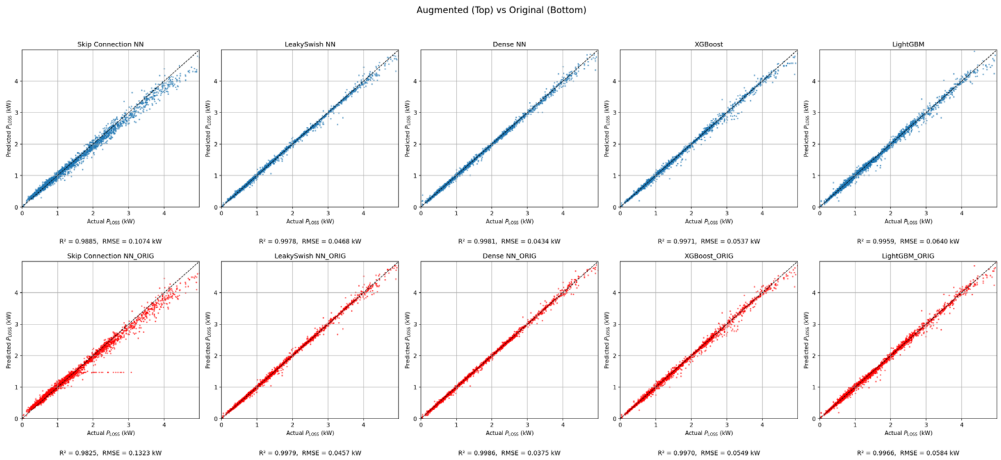


*Figure 3-30. Regression model results for original and augmented trained architectures.*

From the previous branch of graphs, the greatest performance was achieved by the ***DenseNN.*** Their results are now presented adding the relative and absolute error plot distribution (figures 3-31 and 3-32).

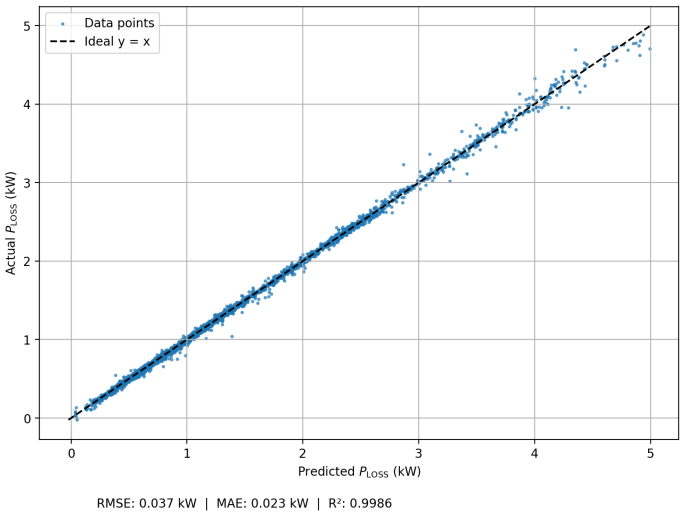


*Figure 3-31. Best regression model results for Pytorch Dense NN.*

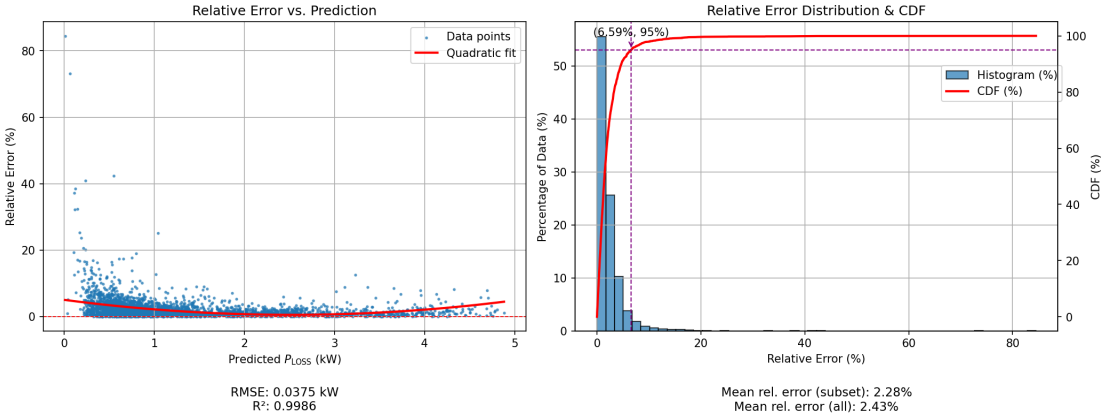


*Figure 3-32. Relative Error results for Pytorch Dense NN.*

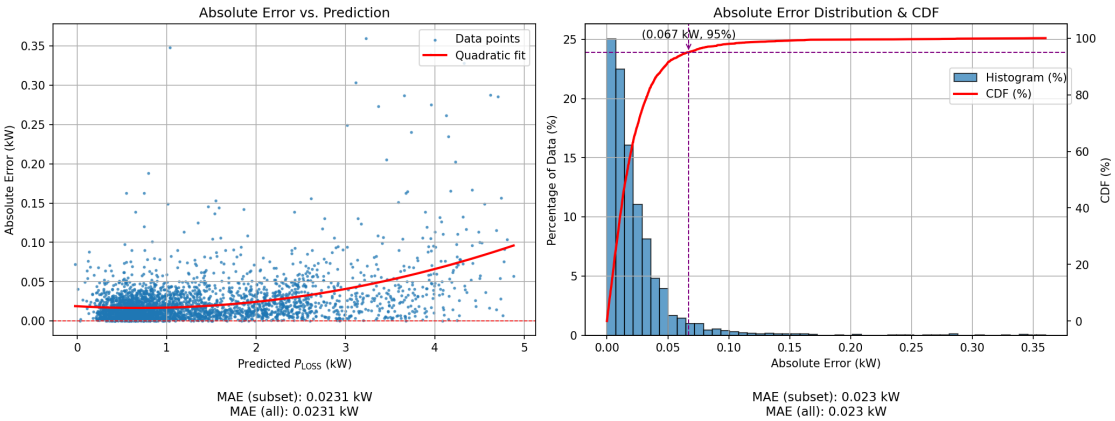


*Figure 3-33. Relative Error results for Pytorch Dense NN.*

### 3.3.4 Complete Model (Classification & Regression)

Once all the architectures were trained and the selection for the best ones was made, the complete model can be built and defined. The next two figures summarize the results and the design of the complete model.

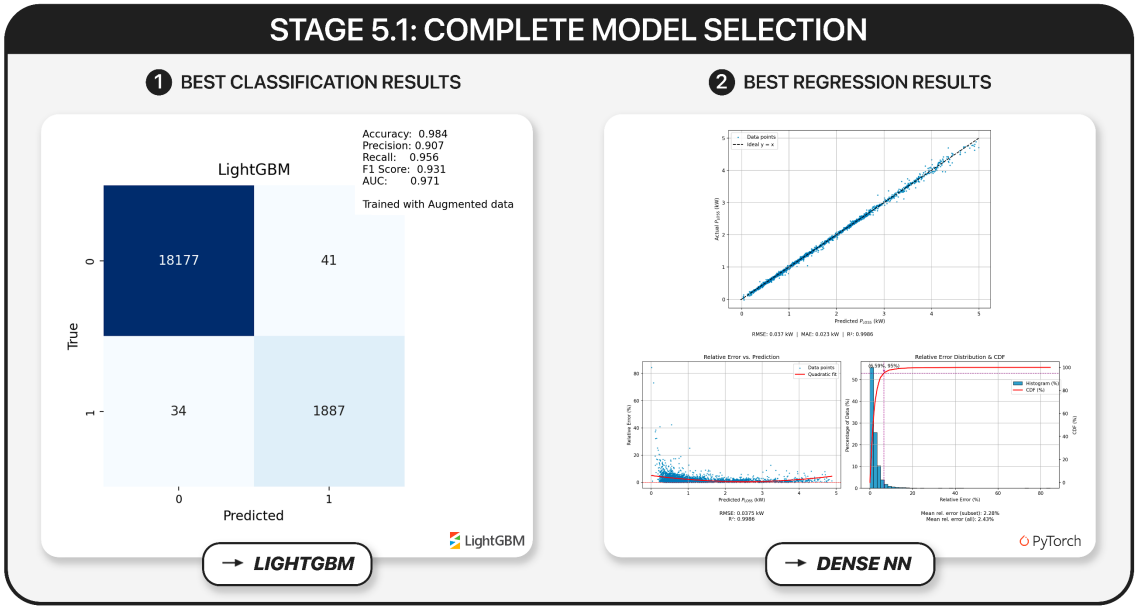


*Figure 3-34. Stage 5.1: Complete model selection.*

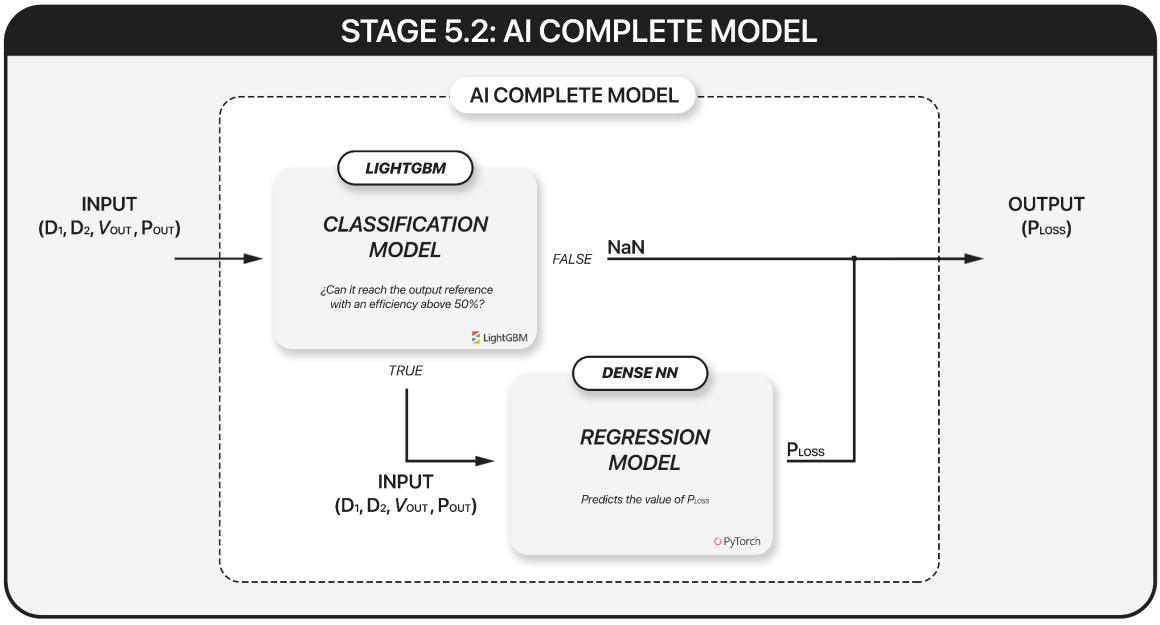


*Figure 3-35. Stage 5.2: AI complete model.*

## A. Results and Validation

To approve the results, **a validation set of 2,500 samples** was gathered and fed to the model. This set was built running simulation (see chapter 3.2.1) on random distribution points inside the ranges defined.

The models were exported on a pickle file with their respective scalers and class definitions. The following function replicates the operation principle of the complete model, where for an input vector (see listing 3-6) it will output the prediction of the power losses if the point satisfies the conditions.

```
def predict_generalized(x_input):
    pred_classify = predict_model_CLASSIFY(x_input)
    if pred_classify == 0:
        return np.nan
    return predict_best_model_LOW(x_input)
```

*Listing 3-16. Complete model prediction function.*

With this function, the validation set was run through, and the following results were found from the prediction of the model. First, **the confusion matrix** for the **classification** task is attached, and then, the results for the regression model on the points predicted as true.

