Università degli Studi di Salerno

Dipartimento di Ingegneria dell'Informazione ed Elettrica e
Matematica Applicata (DIEM)

Dottorato di Ricerca in Informatica ed Ingegneria dell'Informazione
XXIX Ciclo

TESI DI DOTTORATO

Methods and Algorithms for Power Devices Losses Behavioral Modeling

CANDIDATO: **MARIO MIGLIARO**

TUTOR: **PROF. NICOLA FEMIA**

COORDINATORE: **PROF. ALFREDO DE SANTIS**

Anno Accademico 2016 – 2017

Contents

Introduction **1**

1 Power Converter Losses **5**

1.1 Power Conversion Introduction 5

1.2 Power Semiconductor Devices 6

1.2.1 Power Transistors 7

Power MOSFETs 7

Insulated Gate Bipolar Transistors (IGBT) . 9

1.2.2 Transistor Power Losses Calculations 12

Classical Analytical Calculations 12

Other Analytical Models 16

Behavioral Dynamic Models 17

1.3 Power Inductors 19

1.3.1 Main Power Losses 19

1.3.2 Inductor Power Loss Modeling 21

2 Identification Algorithm **25**

2.1 System identification 25

2.2 Static Models and Machine Learning 27

2.2.1 Artificial Neural Network 28

2.2.2 Fuzzy Logic Models 31

2.2.3 Genetic Programming 33

2.3 Parameter Identification 35

2.3.1 Linear in Parameters Models 36

2.3.2 Non-Linear in Parameters Models 36

2.4 Parametric GP 39

2.4.1 Parametric GP tree structure 41
2.4.2 Coefficients optimization 42
Linear GP trees 42
LLS 43
Group Coefficients 43
Nonlinear Least Square 44
2.4.3 Objectives 45
Global Error 46
Complexity Factor 46
Solution Methods 49
2.4.4 Algorithm settings 50
2.4.5 Solution Choice 51

3 Semiconductor Devices Models 57
3.1 Case Study: Half Bridge for Induction Cooking . . 57
3.1.1 Induction Cooking Systems 57
3.1.2 Experimental Tests 62
3.1.3 Results 67
Output models 67
3.1.4 Discussion 68
3.2 Case Study: SMPS transistors 75
3.2.1 Device Characterization Tests 76
3.2.2 Results 79
3.2.3 Discussion 81

4 Inductor Models 91
4.1 Case Study: Partially Saturated Inductor 91
4.1.1 Inductor Power Loss Measurements 93
4.1.2 Training and Validation Set 96
4.1.3 Results 98
4.1.4 Discussion 104

Conclusions and future developments 109

A Multi objective Optimization 111
A.1 Non-Dominated Sorting of a Population 113
A.2 Elitist Non-Dominated Sorting Genetic Algorithm . 114

Introduction

Power electronics is since decades in the focus of very important technology innovations, as the characteristics and the performances of power supplies can severely condition and limit the performances of the system to be fed.

In almost all the applications there is the demand to increase as much as possible the ratio between the maximum power the power supplies can deliver and their volume, defined as the power-density while, at the same time, the cost must be as reduced as possible.

For this reason, electronic system designers have the task of finding, in a reasonable time, ever better performing solutions, choosing the best semiconductor devices and magnetic components. For this purpose, models of power devices are required, but often their inherent non linearity makes them difficult to be modeled, making the job of feasibility assessment very hard.

The power losses in semiconductor devices are classically calculated using analytical models of the components, with current and voltages waveforms modeled as piecewise linear curves during switching , and the device DC behavior modeled as a simple resistor for Field Effect Transistors (FETs) or with a series of a resistance and a voltage generator for Bipolar Devices. The nonlinear dynamic effects that afflict the behavior of the devices during switching, like the voltage dependence of the AC equivalent resistance, the reverse recovery in the diode and the tail currents in the IGBTs, makes these models often very inaccurate.

Other approaches for semiconductor devices are based on simulations, in fact, often SPICE models are provided by the devices manufacturers, but the models obtained are often not accurate because the parameters are calculated in particular operative conditions different from the real ones. Otherwise, if not given, their construction requires the knowledge of geometrical and physical parameters that are not known by the designer. Moreover, the adoption of simulations can be really time consuming if a lot of operative conditions have to be analyzed. Finally, the set up of the best numerical tolerances, to avoid numerical issues that could abort the simulation, could cause a great loss of time, and often a bad construction of the SPICE models makes numerical issues not avoidable.

In magnetic components the method adopted to predict AC power losses in the core is the Steinmetz Law, that is a behavioral model that links operative parameters, like frequency and ac magnetic flux, with the power losses in the magnetic core, having coefficients, provided by the manufacturers data-sheets, depending on the particular material adopted and the specific operative range considered. A problem of this model is that it is constructed using a sinusoidal input, while in switching circuits the current is often piece wise linear, and surely not sinusoidal. Moreover, other effects, like for example the partial saturation of the magnetic core, are not included in the classical Steinmetz models. More complicated models exist, often requiring the knowledge of a lot of coefficients, or their calculation through physical and geometrical parameters not known by the users, making them difficult to be constructed and adopted by the designers.

Moreover, in both semiconductor and magnetic devices data-sheets, some important parameters and curves are often not reported, not accurate, or estimated with particular circuits working in specific operative conditions. This lack of informations makes inaccurate, if not impossible, the construction of the models. For this reason, it should be better if the model parameters could be easily identified from experimental data, so that the designer could identify them through simple electrical measurements. Having models, for both semiconductor devices and magnetic components, in some way similar to the Steinmetz Law, that is simple, accurate, with easily measured inputs and with a minimal physical knowledge of the device materials and geometries, could be very useful. The structure of a model should expresses the main correlations among the losses and physical quantities influencing them, whereas the coefficients express the sensitivity with respect to the physical quantities and their aggregations.

The choice of the model structure and the optimization of its coefficients are the main issues to solve in the identification of a behavioral loss model. The model structure should not be complex, as to provide human designer intelligible information about the system under study, as to allow the implementation of low-computing/low-consumption/low-cost digital controls. In this way, having a model able to give an accurate real time estimation of the power losses, the systems can work as close as possible to their maximum power rating, thus achieving the highest power density in run time operation . The problem of finding an accurate structure is often not trivial, in fact it could be done with a *trial and error* approach, that is constructing a model starting from observation or from physical knowledge of the phenomenon, fitting the data, and accept it if its accuracy is sufficiently good, otherwise trying another model. Obviously, this approach could be very time consuming, in particular if the process to be modeled is very complicated and if the number of possible input variables is high.

In this thesis a general method to identify behavioral models from experimental data, with a trade off between accuracy and complexity, is discussed. The approach adopted is based on Genetic Programming (GP), that is an evolutionary method able to return output models, in order to minimize a fitness function that is a metric of the quality of the solution. In particular, the goal of the algorithm has been to obtain models accurate, but at the same time simple and intelligible for the user. It must be considered that these two desired conditions are often conflicting, being complicated models usually accurate and simple models usually inaccurate, for this reason a Multi Objective (MO) approach has been adopted. Moreover, the goal is to have a model that is as general as possible for similar devices, in terms of physical and geometrical properties. For this reason, the GP has been modified in order to return parametric functions, having the same structure, but different coefficients for all the devices to characterize.

The novelty of this thesis does not consist in the GP algorithm itself, but in its application for the identification of power losses in power devices, considering that similar approaches do not exist in literature. By the way, the GP has been customized for this purpose. In fact, a parametric and multi-objective version has been implemented, and appropriate objectives have been defined and adopted.

The algorithm has been tested on both semiconductor devices and magnetic devices, obtaining accurate, general and intelligible models that could be easily adopted by power engineers.

The Thesis is organized as follows:

In Chapter 1 an overview on losses causes in power converters and the methods classically used to predict them are discussed, in particular for both Semiconductor Devices and Inductors.

In Chapter 2 identification methods and models that have been investigated are discussed and the implementation of the developed algorithm is discussed. Moreover, the choice of the main parameters and of the model is discussed.

In Chapter 3 case studies relative to Semiconductor Devices are discussed, in particular identification of power losses for Insulated gate Bipolar Transistor (IGBTs) for Induction Cooking system. Then the algorithm has been applied in simulation on GaN transistors for a Buck converter to prove the capability to obtain a general model.

In Chapter 4 the application of the method to power inductors is shown, in particular considering how a partial saturation in the magnetic core affects the total power losses.

Chapter 1

Power Converter Losses

1.1 Power Conversion Introduction

In modern power converters, a very big issue is to be able to minimize the power losses of the system, generally for two reasons:

- to increase the system efficiency, in order to minimize the power absorbed from the AC line or from batteries, so reducing the domestic users electricity consumptions and extending the runtime of off-line devices.

- to reduce the power losses on a single device in order to improve its reliability and time of life, avoiding thermal damages

Different causes of power losses are present in power converters, but the main ones are in the semiconductor devices (power transistors and diodes) and in the magnetic components (inductors, transformers, coils). Other minor power losses are in the auxiliary components (drivers, integrated circuits, micro-controllers,etc), the printed circuit parasitic resistances and the capacitors Equivalent Series Resistances (ESRs).

For this reason, a dramatic improvement in the global system efficiency can be obtained if, given a particular application, a valid choice of power semiconductor devices and inductors is done, and if the best operative conditions (like switching frequency, duty cycle, etc.) are chosen in order to minimize their power losses.

Furthermore, size is another important issue in modern power converters, that a general tendency in the years to get always smaller and more integrated. A choice of power transistors not well suited for the particular application could cause the need of over-dimensioned heat-sinks to avoid thermal damages; at the same time, considering that the magnetic components usually occupy most space in the Printed Circuit Boards (PCBs), in particular in Switch Mode Power Supplies (SMPS) applications, it is important to be able to predict their losses, in order to select inductors able to fulfill their power constraints but not too bulky.

In the following sections the main causes of Power Losses in semiconductor devices and in magnetic components are introduced and the method that are classically used by designer to predict their power losses are introduced and discussed.

1.2 Power Semiconductor Devices

Power Transistors are used in all the Power Converters to convert the voltage and currents. In particular, in almost all circuits, semiconductor devices switch to high frequencies to convert energy, so minimizing the size of the magnetic components. For this reason, they are often referred as *switches*.

An ideal switch should work as shown in Fig.1.1, behaving like a short circuit when on and like an open circuit when it is off, with the gray area indicating the ideal operating points of the switch. Moreover, the transitions between on to off state and *vice versa* is instantaneous.

Obviously, real switches do not behave like this and, when they are turned on and current flows through them, a small amount of voltage is present, causing conduction power losses. Furthermore, the non null amount of time necessary to make them commutate causes switching losses. This not instantaneous commutation, makes necessary the adoption of a dead time, where both

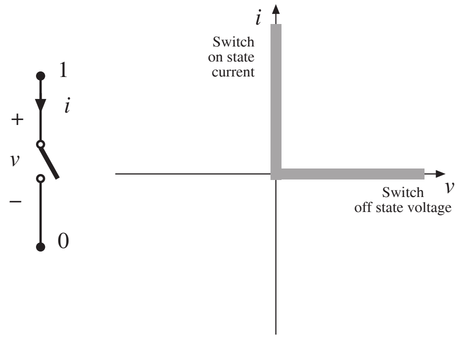


Figure 1.1: Ideal Switch Characteristic

the switches driver signal are off, avoiding the possibility of simultaneous conduction of both the transistors, *punch-through*, that could cause physical damages to the converter and the device. For this reason, in the small amount of time where no transistor can conduct, a diode conducts, increasing the power losses.

1.2.1 Power Transistors

As shown in Figure 1.2, the choice of the switch technology depends on the specific application power ranges and switching frequencies. Medium-Low power applications are considered in this thesis, in particular switching circuits in the range of the Watts, using Metal Oxide Semiconductor Field Effect Transistors (MOSFETs), and Induction Cooking in the kW ranges, using Insulated Gate Bipolar Transistors (IGBTs).

Power MOSFETs

MOSFETs are the most used semiconductor switches in low power applications, because they can be easily driven, and can work at higher frequencies than bipolar devices for their lower commutation times.

The Vertical Diffused Power MOSFET (VDMOS), whose structure is shown in Fig.1.3, is the most used in Power Electronics application, with the device substrate used as drain terminal.

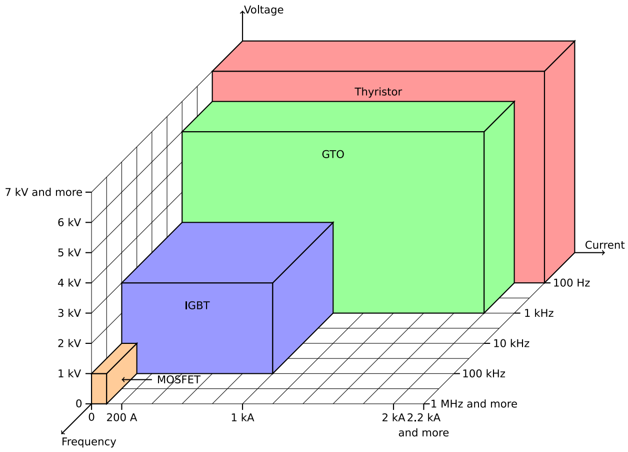


Figure 1.2: Transistors for frequency current and power

The characteristics of a MOSFET transistor are shown in Fig.1.4. The transfer function represents the dependence of the current on the drain-source voltage, having a typical linear behavior for low voltages (*triode mode*) and presenting a saturation (*active mode*) at higher voltages. The trans-characteristic represents the relationship between the drain current and the gate-source voltage, and it is zero before a threshold voltages, and then it presents a quadratic behavior.

In the last times Wide Band-Gap (WBG) Semiconductors MOSFETs are having a great development. WBG [3] allows the device to work at higher temperature, frequencies and voltage than classical Silicon devices. The most used materials are SiC and GaN, having theoretical limits of R_{on} at parity of Breakdown Voltage much lower than classical Si devices, as shown in Fig. 1.5. In particular, GaN has a higher critical electrical field and higher electron mobility than Silicon, allowing smaller sizes at parity of on resistance and breakdown voltage.

The lateral structure of GaN allows the possibility to work at higher frequency (over the MHz range) than Si due to its lateral structure and consequent lower parasitic capacitances, and its lower gate charge Q_g and recovery charge Q_{RR}.

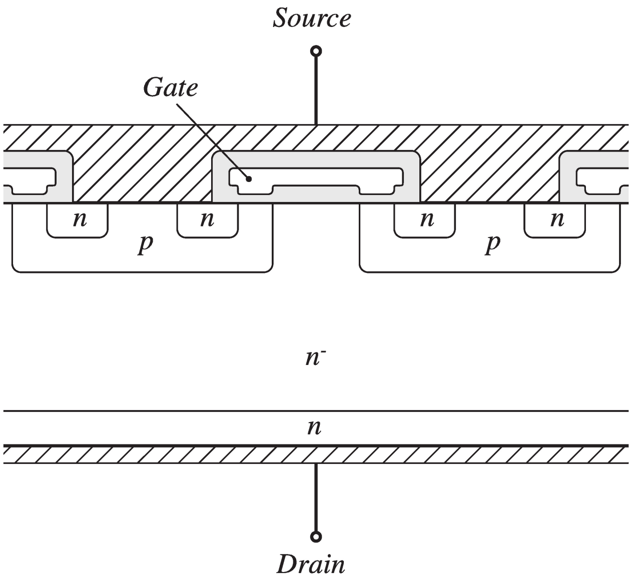


Figure 1.3: Power MOSFET Structure
[1]

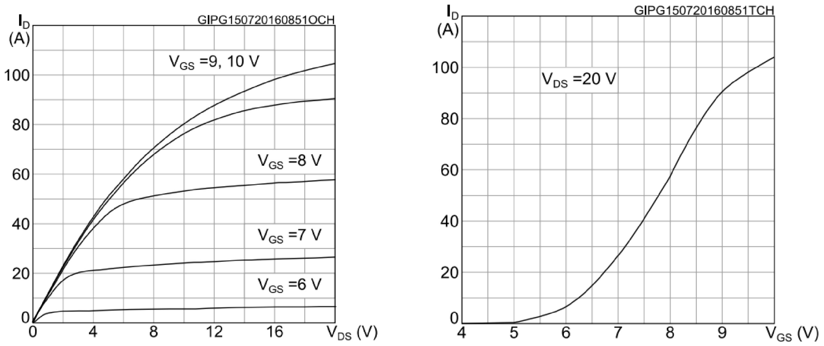


Figure 1.4: MOSFET characteristics. Left: output characteristic. Right:transfer characteristic [2]

Insulated Gate Bipolar Transistors (IGBT)

IGBTs are the most used devices in applications where the power ratings are too high for power MOSFETs. In fact, they combine the MOSFETs advantage of been easily driven with the capability

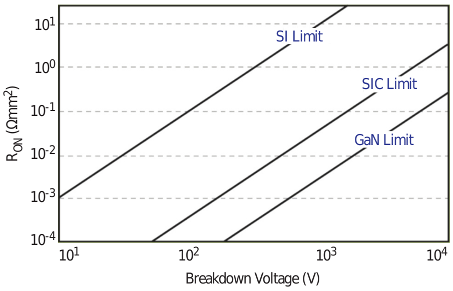


Figure 1.5: On resistance vs Breakdown voltage theoretical limits for different semiconductors [3]

to conduct high current with a low on voltage of the other Bipolar Devices (like BJTs, SCRs and GTOs).

The structure of a IGBTs and its equivalent circuital models are shown in Figures 1.6(a) and 1.6(b) respectively.

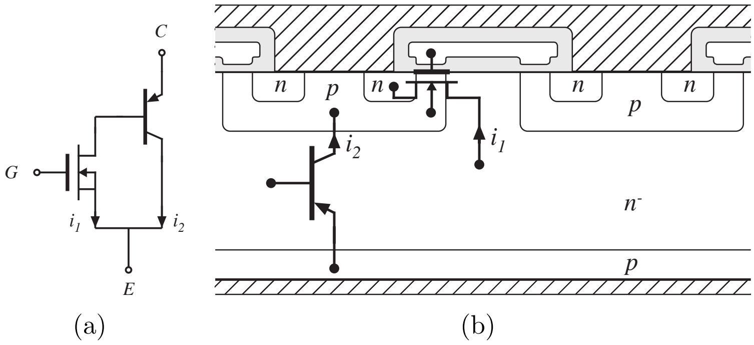


Figure 1.6: IGBT equivalent model (a) and physical structure (b) [1]

The input characteristic is similar to the MOSFET one, while its output characteristic, shown in Fig.1.6(a), can be divided in two zones: for low values of collector-emitter voltage V_{ce} the IGBT can be modeled as a P-i-N diode in series with a MOSFET and, for this reason, the current has an exponential behavior; at higher voltages, the characteristic tends to the MOSFET one shifted of a value equal to the voltage drop on the the parasitic PNP emitter base junction.

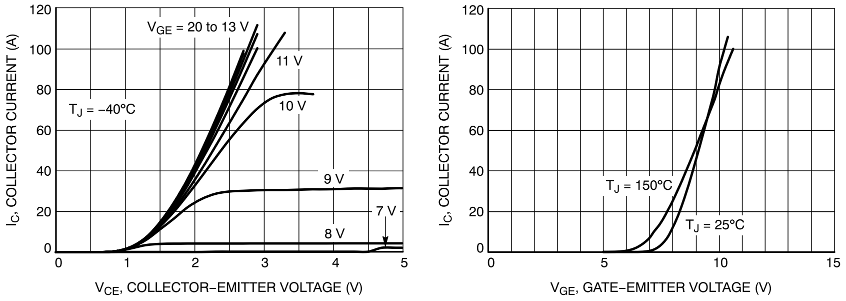


Figure 1.7: IGBT characteristics. Left: output characteristic. Right:transfer characteristic [4]

IGBTs are slower than MOSFET, requiring a switching time in the range of the hundreds of nanoseconds, and moreover, during turn off, there is the additional effect of the *Tail Currents*, shown in Fig.1.8 that can have a dramatic impact on the switching Power Losses.

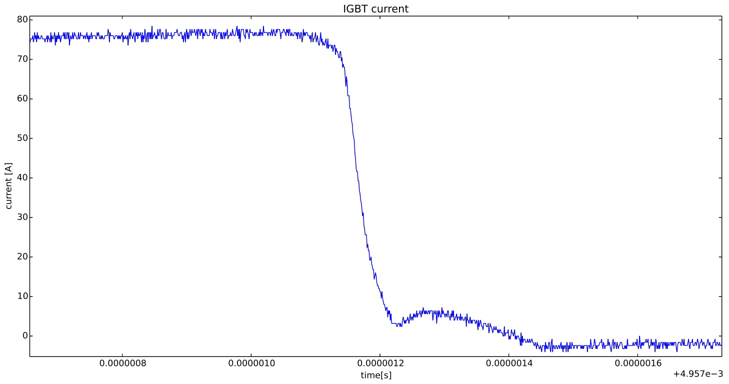


Figure 1.8: IGBT tail current at the turn off

Moreover, IGBTs are unidirectional devices, meaning that they cannot conduct current for negative voltages, and do not have an internal parasitic body diode. In order to allow also negative currents to flow in the device, an anti-parallel diode is usually inserted in the same package of the IGBTs, and for this reason, its losses must be included in the count of the conduction power losses of the IGBT.

1.2.2 Transistor Power Losses Calculations

Transistor Power Losses are usually calculated using one of these two different approaches:

analytical models require a small amount of calculation and always have a solution, but often present simplifications like piecewise waveforms or simplified differential equations that makes their results often inaccurate

behavioral dynamical models require simulations, consisting in the solution of nonlinear differential equations, that can be solved using numerical methods. A correct tuning of the simulation parameters is required to avoid convergence issues, often present if particularly articulated and complex models are adopted.

These two approaches are described in the following of this section.

Classical Analytical Calculations

Power transistor losses are classically divided in two main contributions [1] [5], each one generally decomposed in more terms. These two contributions are:

- Conduction Losses: caused by the nonzero voltage value of the switch during its conduction

- Switching Losses: during commutations the voltage and the current cross between them causing, for a reduced amount of time, instantaneous power losses higher than instantaneous conduction losses, as shown in Fig.1.9.

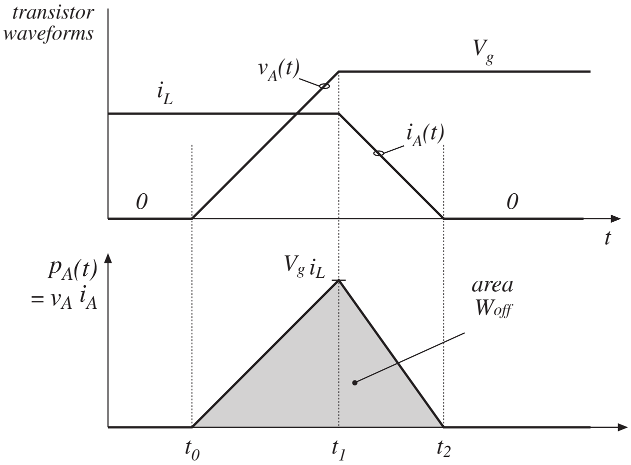


Figure 1.9: Energy dissipated during a real switch turn off

Conduction losses depends on the value of the voltage drop on the devices during their conduction, and are measured as

$$P_{cond} = \frac{1}{T_s} \int_{T_{cond}} v(t) \cdot i(t) dt \tag{1.1}$$

where $v(t)$ and $i(t)$ are the instantaneous value of voltage and current in the switch, T_{cond} is the amount of time where the switch is conducing in a period and T_s is the switching period. Typically, when Pulse-Width Modulation (PWM) is adopted, the conduction time T_{cond} is assumed to be $T_{cond} = DT_s$ for the main switch, with D indicating the value of the Duty Cycle. Actually, the real value of the conduction time T_{cond} is not well defined, in fact it depends on the dead times and the times required for the switching. Furthermore, the Duty Cycle value that depends on the system efficiency and, for this reason, it is not accurately known *a priori*.

In synchronous applications, an auxiliary switch conducts during the remaining amount of time of the switching period; otherwise, a diode conducts the current while the main switch is off. In both the cases, the conduction time is classically considered as $T_{cond} = (1 - D)T_s$.