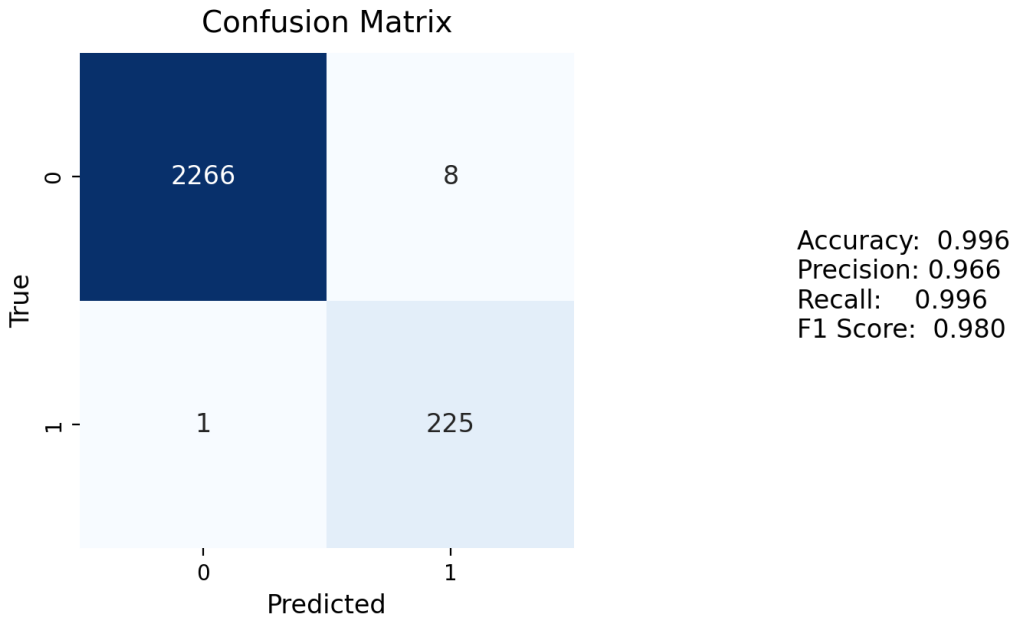


*Figure 3-36. Validation set classification result for Complete Model.*

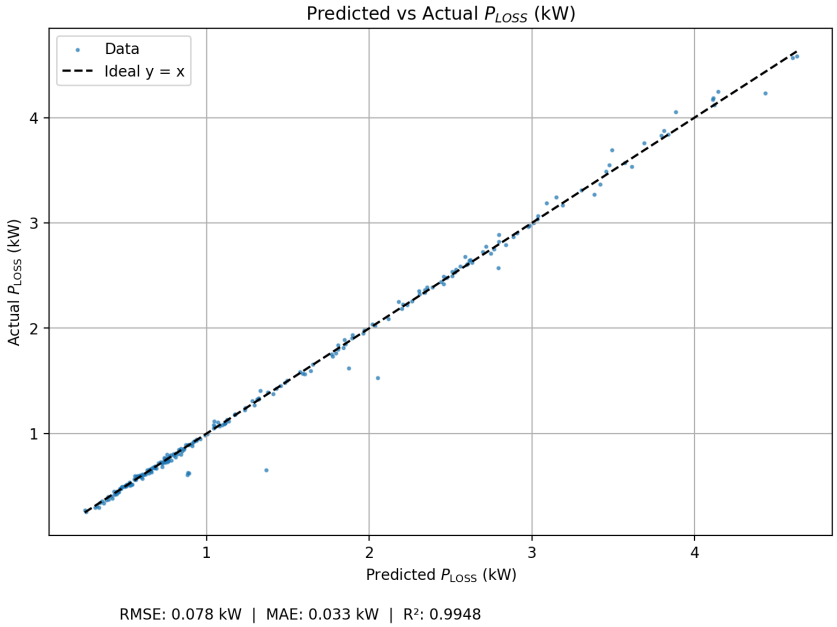


*Figure 3-37. Validation set regression result for complete model.*

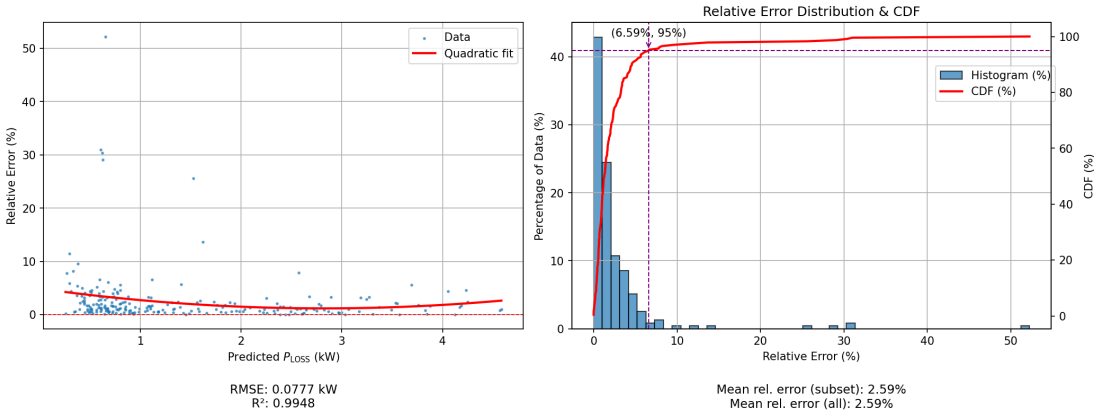


*Figure 3-38. Relative error results for complete model for validation set.*

The results obtained at this stage were considered as valid as the **f1-score** on the classification model at the **98%**, and the **mean relative error** on the regression model was a **2.59%**. Therefore, the complete AI model designed is considered as solvent.

## 3.4 Optimization Solver Prediction Model

With the previous model crafted, it is possible to predict the power losses of a DAB converter within a range of power and voltage output and a combination input of the control variable $D_1$ and $D_2$. Thus, for each combination of power and voltage at the output, each combination of $D_1$ and $D_2$ can be tested to **find the best control parameter** for this working point. Therefore, this solution can be plotted against the load output conditions obtained the most efficient strategy controller for the range of the converter.

Obtaining this solution is the aim of the present chapter, and its procedure is briefly presented on the next figure.

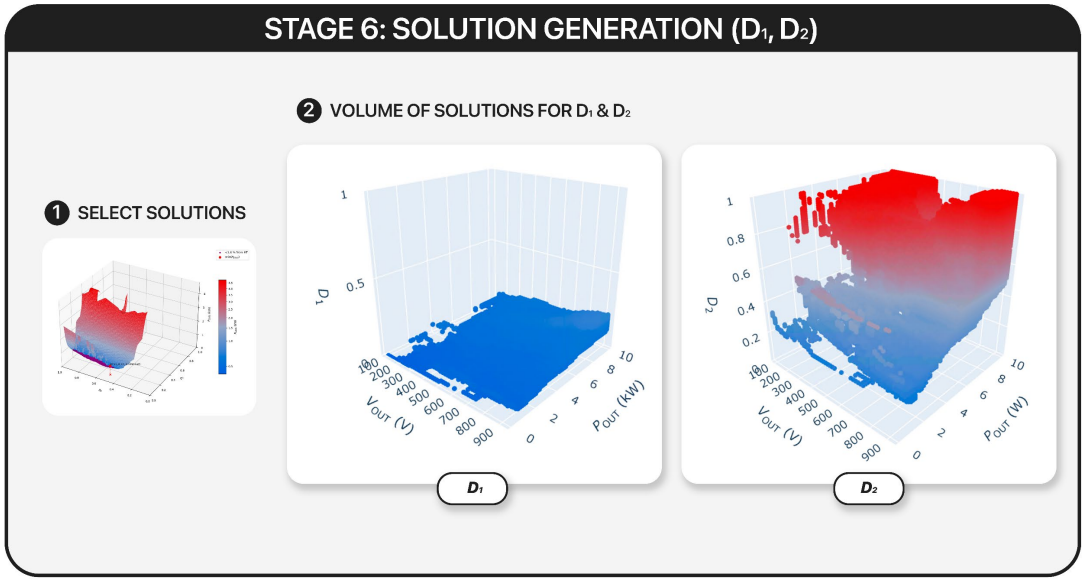


*Figure 3-39. Stage 6: Solution Generation ($D_1$, $D_2$).*

### 3.4.1 Prediction Generation

Following with this idea, the graphs illustrated on figure 3-40 represent the prediction made for a voltage output of 800 V and 8.5 kW where all possible combinations between $D_1$ and $D_2$ with a step grid precision of 0.01. With this prediction surface, the most efficient solution for $D_1$ and $D_2$ in order to achieve the greatest performance of the converter can be found.

This process has to be repeated for the entire range of voltage and power defined (see chapter 3.2.1) with a step parameter. Therefore, a **60x60 grid** of points for $V_{OUT}$ **and** $P_{OUT}$ were predicted using a **0.01 step for** $D_1$ **and** $D_2$ at each point. The results were stored on a pickle file for a total of 36.7 million predictions made—which was a 1.368 GiB amount of storage.

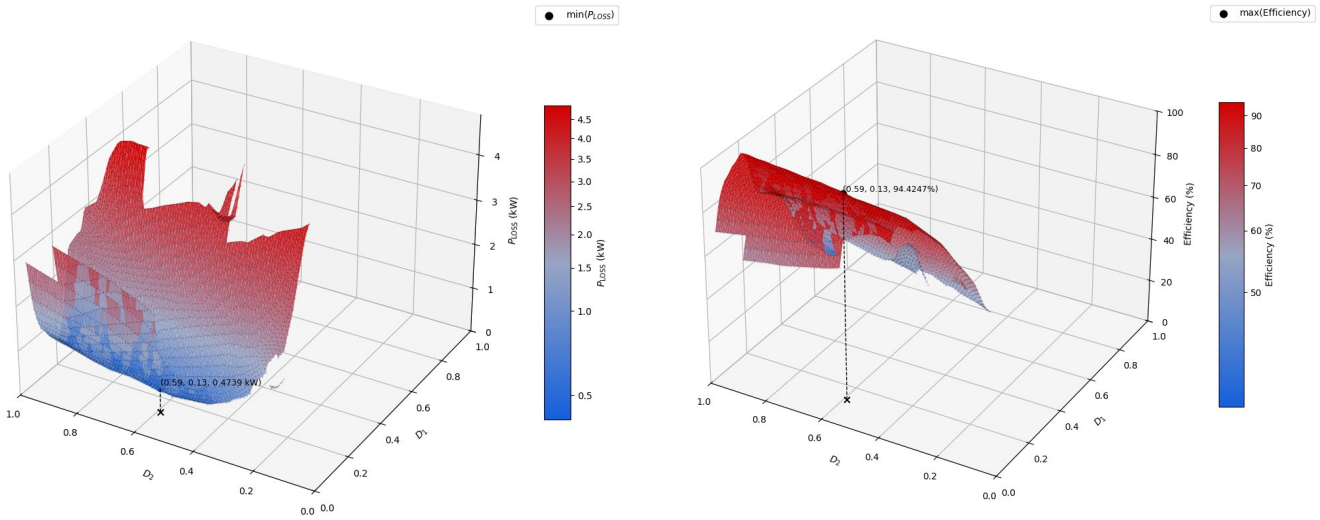


*Figure 3-40. Prediction for $P_{LOSS}$ and efficiency over $D_1$ $D_2$ for $V_{OUT}$ = 800 V $P_{OUT}$ =8. 5kW.*

At this point, could be argued that the predictions for the model and its information gathered may be redundant with what was found using the dataset. It is worth noting that the purpose to build an AI model is to outperform *PLECS* capabilities from a gathering data time-efficient perspective. The next table illustrates the results obtained presents the amount of data gathered and the time consumed comparing *PLECS* with the prediction AI model.

| | ***PLECS SIMULATIONS*** | ***AI MODEL PREDICTIONS*** |
|---|---|---|
| ***DATA GATHERED*** | *~ 100,000 rows* | *~ 36.7 Mill. rows* |
| ***GATHERING TIME*** | *~ 20,000 mins* | *~ 1,500 mins* |

*Table 3-6. PLECS and AI model data gathering comparison.*

Going back to the previous idea, when plotting and observing the results obtained on the experiments presented on figure 3-40, a **high efficiency area** was discovered for **several control combinations** due to the curved nature of the graph where a upper hill surface was spotted. Consequently, despite having a single possible peak efficiency solution, a wide range of other combinations were also delivering high efficiency relative to the peak, therefore, the idea of picking a single solution was not transparent with the results.

Thus, **a range of solutions** were taken as valid by stablishing a one percent maximum deviation relative to the peak for each experiment. The graphs of figures 3-41 and 3-42 represent in purple the range of possible solutions with high efficiency.

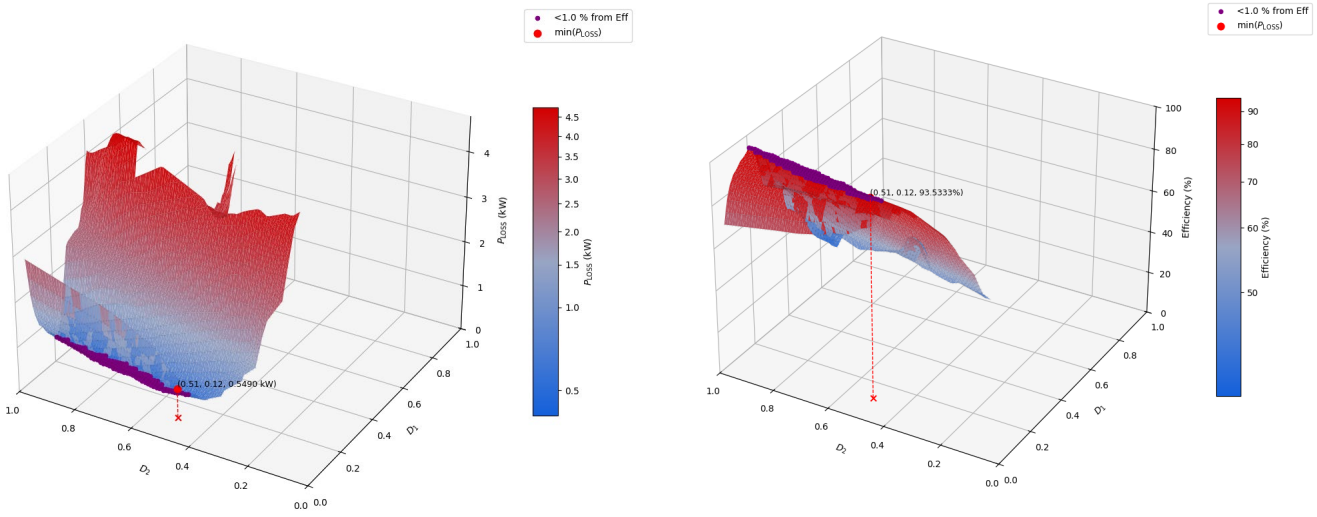


*Figure 3-41. Best prediction results over $D_1$ & $D_2$ for $V_{OUT}$ = 800 V & $P_{OUT}$ =8.5 kW.*

With this adaptation to the results, there is no longer a single possible output, but a range of possible controlling solutions. By gathering the range of high efficiency solutions obtained at each $V_{OUT}$ $P_{OUT}$ combination and plotting those results over an output voltage power grid, the following **volume** represents **the solutions for each $D_1$ and $D_2$ control parameter** that would be delivering the highest efficiency possible for each load demand on the converter.