The voltage $v(t)$ in (1.1) is typically referred as the on voltage

v_{on}, and depends on the device technology, material and structure. For Metal Oxide Silicon Field Effect Transistors (MOSFETs) it is typically calculated (in its linear region) as:

$$v_{on} = R_{on} \cdot i(t), \tag{1.2}$$

with R_{on}, the typical parameter affecting conduction losses, indicating the equivalent resistance present when the FETs is on, calculated as:

$$R_{on} = \left(\frac{\partial i}{\partial v}\right)^{-1}. \tag{1.3}$$

For diodes and Insulated Gate Bipolar Transistor (IGBTs) (in linear region), the current has generally an exponential dependence from the voltage and, for this reason, v_{on} can be approximated as a voltage source in series with a resistance:

$$v_{on} \approx V_{on} + R_{on} \cdot i(t) \tag{1.4}$$

Typically, diodes have a grater v_{on} than MOSFETs, and for this reason, their conduction time should be reduced to diminish the total power losses. Moreover, body diodes participate to the total power losses of their FETs, being on the same die.

Switching losses are present during the commutation of the switches, and depend on many terms. The crossing of the voltage and the current for a small amount of time during the commutations is the main contribution to the total switching losses.

Switching losses are typically calculated as:

$$P_{off(on)} = W_{off(on)} \cdot f_s \tag{1.5}$$

where $W_{off(on)}$ is the energy lost during commutation, calculated as the integral of the transistor current and voltage during the turn off (turn on). Equation (1.5) highlights that switching losses, fixed $W_{off(on)}$, increase linearly with the switching frequency f_s.

In hard switching condition the $W_{off(on)}$ energy could be seen as the area of a triangle having as base the time necessary to turn off (on) $t_{off(on)}$, and as peak the product of the voltage and the current at the turn off(on), indicated as $V_{off(on)}$ and $I_{off(on)}$ respectively, as shown in Fig.1.9.

Assuming a triangular approximation of $W_{off(on)}$, equation (1.5) can be rewritten as:

$$P_{off(on)} = W_{off(on)} \cdot f_s = \frac{1}{2}(V_{off(on)} I_{off(on)}) \cdot t_{off(on)} f_s \tag{1.6}$$

with:

$$t_{off(on)} = \frac{Q_{g,sw}}{I_{g,off(on)}} \tag{1.7}$$

$$I_{g,on} = \frac{V_{dr} - V_{sp}}{R_g} \tag{1.8}$$

$$I_{g,off} = \frac{V_{sp}}{R_g} \tag{1.9}$$

$$V_{sp} = V_{TH} + \frac{I_{off(on)}}{g_{fs}} \tag{1.10}$$

with the threshold voltage V_{TH} and transconductance g_{fs} shown in Fig.1.10.

Other contributions to power losses are:

$$P_{gate} = Q_g f_s V_{dr} \tag{1.11}$$

$$P_{rr} = Q_{rr} f_s V_{in} \tag{1.12}$$

$$P_{Coss} = \frac{1}{2} C_{oss} V_{off}^2 f_s \tag{1.13}$$

where (1.11) is the power given by the gate to charge the transistor with V_{dr} indicating the driver voltage, (1.12) are the reverse recovery losses due to the diode turn off and (1.13) represent the power losses due to the discharge of the parasitic output resistance C_{oss} during turn on.

The main parameter for the switching is represented by the Q_g, that represents the charge on the gate terminal of the MOSFET.

Using gate charge, Q_g, the designer can calculate the amount of current required from the drive circuit to switch the device on in a desired time.

Gate charge and on resistance are inter-related, in fact a lower gate charge means a higher on resistance, for this reason, a performance of a device can be quantified with its figure of merit [6], that is usually indicated as:

$$FOM = \frac{1}{R_{on}Q_g} \tag{1.14}$$

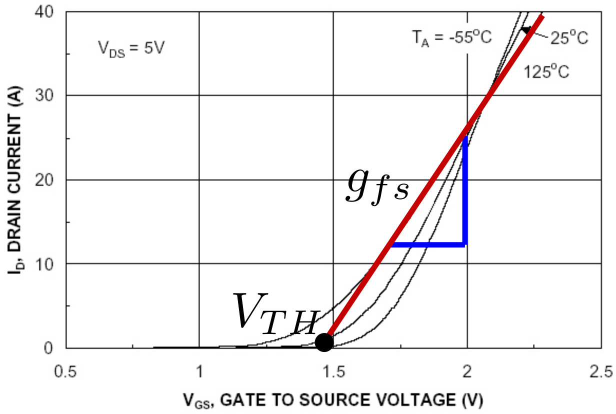


Figure 1.10: MOSFET transfer function curve and parameters

Other Analytical Models

The equations in the previous section are usually adopted to estimate the power losses of MOSFETs. This approach has also be adopted to identify the IGBT power losses: in [7] an analytical method has been applied to estimate the efficiency of an Induction Cooking Half Bridge converter, with the piecewise linear behavior of the IGBT. In this case, a first region represents the classical turn off of the current, and a second region is added to model the tail currents. In [8] several analytical models, indicated as Mathematical models by the author, are described. These models are based on semiconductor physics, and require the solution of physical equations. A main issue is that the knowledge of physical properties of the devices, not given by the manufacturers, is needed for the models construction.

Behavioral Dynamic Models

A typical approach to detect power losses is using dynamic SPICE models, that exist with different levels of detail and can be used in circuit simulators [9]. The knowledge of physical and geometrical parameters, not provided by manufacturers, is needed to construct these models.

Behavioral models are based on the fitting of the curves given by power device manufacturers. The semiconductor devices are usually modeled with the structure shown in Fig.1.11. The DC voltage controlled current source represents the static characteristic of the MOSFET, and the ac parasitic capacitances (Figure 1.12), depending on the voltages across the FETs, model the dynamic behavior of the device.

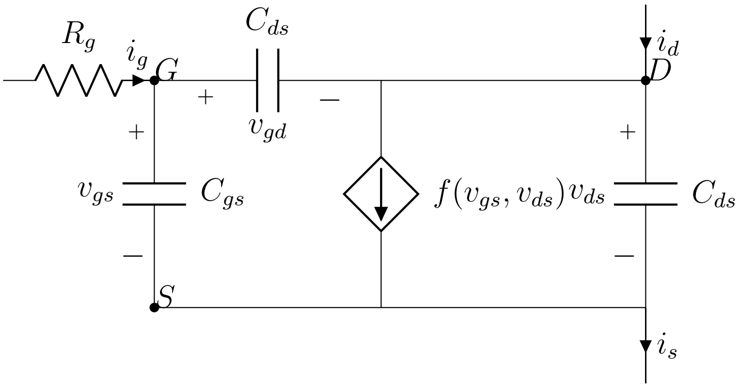


Figure 1.11: Equivalent MOSFET model

The equivalent capacitances are AC values dependent on the voltage drop on the device , as shown in Fig.1.12. They must be well calculated to accurately model the dynamic behavior of the device. In [10] a dynamical model based purely on physical parameters provided in devices data-sheets, has been applied to a

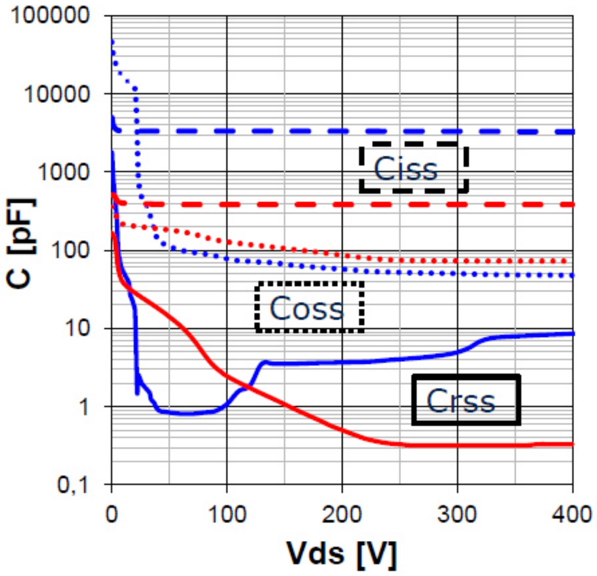


Figure 1.12: Ac capacitances vs Voltage for SJ MOSFETs(blue) and GaN MOSFETs(red)

Switching Cell, with the parasitic capacitance approximated with two exponentials.

The dynamical approach has been applied also to IGBTs, as described in [8] the dynamic IGBTs models can be classified in three different kinds.

Semi-mathematical models are composed by existing BJT and MOSFET models connected between them forming the IGBT structure. These models are similar to the one shown in Fig.1.6(a) but with more components. Other parts are inserted to take into account some specific effects like voltage dependent tail currents, conductivity modulation, nonlinear capacitances between terminals, conductivity modulation lag at turn-on, etc. These models are not very accurate, because the wide base BJT in the IGBT does not resemble any of the existing discrete power BJTs models[8].

Behavioral models do not consider the physical mechanism behind the dynamic of the IGBTs, but have the purpose to simulate their behaviors. They use model structures similar to the one adopted for the MOSFET, like the one in Fig.1.11, but adapting both the Dependent Current Source and the parasitic capacitance to the IGBTs characteristics. The main problems of this approach is that, contrarily to the MOSFETs, the DC transfer function $f(v_{ge}, v_{ce})$ is not known for the IGBTs. Moreover, the dynamic behavior of the IGBTs can not be well approximated by the only AC small signal capacitances, because charge accumulation effects, like tail currents[11], are present.

Finally, in Semi-numerical Models finite element methods are used to model the wide base, while analytical method are adopted to model the other devices parts. Because of the complexity and difficulty in modeling IGBT base, numerical methods are employed in some models to describe it accurately.

The effect of the PCB parasitics must be considered if these kind of models are adopted. In particular, switching power losses actually depend on the parasitic inductance and resistances, that affect the switching waveforms. Their values are not defined until the layout is routed and other components affecting the switching behavior have been fixed, for example the snubber capacitances.

1.3 Power Inductors

1.3.1 Main Power Losses

Inductor Power Losses are typically divided in two main contributions that are the winding losses and the magnetic core losses.

The winding losses are due to the current circulating in the copper windings and are given by (1.15)

$$P_w = R_w \cdot I_{L,RMS}^2 \tag{1.15}$$

where $I_{L,RMS}$ is the rms value of the inductor current $i_L(t)$ and R_w is the winding resistance, which depends on winding geometry, operating temperature and spectrum of the current $i_L(t)$.

In particular while in DC this contribution is small due to the high electrical conductivity of the copper, at higher frequency the effect of eddy currents, shown in Fig.1.13 makes the current circulate only on the external area of the windings, making increase the resistance R_w and consequently the Power Losses with the frequency.

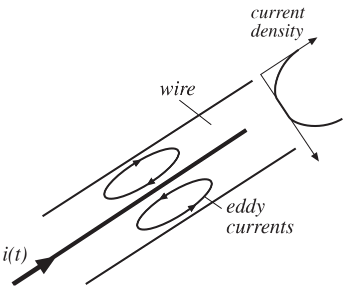


Figure 1.13: Effect of eddy currents in high frequency current [1]

The other losses are in the magnetic core, and are due to the hysteresis loop area in the B-H curves, as shown in Fig.1.14.

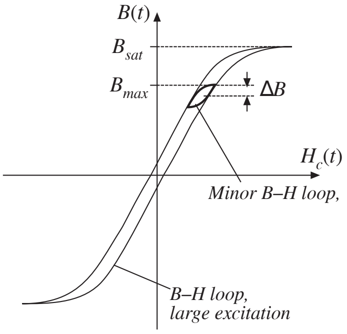


Figure 1.14: minor hysteresis loop [1]

The core losses are typically calculated with the Steinmetz Equation (SE) given in (1.16) that are an empirical behavioral core loss model largely used in SMPS design [12]:

$$P_c = C_m f_s^{\alpha} B_{ac}^{\beta} (A_e l_e) \tag{1.16}$$

with f_s indicating the frequency, B_{ac} the ac magnetic flux density magnitude and A_e and l_e the equivalent cross section and magnetic path length of the magnetic core. The coefficients C_m, α and β principally depend on the magnetic material and are usually provided on the manufacturers data-sheets.

The SE can be re-formulated as a function of the peak-to-peak current ripple Δi_{Lpp} instead of B_{ac}, since Δi_{Lpp} is easier to measure with respect to B_{ac}, as given in (1.17):

$$P_c = K_1 f_s^X (K_2 \Delta i_{Lpp})^Y \tag{1.17}$$

The coefficients K_1, K_2, X and Y depend on material, switching frequency and current ripple range. K_1 depends also on the core volume. Some manufacturers adopt this formulation and provide relevant core losses coefficients [13].

1.3.2 Inductor Power Loss Modeling

An issue in power inductor modeling for SMPSs is how to construct the inductor power loss formula, which has to incorporate winding losses and magnetic core losses .

A review of methods that can be used to predict high-frequency losses in round-wire windings can be found in [14].

In [15], the squared-field-derivative method for calculating eddy-current and proximity-effect losses in round-wire or litz-wire transformer and inductor windings has been derived. In [16], a closed-form expression closely approximating the ac resistance factor has been developed, for planar or foil-wound inductors with a quasi-distributed gap comprising multiple small gaps. However, such method applies only to single winding layers with copper thickness greater than one skin depth. All these methods can be adopted only if the windings cross-section and layers distribution are known. Unfortunately, such data are not available in the data-sheets of commercial parts. The value R_w provided by manufacturers mostly refers to DC test conditions. Using this value in high-frequency conditions can result in unpredictable accuracy of calculated winding loss.

Regarding magnetic core losses, some studies proposed in literature are based on the separation of core losses into hysteresis, eddy-currents and excess loss contributions [17, 18, 19]. However, a common drawback of the proposed models lies in the fact that the relative model parameters are not easy to obtain, as sophisticated experimental measurements are needed to evaluate separately the different loss terms. Both models (1.16) and (1.17) are given for sinusoidal excitations. Several studies have been developed on how to extend the validity of the SE to non-sinusoidal conditions. In [20, 21, 22, 23, 24] enhanced versions of the SE have been proposed, such as the Modified Steinmetz Equation (MSE), Generalized Steinmetz Equation (GSE), improved Generalized Steinmetz Equation (i-GSE), Natural Steinmetz Extension (NSE) and Improved Steinmetz Equation (ISE), valid for both sinusoidal and non-sinusoidal operating conditions and requiring no more parameters than the basic SE. In [25], the i-GSE has been further improved by including a residual core loss term accounting for the relaxation processes in the magnetic material, resulting in an *improved-improved* Generalized Steinmetz Equation (i^2-GSE). Such formulation, however, requires extra model parameters with respect to the SE, so that additional experimental measurements are needed to extract such parameters. On the other hand, the aforementioned formulations neglect the dependence of core losses on dc bias, thus yielding inaccurate core loss estimation in partially saturated condition.

In [26] Kolar *et al.* have presented a study about the influence of pre-magnetization on magnetic material losses and, in particular, on Steinmetz parameters. However, the proposed characterization has been limited only to some magnetic materials, and no analytical formulations have been given which would generalize the SE parameters dependence on dc bias for different materials. In [27], an iron loss map has been created by measuring the dynamic minor loop area traced on the $B - H$ plane of magnetic material in different conditions of applied dc bias, by fixing either ΔB or ΔH values. It has been experimentally verified that, with increasing bias, core losses may either increase if ΔB value is maintained constant, or decrease if ΔH value is fixed. Unfortunately, $B-H$ fields measurements are not allowed on commercial inductors for SMPSs applications.

In [28], Kosai *et al.* have proposed a simple correction to SE for partial saturation operating condition:

$$P_c = C_m f_s^\alpha B_{ac}^\beta (A_e l_e) \cdot \exp[a(\mu_0/\mu - 1)] \tag{1.18}$$

where a is an additional model parameter and the ratio μ_0/μ represents the saturation level of the core material, μ_0 being the magnetic permeability value at zero bias and μ being the effective magnetic permeability in partial saturation condition. However, neither μ_0 nor μ are easily measurable quantities for commercial magnetic parts.

In [29], an FEA approach has been proposed to calculate the magnetic core losses and their distribution for a planar inductor in the presence of dc current. A polynomial curve-fitting model has been proposed for the calculation of core losses:

$$P_c = C_m f_s^\alpha B_{ac}^\beta (A_e l_e) \cdot \sum_{i=0}^{7} a_i H_{DC}^i \tag{1.19}$$

However, only sinusoidal excitations have been analyzed. In [30], Sokalski *et al.* have suggested that the core loss function obeys the scaling law. On the basis of such assumption, a complicated expression has been proposed, including the dc bias H_{DC} influence on core losses:

$$\begin{aligned} P_c = \Delta B^\beta \Bigg\{ & \sum_{i=1}^{4} \Gamma_i \left(\frac{f_s}{\Delta B^\alpha}\right)^{i(1-x)} + \\ & + \sum_{i=1}^{2} \left[\Gamma_{i+5} \left(\frac{f_s}{\Delta B^\alpha}\right)^{(i+y)(1-x)} \tanh(H_{DC} c_{i+1} - r_{i+1}) \right] \Bigg\} \end{aligned} \tag{1.20}$$

where $\{\Gamma_i, c_i, r_i, \alpha, \beta, x, y\}$ represent model parameters.

Compared to (1.18) and (1.19), these models are more accurate but have the problem of have a high number of parameters that are not provided by the magnetic component manufacturers in their catalogs.

In conclusions there is a great number of power loss models for inductors that can be or very simple but with unreliable results, or very accurate but with a lot of parameters that are not easily accessible to the designer and so resulting inapplicable. So the objective of the study in this thesis for power inductors is to find models that are at the same time simple and accurate FCPIs, with parameters and inputs variables the that can be easily identified and measured by the user, and that easily include also the effects of partial saturation of the magnetic core on the total power losses.

Chapter 2

Identification Algorithm

Inferring a model from observation of a physical phenomenon is a kind of problem that is in the class of System Identification. In this chapter, the procedure to construct a model from observation is introduced and, in particular, the classical methods used to identify the parameters of the models and the different kind of models that can be adopted are discussed. Then, the particular problem faced in the Thesis, that combines both parameter and model identification and based on Genetic Programming is described.

2.1 System identification

In many engineering application it is necessary to predict the behavior of a particular physical process, and for this reason, a mathematical expression describing the relationship among the system variables is necessary. The process of identification of a model is shown in Fig.2.1.

In this figure a set of k inputs $\mathbf{x} = x_1, ..., x_k$ is given as input to both the process that must be modeled identify and the mathematical model, that is represented by the equation:

$$f(\mathbf{x}, \boldsymbol{\alpha}) \tag{2.1}$$

where in (2.1) f is the structure of the model considered, and

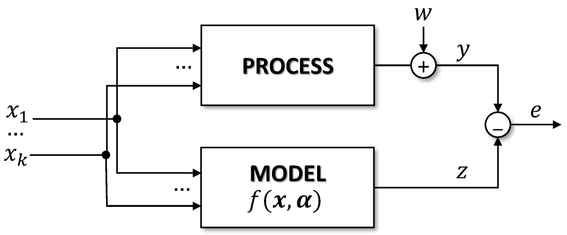


Figure 2.1: Process and Model

$\boldsymbol{\alpha} = \alpha_1, .., \alpha_{n_\alpha}$ is the vector with the parameters of the model. The output of the process is summed to a noise term w, that represents the measurement error. The error e is the difference between the value reconstructed by the model z and the noisy output of the process, and is a quantity that must be minimized to have an accurate model.

The process of identification consists in finding a model structure f and its best parameters $\boldsymbol{\alpha}$ to accurate describe the process behavior.

As described in [31] and [32] the identification process is a loop where different steps must be taken into account with the following order:

1. Choice of the inputs variables
2. Choice of the experiments
3. Choice of the model structure
4. Parameters identification
5. Model Validation

During the first step, starting from physical observation and theoretical knowledge about the phenomenon, the main variables that are supposed to affect the output (in the case of this Thesis the Power Losses) are selected. Then, in the second step, the experiments to identify the model are designed and done. In particular, in a *Supervised Learning* approach, a Training Set is constructed, and the input and output variables are measured. Obviously, the Training Set is constructed in order to identify a model that is valid for certain working condition of the device to identify. Then, a certain model is supposed to represent accurately the process to be modeled in the range of the Training Set.

There can be different models considering how much of the physical process informations are used during their construction. A model can be *White Box* when it is all constructed starting from theoretical knowledge of the phenomenon, *Black Box* when neither the model nor the coefficients are known *a priori*. In this case, system identification is adopted to identify both the model structure both the coefficient through a trial and error approach. A model is *Grey Box* when both theoretical knowledge and mathematical techniques are used. Finally, once a model structure has been chosen and the coefficients have been calculated in order to fit the experimental data the model is validated. The obtained model is given as input to a Validation Set, that must be different from the Training Set, and the error between the output reconstructed by the model and the measured output is calculated. If the behavior on the Validation Set is sufficiently accurate the model is accepted, otherwise the user could change the model structure, or the experiments design, or could change the set of variables used as input, until a good model is reached.

It can be noted that in the first steps (in particular the first two) *a priori* knowledge of the physical process to identify and the condition in which the Device Under Test (DUT) works is requested.

2.2 Static Models and Machine Learning

In this work static models must be identified, in fact in (2.1) the output of the system depends only on the values that the input variables take at that time, and it is not dependent on their past values. White and Gray boxes models, like the ones described in Chapter 1 for the Inductors could also be adopted but, being based on simplified physical considerations, they are certainly less accurate.

In this thesis Black Box approaches are considered, allowing a minimal use of *a priori* knowledge, in order to have a method well suited for different application. For this reason, Soft Computing methods, regarding techniques able to learn from experimental data, have being considered. Machine learning is the subfield of computer science that " gives computers the ability to learn without being explicitly programmed " (Arthur Samuel, 1959) and in the specific the ability of reconstruct a nonlinear function from observed acquisition is called Statistical Learning. The main methods that can be adopted in this kind of problems are Artificial Neural Networks (ANNs) and Fuzzy Logic Models (FLMs), Support Vectors Regression (SVR) or Genetic Methods. These methods are described in this section, focusing on their advantages and their problems.

2.2.1 Artificial Neural Network

Artificial Neural Network are a method that is based on the human brain behavior.

The basic of ANN is a Single Layer Perceptron, that is constituted by a single neuron, called Perceptron, as an example with three input variables x_1, x_2, x_3 and on output y, is shown in Figure 2.2. First, the inputs are multiplied to some weights w_1, w_2, w_3, known as synaptic weights, and summed to a bias b (usually represented as an input with value 1 and a weight of value b). The linear combination of the inputs is then passed through an activate function f, so the input of the activation function u if there are n input variables is $u = \sum_{i=0}^{n} w_i x_i$, with $x_0 = 1$ and $w_0 = b$.

The activate function usually adopted is the *logistic function*:

$$f(u) = \frac{1}{1 + \exp(-k \cdot u)} \tag{2.2}$$

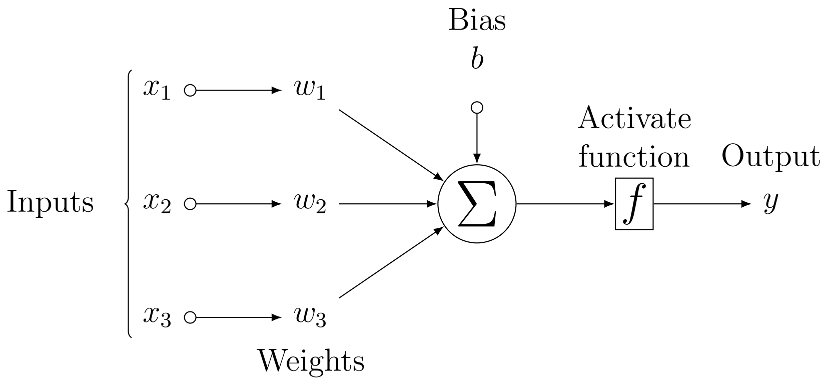


Figure 2.2: Single Neuron (Perceptron)

being $w_0 = b$ and $x_0 = 1$ in eq.(2.2), and k a parameter indicating the smoothness of the function (for high k values the logistic functions approaches a step function).

Obviously, the single layer can represent only a small class of functions. The problem of regression can be solved using a Multi Layer Perceptron (MLP), shown in Fig.2.3, where more inputs of neurons are adopted. This is a *Universal Approximator*[33], meaning that it is able to approximate any smooth function with higher accuracy as a greater number of hidden layer neurons is adopted, and so it can be used in many problem of function approximation.

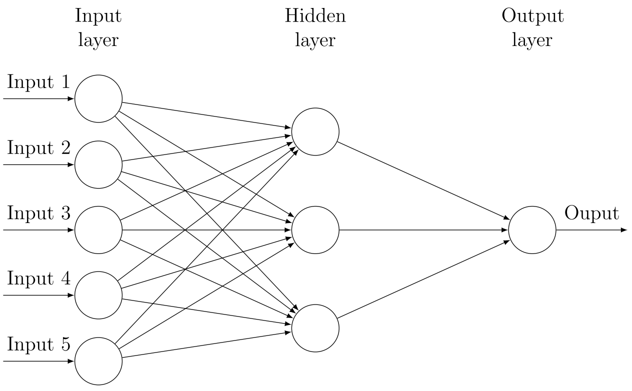


Figure 2.3: Multi Layer Perceptron

Radial Basis Functions (RBFs) are an alternative to MLP, being an universal approximator [34] like MLP. RBFs have an activation like a bell, typically a Gaussian function $f(u) = \exp 1/2u^2$, where in this case u represents a distance of the input vector $\mathbf{x} = [x_1, ..., x_n]^T$ respect to a center vector $\mathbf{c} = [c_1, ..., c_n]^T$ with respect to the norm matrix $\mathbf{\Sigma}$ representing width and rotation of the basis function, so that u is:

$$u = \sqrt{(\mathbf{c} - \mathbf{x})^T \mathbf{\Sigma} (\mathbf{c} - \mathbf{x})} \tag{2.3}$$

The main problems of ANNs for the desired applications are described:

Lack of intellegiblity: a relationship between internal variables cannot be uncovered, and the solution does not increase informations about the process. Moreover the input variables need to be normalized if too different, otherwise variables with a too big absolute value could dominate the others.

Structure Choise : the number of neuron in a level and the number of levels are a parameter that influences the behavior of the ANN. In particular, an excessive number of neurons could cause the net to over-fit the data, losing the generalization capability. On the other hand, a small number could be not sufficient to accurately model the data.

Training : Need long training and learning time, with local minimal and multiple solutions, in particular the coefficients to be calculated are the output layer weights that are linear parameters and the hidden layer weights for the MLP and the centers and standard deviations for the RBF that are non linear parameters.

Resorce consum: a big network could cause a big memory consumption and CPU usage in a micro-controller, in fact the number of coefficients with M hidden layer neurons is $M + 1 + M(n+1)$ for MLP and $2Mn + M + 1$ (with $\mathbf{\Sigma}$ diagonal) for RBF.

2.2.2 Fuzzy Logic Models

Fuzzy Logic, introduced by Zadeh [35] is an extension of Boolean Logic, where a variable can also take inter-medial values between 0 and 1, and so based on human mode of reasoning and communicate, that is usually vague.

Fuzzy Logic models phenomenons using some rules, that are *if-then* relationship between inputs and output variables. For example, for transistor devices in Chapter 1 an example of relationship could be : *if* frequency is high and current is high *then* power losses are high. Both the value of the inputs and the outputs variables and their relationship are given in a linguistic way, like a human being would do. These linguistic terms are defined using some Membership Functions, that define the degree of Membership of the inputs variable to a fuzzy set, so that for example in Fig. 2.4 the variable x is *low* with a factor 0.65, *medium* with a factor 0.3, and *high* with a factor 0, noting that the sum of the membership functions values is not required to be one in each point.

This conversion from numerical value to linguistic value is called *fuzzyfication.* After all the input variables have been *fuzzyfied* the next step is to combine them with fuzzy logic operators, that are the same of classic logic operators (AND,OR) when the variable takes a value 0 or 1 otherwise different metric exist like,

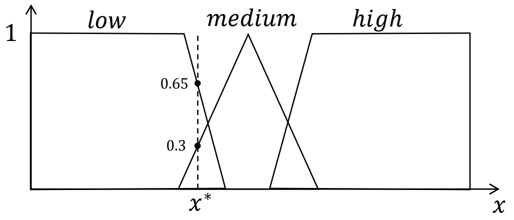


Figure 2.4: Fuzzyfication of the variable x

for example, the minimum between two values for the AND and the maximum for the OR. This gives a degree of fulfillment of a rule, that indicates how much a particular rule is true.

At the end all the rules are used to activate the fuzzy outputs, and finally *defuzzification* techniques can be used to have a numerical output from the system.

In contrast to ANN where no inside view of the effects can be seen, Fuzzy Logic models are classically constructed like a human being with experience could say. Actually FLMs and ANNs are very similar and in fact also them are universal approximators [36]. The main difference is that a FLM is not constructed by experimental data, but by the experience of an expert of the problem. This has the advantage of having very intelligible models where the relationship between input and output variables are explicated and clear, but the disadvantage that a human solution of the problem must exist, and an expert must be able to structure it. This makes the learning very slow, requiring a trial and error construction of the rules. Moreover increasing the number of variables, the complexity of the problem increases, and the possibility that a human being could structure it from his knowledge decreases.

The problems of Fuzzy Models are partially solved by Neuro Fuzzy (NF) Networks where the FLM can be constructed using also experimental data. In particular FLM are drawn in a Neural Structure, and consequently learning methods applied to ANN can be applied also to FLM. The main problem of this approach is that it presents a trade-off between intelligibility and accuracy of the model.

2.2.3 Genetic Programming

Genetic Programming (GP) is an Evolutionary Algorithm introduced by Koza in [37], able to identify model and functions from observed data. Each model is generally represented by a tree structure as shown in Figure 2.5, composed by *terminal nodes* and *non-terminal nodes*: non-terminal nodes are the basic functions that can be used to construct the model, like logarithm, exponential, power, etc.; terminal nodes are composed by the possible inputs to the model functions or by constant value (coefficients).

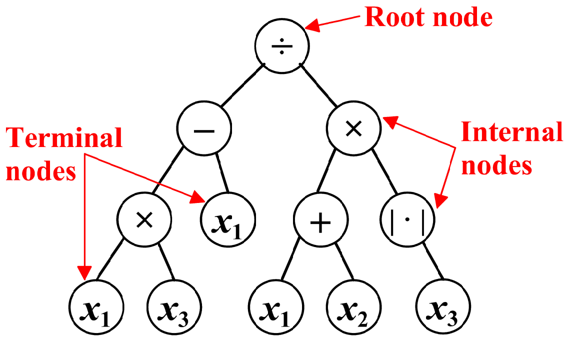


Figure 2.5: Tree representation of a typical GP program

The population, composed by GP models, evolves during generations, using GP operators, that are crossover and mutation, and at the end of the evolution, reached when the maximum number of generation is reached, the best models are returned as output.

The Genetic operators *Crossover* and *Mutation* are applied to individuals that are probabilistically selected based on fitness, so better individuals are more likely to have child models than inferior individuals.

Tournament selection is the method adopted for selecting individuals to which genetic operators are applied: a fixed number of individuals is randomly chosen, and the best of them, based on the fitness, is chosen as parent. When doing crossover, two parents are needed , so two selection tournaments are made.

In *crossover*, as shown in Figure 2.6, given two parents, a crossover point is randomly selected in each parent tree, and then the two subtrees rooted at the crossover point are switched between the two parents and two child GP models are obtained.

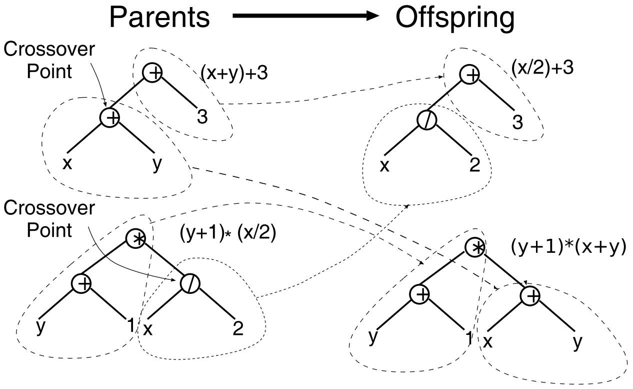


Figure 2.6: GP crossover[38]

There are two kind of mutations, *subtree mutation* and *point mutation*: in *subtree mutation* a mutation point is randomly selected and the subtree rooted is substituted with a randomly generated subtree, as illustrated in figure 2.7.

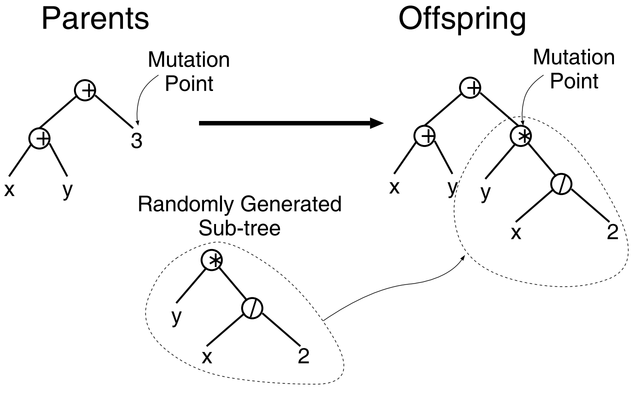


Figure 2.7: GP mutation[38]

In *point mutation* a random node is selected and the function stored there is replaced with a different random function of the same arity taken from the primitive set.

The choice of which of the operators described above should be used to create an offspring is probabilistic. Operators in GP are normally mutually exclusive (unlike other evolutionary algorithms where offspring are sometimes obtained via a composition of operators). The probabilities of their application are called operator rates. Typically, crossover is applied with the highest probability, the crossover rate often being 90% or higher. On the contrary, the mutation rate is much smaller, typically being in the region of 10%.

2.3 Parameter Identification

Once a model structure has been decided, the best coefficients of the model must be identified, and it is done using optimization techniques. As shown in Fig.2.1, given a set of input there is a difference between the model and the process, that is the error e. The error depends principally on two causes: the first is a not good choice of the model, for example a lack of input information or a too simple model, that is a *model error*; the other cause of error is due to the *measurement noise* w, that takes into account the fact that the observed data have a finite accuracy depending on the measurement systems. If the model is supposed to be sufficiently well modeled the second term dominates.

The coefficients are identified in order to minimize a certain function of the error, usually referred as *Cost Function*. This function depends on what principle of minimization is adopted. If the maximum likelihood principle is adopted and the noise is supposed to be *Independent Gaussian Noise with constant variance* it can be demonstrated that the optimal cost function is the Root Means Square Error (RMSE). The assumption of normal noise is due to the fact that the measurement is composed by different noises sources, and for the Central Limit theorem as the number of variables increases the sum of random variables tend to a Gaussian noise.

Several Optimization methods can be used to solve the problem of minimization of this error, classified as Least Square Methods. In particular the methods adopted depends on the kind of parameters, that can be linear or nonlinear.

2.3.1 Linear in Parameters Models

Many models are linear in parameters, and can be generally expressed as:

$$f(\mathbf{x}, \boldsymbol{\alpha}) = \alpha_1 g_1(\mathbf{x}) + ... + \alpha_{n_\alpha} g_{n_\alpha}(\mathbf{x}) \tag{2.4}$$

where with g_i is indicated a function that depends only on the inputs variables, that generally is an input variable itself, or a combination of them, or a constant.

This kind of problem has the important proprieties to have only a minimum point (so only one solution) that can be easily found analytically.

Given a set of n couple of inputs $\mathbf{x}$ and their relatives outputs y, $y = \mathbf{G}\boldsymbol{\alpha}$ where $\mathbf{G}$ is a matrix of size $n \times n_\alpha$, with $n > n_\alpha$ where each element is $G_{ij} = g_j(x_i))$.

The solution of the Linear Least Square (LLS) method is:

$$\boldsymbol{\alpha} = (\mathbf{G}^T \cdot \mathbf{G})^{-1}\mathbf{G}^T\mathbf{y} \tag{2.5}$$

The matrix $(\mathbf{G}^T \cdot \mathbf{G})^{-1}\mathbf{G}^T$ in(2.5) is the *pseudo-inverse* of the matrix G. The solution with the pseudo inverse is generally avoided for two causes: it can give numerical issues and it is not efficient. Other techniques are usually used like the Cholesky or QR factorization , where through factorization the system ($\mathbf{G}\boldsymbol{\alpha} = \mathbf{y}$, the matrix G is reported to a triangular matrix and the solution can be easily obtained through successive approximations.

2.3.2 Non-Linear in Parameters Models

More generally models includes also nonlinear terms, like for example in Fuzzy Logic Models on ANN discussed before. In this case the parameter identification is more difficult than the case of linear in coefficient models discussed in the previous subsection. In fact nonlinear in parameters models generally presents more than one local minimal point, so the maximum global point could also not be found. Moreover, there is not an analytical solution, but solutions must be found using iterative methods, that require some initial conditions.

An important propriety of Least Square Problems is that a cost function of the type:

$$C(\boldsymbol{\alpha}) = \frac{1}{2}\sum_{i=1}^{n} {e_i}^2 = \frac{1}{2}\sum_{i=1}^{n} (y_i - f(\mathbf{x_i}, \boldsymbol{\alpha}))^2 \tag{2.6}$$

must be minimized. From [39], it is shown that this special form of $C(x)$ often makes least-squares problems easier to solve than general unconstrained minimization problems.

If a *residual vector* r is introduced as:

$$r = (e_1(\boldsymbol{\alpha}), e_2(\boldsymbol{\alpha}), ..., e_n(\boldsymbol{\alpha}))^T \tag{2.7}$$

Using this notation, $C(\boldsymbol{\alpha})$ can be rewritten as $C(\boldsymbol{\alpha}) = \frac{1}{2}||r||^2$.

The derivatives of $C(\boldsymbol{\alpha})$ can be expressed in terms of the Jacobian J, which is the nxn_α matrix of first partial derivatives of the residuals, defined by:

$$J = \left[\frac{\partial r_i}{\partial \alpha_j}\right]_{\substack{i=1,...n \\ j=1,...n_\alpha}} = \begin{bmatrix} \nabla e_1^T \\ \nabla e_2^T \\ ... \\ \nabla e_n^T \end{bmatrix} \tag{2.8}$$

with ∇e_i^T the gradient of the $i-th$ error.

The gradient and Hessian of $C(\boldsymbol{\alpha})$ can then be expressed as follows:

$$\nabla C(\boldsymbol{\alpha}) = \sum_{i=1}^{n} e_i(\boldsymbol{\alpha}) \cdot \nabla e_i(\boldsymbol{\alpha}) = J^T \cdot r(\alpha) \tag{2.9}$$

$$\nabla^2 C(\boldsymbol{\alpha}) = J^T \cdot J + \sum_{i=1}^{n} e_i(\boldsymbol{\alpha}) \cdot \nabla^2 e_i \tag{2.10}$$

From (2.9) the Gradient of the cost function can be easily calculated once the residuals and the Jacobian are known. But the best proprieties of the Least Square problems is the Hessian can be calculated without evaluating second derivatives, in fact from (2.10) it can be easily computed once the Jacobian is known as $J^T \cdot J$, considering that the second term in (2.10) is often less important than the first one because the residuals e_i are close to affine if the solution is near ($\nabla^2 e_i$ is small), or because the residuals are small (e_i is small), making the second term in (2.10) negligible compared to the first one. To calculate the Jacobian a derivative is necessary, that can be calculated analytically if it is known , or numerically (two point and three point derivatives). Gauss Newton method is the simplest method that takes advantage of the simplified calculations of the Gradient and the Hessian in Least Square Optimization problems. It is based on the Newton Method, where the solution is found iteratively solving the equation $\nabla^2 C(\boldsymbol{\alpha}(k)) \cdot \boldsymbol{\delta}\alpha = -\nabla I(\boldsymbol{\alpha}(k))$, where $\boldsymbol{\alpha}(k)$ are the coefficients value at the sample time k and $\delta\alpha$ is the search direction, that is a term used to update the parameters, such that $\boldsymbol{\alpha}(k+1) = \boldsymbol{\alpha}(k) + \boldsymbol{\delta}\alpha$.

Using the proprieties of the Least Square problems the search direction can be calculated as:

$$J_k^T J_k \Delta\alpha \approx -J_k^T r_k \tag{2.11}$$

The search of $\Delta\alpha$ is a Linear Least Square problem, which solution can be found analytically using the method described in the previous subsection.

Problems occur if the matrix $J^T J$ is poorly conditioned or singular. Levenberg-Marquardt method is an extension of Gauss Newton Method, where equation (2.11) is modified as:

$$(J^T J + \lambda I)\Delta\alpha = -J^T r \tag{2.12}$$

the addition of λI is used to solve the problem of a poorly conditioned matrix $J^T J$.

When the iteration is close to the optimal solution the approximation of the Hessian as $J^T J$ is good, and so the Gauss Newton can be adopted, this corresponds to a zero value of λ. When we are far from the solution the second order approximation can not be valid, the J^TJ matrix is poorly conditioned and so the second term in (2.12) makes the matrix $(J^TJ + \lambda I)$ positive definite and the decreasing direction is guaranteed. For this reason, the value of λ is continuously updated, in particular it must be higher at the beginning, when far from the solution, and tend to zero as long as the solutions get close to the optimal point. So λ is initialized to a finite positive nonzero value, and if the cost function decrease, then λ is divided by a factor c $(\lambda_{k+1} = \lambda_k/c)$, while if the cost function increase, the search algorithm is not well working and λ get increased by a multiplication with a factor $d(\lambda_{k+1} = \lambda_k \cdot d)$.

In these methods the iterations terminate when one of the following condition is verified:

1. The cost function is lower than a minimum value (ϵ_1)
2. The maximum variation of the parameters is lower than a minimum value (ϵ_2), calculated as $\max_i \left(\dfrac{\Delta\alpha_i}{\alpha_i}\right)$
3. A maximum of iteration is reached $(N_{iter,max})$

2.4 Parametric GP

The objective of this thesis is to find general models, valid for each device used. Obviously, the considered devices represent a family, meaning that they must have similar behaviors and operative ratings. The adopted GP algorithm has been modified in order to make an identification of parametric models possible, meaning that the model structure is the same for all the devices but with different coefficients, optimized for each single device. In this way, the obtained model represents a behavior general for all the devices of the same family, and not a behavior specific of a single component. Moreover, once a general model is known, given a new device, it is necessary only to identify its coefficients and the GP identification procedure is not needed to be relaunched because the model structure is known.

Given that the aim is to infer engineering valuable system-level loss prediction models from experimental data, in the field of power devices, in order to maximize global system efficiency and to avoid thermal damages of the components, let us suppose to have a set $\mathcal{D}$ of m devices, i.e., $\mathcal{D} = \{D_1, \cdots, D_m\}$. Moreover, we suppose that a set of n experimental data have been extracted for each device. Such data consist in a number of physical quantities describing the functional behavior of a power device in terms of input and output variables. As a consequence, such amount of experimental data forms the training set the inference system works with. In particular, the training set $\mathcal{T}$ is made of m data set $\mathcal{T} = \{T_1, \ldots, T_m\}$ each of which is made of n vectors of inputs, each one consisting in k independent variables, i.e., $\boldsymbol{x} = \{x_1, \ldots, x_k\}$ and one output y, i.e., the power loss, representing the dependent variable. In other words, the training set $\mathcal{T}$ is made of n couples $(\mathbf{x}_{ij}, y_{ij})$ of input vectors and their outputs y_{ij}, with $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, m\}$.

The aim of the GP algorithm is to identify a global parametric functional model

$$z_{ij} = f(\mathbf{x}_{ij}, \alpha_j) \tag{2.13}$$

such that, each computed output z_{ij} is as close as possible to the actual output y_{ij}, $\forall i \in \{1, \ldots, n\}$ and $\forall j \in \{1, \ldots, m\}$.

In Eq.(2.13), f represents the parametric power loss expression, which is the same for each training set (device), and $\alpha_j = \alpha(T_j)$ represents the parametric coefficients of the generated expression. Such coefficients are represented in the GP by symbolic constant and are typically different for each training set. The specific values such coefficients can assume within an expression should model the differences existing in the behavior of different devices. As a consequence, the GP aims to find the best expression structure, hopefully the simplest and the most performing for the whole training set $\mathcal{T}$, by using the terminal sets enriched by a unique symbolic coefficient and no numeric constant. In order to instantiate the symbolic coefficients to numeric values within an evolved model for a specific device and, consequently, to evaluate the model on the whole training set $\mathcal{T}$, we need a two–step phase:

i) first we have to estimate the numeric values the coefficients should assume within the parametric expressions on each training set T_j (coefficients optimization),

ii) then we have to estimate the global error over the whole training set $\mathcal{T}$.

In the following section is described how the GP algorithm has been implemented and used.

2.4.1 Parametric GP tree structure

The GP tree structure adopted in this case is modified, in order to have an easier management of the coefficients, and improve convergence with non linear coefficients. There are two different ways to have a coefficient when the GP tree is initially constructed:

1. Function with coefficient (power and exponential), like for example α_3 in Fig.2.8(a).

2. Terminal Nodes

In particular for the second case, each branch connected to a terminal node has a weight, as shown in Fig. 2.8(a) for α_1, α_2 and α_4, and in the particular case that the terminal node is a coefficient, like α_4, it can be considered as a node of value 1 with its weight. This choice is due to different considerations: considering that the aim is to obtain parametric curves able to describe the behavior of physical devices, the inputs and outputs have a physical dimension. If these input variables are summed it is reasonable that a coefficient with a physical dimension equal to the ratio between the output and the inputs is multiplied to it. Moreover, the use of weights is useful to have a better management of the coefficients, in order to improve convergence and avoid indistinguishable parameters during the nonlinear optimization.

2.4.2 Coefficients optimization

The optimization of the coefficients is done on different steps.

- Find Linear Trees and group the coefficients
- Randomly initialize the nonlinear coefficients
- Identify the linear weights of each GP Linear Tree (LT) with a Linear Least Square (LLS) algorithm
- Regroup the coefficients to avoid indistinguishable coefficients and to minimize the number of coefficients
- Identify all the coefficients, optimized for each device, with Nonlinear Least Square (NLS) Algorithm, Levenberg Marquardt.

In the following subsections these steps are discussed in detail.

Linear GP trees

A main problem for nonlinear least square algorithm to converge is to have a good initial condition for the coefficients, in order to have few iteration to convergence and to avoid relative minimal point, if they are present. This is usually achieved starting by physical knowledge of the model that allows to give a plausible starting value for each coefficients, but in this case obviously the algorithm must be general purpose and a priori knowledge can not be used.

For this reason, first the Linear Trees (LT) are found, with the method described in [40], that is: start from the root node, and if a basic function different from sum or a terminal node is found then it is the root of a LT, otherwise if the function is the sum continue with the children nodes until a root of a LT is found, as shown in Fig.2.8(b). At this point the coefficients are recursively grouped and a weight coefficient β_i is given to the i-th node that is root of the i-th LT, as shown in Figure 2.8c.

Each coefficient γ_i belonging to the nonlinear coefficient set $\boldsymbol{\gamma} = \gamma_1, ..., \gamma_{n_\gamma}$, with n_γ the number of nonlinear coefficients, is initialized in this way : because neither its range nor its sign can be known, first with a 50 % probability a minus or plus sign is selected, then a random integer number between a minimum and maximum value in generated, and then ten is powered to this number (for example a random integer number a between -3 and 3 is generated, and the initial value is 10^a).

LLS

The linear coefficient set $\boldsymbol{\beta} = \beta_1, ..., \beta n_\beta$ with n_β the number of LTs, are multiplied to each LT and are calculated with mean of LLS, described in Section 2.3.1, as shown in Fig.2.8(c). Each model tree $f(\mathbf{x})$ then is represented as

$$f(\mathbf{x}) = \beta_1 \phi_{1,\boldsymbol{\gamma}}(\mathbf{x}) + ... + \beta_i \phi_{i,\boldsymbol{\gamma}}(\mathbf{x}) + ... + \beta_{n_\beta} \phi_{n_\beta,\boldsymbol{\gamma}}(\mathbf{x}) \qquad (2.14)$$

where $\phi_{i,\boldsymbol{\gamma}}(\mathbf{x})$, is the $i-th$ LT function calculated once the nonlinear coefficients $\boldsymbol{\gamma}$ have been fixed, Equation (2.14)can be reported to Equation (2.4). The problem to solve is of the kind:

$$\boldsymbol{\Phi} \cdot \boldsymbol{\beta} = \mathbf{y} \qquad (2.15)$$

with $\boldsymbol{\Phi}$ the matrix of sizes $(n \times n_\beta)$. Each element of $\boldsymbol{\Phi}$, Φ_{ij} is the value of the $j - th$ linear GP-tree with the $i - th$ inputs on the training set $\phi_{j,\boldsymbol{\gamma}}(\mathbf{x_{i1}})$, while the $i-th$ element of $\mathbf{y}$, represents the $i - th$ output of the first training set y_{i1}.

As training set is considered only the set of the first device T_1, in fact once the coefficients for the first device are found, they become the initial conditions for the next device (it is reasonable and desirable that the coefficients are not too different between different devices).

Group Coefficients

Before start the nonlinear least square the coefficients are reorganized, in order to minimize the number of coefficients, and to reduce the cases of coefficients repetition, that could cause not biunivocal values of the parameters and convergence problems (for example in $y = a(bx + c)$ a and c are indistinguishable).