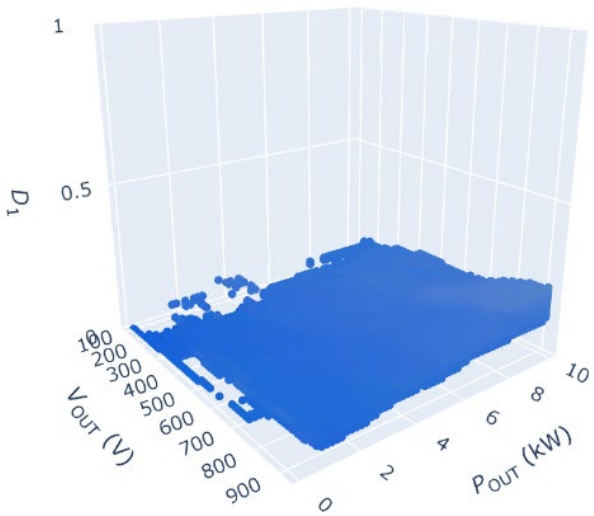


*Figure 3-42. Volume of solutions for $D_1$.*

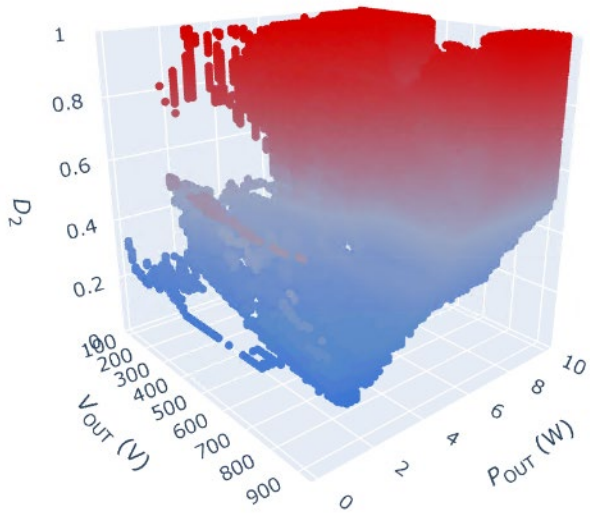


*Figure 3-43. Volume of solutions for $D_2$.*

In sum, figures 3-42 and 3-43 represent the final solutions that were desired throughout the experiment, so these figures summarise and represent the results obtained for the purpose of all the previous stages of research.

### 3.4.2 Generation of single solution

The results on the previous chapter (figures 3-42 and 3-43) finally delivered the solution graphs aiming to achieve during the curse of this project. Meanwhile, those solutions while being more representative from a physical perspective, could not be solvent form a practical standpoint as they represent the whole volume of possible solutions. Therefore, not proposing a control solution could be a big downside from an engineering perspective.

The present chapter propose a single solution solver that extract **a single surface for $D_1$ and $D_2$ out of the volume of point**. Therefore, the obtained solution could be applied on real-time control application for the studied topology.

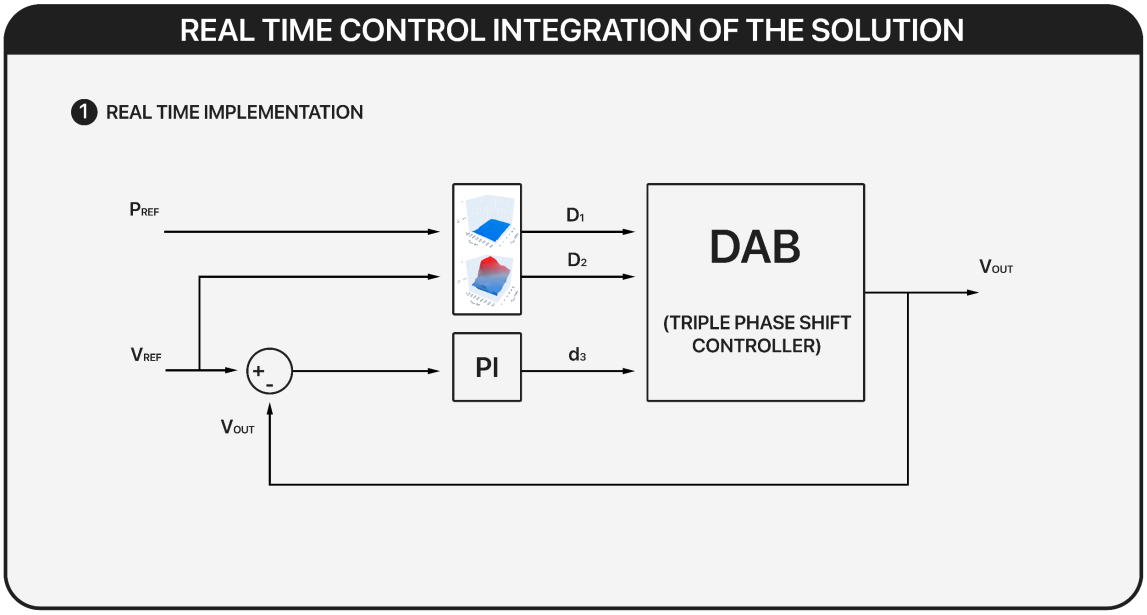


*Figure 3-44. Real Time control integration of the final solution.*

Figure 3-44 represent the close loop diagram where the single solution obtained for the control parameters can be applied to get the highest efficiency possible at the output from each voltage and power at the output.

The content of the chapter is summarised on figure 3-45, which is attached at next page.

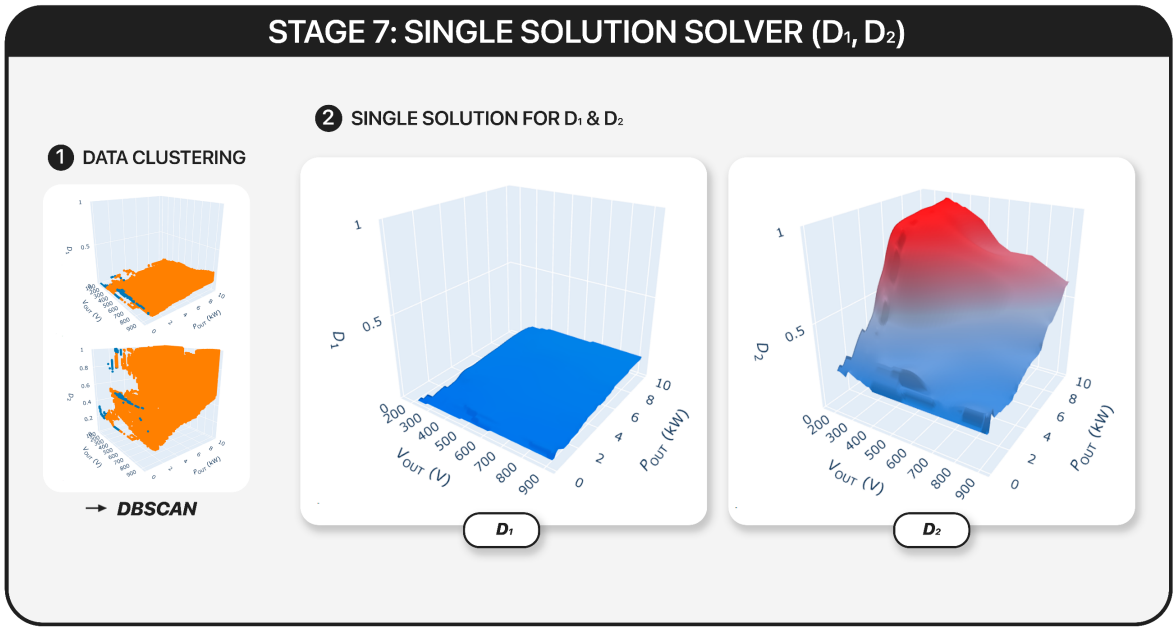


*Figure 3-45. Stage 7: Single solution solver.*

As a starting point, it is worth mentioning that this process can be driven by proposing many other alternative methodologies and strategies. Therefore, the presented methodology aims to represent the solution in a **smooth slope optimized surface** that avoid generating false solutions and enhance performance on real time implementation as variation between load condition are minimized.

In order to achieve that, it was noticed that, in critical region, bigger gaps between possible solutions for a power and voltage combination were appearing. In order to group close by solutions and isolate noisy points, a **data clustering technique** was applied using ***DBSCAN*** algorithm.

Density-Based Spatial Clustering of Applications with Noise (*DBSCAN*) is a non-parametric, density-based algorithm that groups points having at least a minimum number of **neighbours** inside a user-set radius ($\varepsilon$), designating such dense regions as clusters and labelling sparse points as noise. It begins from an arbitrary seed point and recursively expands clusters through density-reachability, allowing it to uncover clusters of arbitrary shape without requiring you to pre-specify how many clusters there should be [75–77].

The next figures represent the volume of solutions for $D_1$ and $D_2$ with the noisy points highlighted in blue. Grouping the closes data facilitate the smoothing principle afterwards and contributes to generate a much realisable solution.

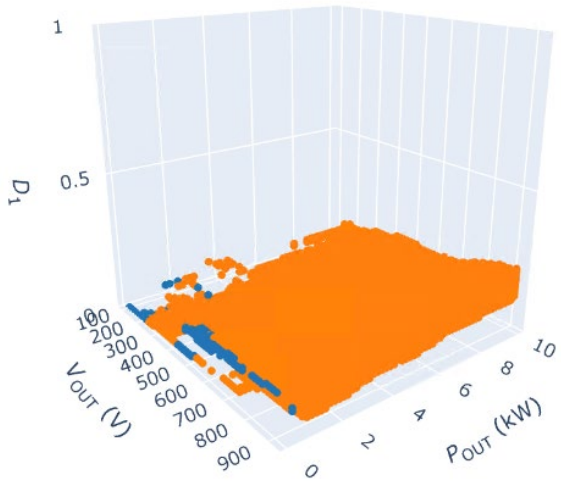


*Figure 3-46. Data clustering $D_1$ with DBSCAN.*

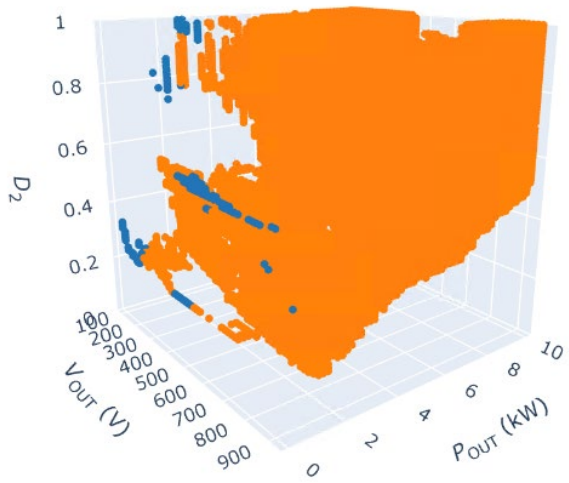


*Figure 3-47 Data clustering for $D_2$ with DBSCAN.*

The parameters for the visualized results were the following.

```
X = df[['Vout', 'Pout', 'D1']].values
Xs = StandardScaler().fit_transform(X)
clustering = DBSCAN(eps=0.55, min_samples=150).fit(Xs)
```

*Listing 3-17. DBSCAN Clustering definition.*

Once the data is clustered, a slope smoothing algorithm is implemented for the resulting cluster. The smoothing algorithm uses **simulated annealing**—a stochastic, temperature-driven search algorithm—to pick one candidate *z* value per grid cell with the aim of resulting a surface as smooth as possible. The next group of figures illustrates two different angles of the 3-dimensional view of the surface results for $D_1$ and $D_2$.

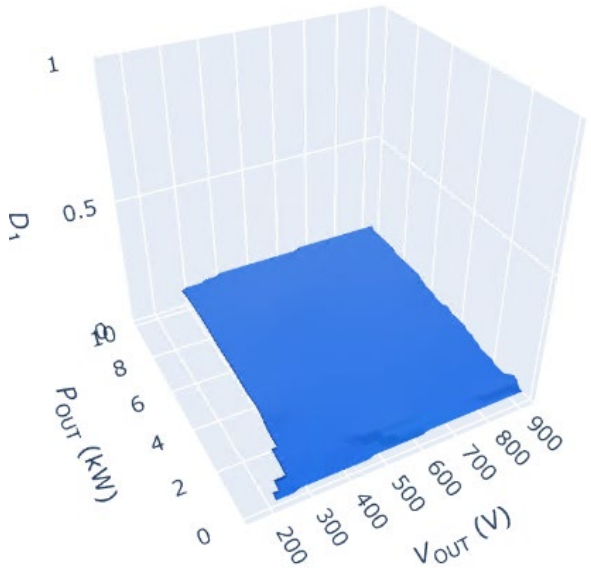


*Figure 3-48. Smoothed single solution for $D_1$ (1).*

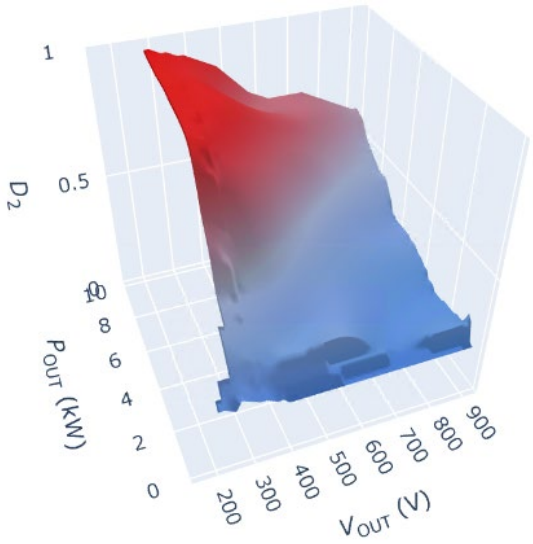


*Figure 3-49. Final smoothed single solution for $D_2$ (1).*

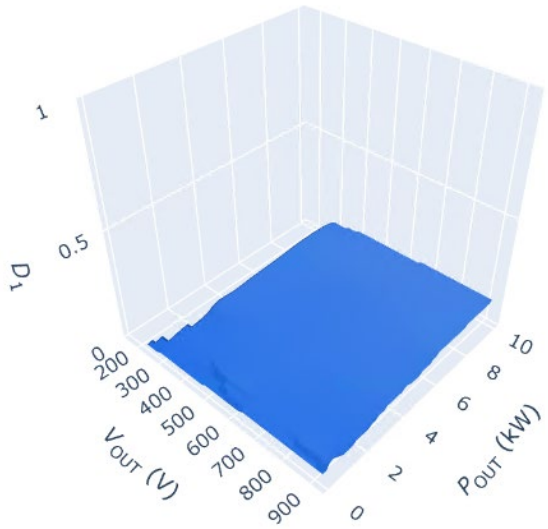


*Figure 3-50. Smoothed single solution for $D_1$ (2).*

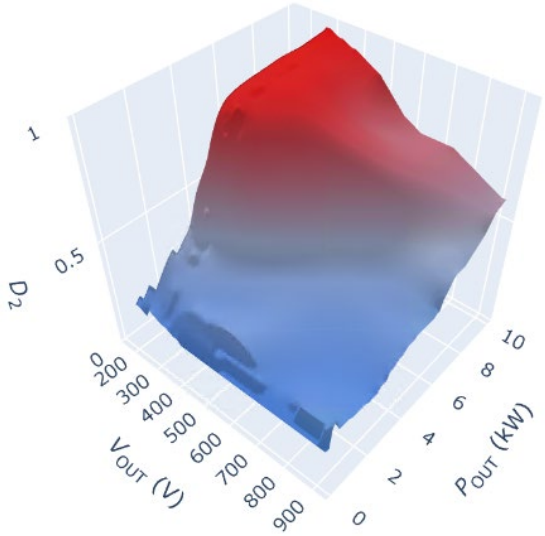


*Figure 3-51. Final smoothed single solution for $D_2$ (2).*

Finally, a bi-dimensional top view of the solution is presented on the following **heatmap** to facilitate the visualization of continuous surface and boundaries.

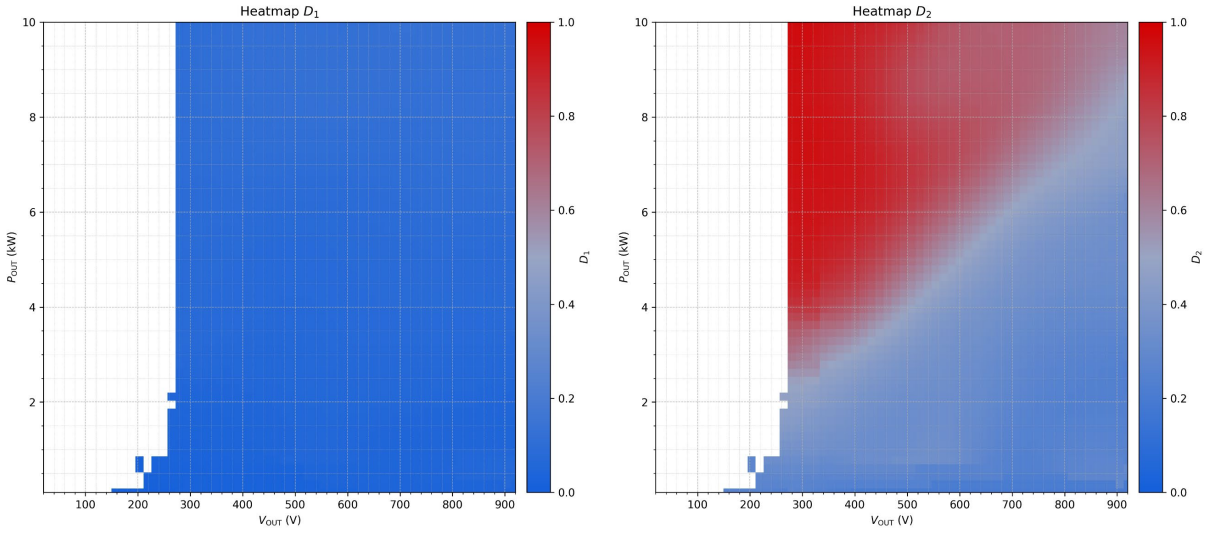


*Figure 3-52. Heatmap top view of smooth mesh solution for $D_1$ & $D_2$.*

Furthermore, it is worth noting that this solution culminates the purpose of the research, which was to design a methodology to obtain a practical solution for a real time control of a DAB converter with a TPS controls strategy.

# 4 Realisation of Solution

In the previous chapter, it was discussed the whole methodology and technical approach. In this chapter, the aim is shifter to analyse from a study perspective the meaning of the results and the findings.

## 4.1 Regions and Waveforms Analysis

As explained on previous chapters, and when visualizing the volume of feasible solutions for $D_1$ and $D_2$ (see chapter 3.4.1), the converter cannot reach the reference at every required load. This presents the idea of have a stable and a non-stable region. Furthermore, the surface varies throughout the whole stable surface, providing a different behaviour at any point.

Therefore, it is worth investigating the waveforms of the inductance current at those regions. First, we would present the regions founds, compare their waveforms; and then analyse each one thoroughly to finally take a deep look at the border between regions.

### 4.1.1 Region Presentation

Based on the work presented, we would use the heatmap to classify two regions found. Obviously, those regions are the same for both maps as they are corelated, and the border can clearly be appreciated for the absence of colour on the non-stable region.

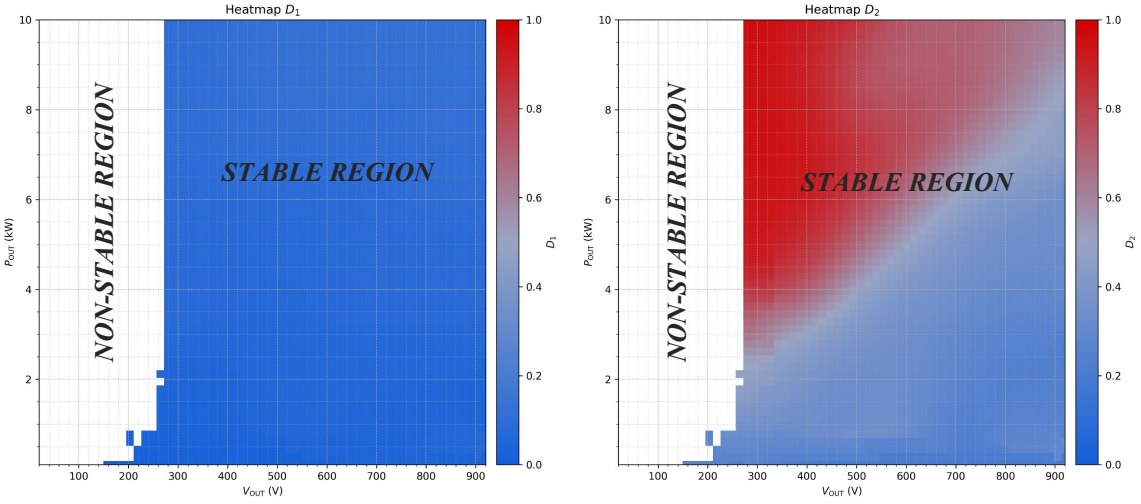


*Figure 4-1. Region definition on the Heatmap.*

The regions identified stands as the following. Further discussions for each one individually are made at their specific subchapter:

- ***Non-stable region:*** This region corresponds to the blank part mostly on the left of the map for low voltage condition. This section has no data results for neither $D_1$ nor $D_2$ as the classification model has not found any true solution.

- ***Stable region:*** This is the largest region found. It remains almost steady for $D_1$, where it has a small value close to 0.1. On the other hand, $D_2$ has a much broader variation and goes all the way from top to bottom. It covers the whole aera of power and medium-high voltage.

### 4.1.2 Non-stable Region

As previously introduced, this region stands for the area where there were not data points predicted. From a technical perspective, this is because the **classification model** is not predicting any potential highly efficient value (see chapter 3.3.2), therefore there are no solutions presented there. It is worth noting that the classification model predicts as true the working points that achieve an efficiency rate over 50% while reaching the reference voltage output. Therefore, at this region we can find both cases:

- Points that due to converter limitations cannot reach the reference output of power and voltage.

- Points that, while reaching the reference output, the efficiency is not great enough to get over 50%

It is great realising the reasons for these events with an explanation based on the principles of the converter. Thus, from an engineering standpoint, this region covers the area of low voltage (**mostly below 280 Volts**), and it is notable how the area grows as the power does until a point where it remains straight, generating a line as the border.

Multiples reason argues the existence of this border. Firstly, the value of the voltage at the input remains constant for all gathered datapoints at 800 Volts. Consequently, when the reference voltage output is small, the big step difference induces high conduction losses, mostly in the secondary side, where a great voltage reduction has to be done (it is also great reminding that the converter taken for simulation has a turn-ratio of 4). In addition, for any power output, if the voltage is low, the current will consequently be higher which **increases the current stress** on the inductance by requiring higher RMS —to achieve the referenced power—and therefore inducing in higher losses.

This current limitation can be shown at simulations by visualizing the voltage output and inductive current evolution over time. On figure 4-2, the voltage at the output is

represented in green, with the reference value in red. It can be easily observed how the voltage output ramps up until it hit a peak value and goes back to zero, not reaching the output reference value of the close loop. The simulation parameters at attached on the top right of the figure.

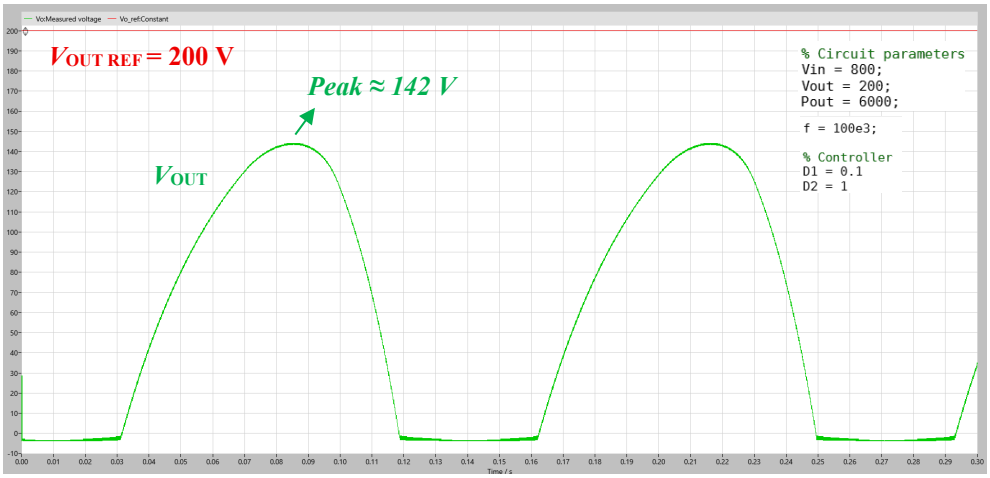


*Figure 4-2. Voltage output not reaching the output reference.*

If we look also to the current evolution at the inductance, it is clear how the amplitude on the current goes higher as the transfer tries to achiever a greater value of voltage but hit a top boundary limitation from where it starts decreasing again as it cannot deliver a greater value of current.