The following simplifications and grouping of the coefficients are done:

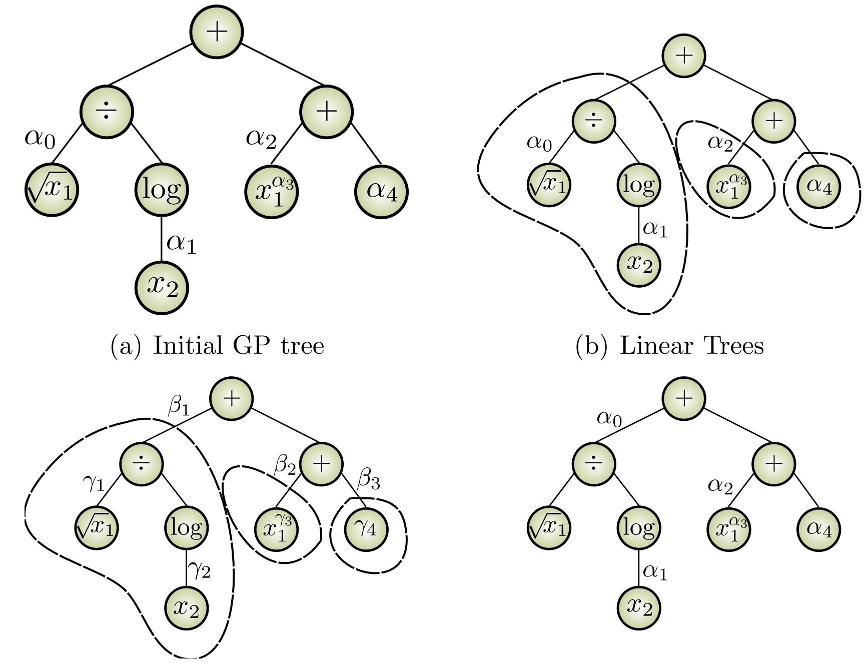
- $\gamma_1 \cdot (\gamma_2 f_1(\mathbf{x}) + \gamma_3 f_2(\mathbf{x})) \to \alpha_1 \cdot (f_1(\mathbf{x}) + \alpha_2 f_2(\mathbf{x}))$
  with $\alpha_1 = \gamma_1 \cdot \gamma_2$ and $\alpha_2 = \gamma_3/\gamma_2$
- $\gamma_1 \cdot (\gamma_2 f_1(\mathbf{x}) \cdot \gamma_3 f_2(\mathbf{x})) \to \alpha_1 \cdot (f_1(\mathbf{x}) \cdot f_2(\mathbf{x}))$
  with $\alpha_1 = \gamma_1\gamma_2\gamma_3$
- $\gamma_1 \cdot \dfrac{\gamma_2 f_1(\mathbf{x})}{\gamma_3 f_2(\mathbf{x})} \to \alpha_1 \cdot \dfrac{f_1(\mathbf{x})}{f_2(\mathbf{x})}$
  with $\alpha_1 = \dfrac{\gamma_1\gamma_2}{\gamma_3}$
- $(\gamma_1 f(\mathbf{x}))^{\gamma_2} \to \alpha_1 f(\mathbf{x})^{\gamma_2}$
  with $\alpha_1 = \gamma_1^{\gamma_2}$

This process is shown as example in the transition from Fig.2.8(c) to Fig2.8(d). It is important to note that what is changed is only how the coefficients are organized, but the structure of the models is always maintained the same.

Nonlinear Least Square

Finally the coefficients optimization is completed with a NLS algorithm, that is done on all the coefficients, both linear and nonlinear. A main problem is that the model on what the coefficient optimization is done is not known a priori, so a general purpose model must be adopted: the method adopted is Levenberg-Marquardt, described in Section 2.3.2 that is accurate, has a rapid convergence and requires less calculations than other optimization methods, which could impact the execution time of the algorithm being them in the loop.

The chi-squared error criterion [41] is applied between the simulated data y_{ij} and the estimated data:



(a) Initial GP tree

(b) Linear Trees

(c) Linear Coeffcients

(d) Final Coeffcients

Figure 2.8: GP tree coefficients management

$$\chi_j{}^2 = \frac{\sum_{i=1}^{n}((f(\mathbf{x}_{ij},\alpha(T_j)) - y_{ij})^2}{n} \tag{2.16}$$

where x_{ij} and y_{ij} are the inputs and the outputs of the training set T_j. The coefficient optimization is done on the first training set T_1, with the coefficients initialized with random values. Then, the optimization is done on the other sets, with the values of the coefficients previously found as initial values, in order to start from pre–optimized coefficients

2.4.3 Objectives

Different solutions able to give a good approximation of the data exist and can be found by the GP algorithm. There exist possible solution having a very high level of accuracy on the data inserted in the GP algorithm, but often these models are very complicated and rich of terms and coefficients that makes them difficult to be interpreted and adopted by the final user. Moreover, complicated functions can be cause of over-fitting problems, that means that also the noise is fit, and the obtained model is not able to generalize solutions also slightly different from the ones used during identification. On the other hand, functions very easy exist, but with levels of error too high and so not acceptable for the final user.

This means that two objectives must be calculated and then minimized at the same time:

- the Global Error over the training data set.
- the Model Complexity.

Global Error

In order to have a model sufficiently accurate for all the Devices Under Test (DUTs) as objective a global Root Mean Square Error ($RMSE$) between the experimental and GP-simulated losses has been introduced as a metric, that from (2.16) is:

$$RMSE = \sqrt{\frac{\sum\limits_{j=1}^{m} {\chi_j}^2 n_j}{\sum\limits_{j=1}^{m} n_j}} = \sqrt{\frac{\sum\limits_{j=1}^{m} {\chi_j}^2}{m}} \tag{2.17}$$

with n_j and ${\chi_j}^2$ respectively the number of element of the training set and the χ^2 returned by the NLS algorithm for the j-th DUT.

Complexity Factor

The elements used to construct the GP trees are shown in Table 2.1. The terminal set is composed of the inputs and the coefficients used by the GP Algorithm. The non terminal set is composed of the elementary functions used by the GP algorithm.

Table 2.1: Non–Terminal and Terminal Sets

#operands	Non-terminal	Description	Complexity
2	sum	$f + g$	1
2	multiplication	$f \cdot g$	1
2	power	f^g	cf
2	division	f/g	cf
1	logarithm	$\log(f)$	cf
1	natural exp.	$\exp(f)$	cf
1	power	f^α	cf
1	exponential	α^f	cf
1	square root	$\sqrt{f}$	cf
1	hyperbolic tangent	$\tanh f$	cf
1	mult.inv.	$1/f$	cf

#operands	Terminal	Description	Complexity
0	input	$\mathbf{x}$	0.6 (for multiplications and divisions) 1.0 (all other operations)
0	coefficient	α	1

Given a model structure and relevant parameters generated by the GP, the goodness of the model is evaluated based on a global fitting error over the whole training set (accuracy of the model) and on the model structure (complexity). To quantify the complexity of the model, the term *Complexity* has been introduced, based on the values given in Table 2.1 and calculated in the following way: each element of the terminal set has been assigned a complexity factor equal to one, whereas each element of the non-terminal set has been assigned a complexity factor cf equal or greater than one. Obviously, the choice of cf affects the complexity of each model and consequentially the choice of the optimal solutions by the algorithm, a low value of cf (between 1.2 and 1.5) promotes solutions more articulated and rich of nonlinear terms but for this reason, the solution obtained are less repeatable, while a value of 2 promotes polynomial solutions, for this reason, a value acceptable and typically used has been 1.5, but generally greater is the cf of a single elementary function lower is the probability that it is returned by the GP algorithm, and *vice versa*. The $Complexity$ value has been attributed to each function of the non-terminal set on the basis of its intelligibility and usefulness in the modeling problem. Higher $Complexity$, that means more penalty, is given to models that reduce the intelligibility of functional dependencies. Then the global $Complexity$ of the GP model has been defined based on the following criterion:

- if a function (non-terminal symbol) is the argument of another function, the complexity factors of the two functions are multiplied;

- if two functions are multiplied or summed, the complexity factors are summed and subsequently multiplied by the complexity factor of a sum or a product, respectively.

In the first case, a vertical development of the models (i.e., involved functions of functions) is prevented, especially for the functions having a great complexity. In the second case, a horizontal development of the models is prevented, i.e., models composed of many simple functions multiplied or summed between them, when a single more complicated function (e.g., exponential or power function) could be sufficient to model the quantity of interest. Complexity factors of quadratic and cubic function nodes have been assigned a value lower than 1. In particular each multiplication is assigned a complexity factor 0.6 (see Table 2.1). This prevents an excessive penalization of quadratic and cubic terms, which are desirable in loss formulas as they are quite compact and well intelligible. To prevent an excessive penalization of quadratic and cubic terms of the input nodes x_i of the terminal set, in case of multiplication complexity factors of such nodes have been assigned a value lower than 1, in particular 0.6 (see Table 2.1). For instance, using *Complexity* coefficients from Table 2.1, with $cf = 1.5$ for all the primitive functions, the model:

$$p_0(1 + x_1 \cdot \log(x_2) + \exp(\log(x_3) + (x_2 \cdot x_1)/x_3)) \tag{2.18}$$

has a complexity of:

$$Complexity = 1+(0.6\cdot 2)+2\cdot(2+(1.5\cdot(0.6+0.6)\cdot 1)) = 9.8. \tag{2.19}$$

Solution Methods

The algorithm has been first tested with a single objective optimization, where the fitness function was a combination of three terms:

$$fitness = a \cdot RMSE + b \cdot Complexity + c \cdot n_\alpha \tag{2.20}$$

A different set of the values a, b, c can give a different solution, obviously a greater the value of a lower is the error but grater are the number of coefficients and the complexity of the obtained model, while a greater value of b and c could mean a very simple solution but with a not acceptable error. So the values of a,b and c weights should be tuned using different combination and finding the bes trade off between complexity and accuracy.

To overcome the problem of the choice of the best values of a,b and c, and in order to discover models offering a trade-off between $RMSE$ and *Complexity* values this problem successively Multi-Objective Optimization (MOO) approach has been adopted. In particular, an elitist Non-dominated Sorting Genetic Algorithm (NSGA-II) [42], based on the Pareto-optimality principle, has been used, and is described in Appendix A. Such algorithm returns a Pareto front containing the non-dominated solutions present in the population, i.e., the solutions outperforming the other elements of the front in at least one objective, being worse in some other objectives. Herein, $RMSE$ and *Complexity* have been considered as objective functions for minimization.

2.4.4 Algorithm settings

Each GP execution has been repeated 10 times, and after a tuning phase the following settings for th GP have been adopted:

- population size: 400
- number of generations: 300
- Mutation probability: 20%

  Point mutation probability: 90%

  Subtree mutation probability: 10%
- Crossover probability: 80%
- Tournament Size: 2

These settings must be set in order to have a good evolution of the solutions during the generations. In the first generations, the models present big errors and complexity factors. In order to have more optimal elements on the Pareto front, the population must be sufficiently numerous, causing a quicker and better evolution. A small population size could cause the elements to collapse to the few values positioned in the Pareto front. Consequentially, the algorithm gets stuck in a local minimum value.

The number of generations represents the terminal condition of the algorithm, and to choose its value, it must be considered that its multiplication with the population size represents the total number of the solutions to be evaluated, that is the main factor affecting the computation time of the algorithm. An excessive reduction of this factor could cause non-optimal final solutions, while a big value could cause useless model evaluations having no improvement on the final solutions.

The mutation and crossover probabilities affects the evolution, and several tests could be done to verify their values. A big mutation probabilities can make the algorithm too random, while a big crossover probability could make the solution locked in local minimum solutions. The mutation probability is typically between 10% and 30%.

Finally, the tournament size affects the elitism. In fact, if more elements participate to the tournament, it is more probable that the Pareto Solution are drawn, penalizing the worse solution, that will win less tournaments. Consequentially, the models in the new generations will be more similar to the Pareto solutions. The minimum value of the tournament has been chosen to minimize this effect and have a better evolution, avoiding local minimum solutions.

2.4.5 Solution Choice

Being the GP an evolutionary algorithm, the solution depends on the initial population elements(that are randomly created) and how the population evolved during the generations. For this reason, each time the GP algorithm is executed a different Pareto front is returned. An example of GP results from a single run can be seen in Fig. 2.9, where the RMSE on the training (black circles) and on the validation (blue circles) are shown.

The easiest function on the top have a great error on both the training and the validation set, while as the complexity of the function increase the error on the training set decreases. The best element, as RMSE on the validation set, for this example is highlighted in red (model 4).

Because GP is an evolutionary algorithm the final solutions depends on how the population evolved during the generations and considering that there are a lot of possible models that can describe the behavior of the process, the returned formulas are different each time the algorithm is launched.

In Figure 2.10 Pareto fronts returned after different runs of the GP algorithm are represented, and in fact it is shown that the most of the solution returned are different because the evolution, but there are also some solutions that are returned in more than one GP execution, that are the ones overlapping in Fig. 2.10.

The Pareto fronts obtained after the execution of ten indepen-

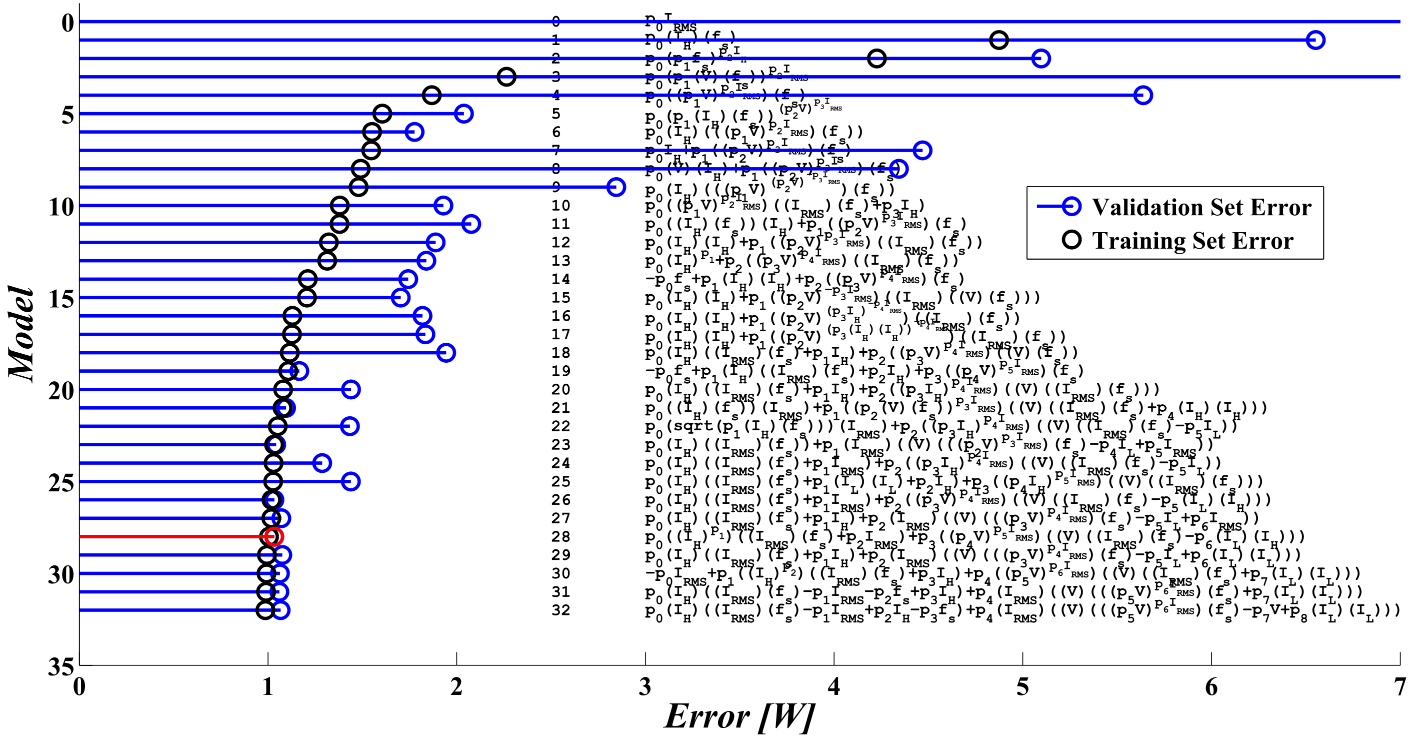


Figure 2.9: Pareto front returned by a GP run, with error on Training set (black) and Validation sets (blue), and relatives models

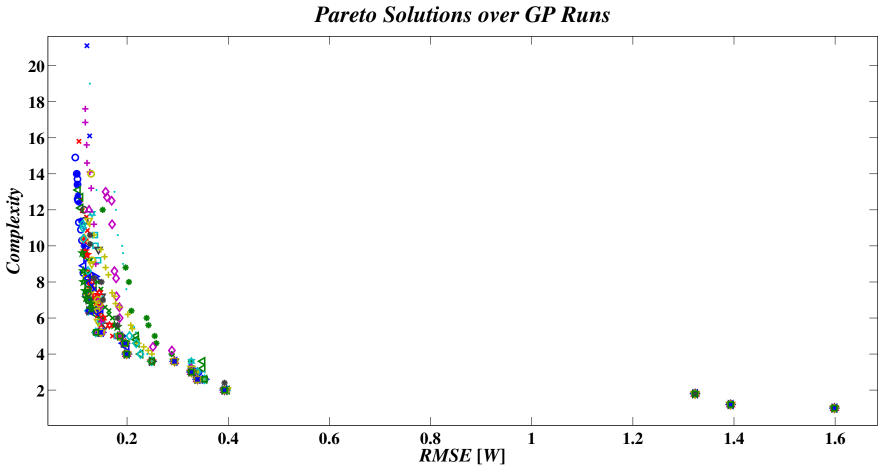


Figure 2.10: Pareto Fronts obtained after 20 GP runs. Each point represents a model in the Pareto Front, and each marker type represents a GP run

dent runs of the GP algorithm are shown in Fig. 2.11, where the number of times a model is returned on all the runs and the average age of its repetition are shown for each model. A high mean age indicates that the solution has been found at the beginning of the evolution, and it is usually true for the simplest solutions.

The two objective functions used during the Multi-Objective Optimization – namely, the $RMSE$ evaluated on the training data set and the $Complexity$ factor – are shown on the x and y axes, respectively. The solutions at the bottom-right side of the plot are characterized by very simple structures (typically constants or linear functions) and very high prediction errors with respect to experimental ac losses. Conversely, the solutions at the top-left side of the plot are the ones presenting the lowest errors at the cost of very complicated structures, which can be difficult to interpret and could cause data over-fitting. The optimal solutions, presenting a trade-off between prediction error and model complexity, are those situated in the left-bottom side of the graph, with acceptable errors and compact structures. It is worth not-

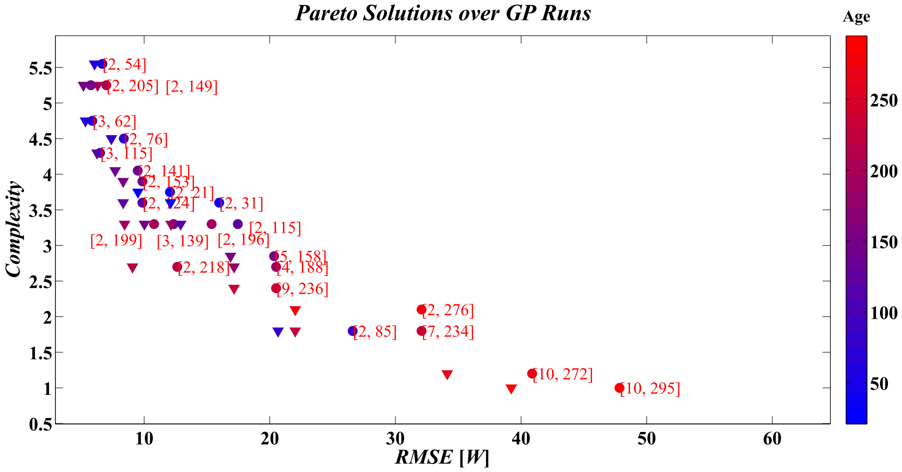


Figure 2.11: example of most repeatable solutions found after 10 runs. Circles: RMSE on the training set; Triangular: RMSE on the validation set; first coefficient in the bracket number of repetitions; second coefficient average age

ing that no practical benefits are achieved by using too evolved models, as only a minor $RMSE$ improvement is obtained at the expense of too complicated model expressions.

The simplest models, that are the easiest to be found and so are always returned by the algorithm and have a greater age, but present also too high error values.

In order to select an optimal ac-loss model among all the obtained Pareto-optimal solutions, several metrics have been introduced to quantify the model's goodness for practical purposes:

- $RMSE_{train}$: indicates the error on the training set adopted during the model construction, and must be minor of a certain acceptable value .
- $RMSE_{valid}$: the error on the validation set is used to verify that the model obtained can generalize and so must be minor of a certain acceptable value.
- *Repeatability*: indicates how many times a solution is re-

peated during different executions of the algorithm. If two solutions have an acceptable error on both the training and the validation set, the one with the higher repeatability is preferred because can be considered more reliable and simples. In fact, if there are terms summed that have not a great impact on the model error, it is probable that they do not repeat them self in more than one run.

- *Mean Age*: when the same solution is repeated more times during different GP runs, each time it has an age value, representing how many generation the solution has been present in the population. This value is the mean value on the ages of the repeated solutions. As can be seen in Fig.2.11 the simplest solution are more easily found by the algorithm and so have a greater age(red) than the most complex solutions that are usually obtained in the final generations (blue). For this reason, this value can be intended as an indicator of the simplicity of a model.

Chapter 3

Semiconductor Devices Models

The identification algorithm described in the previous Chapter has been applied to identify the power losses model of semiconductor devices. In particular two applications are considered: first, the algorithm has been applied to IGBT for induction cooking application. Then, GaN MOSFET for SMPS have been tested in simulation to prove the method.

3.1 Case Study: Half Bridge for Induction Cooking

3.1.1 Induction Cooking Systems

Induction Cooking is the process to heat metallic vessels for domestic application (like a pot or a pan) by electromagnetic induction. If a high frequency (above 20 kHz) Magnetic field is generated, eddy currents circulates in the vessel, heating it by Joule effect. Heat may also be generated by magnetic hysteresis losses in materials that have significant relative permeability.

So, respect to a classical gas hub, there is no heat transfer, but the heat is generated in the pot itself.

The main advantages of induction cooking respect to other techniques are principally the speed, the safety (there is no gas), the kitchen is not heated because there is not heat transfer and the controllability of the system. The main disadvantages of this technology is that it can be applied only to some kinds of pots, in particular the ones composed by ferromagnetic material, like iron and its alloys.

In order to prefer this kind of application on classical hubs it is important also have a good reliability, and a great energy efficiency, for environment problems and in order to minimize the consumed electrical energy.

Resonant converters are typically adopted in this kind of application, in particular the two most adopted are the Half Bridge Fig.3.1 and the Quasi Resonant converters. Resonant converters are adopted because they have the properties to minimize the switching losses of the semiconductor devices, then improving the energy efficiency.

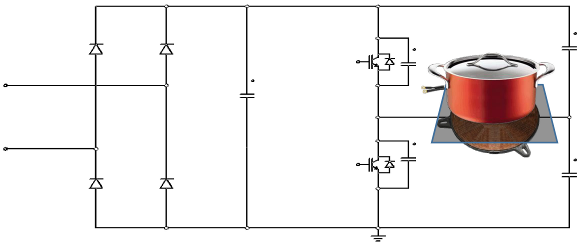


Figure 3.1: Half Bridge Circuit for induction cooking

The load is composed by the coils and the pots, that are magnetically coupled. It can be easily seen as a power transformer with the pot as its secondary side, and the easiest way to model it is as a series of a resistance R_{eq}, that is the secondary resistance reported to the primary side, and an inductance L_{eq}, that is the coil inductance [43]. In literature more complex ways to model the loads exist, as described in [44].

The load depends on different parameters, and cannot be known *a priori*, in fact it depends on the coils size, the kind of pot adopted, that is the material and the size, but also by operative parameters like its temperature, and the magnetic field frequency. Moreover, the position of the pot influences the load. In fact, the more the pot and the coil gets mismatched, meaning that the vessel is not well centered on the coils, the more the inductive component L_{eq} of the loads dominates on the resistive component R_{eq}. This concept is shown in Figures 3.2 and 3.3 (from [45]), where the values of these equivalent components are shown in function of the frequency for different pot material, and with L_0 and R_0 indicating the equivalent load parameters when no pot is positioned on the coil.