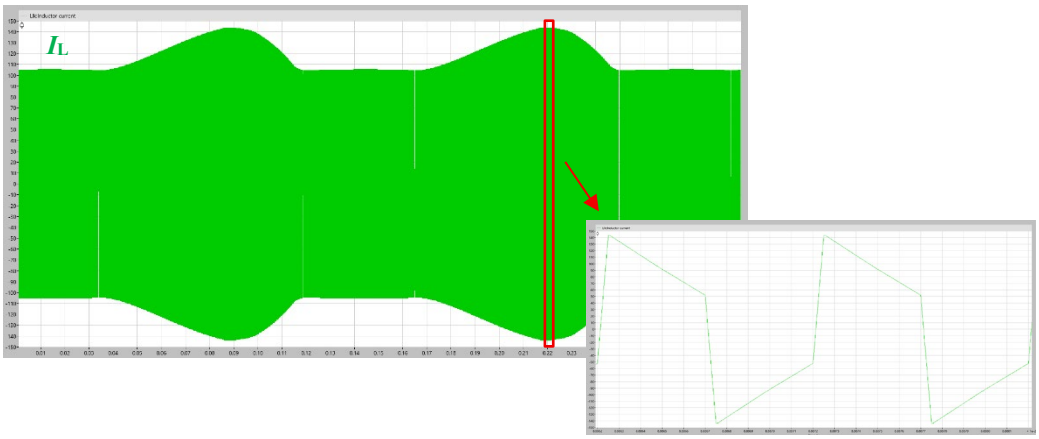


*Figure 4-3. Inductance current for working point not reaching the reference.*

This limitation can be explained with the equation 2-2 mentioned on chapter 2.1 that describes the **power transmission** through the primary and secondary bridge. In order to achieve a greater amount of current, the power limitation must be raised which means that the value of the switching frequency must be higher. By doing so, the inductance has more time to charge and can reach a higher peak value, and therefore, deliver a higher amount of power.

To overcome the limitation, also the control strategy can be change, indeed, the failure on achieving the reference could be argued to be on the performance of the PI close loop controller. Nonetheless, all the simulation were gathered considering a perfect sampler controller for the close loop as its sample time is equal to the frequency of the system (see chapter 3.2.1). The previous explanation regarding figures 4-2 and 4-3 validates the first argument of the two proposed.

Furthermore, when the voltage and power are low (under 2kW), there is a small cloud of points drawing a curve where solvent control parameters were found. Those points were tested on simulation and proved as real solutions. The explanation come in line with the previous idea, knowing the fact that if the power is smaller, **the current stress also decreases**, enabling the converter to be operable at low power and voltage conditions.

On the other hand, as the power gets higher, a few points were spotted that under a small range of values of $D_1$ and $D_2$ **can reach the reference** output while not being represented on the heatmap. Nevertheless, they achieve date with a **poor efficiency rate** (below 65%). Figure 4-4 plots the voltage output achieve for a combination of power and output on the non-stable region. It is worth noting the parameters used and the efficiency rate obtained.

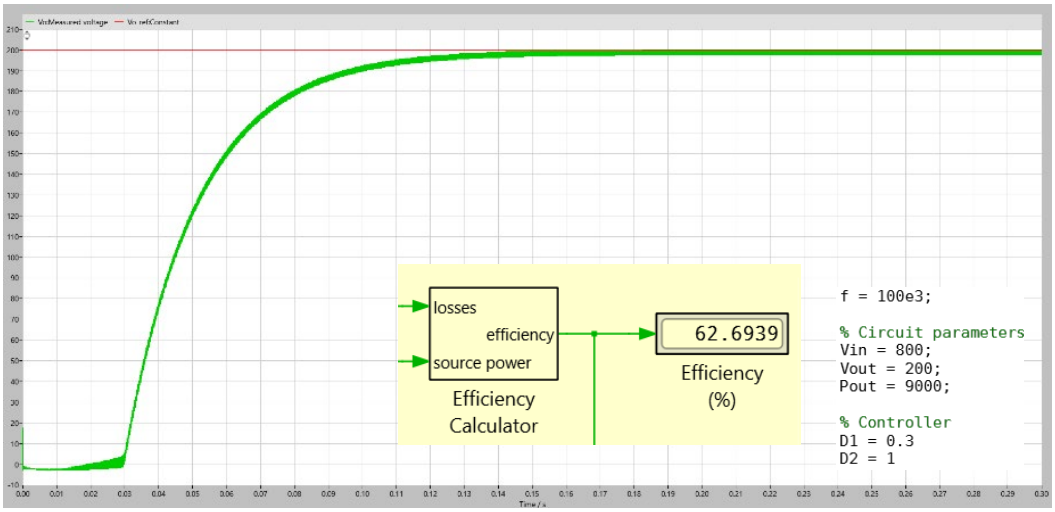


*Figure 4-4. Low efficiency point region.*

Because the classification model classifies as true based on an efficiency threshold, those points are on the critical region for the classifier and are therefore not being consistently predicted as true. In addition, the single solution filtering and smoothing has cleaned this noisy region. At any rate, this area does not prove to have high efficiency reliable working points consistently, so running the converter on this section does not have much interest for the presented purpose. With that being said, we can

confirm that there are operable points in this region, but the efficiency is so low that they are not classified as true by the AI model, which argues the second statement proposed.

### 4.1.3 Stable Region

The area named as stable region covers the major region of the heatmap. It encompasses the whole range of power and a section of mid-high voltage. During all its extension, the value for ***D*₁ remains almost stable**, whereas for ***D*₂, it has a smooth decrease** as the voltage increases and the power decreases—going from a maximum value close to 1 to a minimum of 0.15 approximately.

The first thing worth noticing is, as previously introduced, the graph of the efficiency for every possible combination of $D_1$ and $D_2$ represents a curve with a peak region where the values of efficiency are close by. This region is located at a specific range of $D_1$ that enables to find the peak of the slope. Once at that point**, the variation of *D*₂ is much more freely adjustable**, while still being restricted.

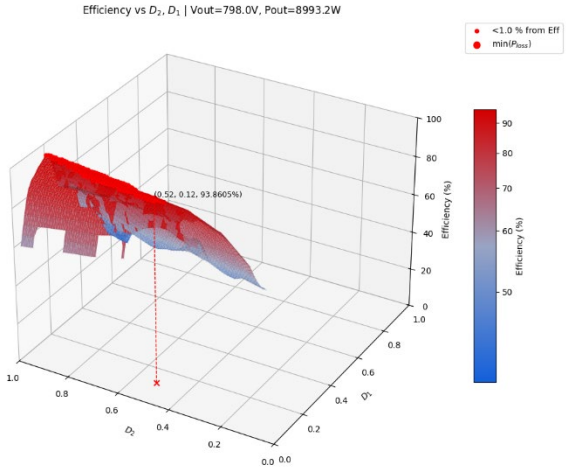


*Figure 4-5. Efficiency graph for $D_1$ and $D_2$.*

Thus, this means that in this region, the efficiency of the converter is **much more sensitive to the variation on the duty cycle of the primary side** rather than the second one. This explains why the cloud of points plotted at the volume of solution for $D_1$ was much narrower and more restricted than the possible output of $D_2$ (see figure 4-5).

This shape for the efficiency curve is found within the whole region, except for the fact that the value $D_2$ becomes more restricted as the power output decreases. The following graph on figure 4-6 compares the power losses output for the same voltage at a high and low power output (9 kW and 4 kW).

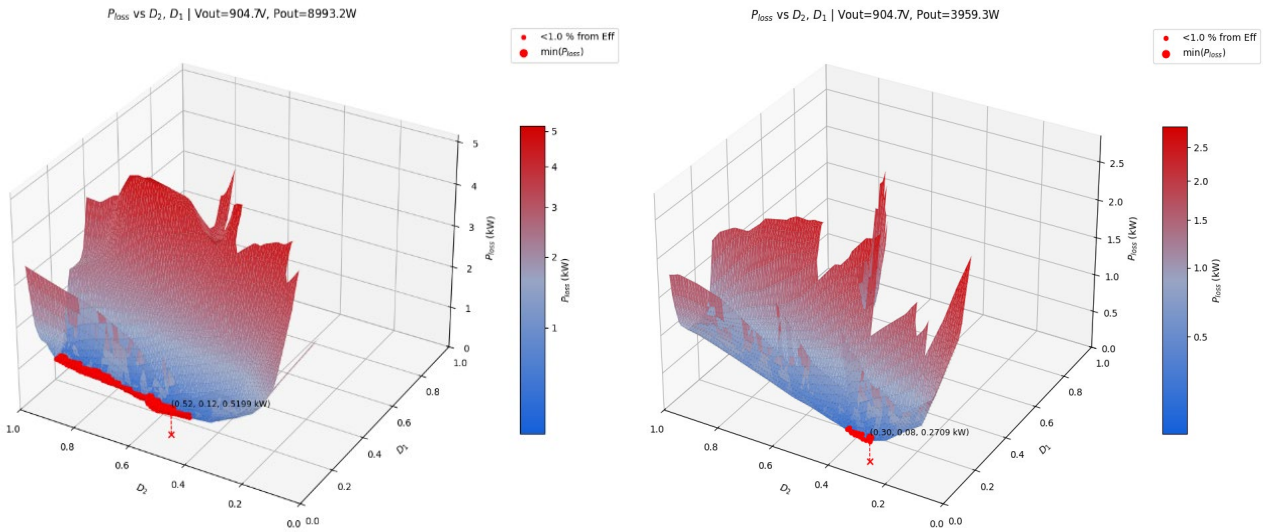


*Figure 4-6. Compare of power loss for high and low power output.*

It can be easily visualized on the previous figure how the shape of the curve remains similar, but at a low power, the hill also has a slope related to the secondary duty cycle, making the range of solution for the secondary duty cycle much narrower.

Once this was realized, it is worth having a look at the waveforms for the inductance current at different points of the region. As previously discussed, it was argued that the region had a similar working operation for the whole surface, but actually a slight variation on the behaviour of the converter due to the load condition was founded.

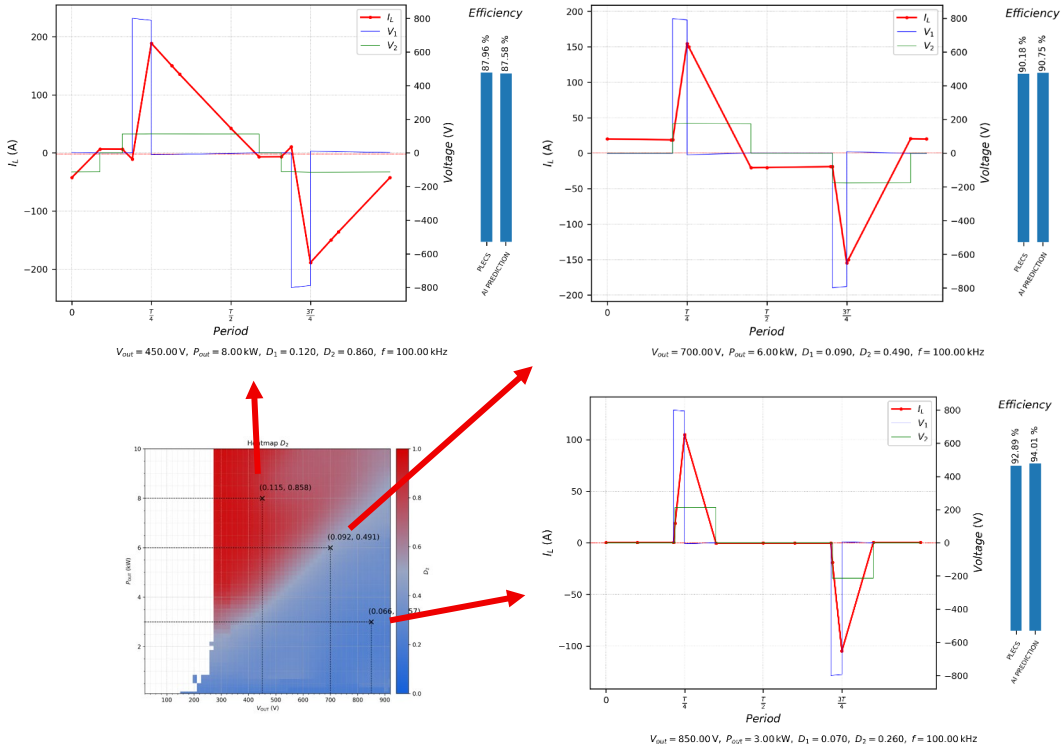


*Figure 4-7. Inductance current waveforms for region 1.*

On the previous figure, there are compared three operation points within the region for which the waveform for the inductance current is attached. First, it is worth noticing that the chart also describes the voltages on both sides of the inductance ($V_1$ and $V_2$), and the efficiency obtained at the simulation is compared to that obtained by the prediction of the model.

The **prediction of the efficiency** for the three cases attached is highly close to the efficiency obtained by the *PLECS* simulation, which means that the prediction model is working successfully. Therefore, this validates the performance of the AI prediction model.

Moving on to analyse the waveforms, it is clear that all three charts have a similar shape, having a rapid increase to get to the peak current, and a slower decrease. This rapid increase is due to the fact that the primary duty cycle is similar for all plotted charts. Meanwhile, the decrease depends on the secondary duty cycle, which gets smaller from the first chart to the following.

Furthermore, **the overall efficiency increases as the power does**. This can be explained as the converter not needing to deliver a higher current peak, which minimizes the RMS of the inductance current and therefore minimizes the power losses of the converter [10, 78]. Furthermore, this can be easily contrasted with the idea of the secondary duty cycle also decreasing with the power output, which means that the inductance has a much quicker decrease time. In addition, there are too small or no sight of backflow current on all three charts as following the definition given on references [37, 79].

Also the fact that $I_L$ is positive when the high branch transistors are turned on and negative when the low branch transistors are turned on minimize the leakage inductance current which enhance the condition for having ZVS in both bridges, which minimize the switching losses of the converter [10, 78, 80, 81]. This ZVS can also be visualized on the fact that the current remains almost or close to be zero right before the voltage on the primary side turns on, therefore this produces this soft switching on the switches that drastically reduce the switching losses.

Finally, the whole research work was done by optimizing the selection of the parameters $D_1$ and $D_2$, where for a TPS control strategy the $d$ parameter still freely adjustable. On figure 4-7 it is also worth noticing that not only $D_1$ and $D_2$ were responding to the load condition, but also d, which had a value close to 180 degrees for low power conditions, and a greater value as the power increased.

### 4.1.4 Critical Regions

To validate the whole performance of the solution, the regions where the behaviour of the controller may be critical has to be analysed. Therefore, the aim of this chapter is to

visualize the most critical region to evaluate thoughtfully the quality of the solution under stress conditions. Consequently, the waveforms on the most critical areas of the surface will be now analysed.

Fist, two experiments for **high power and low voltage** conditions are evaluated. Those points are close to the border, so the efficiency achieved is expected to be smaller than in previous presented regions.

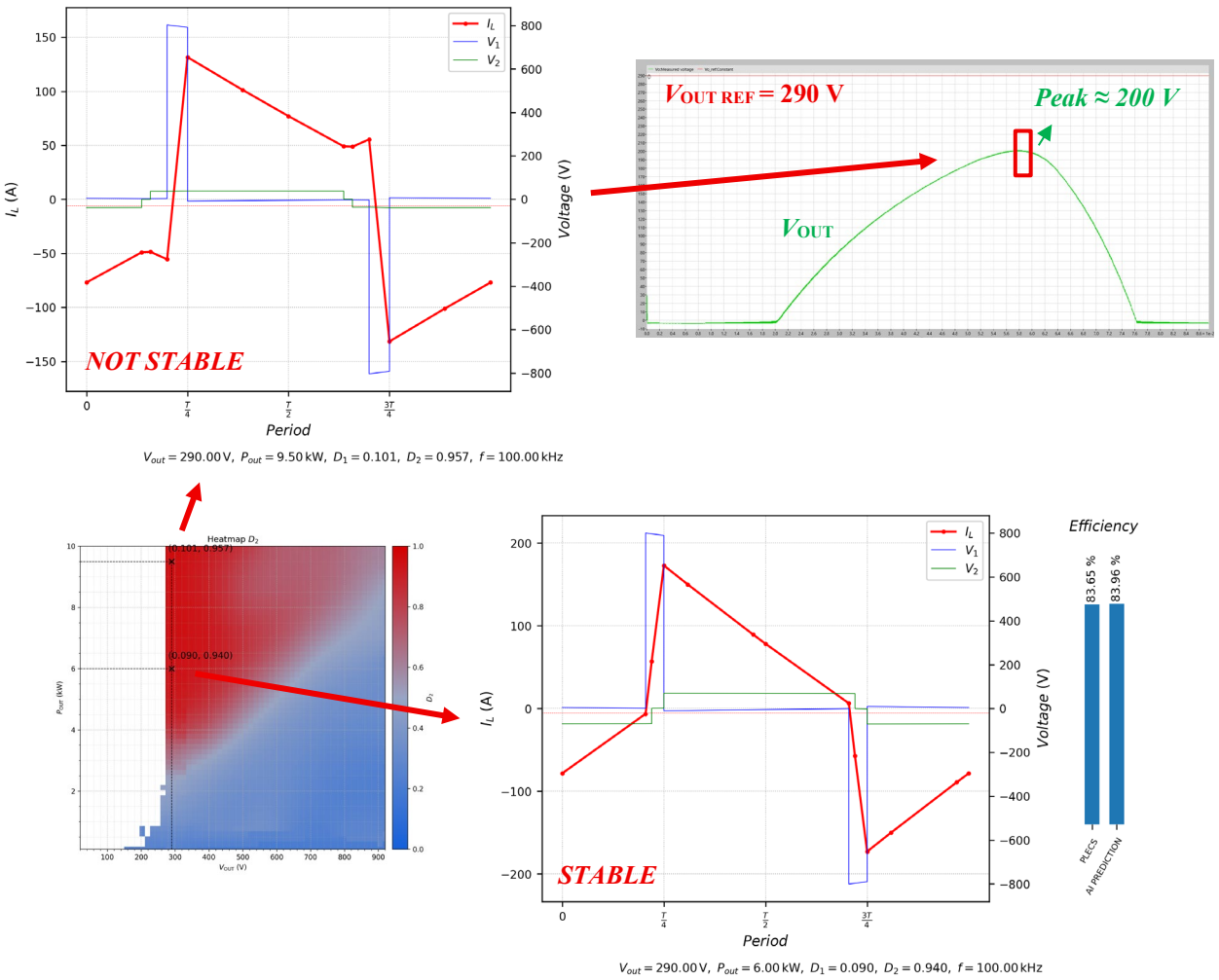


*Figure 4-8. Inductance current waveforms for the border on high power conditions.*

First thing noticing is that for the experiment at **9.5 kW**, the transformer **is not able to reach the referenced voltage reference** nor power reference. This is caused by current stress limitation on the conductance. It can be observed on the plot that the RMS on $I_L$ is apparently high, which is causing high conduction losses which lead to an unstable response. On the other hand, when the output power was set to 6 kW, the solution was

reaching the voltage reference, and the efficiency was considerably high. Meanwhile, if compared with the experiments on figure 4-7, that efficiency rate is smaller than the other one.

Furthermore, the fact that the solution was not stable for the higher power and then it was stable for the slower power condition, reinforce the statement that the performance of the DAB is hardly limited by the **current stress** on the converter. Therefore, when higher peak current is needed at the inductance, higher losses are found and even unstable conditions to reach the references.

Moving on to the second scenario presented on figure 4-9 two **low power** points were plotted for low and high voltage conditions. Here, the current stress is minimal.

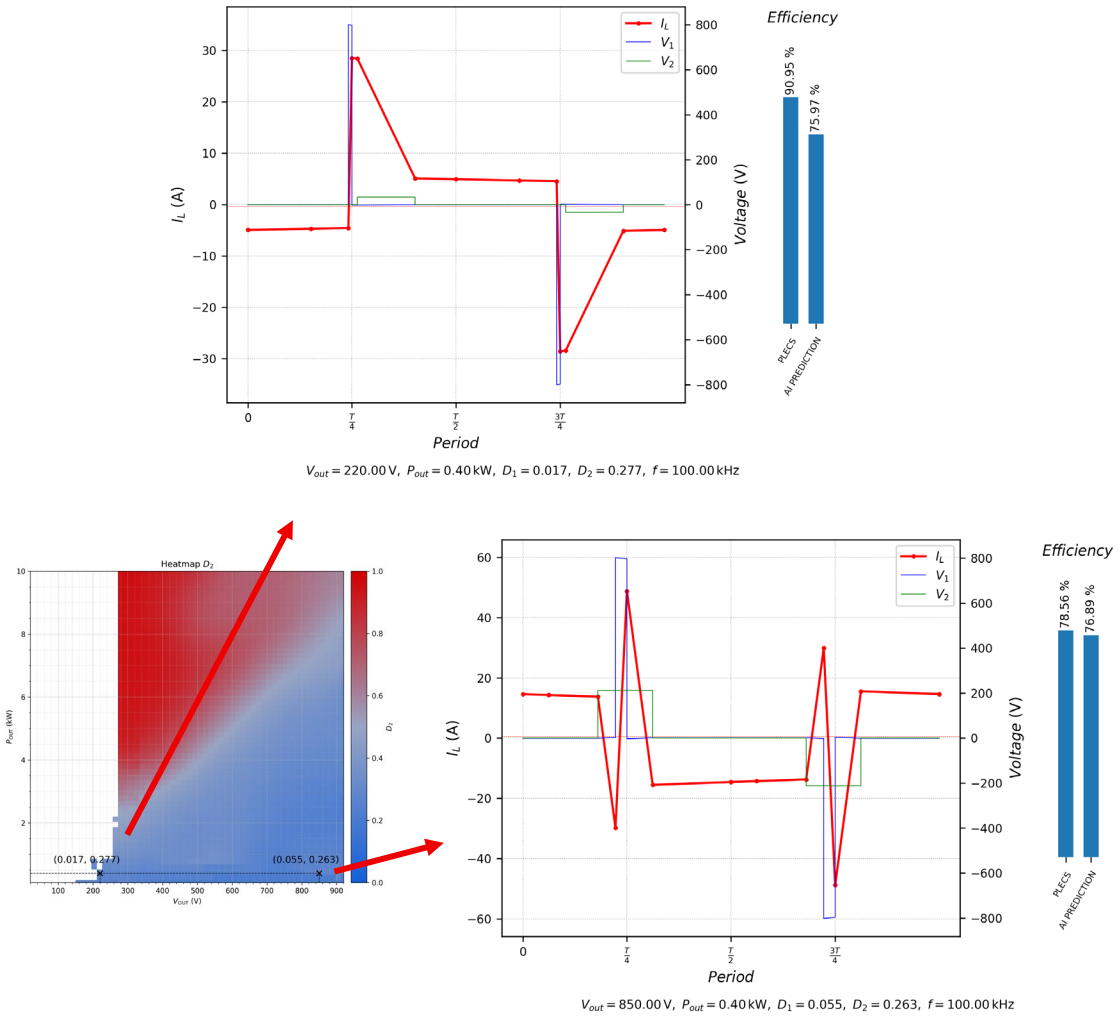


*Figure 4-9. Inductance current waveforms for low voltage conditions.*

Firstly, because of the small power demanded at the output, the duty cycles for both experiments are visually small, so the final current at the output is also small enough to reach the reference. Furthermore, both waveforms resemble to those of figure 4-7, remarking the idea that the DAB used the same operating principle during the whole plotted surface to achieve low power losses. For the second graph, the current waveform of the inductance has even several crosses from positive to negative to optimize the reduction on the current RMS. This point has been evaluated on the region where the current output is smaller, so it is worth appreciating that the whole area covers a small value for the secondary duty cycle. This statement also reinforces the reason for the decrease on $D_2$ while moving form high-power low-voltage point to opposite low-power and high-voltage region.

Lastly, on the next figure, the **efficiency results** for the control parameters obtained is attached. A Gaussian smoothing filtering was applied to the surface to enhance the representability, but also the original points conforming the surface are represented in grey. The idea is to visualize the efficiency obtained for the control parameters combinations to verify that high efficiency points are being obtained.

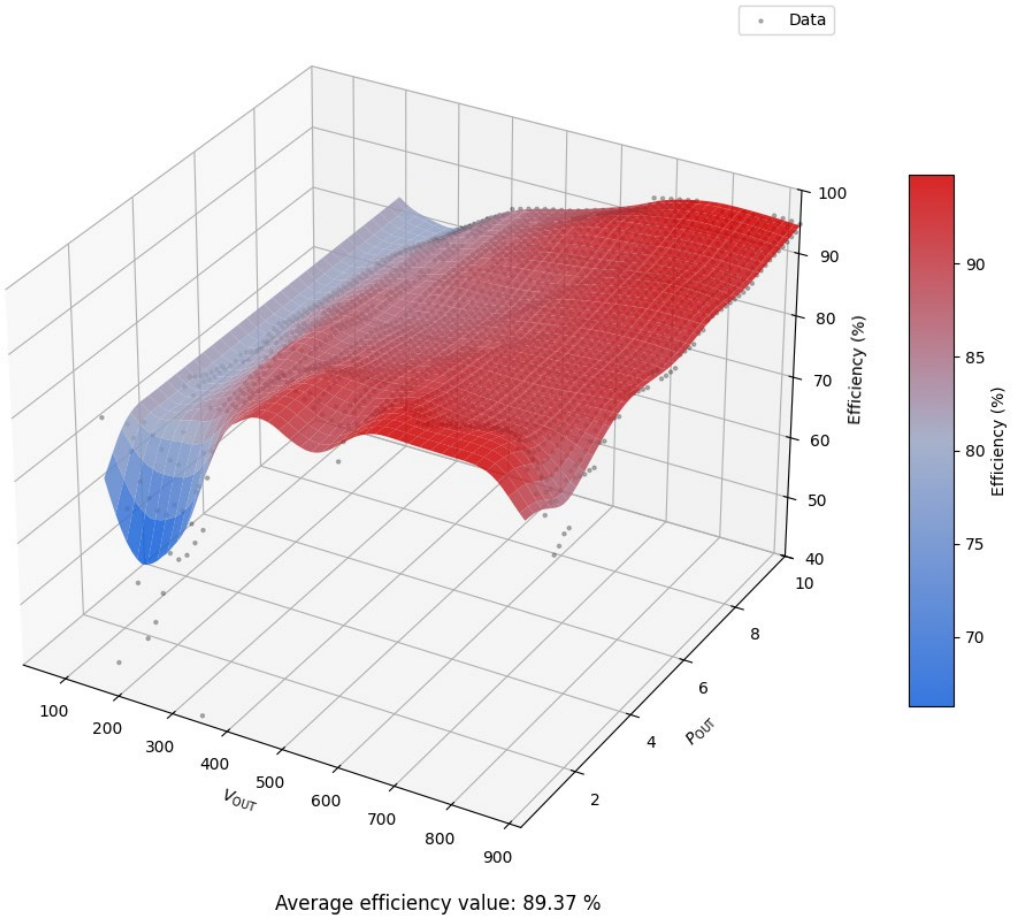


*Figure 4-10. Best efficiency reached for load conditions grid*

At the bottom of the previous figure, the average efficiency value is calculated delivering an **overall 89.37% rate**. This value is reasonable for the DAB operating range analysed and can validate the reliability of the results. Hence, it can also be visualized how the overall efficiency tends to decrease as we move into low voltage conditions where the border between the studied regions was found.