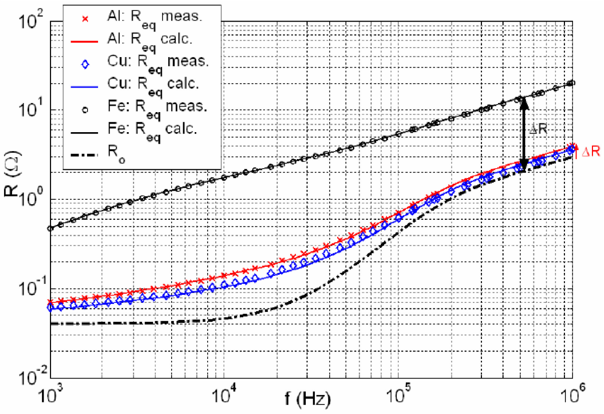


Figure 3.2: Equivalent resistance R_{eq} vs frequency curves for different materials [45]

If Fig.3.2 it can be seen as the presence of the pot increases the value of the equivalent resistance, that increases with the frequency for the skin effect in the coils. From Fig.3.3 it is shown that the presence of the pot typically causes a decrease in the equivalent inductance value, in fact L_0 is the self inductance of the coils windings.

As shown in [43] and [47], also the distance of the pot from the

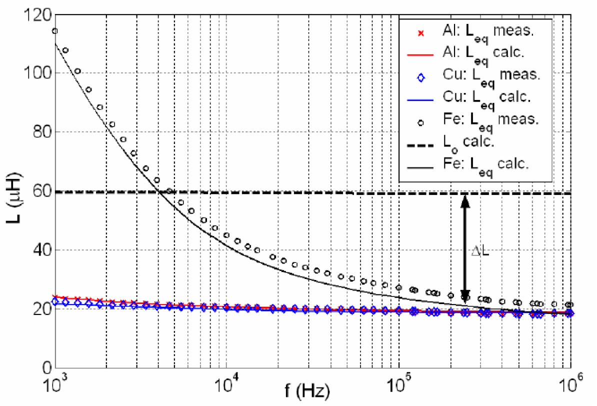


Figure 3.3: Equivalent inductance L_{eq} vs frequency curves for different materials [45]

coils and its temperature have an impact in the load parameters.

In the circuit, shown in in Fig.3.1, the 50 Hz line voltage gets rectified by a uncontrolled diode bridge. Then, in the Half Bridge the two IGBTs switch to high frequency in the range of 20 kHz-50 kHz, causing a high frequency current to circulate in the coils generating a magnetic field that couples with the ferromagnetic pot causing its heating.

In Figures 3.4 and 3.5 the typical waveforms in a Half Bridge inverter during a switching period are shown for different switching frequencies.

As can be seen in Fig.3.4, as closer to the resonance frequency is the switching frequency, as the peak current in the load current gets higher, because the load impedance is lower.

The frequency range of a HB inverter is typically between 20 kHz and 50 kHz. The lower limit of 20 kHz depends on the avoidance of acoustic noise.

If the switching frequency is too close to the resonance frequency the current and and voltage phase lag approaches zero and, as can be seen from Fig.3.5 the IGBT turn off current approaches zero. At higher frequencies the phase lag increases and the turn off current is the peak current. When the frequencies are

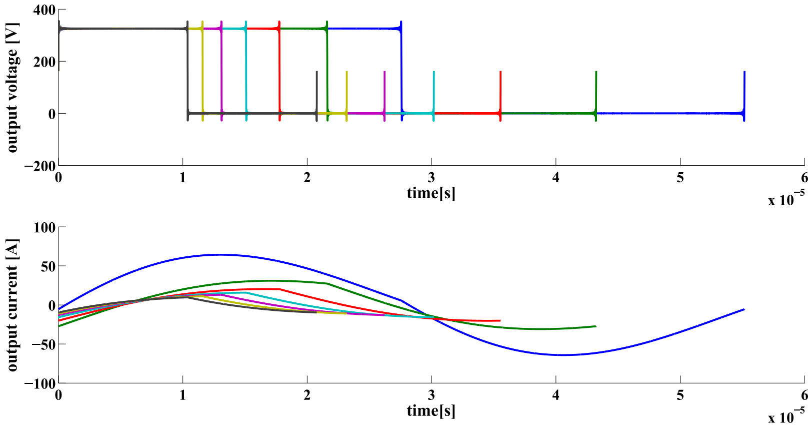


Figure 3.4: Mid-bridge voltage v_o and load current i_o in a switching period for different switching frequencies

too high the peak current approaches zero, so also in this case the turn off current becomes very small.

An excessive small value of the turn off current must be avoided to maintain the soft switching condition. In fact the two conditions that must be respected are:

$$\begin{cases} I_{off} > \dfrac{2C_{snub}V_{off}}{t_d} \\ \\ \dfrac{1}{2}(L_{eq})I_{off}^2 > \dfrac{1}{2}(2C_{snub})V_{off}^2 \end{cases} \tag{3.1}$$

where the first equation in 3.1 indicates that the time necessary to discharge/charge the two snubber capacitor must be lower than the dead time t_d, and the second equation specifies that the energy magnetized in the load is enough to discharge the capacitor.

The snubber capacitors positioned in parallel to the IGBTs are required to limit the turn off power losses of the devices in fact, as shown in Fig.3.6, the tail currents increase the power losses. A bigger snubber capacitances causes a lower value of the voltage during switching and consequentially a lower value of the switching energy E_{off}, but an excessive value of the snubber could cause

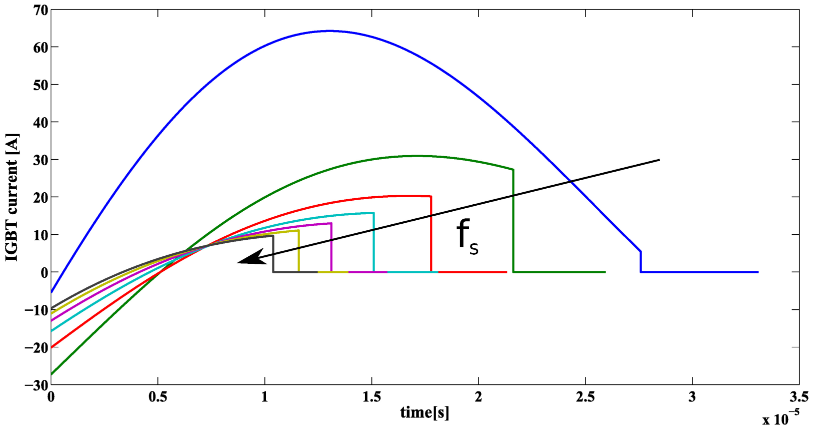


Figure 3.5: Diode and IGBT currents in a switching period for different switching frequencies

hard switching because the time necessary to switch off the IGBTs increases proportionally to the capacitor value and the energy required to discharge and charge the snubbers is higher, so the equations 3.1 are more difficult to be fulfilled.

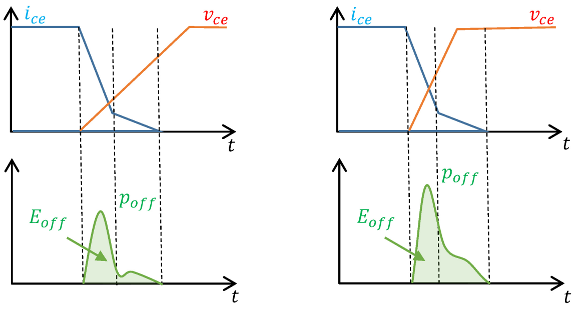


Figure 3.6: IGBT turn off with a High Snubber capacitor value (left) and a Low Snubber capacitor value (right)

3.1.2 Experimental Tests

The impossibility to known the load, as specified in the previous section, and the variability of the operative parameters, like the switching frequency and the line voltage, are the main issues that must be afforded when a power loss model of the IGBT must be constructed for this application. As discussed in the previous section, a model of the load is difficult to be constructed, considering that the RL model is too simplified and frequency dependent. For this reason, the idea has been to find the power losses inferring a model that has as inputs the measurable electrical data. In this way, a model of the load is not required to construct the loss formulas. In fact, the load is modeled by its effects on the current waveforms. I.e. given a particular load, and imposing the switching frequency f_s and having a known line voltage V_{line}, the load current waveform is known. Consequentially, the power losses in the devices are known too.

For this reason, the total IGBT losses depend on the following quantities:

1. Switching Frequency (f_s) [kHz]
2. Line Voltage RMS (V_{line}) [V]
3. Turn Off Current (I_{off}) [A]
4. Load Current RMS (I_{RMS}) [A]
5. Phase (ϕ) [deg]

Switching Frequency is the only quantity determined by the system controller, based on the load power requirement. The Line Voltage can vary between the AC grid regulation limits, depending on local grid conditions, that depend on the country and are typically about +/- 10 % of the nominal value. The Turn Off Current of the IGBT is measured when the line voltage is at its peak value (at the middle of the period). The root mean square Load Current is averaged over one rectified AC line Voltage period. The Phase represents the lag between the load voltage and current, calculated as the arc-cosine of the load power factor.

The last three terms are dependent on the load seen by the inverter, namely the particular pot and its positioning on the coils, and on the coils size. In fact, f_s and V_{line} are variables imposed by the system, while I_{off}, I_{RMS} and ϕ depends on the current waveform, and consequentially on the load. It is more clear making a first harmonic approximation and supposing a RL model. The phase can be expressed as $\phi \approx \tan^{-1}(\frac{\omega_s L_{eq}}{R_{eq}})$, so it depends on the f_s and the load. The load current amplitude, proportional to the I_{RMS}, is about $V_{line}/|Z(\omega_s)|$, where $Z(\omega_s)$ is the impedance of the load at $\omega_s = 2\pi f_s$, that is $|Z(\omega_s)| = \sqrt{(R_{eq}^2 + (\omega_s L_{eq})^2)}$.

Tests have been done using the following instruments: Lecroy HD06054 12 bit oscilloscope with a Rogowski coil that has been used for IGBT current measurement and Newtons4th N4L PPA5500 Precision Power Analyzer to measure the load current RMS and phase.

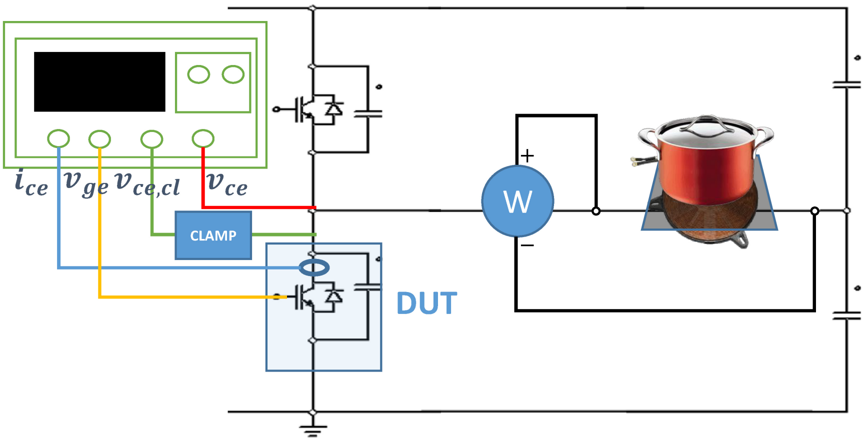


Figure 3.7: Measurement Setup

The experimental setup is shown in Fig.3.7, with the total power losses measured electrically from the oscilloscope, acquiring in a line voltage period (0.01 s) the collector current i_{ce}, the collector voltage v_{ce}, the clamped collector voltage v_{ce}(to measure more accurately the conduction phase [48]) and the gate emitter voltage v_{ge} (for synchronization). An example of the acquired waveforms is shown in Figure 3.8. Fig 3.8c it can be seen that there are four contributes to the total power losses, that are the initial Diode Conduction phase, the IGBT conduction phase, and the turn off losses. Turn off losses can be divided in two contributions: in the first phase the current decreases and the voltage gets higher, and the instantaneous turn off power losses are given by their product, presenting the typical bell waveform; in a second phase, with the current approaching zero and the voltage already high, the current get increased due to the tail currents, causing a great amount of instantaneous power losses.

The training set has been organized as a Plackett-Burman fractional factorial Design Of Experiment (DOE). This will ensure a uniform, orthogonal and balanced exploration of the inference space under investigation. As shown in Table 3.1, tests have been done with two different pots, the Tefal and the Demeyere, that have very heterogeneous magnetic behavior, two different coil size (145 mm and 210 mm), and two different conditions of matching, that are pot well centered on the coil and in mismatch conditions (pot not well centered, in particular centered at a quarter of the coil diameter). The adopted voltage levels are the extreme ranges of the European limits of the RMS voltage levels. Tests have been repeated for the even switching frequency values from 28 kHz to 38 kHz.

In the validation set, shown in Table 3.2, an inter medial pot has been used, which is the Normative pot, that has an intermediate behavior in terms of mirrored resistance; the adopted coil is of the intermediate 180 mm, with the pot positioned both in centered position and in mismatched position. The line voltage is at the nominal European value of 230 V. The switching frequencies are taken both inside and outside the internal range of the training set, to verify the capacity of extrapolation of the model.

Two kind of model generalization must be considered in this kind of applications: *interpolation* is the capability of the model to generalize solutions that are internal to the training set ranges; *extrapolation*, on the opposite, represents how the model can generalize solutions outside the training set range.

Extrapolation is important in this case, because this analy-

Table 3.1: Training set

C_{snub}[nF]	V_{line}[V]	f_s[kHz]	*pot*	*coil size*[mm]	*mismatch*
22	196	28:2:38	Tefal	145	0
22	196	28:2:38	Demeyere	210	0
22	196	28:2:38	Tefal	210	quarter
22	196	28:2:38	Demeyere	145	quarter
22	254	28:2:38	Tefal	145	0
22	254	28:2:38	Demeyere	210	0
22	254	28:2:38	Tefal	210	quarter
22	254	28:2:38	Demeyere	145	quarter
33	196	28:2:38	Tefal	210	0
33	196	28:2:38	Demeyere	145	0
33	196	28:2:38	Tefal	145	quarter
33	196	28:2:38	Demeyere	210	quarter
33	254	28:2:38	Tefal	210	0
33	254	28:2:38	Demeyere	145	0
33	254	28:2:38	Tefal	145	quarter
33	254	28:2:38	Demeyere	210	quarter

sis is principally oriented in the prediction of the power losses in order to avoid damages to the IGBTs. For this reason, it is not always possible make work the IGBT in potentially dangerous conditions during characterization. Moreover the possible loads are very numerous, and it could be not physically possible make all the necessary tests.

The tests have been done on the same semiconductor device with two different snubber capacitors values (22 nF reported as DUT 1 and 33 nF reported as DUT 2), which affect the dynamical behavior of the devices and so the turn off losses. Consequently, different model's coefficients are obtained depending on the snubber capacitor used.

Table 3.2: Validation set

C_{snub}[nF]	V_{line}[V]	f_s[kHz]	*pot*	*coil size*[mm]	*mismatch*
22	230	37,34,32,31 30,28,25,22	Normative	180	0
22	230	37,34,32,31, 30,28,25,22	Normative	180	quarter
33	230	37,34,32,31, 30,28,25,22	Normative	180	0
33	230	37,34,32,31, 30,28,25,22	Normative	180	quarter

3.1.3 Results

Output models

The most repeatable results after 10 executions are shown in Fig.3.9.

A trend in all the returned model is that of the five inputs variables used, only the line voltage V_{line}, the I_{rms} and the switching frequency f_s appear in the main models, so they are the variable that have the greater impact on the power losses. From the models shown in figure 3.9, the chosen model is the one indicated as [8,51], that is written in Eq.3.2, and with coefficients for two different snubber capacitors in Tab.3.3.

$$P_{\text{loss}} = p_0 \cdot (p_1 I_{RMS})^{p_2 f_s} - p_3 V_{line} + p_4 I_{RMS} \qquad (3.2)$$

Table 3.3: Coefficients

DUT	p_0	p_1	p_2	p_3	p_4
1	4.76e-1	8.06e-2	8.64e-2	7.49e-1	-9.8e-3
2	6.72e-2	1.08e-1	1.43e-1	6.58e-1	-6.77e-3

This model has been chosen for its small error (lower than 0.5 W) both on the training set and in the validation set, has a great repeatability and age, meaning that it is a model simple and accurate.

The RMSE of (3.2) on the Training Set is 200 mW (190.7 mW for DUT 1 and 207 mW for DUT 2), while on the Validation Set it is 472 mW (496 mW for DUT 1 and 447 mW for DUT 2).

The reconstructed values of the training and validation set, and the relative errors for the model (3.2) are shown in Figures 3.10, 3.11 ,3.12, 3.13, 3.14, 3.15.

The Training Set Values are all perfectly reconstructed by the model, with errors included in $\pm 0.5W$. In the validations set the error is greater than 0.5 W in the samples 7,8, 15 and 16. These samples have been inserted to valuate how well the algorithm performs out of the range of the training set. Their error is larger than the other samples, but still in an acceptable range (below 1W). The results is positive considering that both the load and the system variables in the validation set are different from the ones adopted in the training set.

3.1.4 Discussion

Many considerations can be done observing the most repeatable functions in Fig.3.9. The turn off current I_{off} and the phase ϕ never appear. This does not mean that the total power losses are independent of these values, but means that the effects of these inputs is included in other variables. In fact, all the variables are correlated between them. For example, observing Fig.3.5 it is clear that, a decrease in the switching frequency f_s causes a high decrease in the RMS current I_{RMS}, and a slow decrease in the turn off current I_{off}.

Some models containing all the variables certainly exist and have appeared during evolution, but the increase of the complexity is not justified by a best data fit and, consequentially, these solution are dominated. The GP simply returns the simplest combination of the input variables.

Not inserting some variables because it is supposed that they present a small effect on the power losses than the other ones is not usually a correct approach. It is *a priori* knowledge, but could be not correct and, consequentially, it could lead to not optimal models. It is the task of the GP decide if a variable deserves or not to be included in the returned models.

The terms that appear more frequently are power, linear an quadratic term of I_{RMS}, that in fact is the variable that has more information about the load. The linear and quadratic term of I_{RMS} could be seen as more representative of the conduction losses, while the term that are a combination of f_s and I_{RMS} could represent more the switching losses (I_{off} is correlated with I_{RMS}).

The main application of this approach is to implement it on an embedded system in order to have a real time estimation of the devices power losses and so find the right control in order to reduce the power losses and avoid thermal damages. The small number of coefficients to be memorized and the simplicity of the models make this real time solution quick not requiring too time expensive and memory expensive calculations.

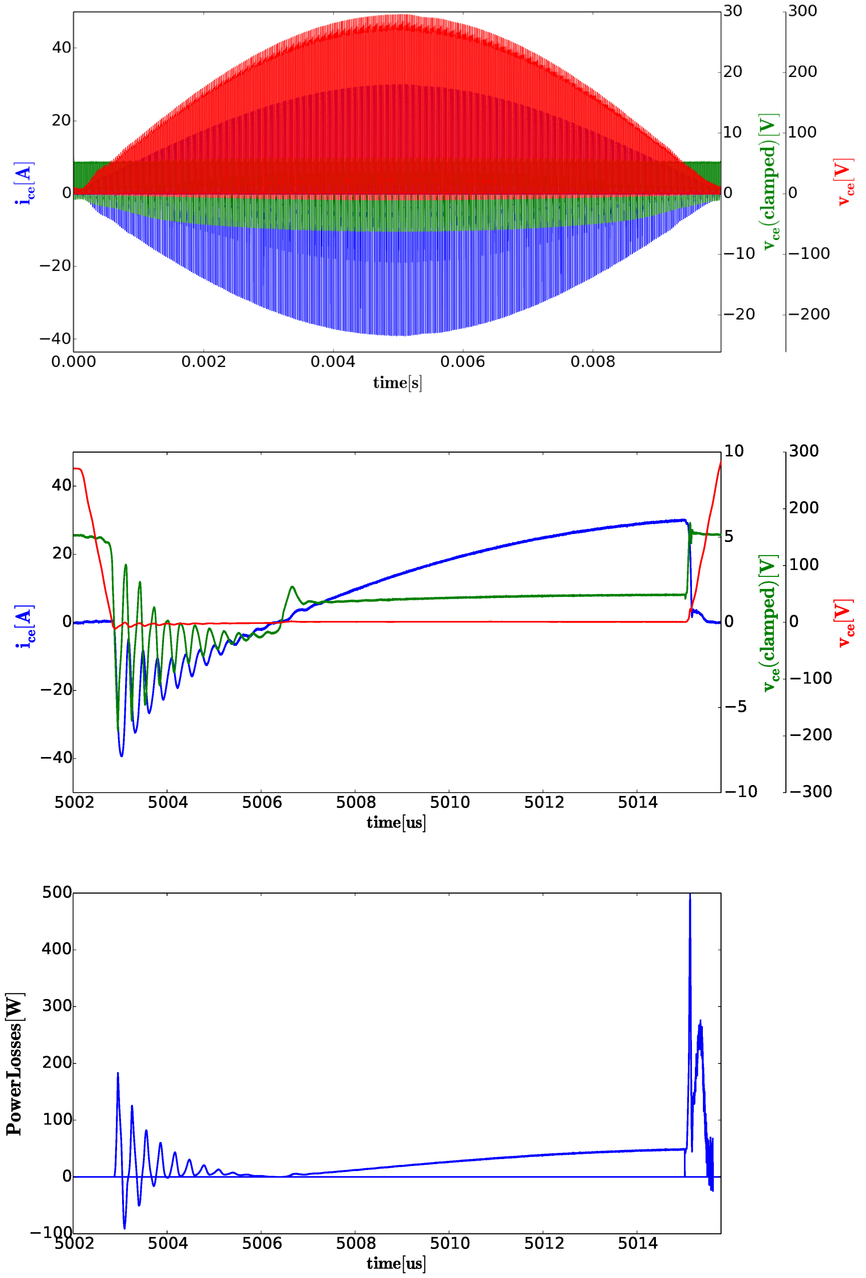


Figure 3.8: IGBT acquired waveforms. Top:AC Line period; Middle:On Period; Bottom:power losses during On Period

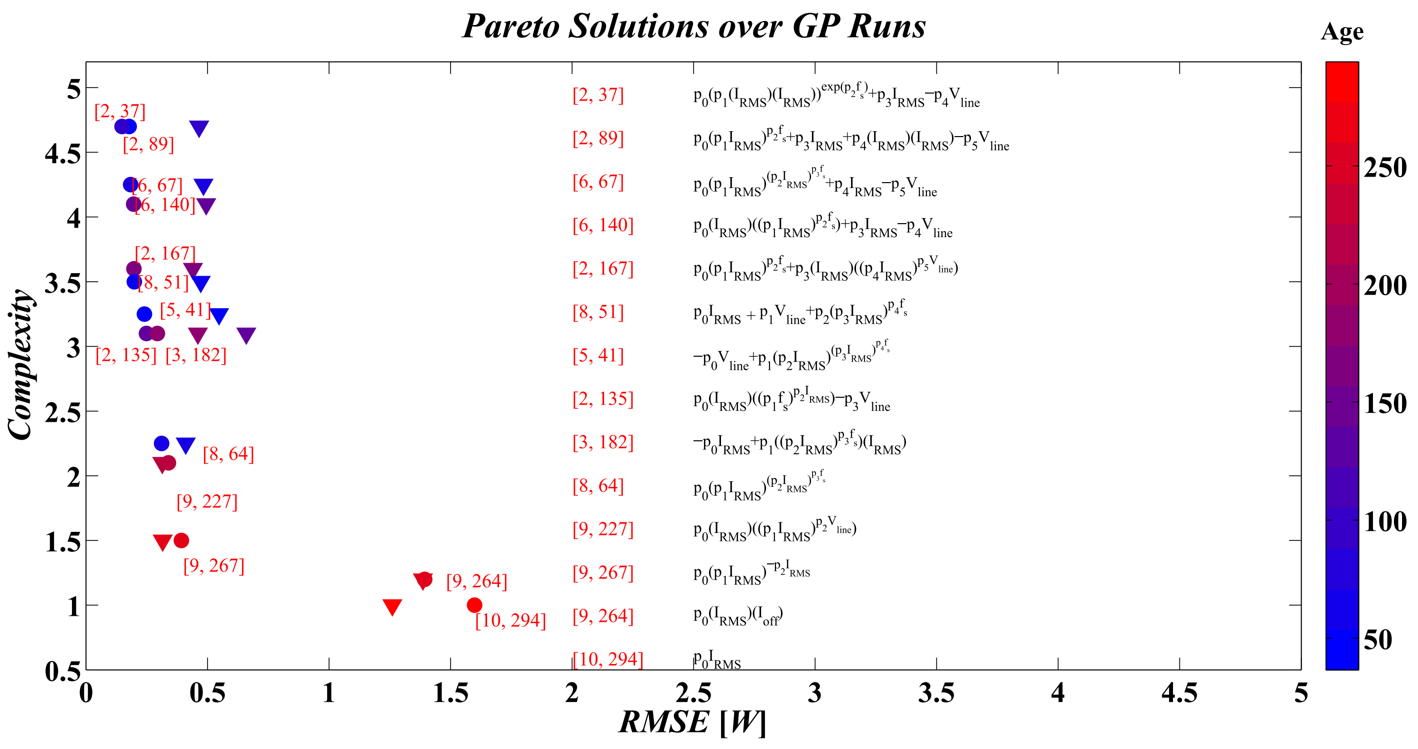


Figure 3.9: Pareto Front with the most repeated models after 10 executions, the number in brackets represent the number of times the model is repeated on all the GP runs, and the average of the ages

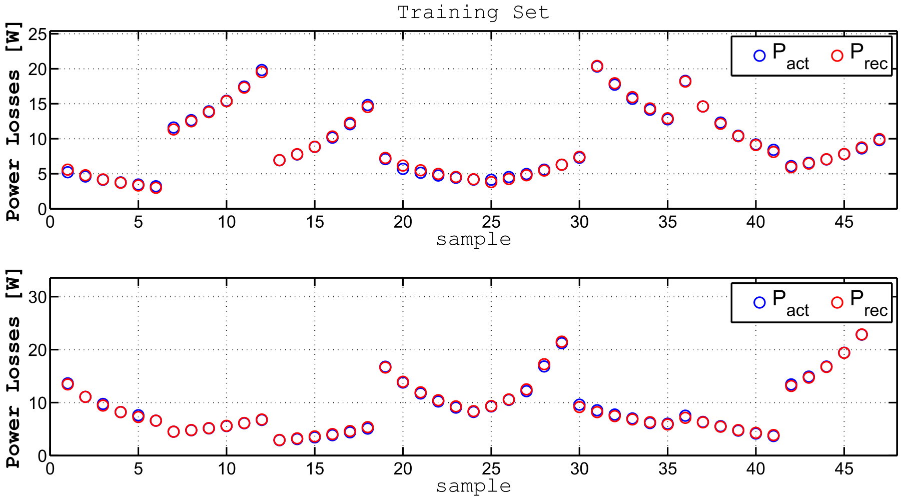


Figure 3.10: Training Set actual and reconstructed data for DUT1 (top) and DUT2 (bottom)

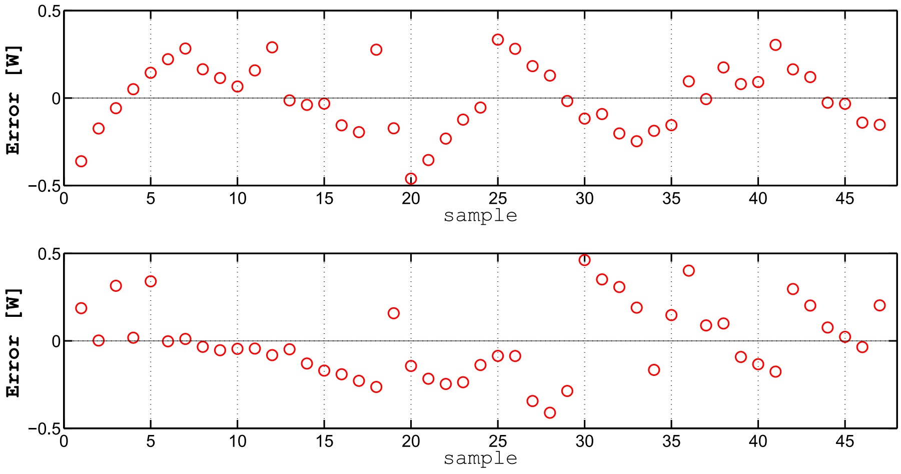


Figure 3.11: Training Set errors for DUT1 (top) and DUT2 (bottom)

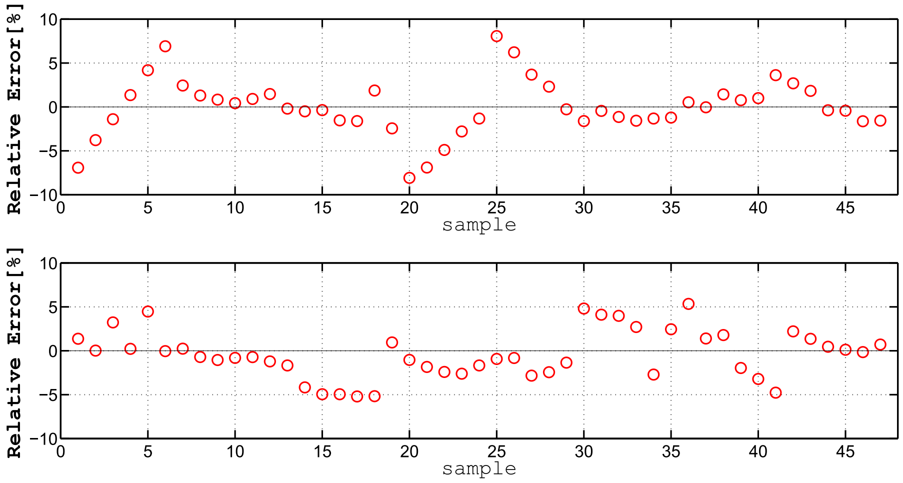


Figure 3.12: Training Set relative errors for DUT1 (top) and DUT2 (bottom)

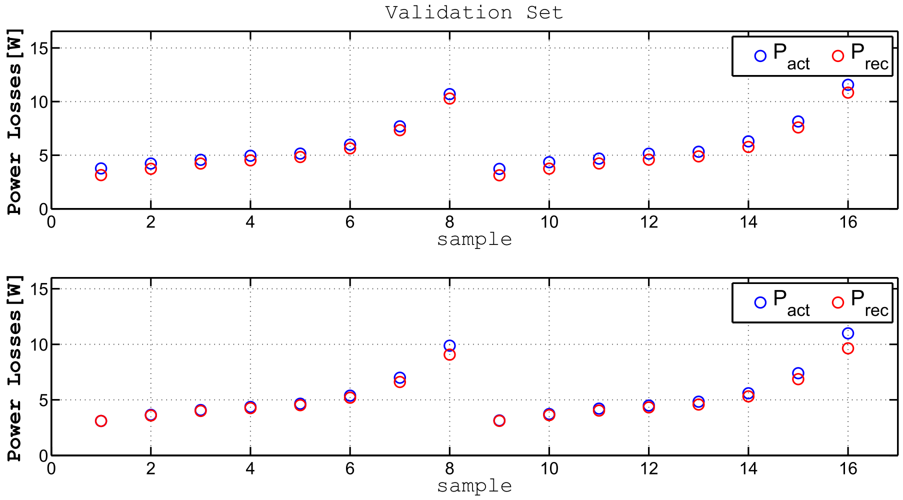


Figure 3.13: Validation Set actual and reconstructed data for DUT1 (top) and DUT2 (bottom)

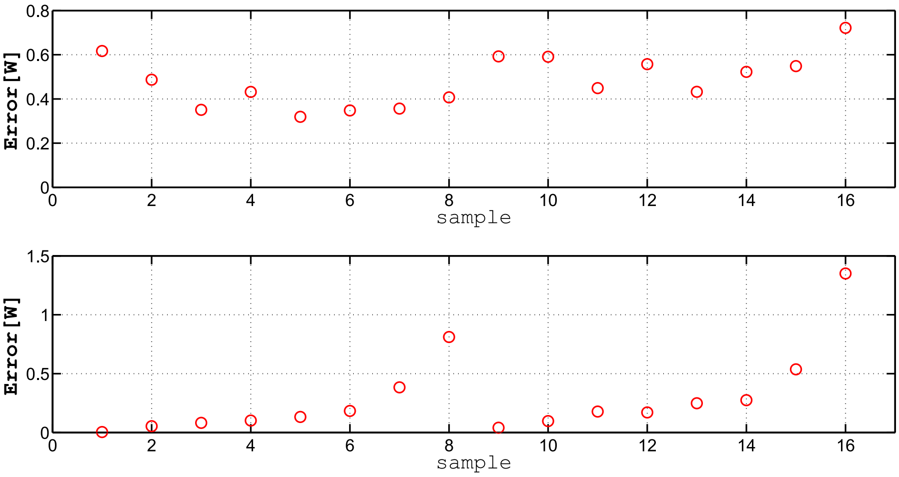


Figure 3.14: Validation Set errors for DUT1 (top) and DUT2 (bottom)

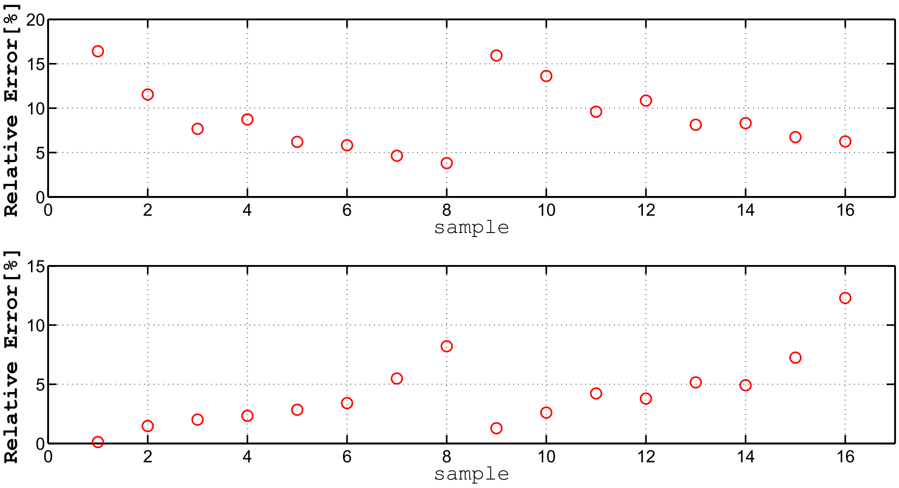


Figure 3.15: Validation Set relative errors for DUT 1 (top) and DUT 2 (bottom)

3.2 Case Study: SMPS transistors

The identification algorithm has been applied to the Buck circuit shown in Fig.3.16 to calculate the power losses of the MOSFETs, indicated by M_H and M_L. In particular the total power losses of the two MOSFETs, constituting a Switching Cell (SC), have been calculated in function of electrical parameters that characterize the behavior of the switching converter.

The choice to identify power losses in a SC respect to a single device depends on two considerations:

1. Measure power losses in a switching cell could be usually easier, in fact it could be measure as the difference between the input power and the power given to the inductor
2. From the point of view of the efficiency it is not necessary to know how the power is divided by the two devices but the total power dissipated
3. Some loss contributions of a device depends on the other device characteristic

In absence of an experimental setup and in order to prove the quality of the algorithm and develop a method that can be then applied to real GaN switching cell or also to other semiconductor devices operating in a SMPS, the data have been identified through simulation done by SPICE simulator.

So there must be a clarification: the aim of this study is not to identify the model of real GaN, in fact usually SPICE models of semiconductor devices can give a dynamical and static behavior different from the real transistor. In this application the GaN model is considered an artificial device, and the aim is to identify the power losses model of these artificial devices. At the same time, in order to have the results as close as possible to the real behavior of a Switching Converter, a measurement noise has been added to the power losses calculated by simulation, that is a Gaussian Noise with standard deviation of 1W. As known by the statistic

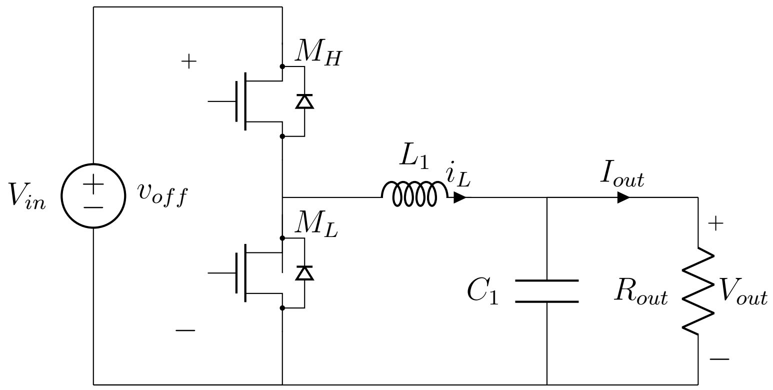


Figure 3.16: Buck converter circuit

theory, the RMSE approaches the error standard deviation for an increasing number of samples, under an hypothesis of experimental data having noise with an uniform white Gaussian distribution.

Moreover parasitic inductance and resistance due to Printed Circuits have been inserted in the simulator, in fact they affect the dynamic behavior of the devices and so the switching power losses.

3.2.1 Device Characterization Tests

As done with the IGBT in section 3.1 GP has been utilized to predict the power losses of the Devices Under Tests, identify a model depending on electric values that characterize the behavior of the system.

The quantities adopted as inputs are :

1. Root Mean Square of the inductor current $i_L(t)$, I_{RMS}[A]

2. High Peak of the current I_H[A]

3. Low Peak of the current I_L[A]

4. Turn off Voltage (determines the voltage at the switching) V_{off}[V], that in the case of a Buck Converter corresponds to the Input Voltage V_{in}[V]

5. Switching Frequency f_s[kHz]

f_s, V_{off} and the current peaks I_H and I_L have the main effects of the power losses, while I_{RMS} affects principally the value of the conduction losses. The Duty Cycle D has not been used because it affects the conduction losses of a single device, but if the same devices are considered it does not affect the total power losses values. From the computational point of view, when the ripple is small, it is possible that the high and low inductor current values can be numerically confused with the average inductor current. For this reason, also higher ripple conditions have been considered, changing the inductance values.

During the training of the algorithm some of the inputs can be set directly by the user during the model construction, that are the input voltage V_{in} (with a power supply) and the switching frequency f_s (with the controller), while the other input variables, once f_s and V_{in} are fixed, depends on the value of the inductance adopted and on the Duty Cycle D, in fact a lower inductance causes a higher peak value and a lower valley value in the inductor current, and an increase in the RMS of the current.

In SMPS the main variations during the work are in the load current I_{out} shown in figure 3.16, that depends on the circuits that follows the switching circuit, and in the Input Voltage V_{in}.

For this reason, the quantities varied to construct a training and a validation set are:

1. Switching Frequency f_s[kHz]

2. Duty Cycle D[%]

3. Input Voltage V_{in}[V]

4. Output Current I_{out}[A]

5. Inductor $L[\mu\text{H}]$

Several simulated tests have been done to characterize the DUTs power losses, in particular in each experiment variables that influence the circuits waveforms have been changed, and the Training and Validation set adopted are reported in Table 3.4 and Table 3.5 respectively.

Table 3.4: Training set employed.

f_s[kHz]	D[%]	V_{in}[V]	I_{out}[A]	L[μH]
100, 600, 1000	0.2, 0.5, 0.8	10, 20, 30	5, 10, 15	11, 6, 1

The devices adopted are GaN taken by the library in [49] with the characteristics reported in Table 3.6.

In Section 1.2 it is described how the geometry of the devices affects their performances, and it is known that there is a trade off between resistance, capacitance, and voltage and current ratings that affect conduction and switching losses. For this reason, the devices have been chosen having similar voltage ratings but different electrical parameters, in fact DUT 2 should have the best conduction losses(low on resistance) and the worst dynamical behavior (high gate charge),DUT 3 has the best switching and intermediate conduction and DUT 1 the worst conduction and intermediate switching. The three devices are heterogeneous but all could be selected for an application of 60 V maximum of input voltage and 25 A maximum of output current.

Table 3.5: Validation set employed.

f_s[kHz]	D[%]	V_{in}[V]	I_{out}[A]	L[μH]
300 : 200 : 1100	0.3, 0.6, 0.9	15, 25	7, 13, 19	3, 9

Table 3.6: Devices Under Test Characteristics

DUT	DEVICE	$R_{on}[m\Omega]$	$Q_g[nC]$	$V_{max}[V]$	$I_{max}[A]$
1	EPC2001	7	10	100	25
2	EPC2020	2.2	20	60	90
3	EPC2102	4.4	6.8	60	23

3.2.2 Results

After 10 runs the most repeatable models are shown in Fig.3.17, with the model structure specified only for the models having a training set RMSE sufficiently low (lower than 1.5 W).

As shown in Fig. 3.17 the models with lower error on the training present a greater error on the validation set (indicate by the triangles). In this case study the same considerations about extrapolation and interpolation generalization discussed in the previous section of the IGBT must be done: the most complicated model are not able to give a good generalization of the samples outside the range considered in the training set. This error is certainly not due to over-fitting, intended as the fit of the noise, because it is still greater than 1 W for all the three device.

The models with a low error both on the train and the validation set are the [2,28], [2,99], [2,162] and [5,220], with the last one chosen because its higher number of repetitions and average age. The model chosen has been reported in equation 3.3, with the coefficients for each DUT shown in Table 3.7.

$$P_{loss} = p_0 \cdot (p_1 \cdot V_{off})^{-p_2 \cdot I_H} f_s \cdot V_{off} \cdot I_{RMS} \tag{3.3}$$

The reconstructed train and validation sets sample and the er-

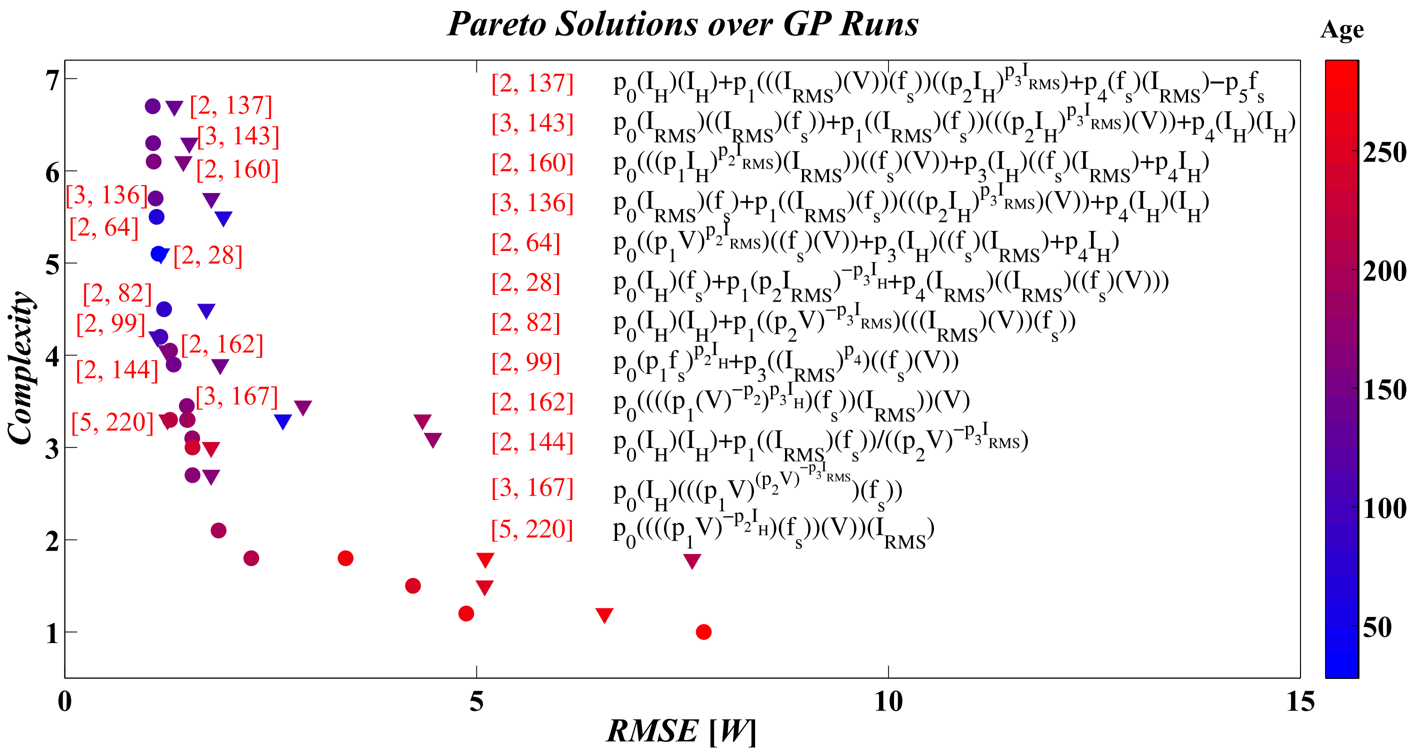


Figure 3.17: Pareto front with the most repeated models after 10 executions, with training RMSE(circles) and Validation RMSE (triangles) and the number in brackets represent the number of times the model is repeated on all the GP runs and the average of the ages

Table 3.7: GP models coefficients

DUT	p_0	p_1	p_2
1	3.90e-5	4.45e-3	2.29e-2
2	4.30e-5	1.84e-3	1.42e-2
3	4.11e-5	4.72e-3	2.16e-2

rors are shown in Figures 3.18, 3.19, 3.20, 3.21, 3.22, 3.23, 3.24and 3.25.

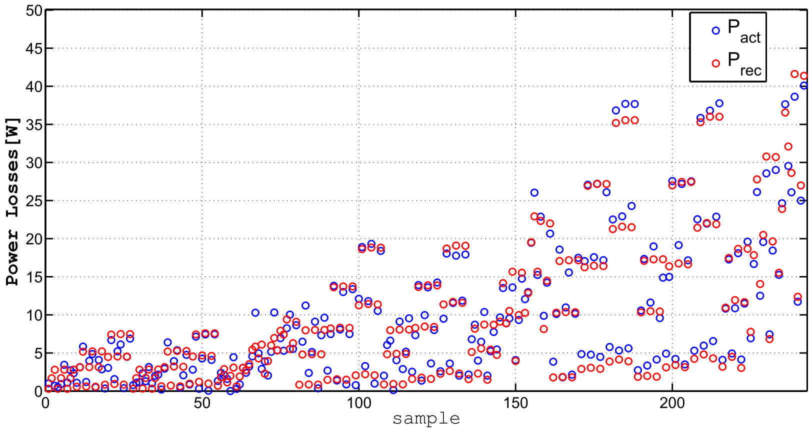


Figure 3.18: actual and reconstructed power losses for the training set (DUT1)

3.2.3 Discussion

From Fig.3.17 many considerations can be done: of the five inputs variables adopted all appeared except for the lower peak of the switching current I_L, this can be justified because in the majority of conditions (when i(t) never becomes negative) while the high peak I_H determines the turn off losses of the high side switch, while the I_L represent the turn on losses of the high side.

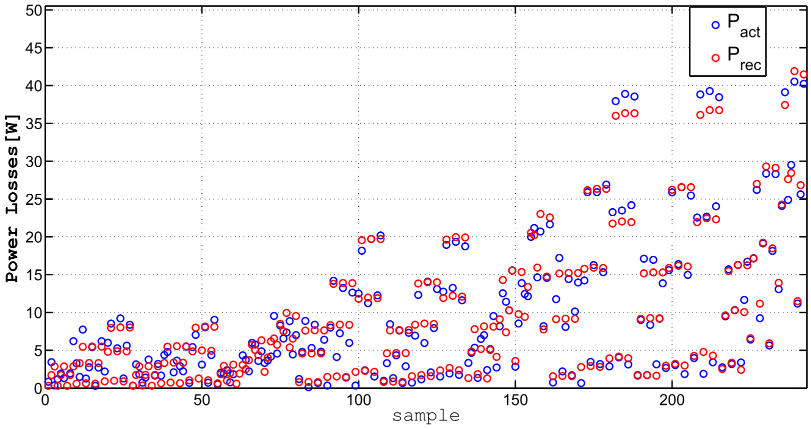


Figure 3.19: actual and reconstructed power losses for the training set (DUT2)

A term of the kind I_H^2 appears in many model, and can be seen as an overestimation of the conduction losses that are proportional to the square of the I_{RMS}.

The best model (3.2) can be rewritten as:

$$P_{loss} = p_0 \cdot {p_1}^{-p_2 \cdot I_H} \cdot {V_{off}}^{1-p_2 \cdot I_H} f_s \cdot I_{RMS}, \qquad (3.4)$$

where the terms with V_{off} are put together for a best interpretation.

As in the case of the IGBTs for Induction Cooking not all the variables appear in the solutions, for a correlation between them. The model (3.4) can be interpreted in this way: a higher value in the RMS current I_{RMS} causes an increase in the conduction losses and at the same time bigger switching losses are present if there are higher values of the switching frequency f_s and of the voltage V_{off}. Moreover, the switching currents I_H and I_L are dependent on the average inductor current and consequentially, on the RMS current I_{RMS}. For this reason, I_{RMS} has an indirect effect also on the switching losses.