# 5 Conclusions and Future Work

In the previous chapters, the solutions found by the AI algorithm were discussed and explained from an electrical-theorical perspective. Whereas the main developments of the present work were made without taking direct care of many of those concepts (current stress minimizing, ZVS…), the solutions obtained have reach to high efficiency point which can be analysed and explained with the theory.

Accurately speaking, the AI algorithm was not explicitly trained to identify or minimize parameters nor regions that are normally encountered when approaching the optimization of a DAB from a traditional perspective—which can involve performing a piecewise or harmonic analyse heavy-biased in mathematics and electrical principles (see chapter 2.1 and 2.2).

This situation is in line with the totality of the research, as this was presented as a methodology to get to find and optimize high efficiency points by leveraging on computer calculation and cutting-edge technology tools. Indeed, the aim of the research was to stablish a workflow that enables to predict with AI the behaviour of the converter to strive for efficiency points and data that visually and thoroughly help understanding the working principles of the converter.

Overall, and after having done the theoretical approach to the solutions, it is clear that the model proposed has been able to accurately find efficient combinations of the control parameters for the DAB, providing a continuous solution for a wide range of operating conditions typical for a EV bidirectional charging purpose. Furthermore, the solutions found have a physical meaning and logical explanation from and engineering and power electronics standpoint and have obtained reasonable results, which validate the design of the solution.

As the tendency on both EV and AI still growing, the current presented work must be properly treated to keep it up to date with the new disruptive technologies, needs and trending. From a critical point of view, the work resulted is a great representation of the viability of applying AI calculation on a DAB controller, meanwhile, further research can be led by expanding the range and combination of variable and investigating other critical efficiency-related aspect of the converter such as the losses on the core which were not mainly represented on this project.

At any rate, this work aimed to test the suitability of trending AI tool and algorithms to the power electronics field with an EV charging optimization purpose, and it has always been a concern the crafting of a flexible solutions that could be implemented and complemented with future research. Consequently, the scope of the investigation can consider as validated and the methodology propose conclude on successful and practical results for a real-time control optimization purpose.

# 6 References

References


[1] D. Gielen, F. Boshell, D. Saygin, M. D. Bazilian, N. Wagner, and R. Gorini, "The role of renewable energy in the global energy transformation," *Energy Strategy Reviews*, vol. 24, pp. 38–50, 2019, doi: 10.1016/j.esr.2019.01.006.

[2] Dr. Hartwig Stammberger and Prof. Dr.-Ing. Holger Borcherding, "Project introduction: DC-INDUSTRIE2," 2020. [Online]. Available: https://dc-industrie.zvei.org/fileadmin/DC-Industrie/Praesentationen/DCI2_Project-presentation_en.pdf

[3] D. V. Pelegov and J. Pontes, "Main Drivers of Battery Industry Changes: Electric Vehicles—A Market Overview," *Batteries*, vol. 4, no. 4, p. 65, 2018, doi: 10.3390/batteries4040065.

[4] A. Janissek, "CharIN Position Paper," [Online]. Available: https://www.charin.global/media/pages/technology/knowledge-base/bb62e2344d-1686142056/20230605-power-classes-bidirectional-v1.07.pdf

[5] F. Adegbohun, A. von Jouanne, E. Agamloh, and A. Yokochi, "A Review of Bidirectional Charging Grid Support Applications and Battery Degradation Considerations," *Energies*, vol. 17, no. 6, p. 1320, 2024, doi: 10.3390/en17061320.

[6] T. B. R. Company, *Bidirectional Electric Vehicle Charger Market Analysis: Key Insights on Growth Rates, Trends, and Major Opportunities - Latest Global Market Insights.* [Online]. Available: https://blog.tbrc.info/2025/02/bidirectional-electric-vehicle-charger-market-share (accessed: Apr. 18 2025).

[7] D. Team, *Bidirectional Charging: Future Trends & Use Cases.* [Online]. Available: https://driivz.com/blog/emerging-trends-and-future-use-cases-for-bidirectional-charging (accessed: Apr. 18 2025).

[8] Plexim, *Controller HIL Testing for GaN and SiC Converters.* Accessed: Mar. 18 2025.

[9] S. J. Ríos, D. J. Pagano, and K. E. Lucas, "Bidirectional Power Sharing for DC Microgrid Enabled by Dual Active Bridge DC-DC Converter," *Energies*, vol. 14, no. 2, p. 404, 2021, doi: 10.3390/en14020404.

[10] G. Park, H. Kim, B.-G. Cho, and S. Cui, "ZVS-Enhanced and RMS-Current-Minimized Optimal Modulation Scheme of Dual-Active Bridge Converter With Comprehensive ZVS Analysis," *IEEE Trans. Power Electron.*, pp. 1–15, 2025, doi: 10.1109/TPEL.2025.3541847.

[11] G. G. Oggier, G. O. GarcÍa, and A. R. Oliva, "Switching Control Strategy to Minimize Dual Active Bridge Converter Losses," *IEEE Trans. Power Electron.*, vol. 24, no. 7, pp. 1826–1838, 2009, doi: 10.1109/TPEL.2009.2020902.

[12] S. Shao *et al.,* "Modeling and Advanced Control of Dual-Active-Bridge DC–DC Converters: A Review," *IEEE Trans. Power Electron.*, vol. 37, no. 2, pp. 1524–1547, 2022.

[13] A. S. Babokany, M. Jabbari, G. Shahgholian, and M. Mahdavian, "A review of bidirectional dual active bridge converter," in *2012 9th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, Phetchaburi, Thailand, 2012, pp. 1–4.

[14] A. Rodriguez Alonso, J. Sebastian, D. G. Lamar, M. M. Hernando, and A. Vazquez, "An overall study of a Dual Active Bridge for bidirectional DC/DC conversion," in *2010 IEEE Energy Conversion Congress and Exposition*, Atlanta, GA, 2010, pp. 1129–1135.

[15] *Dual Active Bridge.* [Online]. Available: https://www.mathworks.com/discovery/dual-active-bridge.html (accessed: Apr. 21 2025).

[16] G. Fernandez, *Dual Active Bridge converter modulation techniques - imperix.* [Online]. Available: https://imperix.com/doc/implementation/dual-active-bridge-control (accessed: Apr. 21 2025).

[17] Ali Ahmadi Chamyousefali, "Analysis of Dual Active Bridge Converter," Politecnico Milano 1863. Accessed: Jun. 5 2025.

[18] Texas Instruments, Incorporated [TIDUES0, and E ], "Bidirectional, Dual Active Bridge Reference Design for Level 3 Electric Vehicle Charging Stations (Rev. E),"

[19] J. He, Y. Chen, J. Lin, J. Chen, L. Cheng, and Y. Wang, "Review of Modeling, Modulation, and Control Strategies for the Dual-Active-Bridge DC/DC Converter," *Energies*, vol. 16, no. 18, p. 6646, 2023, doi: 10.3390/en16186646.

[20] M. F. Fiaz, S. Calligaro, M. Iurich, and R. Petrella, "Analytical Modeling and Control of Dual Active Bridge Converter Considering All Phase-Shifts," *Energies*, vol. 15, no. 8, p. 2720, 2022, doi: 10.3390/en15082720.

[21] B. Khanzadeh, T. Thiringer, and M. Kharezy, "Multilevel Dual Active Bridge Leakage

Inductance Selection for Various DC-Link Voltage Spans," *Energies*, vol. 16, no. 2, p. 859, 2023, doi: 10.3390/en16020859.

[22] D. Wang, F. Peng, J. Ye, Y. Yang, and A. Emadi, "Dead-time effect analysis of a three-phase dual-active bridge DC/DC converter," *IET Power Electronics*, vol. 11, no. 6, pp. 984–994, 2018, doi: 10.1049/iet-pel.2017.0701.

[23] A. Fernandez-Hernandez, F. Gonzalez-Hernando, A. Garcia-Bediaga, I. Villar, and G. Abad, "Design Space Analysis of the Dual-Active-Bridge Converter for More Electric Aircraft," *Energies*, vol. 15, no. 24, p. 9503, 2022, doi: 10.3390/en15249503.

[24] F. D. Esteban, F. M. Serra, C. H. de Angelo, and O. D. Montoya, "Nonlinear control for a DC–DC converter with dual active bridges in a DC microgrid," *International Journal of Electrical Power & Energy Systems*, vol. 146, p. 108731, 2023, doi: 10.1016/j.ijepes.2022.108731.

[25] F. Lin, "The dual active bridge converter design with artificial intelligence," Nanyang Technological University. [Online]. Available: https://dr.ntu.edu.sg/handle/10356/163850

[26] R. de Doncker, D. M. Divan, and M. H. Kheraluwala, "A three-phase soft-switched high-power-density DC/DC converter for high-power applications," *IEEE Trans. on Ind. Applicat.*, vol. 27, no. 1, pp. 63–73, 1991, doi: 10.1109/28.67533.

[27] ETHW, *Rik W. De Doncker.* [Online]. Available: https://ethw.org/Rik_W._De_Doncker (accessed: May 5 2025).

[28] S. Ghamari, D. Habibi, M. Ghahramani, and A. Aziz, "Design of a Robust Adaptive Cascade Fractional-Order Nonlinear-Based Controller Enhanced Using Grey Wolf Optimization for High-Power DC/DC Dual Active Bridge Converter in Electric Vehicles," *IET Power Electronics*, vol. 18, no. 1, 2025, doi: 10.1049/pel2.70056.

[29] F. Corti, V. Bertolini, A. Reatti, E. Cardelli, and M. Giallongo, "Comparison of Control Strategies for Dual Active Bridge Converter," in *2022 IEEE 21st Mediterranean Electrotechnical Conference (MELECON)*, Palermo, Italy, 2022, pp. 902–907.

[30] S. M. Akbar, A. Hasan, A. J. Watson, and P. Wheeler, "Model Predictive Control With Triple Phase Shift Modulation for a Dual Active Bridge DC-DC Converter," *IEEE Access*, vol. 9, pp. 98603–98614, 2021, doi: 10.1109/access.2021.3095553.

[31] C. Calderon *et al.,* "General Analysis of Switching Modes in a Dual Active Bridge with Triple Phase Shift Modulation," *Energies*, vol. 11, no. 9, p. 2419, 2018, doi: 10.3390/en11092419.

[32] Changjiang Sun, Xin Zhang, and Xu Cai, *ECCE 2019: IEEE Energy Conversion Congress & Expo : Baltimore, MD, Sept. 29-Oct. 3*. Piscataway, NJ: IEEE, 2019. [Online]. Available: https://ieeexplore.ieee.org/servlet/opac?punumber=8897530

[33] Robert Lenke, Florian Mura, and Rik W. De Doncker, "Comparison of non-resonant and super-resonant dual-active ZVS-operated high-power DC-DC converters,"

[34] A. Mirtchev and E. Tatakis, "Design Methodology Based on Dual Control of a Resonant Dual Active Bridge Converter for Electric Vehicle Battery Charging," *IEEE Trans. Veh. Technol.*, vol. 71, no. 3, pp. 2691–2705, 2022, doi: 10.1109/TVT.2022.3142681.

[35] J. Lee, S. Choi, and J.-I. Ha, "Design and Analysis of Resonant Network for Dual Active Bridge Converter," *IEEE J. Emerg. Sel. Topics Power Electron.*, vol. 12, no. 4, pp. 3784–3799, 2024, doi: 10.1109/JESTPE.2024.3406953.

[36] W. Malan, D. M. Vilathgamuwa, and G. Walker, "Modeling and Control of a Resonant Dual Active Bridge with a Tuned CLLC Network," *IEEE Trans. Power Electron.*, p. 1, 2015, doi: 10.1109/TPEL.2015.2507787.

[37] K. Wang, I. Laird, and J. Wang, "Multi-Objective Efficiency-Oriented Optimization for DAB Converters Minimizing Current Stress and Backflow Power with Soft-Switching Assurance," in *2025 IEEE Applied Power Electronics Conference and Exposition (APEC)*, Atlanta, GA, USA, 2025, pp. 2088–2095.

[38] Texas Instruments, Incorporated [SLYT664, and *], "MOSFET power losses and how they affect power-supply efficiency," [Online]. Available: https://www.ti.com/lit/an/slyt664/slyt664.pdf?ts=1753686979638

[39] L. Shu, W. Chen, D. Ma, and G. Ning, "Analysis of Strategy for Achieving Zero-Current Switching in Full-Bridge Converters," *IEEE Trans. Ind. Electron.*, vol. 65, no. 7, pp. 5509–5517, 2018, doi: 10.1109/TIE.2017.2779443.

[40] R. H. Ashique *et al.,* "A Comparative Analysis of Soft Switching Techniques in Reducing the Energy Loss and Improving the Soft Switching Range in Power Converters," *Electronics*, vol. 11, no. 7, p. 1062, 2022, doi: 10.3390/electronics11071062.

[41] R. Huang and S. K. Mazumder, "A Soft-Switching Scheme for an Isolated DC/DC Converter With Pulsating DC Output for a Three-Phase High-Frequency-Link PWM Converter," *IEEE Trans. Power Electron.*, vol. 24, no. 10, pp. 2276–2288, 2009, doi: 10.1109/TPEL.2009.2022755.