Considering that the power losses are not strictly linear with I_{RMS}, V_{off} and f_s, the terms containing I_H act as corrective terms: an increase in the V_{off} causes an increase in the turn off current

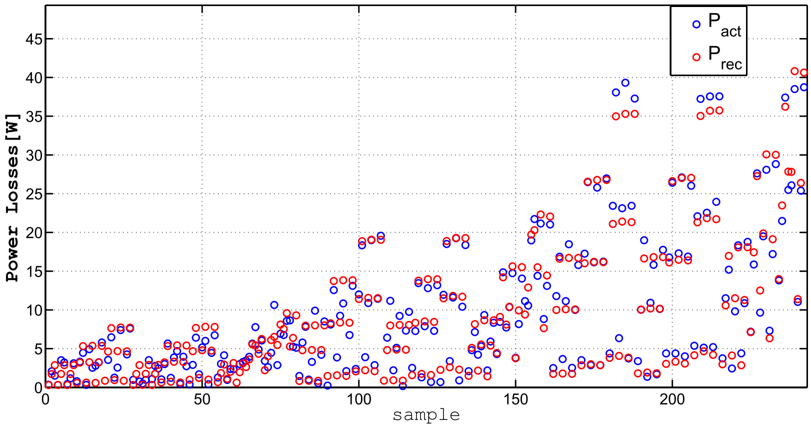


Figure 3.20: actual and reconstructed power losses for the training set (DUT3)

I_H, so its power to $(1 - p_2 \cdot I_H)$ compensates these effects. At the same time, greater is I_{RMS}, greater is the average current and consequentially I_H increases, so the negative power term $p_1{}^{-p_2 \cdot I_H}$ takes into account the dependency of I_H from I_{RMS}. The losses due to I_L are similar to the losses due to I_H for low ripple (high inductance L), while are dominated by the turn off losses in high ripple conditions (low inductance L), and probably for this reason, the contribution of I_L does not appear.

As shown in Table 3.8 the errors on the training and in the validation are fo about 1.2-1.3 W. The value of the RMSE should approach the value of the standard deviation, so about 1 W. There are in fact models that have about 1W on both the training and the validation set, but they are too complicated or with too many terms, and so don't repeat themselves during different GP runs.

Table 3.8: GP models errors

	$DUT1$	$DUT2$	$DUT3$	TOT
Training Set	1.26	1.27	1.3	1.28
Validation Set	1.38	1.16	1.19	1.25

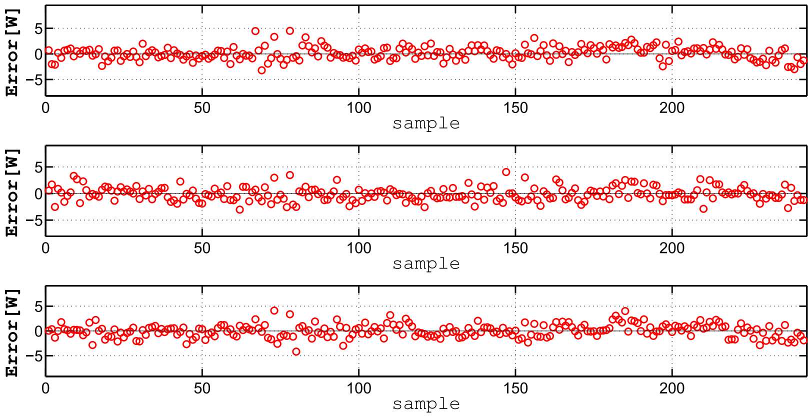


Figure 3.21: power losses errors for the training set (top: DUT 1, middle : DUT2, bottom: DUT 3)

Now the question is, it is necessary to run the GP if a new device must be characterized? If a normal GP would be run there could not be models that are well suited for all the devices adopted. Otherwise in a Parametric GP the solution is forced to be equal for all the Devices characterized. For this reason, if a device having behaviors similar and inter-medial to the ones adopted during GP, it is reasonable that the model returned by the GP can give a good representation of its behavior. This has been verified modeling another device, the EPC2029, with electrical characteristics described in Table 3.9.

Table 3.9: EPC2029 Characteristics

$R_{on}[m\Omega]$	$Q_g[nC]$	$V_{max}[V]$	$I_{max}[A]$
3.2	16	80	48

The coefficients can be easily found utilizing some of the NLS techniques described in Section 2.3.2 on the training set of the new device, and using as initial conditions the coefficient of one of the devices from Table 3.7. The resulting coefficients are shown in

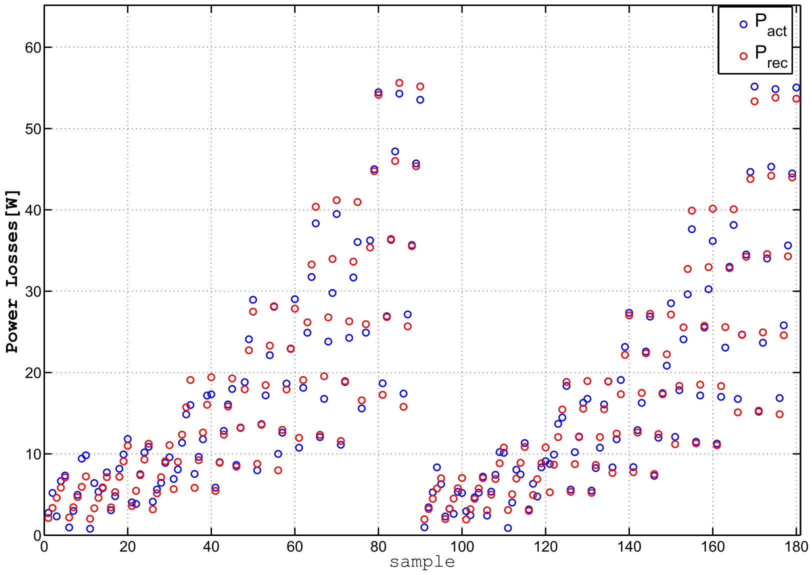


Figure 3.22: actual and reconstructed power losses for the validation set (DUT1)

Table 3.10: EPC2029 coefficients

p_0	p_1	p_2
4.29e-5	2.24e-3	1.53e-2

Table 3.10. The total error on the training set is 1.25W and the error on the validation is 1.11W, that are values in line with the ones of the other devices from Table 3.8.

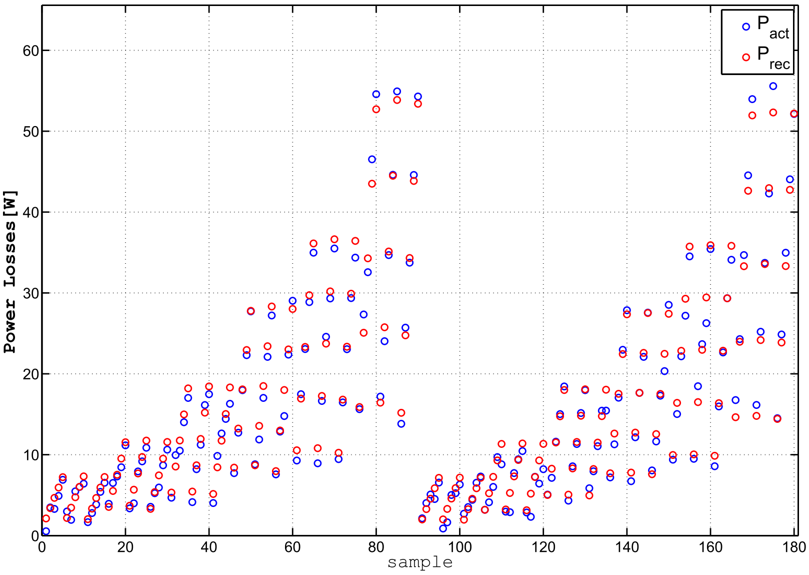


Figure 3.23: actual and reconstructed power losses for the validation set (DUT2)

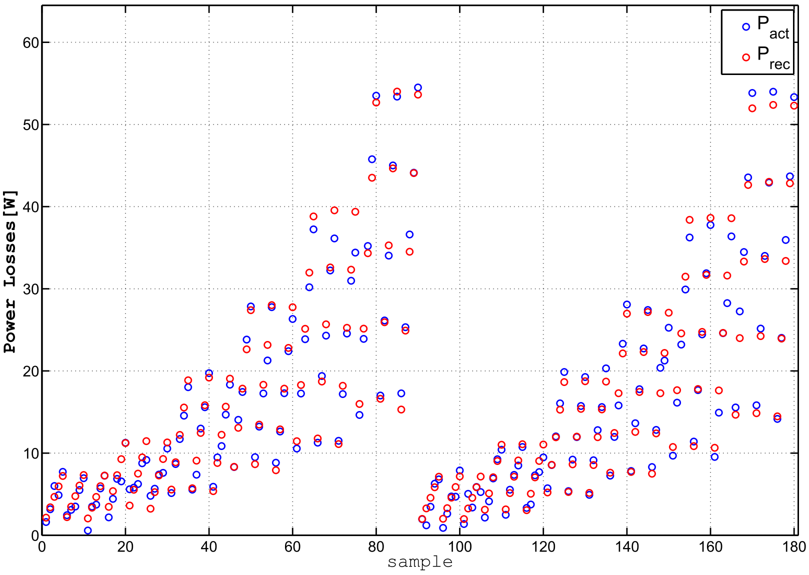


Figure 3.24: actual and reconstructed power losses for the validation set (DUT3)

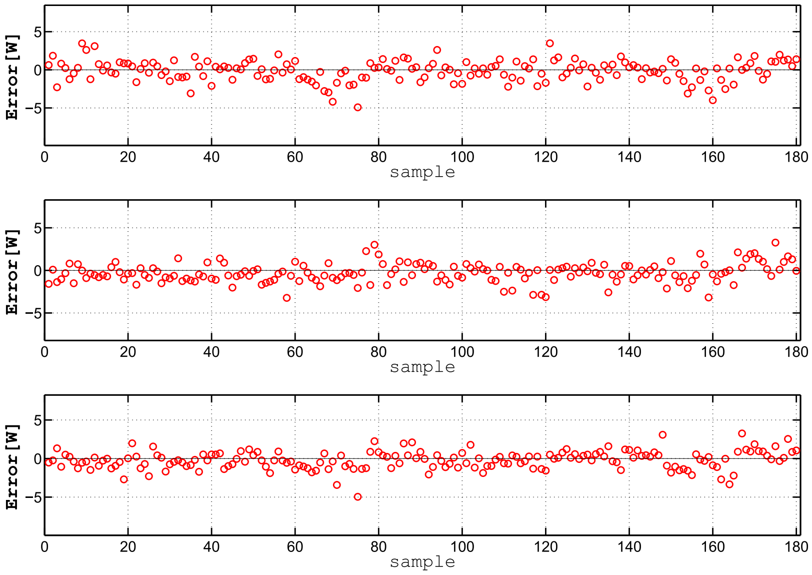


Figure 3.25: power losses errors for the validation set (top: DUT 1, middle : DUT2, bottom: DUT 3)

The reconstructed training and validation set and their relative errors on the new DUT are shown in Figures 3.26, 3.27, 3.28 and 3.29.

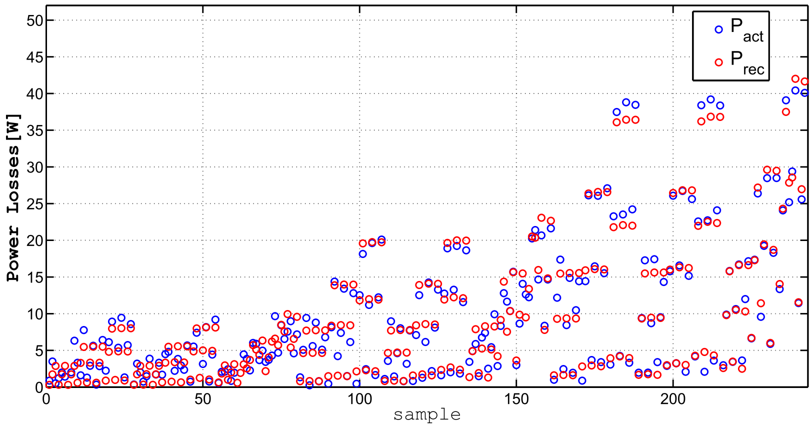


Figure 3.26: Actual and Reconstructed Power Losses on the Training Set (DUT 4)

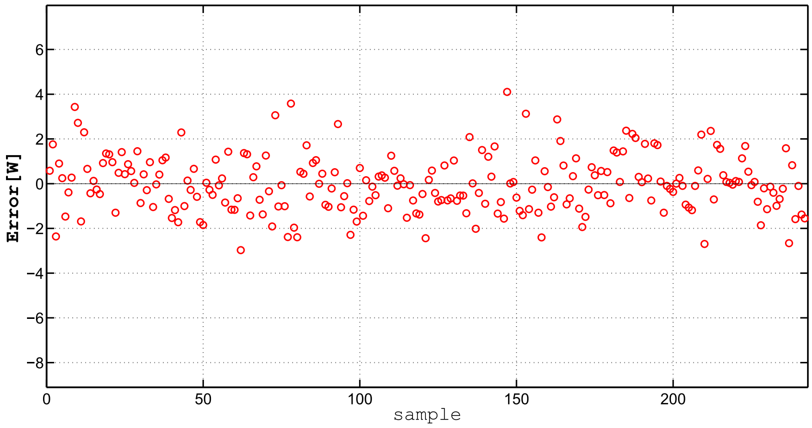


Figure 3.27: Errors on Power Losses on the Training Set (DUT 4)

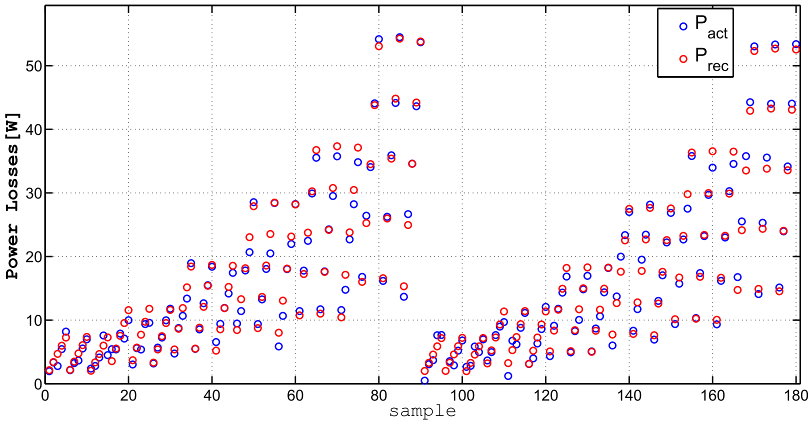


Figure 3.28: Actual and Reconstructed Power Losses on the Validation Set (DUT 4)

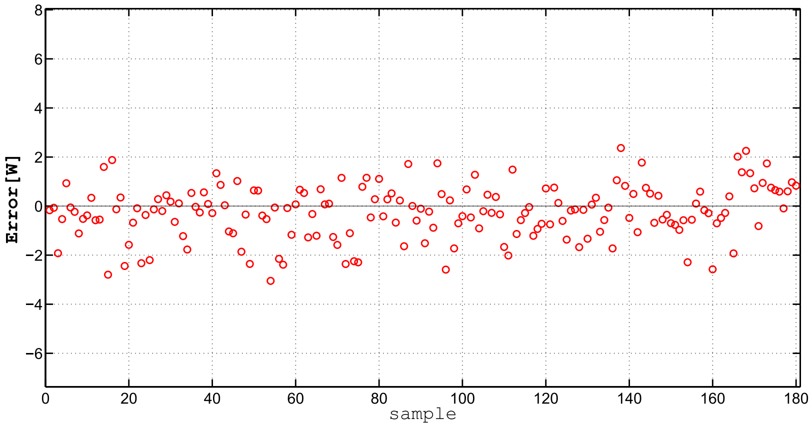


Figure 3.29: Errors on Power Losses on the Training Set (DUT 4)

Chapter 4

Inductor Models

4.1 Case Study: Partially Saturated Inductor

Power inductors strongly impact high-power-density Switch Mode Power Supplies (SMPSs) power losses and size. Saturation is an important characteristic of such components. It is a non-linear phenomenon, which yields a progressive inductance decrease while the average inductor current increases. Ferrite Core Power Inductors (FCPIs) are conventionally selected in SMPSs design when high efficiency and high power density are required. FCPIs exhibit a rapid transition to saturation, when the current exceeds a certain threshold. It is commonly considered good practice to select FCPIs in SMPS design such that they operate in the region of weak saturation (within about 20% inductance drop). Nevertheless, saturation is not a real issue, neither for the inductor nor for the entire converter, if the current ripple, the power losses and the temperature rise fall within limits suitable for the device and for the application. Moreover, some recent studies have highlighted that accepting a partial saturation of FCPIs allows to achieve a reduction of the inductor size and to increase SMPSs power density [50, 51, 52]. To effectively and safely exploit the advantages offered by such unconventional design approach, appropriate saturation models and power losses models are needed for reliable current ripple and power loss prediction in partial saturation.

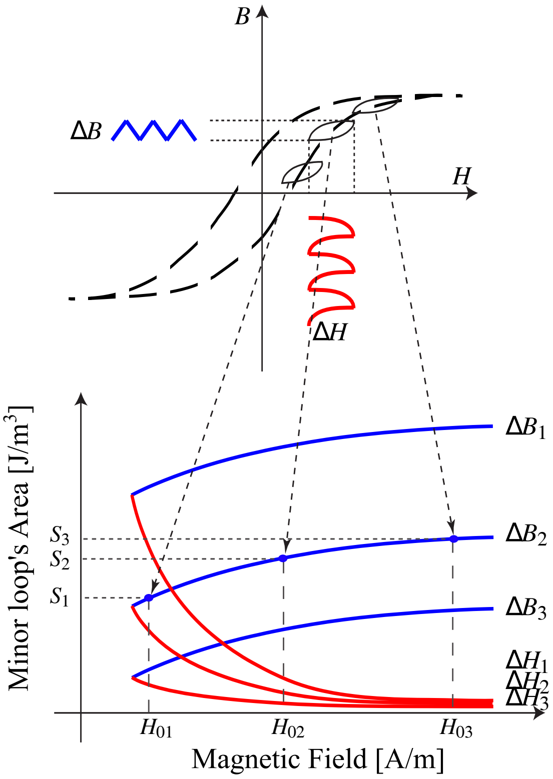


Figure 4.1: Minor Hysteresis loops changing the average current

The models mostly used in partial saturation are based on variants of Steinmetz equation, formulated at different detail levels and taking into account the core loss dependence on the excitation waveform [12, 20, 21, 22, 23, 24, 25, 53]. Some of them have been extended to include inductor core loss dependence on DC bias [54, 55, 56, 28], resulting suitable for power loss estimation in partial saturation. Other models, still including DC bias dependence, have been formulated so as to be independent of the excitation waveform [30, 57]. Conversely, physical reliable models for losses are not viable in SMPSs applications, for several reasons:

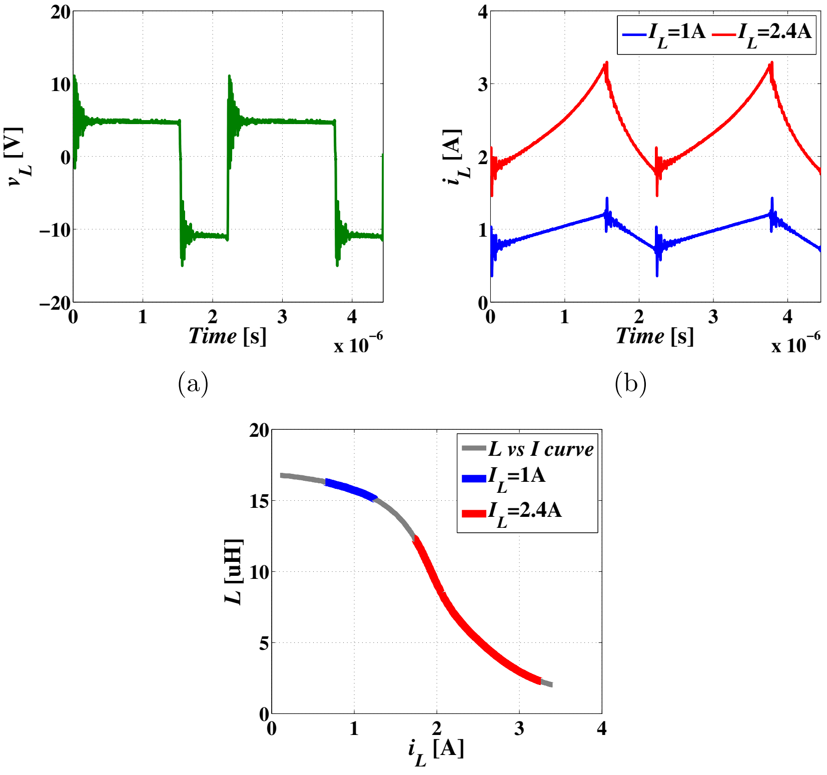
- physical phenomena occurring in square-wave large-signal conditions are difficult to model;
- for SMPSs design validation, loss model at component level is needed, which depends on:
  - the core geometry, which is mostly characterized by sharp edges not allowing easy analytical modeling;
  - the core materials, whose nature and characteristics are not disclosed by inductors manufacturers;
  - the arrangement of windings, where skin and proximity effects are involved as Litz wires are not used for cost reasons, and modeling of relevant losses is impossible as uniform accommodation of layers is not guaranteed by manufacturing processes;
- experimental validation of physical models is not allowed as the effects of different loss contributions cannot be separately measured in inductors for SMPSs applications.

The key point is that the saturation and loss models need to be coherent between them, and both have to be consistent with the large-signal square-wave conditions imposed by SMPSs operation to inductors, including the effect of saturation. Moreover, they have to be based only on measurable quantities like voltage, current and switching frequency imposed by the SMPS operation to the component, rather than on quantities, like the ripple current and the magnetic flux density, which represent the response of the inductor to the square-wave voltage imposed to it.

4.1.1 Inductor Power Loss Measurements

Experimental data needed for GP-based inductor loss model identification have been obtained by means of the MADMIX system [58]. The MADMIX allows to test power magnetics in a wide

range of SMPS operating conditions, providing voltage and current waveforms and the resultant power losses of magnetic devices. Typical inductor voltage and current waveforms obtained by means of such system are shown in Fig. 4.2, for the 18μH Coilcraft MSS7341-183 inductor operating in a buck topology with V_{IN}=16V, D=0.7, f_s=450kHz, at two DC current levels:
I_{OUT}=I_L=1A, involving inductor operation in the weak saturation region (blue line in Fig. 4.2), and I_{OUT}=I_L=2.4A, involving inductor operation in the saturation roll-off region (red line in Fig. 4.2)).



(a)

(b)

(c)

Figure 4.2: MADMIX-based inductor waveforms: (a) voltage and (b) current, and (c) L *vs* I curve with relevant operating regions

Being the inductor a bipole, its power losses are given by the mean value of the instant power absorbed by the device over the switching period T_s:

$$P_{tot} = f_s \int_0^{T_s} v_L\left(t\right) \cdot i_L\left(t\right) dt \tag{4.1}$$

where $v_L\left(t\right)$ and $i_L\left(t\right)$ are the instantaneous inductor voltage and current, respectively. Total losses can be separated into dc term and ac term. The dc term depends on the dc components of inductor voltage (V_L) and current (I_L), and represents the dc winding losses:

$$P_{dc} = V_L \cdot I_L = DCR \cdot I_L^2 = P_{w,dc} \tag{4.2}$$

where DCR represents the dc winding resistance. The ac term depends on the ac components of the inductor voltage $v_{L,ac}(t)$ and current $i_{L,ac}(t)$:

$$P_{ac} = f_s \int_0^{T_s} v_{L,ac}\left(t\right) \cdot i_{L,ac}\left(t\right) dt \tag{4.3}$$

The above power loss terms can be evaluated separately, and the total losses can be subsequently obtained by summing the dc and ac loss contributions. The advantage of this approach is that the power losses are obtained starting from inductor voltage and current measurements realized under SMPS operating conditions. This is the approach adopted by the MADMIX system.