[42] J. Riedel, D. G. Holmes, B. P. McGrath, and C. Teixeira, "ZVS Soft Switching Boundaries for Dual Active Bridge DC–DC Converters Using Frequency Domain

Analysis," *IEEE Trans. Power Electron.*, vol. 32, no. 4, pp. 3166–3179, 2017, doi: 10.1109/TPEL.2016.2573856.

[43] MIT Sloan, *Machine learning, explained | MIT Sloan.* [Online]. Available: https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained (accessed: Jul. 28 2025).

[44] L. Wuttke, *Machine Learning vs. Deep Learning: Wo ist der Unterschied?* [Online]. Available: https://datasolut.com/machine-learning-vs-deep-learning/ (accessed: Jul. 28 2025).

[45] M. S. Jim Holdsworth, *What Is Deep Learning? | IBM.* [Online]. Available: https://www.ibm.com/think/topics/deep-learning (accessed: Jul. 28 2025).

[46] S. Singh, R. Kumar, S. Payra, and S. K. Singh, "Artificial Intelligence and Machine Learning in Pharmacological Research: Bridging the Gap Between Data and Drug Discovery," *Cureus*, vol. 15, no. 8, e44359, 2023, doi: 10.7759/cureus.44359.

[47] Nilesh Barla, *A Gentle Introduction to Deep Learning—the ELI5 Way.* [Online]. Available: https://www.v7labs.com/blog/deep-learning-guide (accessed: Jul. 24 2025).

[48] Pragati Baheti, *Activation Functions in Neural Networks [12 Types & Use Cases].* [Online]. Available: https://www.v7labs.com/blog/neural-networks-activation-functions (accessed: Jul. 24 2025).

[49] Adnan Mazraeh, "Tabular Data Augmentation: From Basics to Advanced Techniques," 2025. [Online]. Available: https://medium.com/@adnan.mazraeh1993/data-augmentation-from-basics-to-advanced-concepts-179963c25a9a

[50] PLECS, *Dual Active Bridge Converter.* [Online]. Available: https://www.plexim.com/support/application-examples/1506 (accessed: Mar. 5 2025).

[51] X. Li, X. Zhang, F. Lin, C. Sun, and K. Mao, "Artificial-Intelligence-Based Triple Phase Shift Modulation for Dual Active Bridge Converter With Minimized Current Stress," *IEEE J. Emerg. Sel. Topics Power Electron.*, vol. 11, no. 4, pp. 4430–4441, 2023, doi: 10.1109/jestpe.2021.3105522.

[52] F. Lin, X. Zhang, X. Li, C. Sun, W. Cai, and Z. Zhang, "Automatic Triple Phase-Shift Modulation for DAB Converter With Minimized Power Loss," *IEEE Trans. on Ind. Applicat.*, vol. 58, no. 3, pp. 3840–3851, 2022.

[53] Plexim, *XML-RPC Interface of PLECS.* [Online]. Available: https://www.plexim.com/sites/default/files/tutorials_categorized/plecs/xmlrpc_controller_design.pdf (accessed: Oct. 4 2025).

[54] Texas Instruments, Incorporated [TIDUES0, and E ], "Bidirectional, Dual Active Bridge Reference Design for Level 3 Electric Vehicle Charging Stations (Rev. E),"

[55] A. Mathew, U. R. Prasad, M. G. M, N. Naik, C. Vyjayanthi, and B. Subudhi, "Performance Analysis of a Dual Active Bridge Converter in EV Charging Applications," in *2022 International Conference for Advancement in Technology (ICONAT)*, Goa, India, 2022, pp. 1–6.

[56] STMICROELECTRONICS, "25 kW, dual active bridge bidirectional power converter for EV charging and battery energy storage systems - User manual,"

[57] T. Jin, Y. Wang, W. Ming, I. Lüdtke, A. Lewis, and S. Wang, "AI-driven design approach for dual active bridge converters with increased explainability," *IET Conf. Proc.*, vol. 2024, no. 3, pp. 329–336, 2024, doi: 10.1049/icp.2024.2175.

[58] Imbalanced Learn, *SMOTEENN.* [Online]. Available: https://imbalanced-learn.org/stable/references/generated/imblearn.combine.SMOTEENN.html (accessed: Jun. 29 2025).

[59] L. Cui, H. Li, K. Chen, L. Shou, and G. Chen, "Tabular Data Augmentation for Machine Learning: Progress and Prospects of Embracing Generative AI," Jul. 2024. [Online]. Available: http://arxiv.org/pdf/2407.21523v1

[60] A. Shmuel, O. Glickman, and T. Lazebnik, "Data Augmentation for Deep Learning Regression Tasks by Machine Learning Models," Jan. 2025. [Online]. Available: http://arxiv.org/pdf/2501.03654v1

[61] F. Lin *et al.,* "AI-Based Design With Data Trimming for Hybrid Phase Shift Modulation for Minimum-Current-Stress Dual Active Bridge Converter," *IEEE J. Emerg. Sel. Topics Power Electron.*, vol. 12, no. 2, pp. 2268–2280, 2024, doi: 10.1109/jestpe.2022.3232534.

[62] XGBoost, *XGBoost Documentation.* [Online]. Available: https://xgboost.readthedocs.io/en/stable/ (accessed: May 28 2025).

[63] Microsoft Corporation, *LightGBM's documentation.* [Online]. Available: https://lightgbm.readthedocs.io/en/stable/ (accessed: May 31 2025).

[64] Mohammed Shammeer, *Scikit-learn pipelines: A comprehensive guide.* [Online]. Available: https://shammeer.medium.com/mastering-scikit-learn-pipelines-4a54ef423c75 (accessed: May 18 2025).

[65] scikit-learn developers, *Common pitfalls and recommended practices.* [Online]. Available: https://scikit-learn.org/stable/common_pitfalls.html (accessed: Jul. 17 2025).

[66] Data604, *Hyperparameter Tuning Part4: Informed Search — Coarse to fine.* [Online].

Available: https://medium.com/@Mustafa77/hyperparameter-tuning-part4-informed-search-coarse-to-fine-716398a9f225 (accessed: Jul. 17 2025).

[67] Ferdinand Che, "Practical Machine Learning Hyperparameter Optimization By A Coarse-To-Fine Search," 2024. [Online]. Available: https://medium.com/@ferdinand.che/practical-machine-learning-hyperparameter-optimization-by-a-coarse-to-fine-search-0fa68c4ce606

[68] Nikolas Adaloglou, *Intuitive Explanation of Skip Connections in Deep Learning.* [Online]. Available: https://theaisummer.com/skip-connections/ (accessed: Jul. 8 2025).

[69] Sivaram, *What are Skip Connections in Deep Learning?* [Online]. Available: https://www.analyticsvidhya.com/blog/2021/08/all-you-need-to-know-about-skip-connections/ (accessed: Jul. 16 2025).

[70] Abhishek Jain, *SILU and GELU activation function in transformers | by Abhishek Jain | Medium.* [Online]. Available: https://medium.com/%40abhishekjainindore24/silu-and-gelu-activation-function-in-tra-a808c73c18da (accessed: Jul. 26 2025).

[71] Diganta Misra, *The Swish Activation Function | DigitalOcean.* [Online]. Available: https://www.digitalocean.com/community/tutorials/swish-activation-function (accessed: Jul. 26 2025).

[72] SuperAnnotate, *Activation functions in neural networks [Updated 2024] | SuperAnnotate.* [Online]. Available: https://www.superannotate.com/blog/activation-functions-in-neural-networks (accessed: Jul. 26 2025).

[73] Alejandro Ito Aramendia, *DenseNet : A Complete Guide. Extending the ResNet to improve… | by Alejandro Ito Aramendia | Medium.* [Online]. Available: https://medium.com/%40alejandro.itoaramendia/densenet-a-complete-guide-84fedef21dcc (accessed: Jul. 26 2025).

[74] G. Huang, Z. Liu, G. Pleiss, L. van der Maaten, and K. Q. Weinberger, "Convolutional Networks with Dense Connectivity," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 12, pp. 8704–8716, 2022, doi: 10.1109/TPAMI.2019.2918284.

[75] GeeksforGeeks, *DBSCAN Clustering in ML - Density based clustering - GeeksforGeeks.* [Online]. Available: https://www.geeksforgeeks.org/machine-learning/dbscan-clustering-in-ml-density-based-clustering (accessed: Jul. 27 2025).

[76] H. Idrees, *K-Means vs. DBSCAN: Clustering Algorithms for Grouping Data | by Hassaan Idrees | Medium.* [Online]. Available: https://medium.com/%40hassaanidrees7/k-means-vs-dbscan-clustering-algorithms-for-grouping-data-a4969034cfcc (accessed: Jul. 27 2025).

[77] Rajesh Kumar, *A Guide to the DBSCAN Clustering Algorithm | DataCamp.* [Online]. Available: https://www.datacamp.com/tutorial/dbscan-clustering-algorithm (accessed: Jul. 27 2025).

[78] K. Shen *et al.,* "ZVS Control strategy of dual active bridge DC/DC converter with triple-phase-shift modulation considering RMS current optimization," *The Journal of Engineering*, vol. 2019, no. 18, pp. 4708–4712, 2019, doi: 10.1049/joe.2018.9341.

[79] F. Xiong, J. Wu, L. Hao, and Z. Liu, "Backflow Power Optimization Control for Dual Active Bridge DC-DC Converters," *Energies*, vol. 10, no. 9, p. 1403, 2017, doi: 10.3390/en10091403.

[80] F. Xu, F. Zhao, Q. Shi, and X. Wen, "Studies of ZVS soft switching of dual-active-bridge isolated bidirectional DC-DC converters," in Busan, South Korea, 2018, p. 20013.

[81] B. Liu, P. Davari, and F. Blaabjerg, "Enhanced Zero-Voltage-Switching Conditions of Dual Active Bridge Converter Under Light Load Situations," in *2020 IEEE Applied Power Electronics Conference and Exposition (APEC)*, New Orleans, LA, USA, 2020, pp. 1374–1381.

# 7 List of Appendices

**APENDIX A: PLECS MODELS**

**A.1 DAB *PLECS* Demo Model**
dual_active_bridge_converter.plecs

**A.2 DAB *PLECS* Complete Model**
dual_active_bridge_converter_TPS.pdf
dual_active_bridge_converter_TPS (TPS controller).pdf
dual_active_bridge_converter_TPS.plecs

***APENDIX B: DATASET***

**B.1 TRAINING**
Data_SWEEP_00.csv
Data_SWEEP_01.csv
Data_SWEEP_02.csv
Data_SWEEP_D1_50_D2_50.csv
Data_SWEEP_SMALL_D1_00.csv

**B.2 VALIDATION**
Data_RANDOM_VAL_00.csv

***APENDIX C: CODE***

**C.1 PLECS SIMULATION**
RunSimulations_PLECS.py
GenerateValues.py

**C.2 AI**
Model.ipynb

**C.3 VALIDATION**
Validation.ipynb

### C.4 OPTIMIZATION

Optimization.ipynb

## *APENDIX D: AI TRAINED MODELS*

### D.1 CLASSIFICATION

*xgb*

- xgb.pkl
- xgb_ORIG.pkl

*lgbm*

- lgbm.pkl
- lgbm_ORIG.pkl

*pytorch*

- dense_nn.pt
- dense_nn_ORIG.pt
- leakyswish_nn.pt
- leakyswish_nn_ORIG.pt
- skip_connection_nn.pt
- skip_connection_nn_ORIG.pt
- scaler_X.pkl

feature_names.json

### D.2 REGRESSION

*xgb*

- xgb.pkl
- xgb_ORIG.pkl

*lgbm*

- lgbm.pkl
- lgbm_ORIG.pkl

*pytorch*

- dense_nn.pt
- dense_nn_ORIG.pt
- leakyswish_nn.pt
- leakyswish_nn_ORIG.pt
- skip_connection_nn.pt
- skip_connection_nn_ORIG.pt
- scaler_X.pkl
- scaler_y.pkl

feature_names.json

## *APENDIX E: OPTIMIZATION RESULTS*

*Results_Optimization.pkl*

# IX. Erklärung

Ich versichere,

dass ich die Arbeit selbstständig verfasst habe, dass ich keine anderen als die angegebenen Quellen benutzt habe und ich alle wörtlich oder sinngemäß aus anderen Werken übernommenen Aussagen als solche gekennzeichnet habe,

dass die eingereichte Arbeit weder vollständig noch in wesentlichen Teilen Gegenstand eines anderen Prüfungsverfahrens gewesen ist, dass ich die Arbeit noch nicht veröffentlicht habe und dass das elektronische Exemplar mit dem gedruckten Exemplar übereinstimmt.

Ich erkläre mich damit einverstanden, dass die bibliographischen Daten (Titel, Autor, Erscheinungsjahr, etc.) meiner Arbeit auf der Institutswebseite des Instituts für elektrische Energiewandlung (IEW) veröffentlich werden dürfen.

Der Universität Stuttgart übertrage ich ein kostenloses, zeitlich und örtlich unbeschränktes, einfaches Nutzungsrecht meiner Arbeit und den erzeugten Arbeitsergebnissen für Zwecke der Forschung und der Lehre.

Stuttgart, 7.08.2025 Diego San Jose Vega