The dc loss term P_{dc} can be estimated from inductor DCR and I_L measurements. The ac loss term P_{ac} can be estimated from the ac voltage and current waveforms over the switching period. In particular, the P_{ac} term contains the magnetic core losses P_c and the ac winding losses $P_{w,ac}$:

$$P_{ac} = P_c + P_{w,ac} \tag{4.4}$$

The term $P_{w,ac}$, relevant to the winding, can be expressed as in (4.5):

$$P_{w,ac} = R_w \cdot I_{ac,rms}^2 \tag{4.5}$$

where $I_{ac,rms}$ represents the rms value of the inductor ripple current. Starting from the inductor voltage and current waveforms, the P_{ac} term can be modeled as a function of the variables representing the operating conditions directly imposed to the inductor by the SMPS, that is:

1. converter switching frequency f_s;

2. average inductor current I_L;

3. equivalent inductor voltage $V_{eq} = V_{L,on}D$,

where D is the converter duty-cycle and $V_{L,on}$ is the average inductor voltage during the on-time interval. The main innovation characterizing the modeling approach proposed in this thesis lies in the use of the two variables V_{eq} and I_L, instead of the ac component of the magnetic flux density B_{ac} or the peak-to-peak ripple current Δi_{Lpp}, used in formulas (1.17) and (1.16). The two variables V_{eq} and I_L, indeed, jointly influence the magnitude of the ripple Δi_{Lpp}, the former because of its impact on the increase of the inductor magnetic flux, the latter because of its influence on inductor saturation. The quantities $\{f_s, V_{eq}, I_L\}$ have been used as the GP input variables, whereas P_{ac} has been assumed as the GP output variable.

4.1.2 Training and Validation Set

A training set has been assembled to run the GP algorithm. Three different Coilcraft inductors have been used during the investigation:

- DUT #1: MSS7341-183;

- DUT #2: MSS1038-183.

- DUT #3: MSS7341-153.

Table 4.1: Reference inductors and their main characteristics

DUT	Part Number	L	DCR_{max}	I_{sat} [A]	
		[μH]	[mΩ]	30% drop	70% drop
#1	MSS7341-183	18	75	1.20	1.62
#2	MSS1038-183	18	65	2.44	3.52
#3	MSS7341-153	15	55	1.78	

Their main datasheet characteristics are compared in Table 4.1.

For each inductor, the operating conditions $\{f_s, V_{IN}, D, I_{OUT}\}$ given in Table 4.2 and relevant to a buck topology have been fixed by means of the MADMIX system. All the possible combinations of such values have been tested. As a result, the quantities of interest $\{f_s, V_{eq}, I_L\}$ and P_{ac} have been measured and used to compose the training set. A wide variation of the load current $I_{OUT} = I_L$ has been imposed for each DUT, in order to cover both weak saturation region and roll-off region of the inductance *vs* current (L *vs* I) curve shown in Fig. 4.3, and guarantee power loss characterization also for inductor operations in partial saturation. Then a validation set has been created to check the GP models predictions fidelity with respect to the data given in Table 4.3.

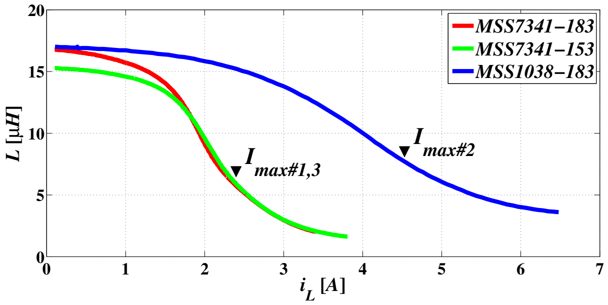


Figure 4.3: Inductors L *vs* I curves and maximum tested load currents ($I_{max\#1} = 2.4A$, $I_{max\#2} = 5A$,$I_{max\#3} = 2.4A$)

Table 4.2: Training set

DUT	f_s[kHz]	V_{in}[V]	$Duty$[%]	I_{out}
1	250, 350, 450	8, 12, 16	0.3, 0.4, 0.5	1: 0.2: 2.4
2	250, 350, 450	8, 12, 16	0.3, 0.4, 0.5	1.5: 0.5: 5
3	250, 350, 450	8, 12, 16	0.3, 0.4, 0.5	1: 0.2: 2.4

Table 4.3: Validation set

DUT	f_s[kHz]	V_{in}[V]	$Duty$[%]	I_{out}
1	300, 400, 500	10, 12, 14	0.4, 0.6	1: 0.2: 2.4
2	300, 400, 500	10, 12, 14	0.4, 0.6	1.5: 0.5: 5
3	300, 400, 500	10, 12, 14	0.4, 0.6	1: 0.2: 2.4

4.1.3 Results

After 10 times the GP algorithm has been executed the models that are repeated more than 2 times are shown in Fig.4.4, with shown the model structures of all the models sufficiently accurate (an error on the training set lower than 10 mW).

The model chosen in term of repeatability and age is $[3, 115]$, compact ac loss model identified by the GP is given in (4.6).

$$P_{ac} = p_0(p_1 + p_2 I_L^{p_3})\frac{V_{eq}^2}{f_s} \tag{4.6}$$

Actually p_1 in 4.6 is not a real parameter, but is considered by GP a coefficients of value 1, because otherwise p_0, p_1 and p_2 could not be identified in a univocal way.

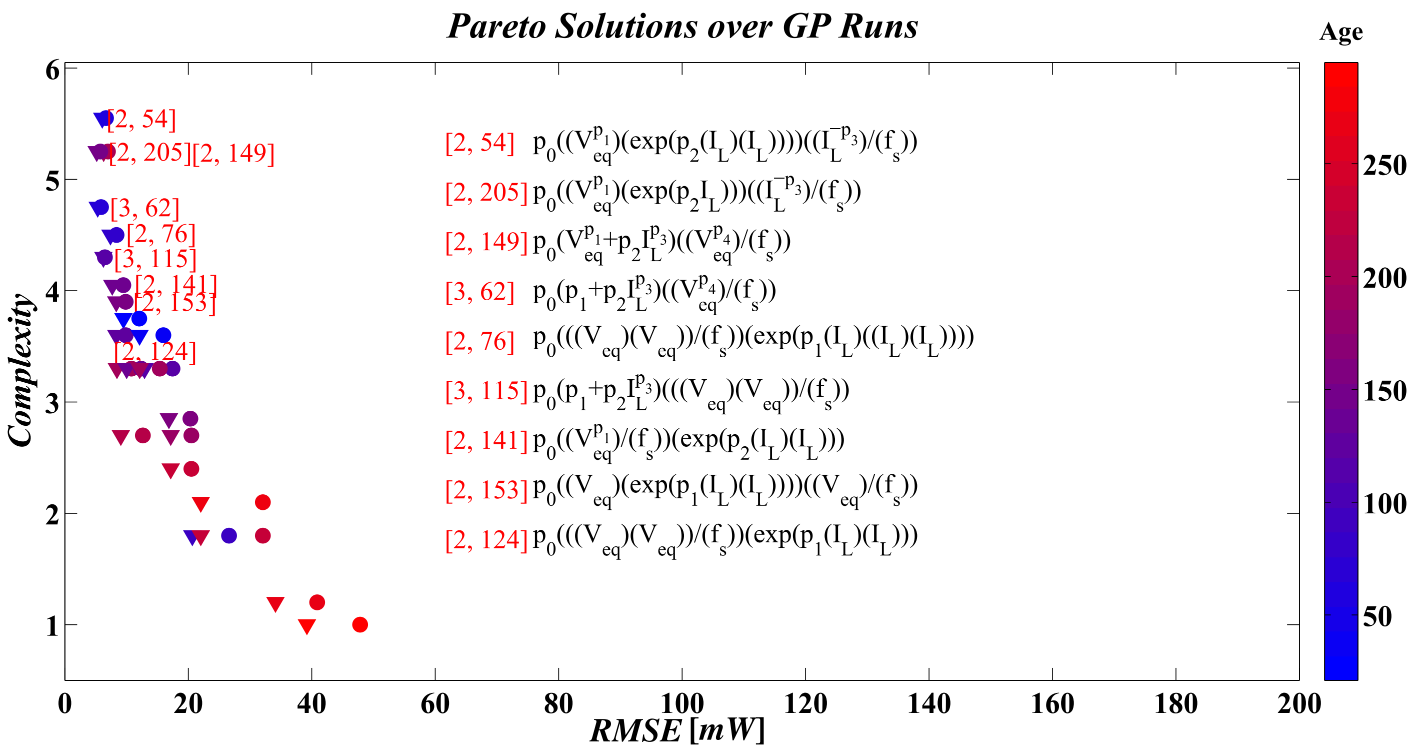


Figure 4.4: Most Repeatable solution on 10 GP runs. Circle:Training Set RMSE. Triangle: Validation Set RMSE

Table 4.4: GP models coefficients

DUT	p_0	p_1	p_2	p_3
1	2.82e+3	1	5.93e-3	6.1
2	1.04e+3	1	8.35e-4	5.3
3	3.33e+3	1	2.81e-4	9

Table 4.5: GP models errors

	$DUT1$	$DUT2$	$DUT3$	TOT
Training Set	5.16	6.9	7.2	6.5
Validation Set	5.3	6.7	6.65	6.25

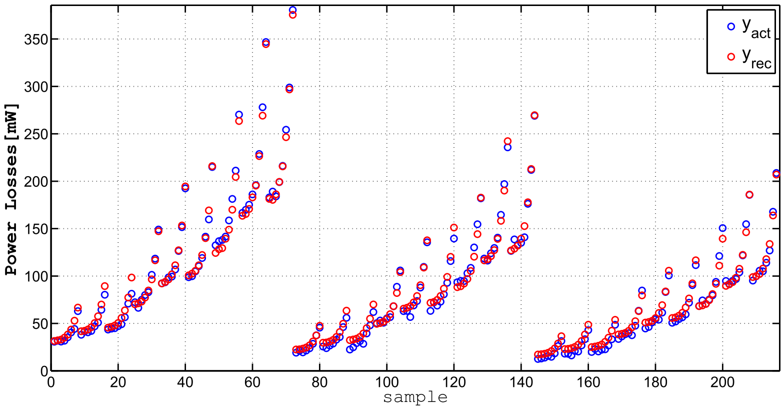


Figure 4.5: Actual and Reconstructed Power Losses for the Training Set (DUT 1)

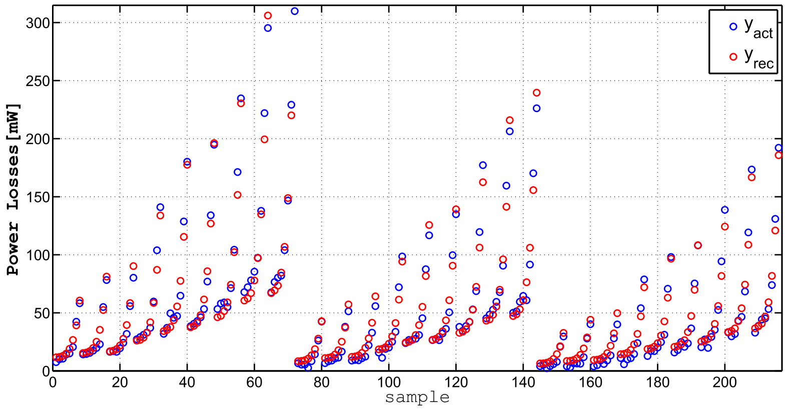


Figure 4.6: Actual and Reconstructed Power Losses for the Training Set (DUT 2)

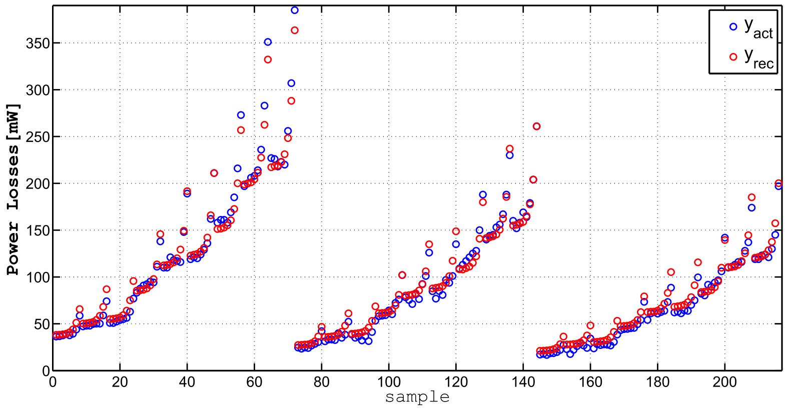


Figure 4.7: Actual and Reconstructed Power Losses for the Training Set (DUT 3)

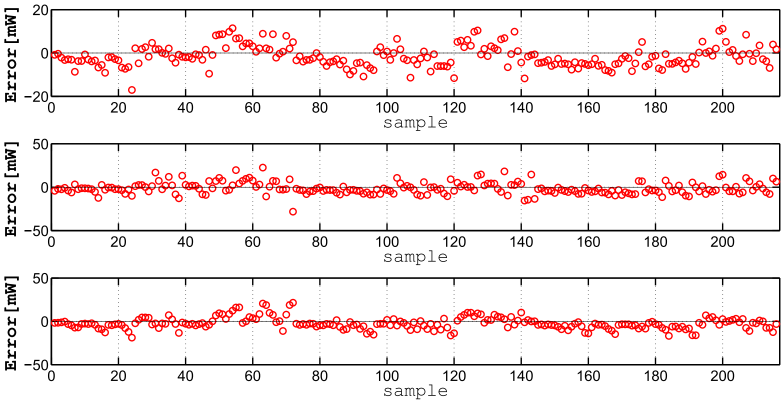


Figure 4.8: Errors on Power Losses for the Training Set (Top: DUT 1, Middle: DUT 2, bottom: DUT 3)

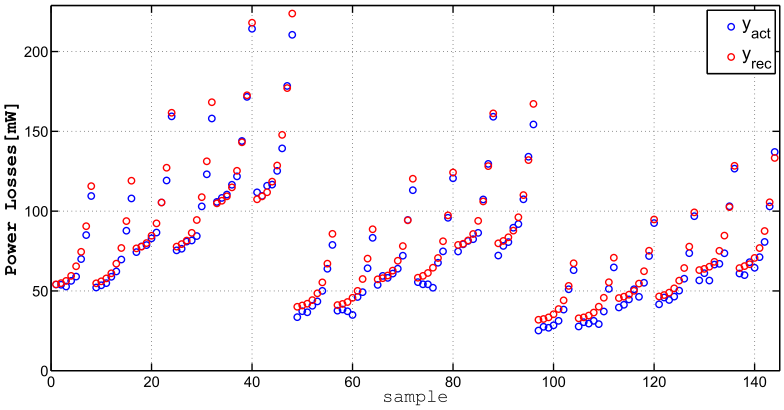


Figure 4.9: Actual and Reconstructed Power Losses for the Validation Set (DUT 1)

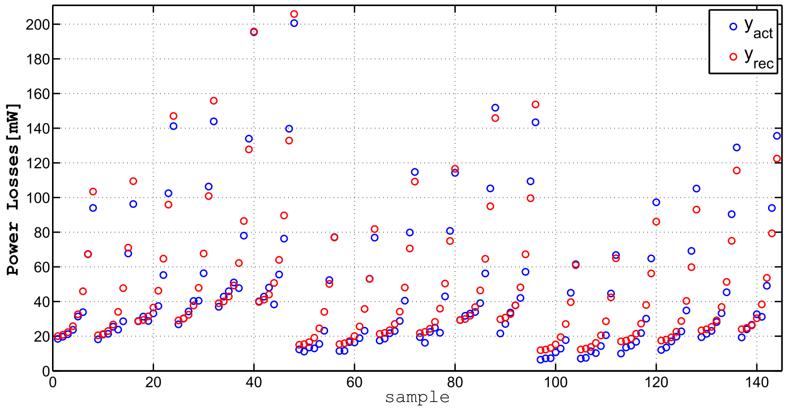


Figure 4.10: Actual and Reconstructed Power Losses for the Validation Set (DUT 2)

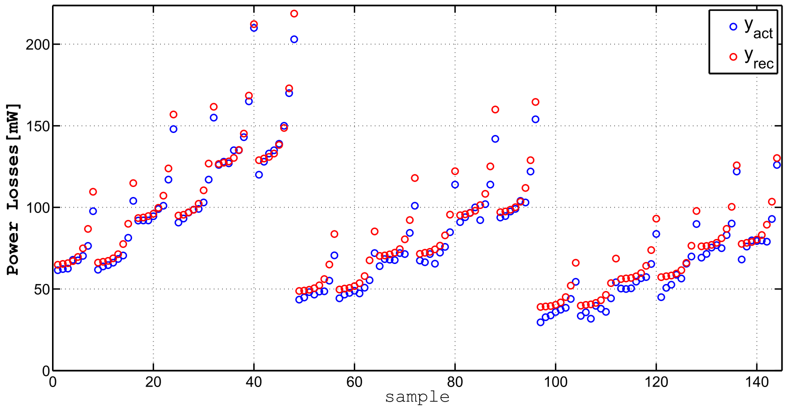


Figure 4.11: Actual and Reconstructed Power Losses for the Validation Set (DUT 3)

4.1.4 Discussion

All the solution are generally composed by two contributions multiplied by each other:

1. a first contribute models principally the losses effect in the weak saturation region:

$$P_{ac,weaksat} = \frac{p_0}{f_s}\left(v_{eq}\right)^2 \tag{4.7}$$

   representing the inductor working in its linear region of the L vs I characteristic.

2. a second contribute dependent on the average inductor currents, that gets higher as bigger is the current. This is represented or by an exponential or a power of the current.

For the parts under investigation, the model (4.6) highlights that the power losses are inversely proportional to the switching frequency. This can be explained as follows: the ac-winding losses are expected to decrease with the increasing frequency, at fixed V_{eq} and I_L values. In fact, as the frequency increases, the peak-to-peak current ripple decreases, resulting in lower $I_{ac,rms}$ value and, consequently, lower ac-winding losses. However, the R_w value could increase at higher frequency due to skin and proximity effects, thus involving higher losses. Thus, the winding losses can increase or decrease with the frequency, depending on the windings arrangement. The smaller peak-to-peak current ripple determined by a higher frequency involves a smaller ΔB magnitude and a smaller area of the H-B loop, which facilitate loss reduction. However, the higher frequency involves that the H-B loop is repeated more times per second, and this plays as a loss increase factor. As a result, the overall frequency impact on core losses could be as increasing as decreasing. For the investigated DUTs, the formula (4.6) confirms that the benefits of a higher frequency are dominant.

The model (4.6) also shows a square law dependence on the applied V_{eq} value. Such result is in agreement with the expected loss trend, as V_{eq} is directly proportional to the peak-to-peak current

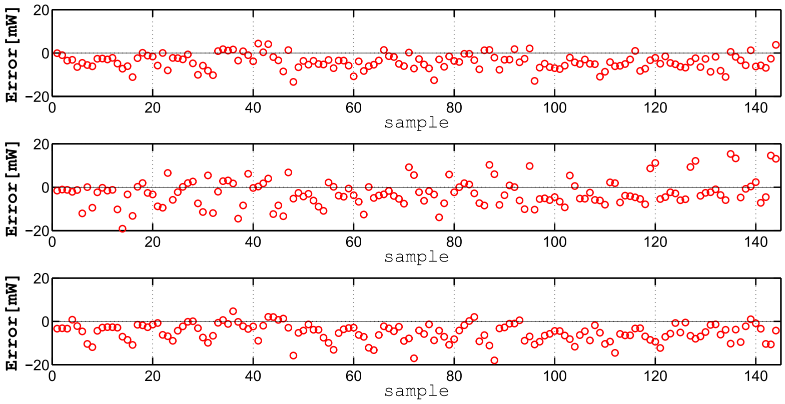


Figure 4.12: Errors on Power Losses for the Validation Set (Top: DUT 1, Middle: DUT 2, bottom: DUT 3)

ripple magnitude. The ac-winding losses are proportional to the square rms ripple current (see eq. 4.5), which, in case of triangular or cusp-like inductor waveform can be fairly well represented by the square of the peak-to-peak ripple magnitude. Moreover, the value V_{eq} is proportional to the magnetic flux density magnitude B_{ac}. Since the core losses increase with B_{ac} more than linearly, the obtained square law dependence on V_{eq} is well justified.

Finally, the dc-bias dependence of ac losses is represented in the model (4.6) by the term $(p_1 + p_2 I_L^{p_3})$. As the average inductor current increases and the roll-off region of the L *vs* I curve is approached, the inductance of the component decreases non-linearly with I_L, resulting in a non-linear increase of the current ripple and of the overall ac losses. In particular, the ac-winding losses increase due to the higher ripple magnitude, whereas the core losses may increase due to the higher H-B loop size, which depends on the peak-to-peak current ripple.

The ac loss predictions for the three DUTs, obtained by means of the model (4.6) and applied over the given validation data set, are shown in Fig. 4.13, compared to the relevant experimental ac losses. Quite good fittings of power losses are ensured for all the analyzed operating conditions, even if the frequency, duty-cycle, input voltage and output current conditions are different from the training set conditions. Power loss predictions obtained by using classical formulas (4.5) and (1.17) for ac-winding and core loss evaluation are also shown in Fig. 4.13, which provide meaningful results within the weak-saturation region only (low-power).

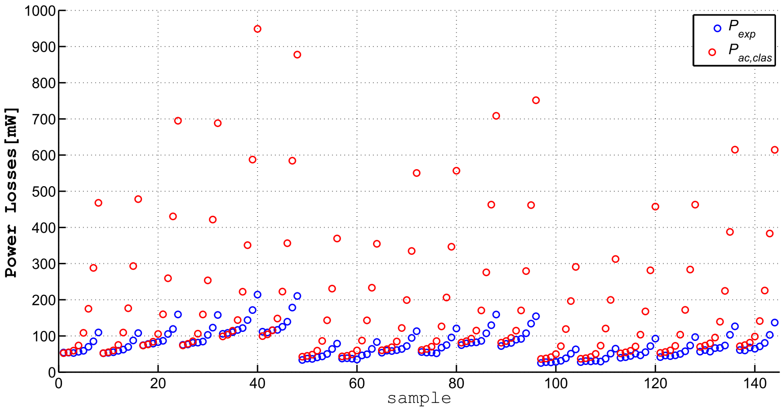


Figure 4.13: Actual and Reconstructed Power Losses with classical model on the Validation Set (DUT1)

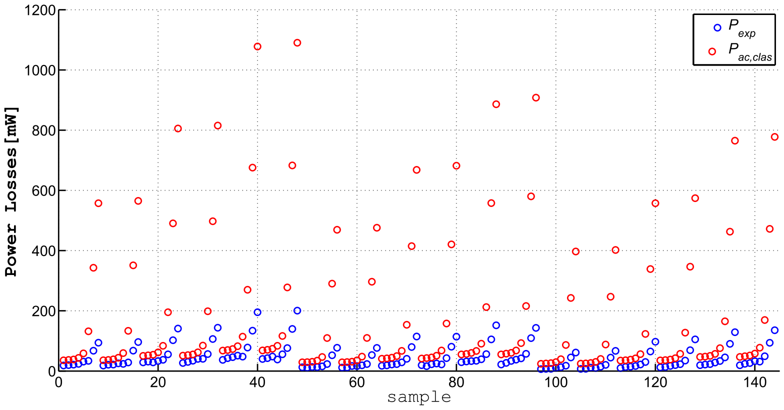


Figure 4.14: Actual and Reconstructed Power Losses with classical model on the Validation Set (DUT2)

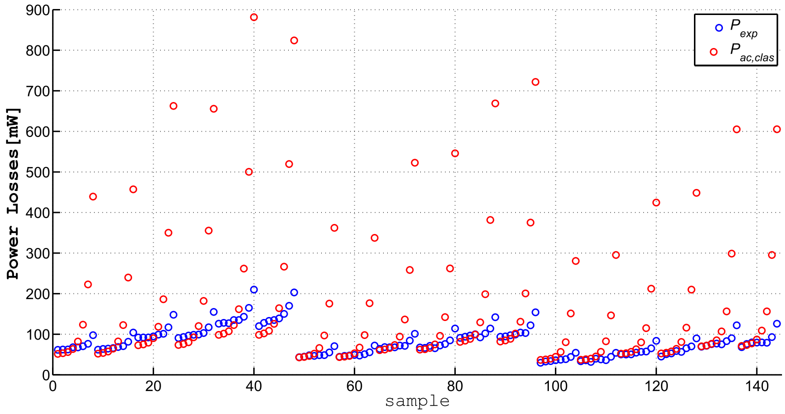


Figure 4.15: Actual and Reconstructed Power Losses with classical model on the Validation Set (DUT3)

As expected, when the partial saturation is approached, classical formulas do not assure a reliable power loss prediction. Strong overestimation of ac losses has been obtained, predicting simulated losses nearly four times (DUT #1) and five times (DUT #2) the experimental ones. In fact, if $\Delta i_{Lpp}/\Delta H$ values are maintained constant, core losses are expected to decrease with increasing current [27], whereas formula (1.17) does not include dc-bias dependence and, as a result, overestimates the core losses. In particular, K_2 coefficient in formula (1.17) depends on the magnetic material permeability, or, equivalently, on the inductance value, as given in (4.8):

$$K_2 = \frac{L}{2nA_e} \tag{4.8}$$

Such value may become much de-rated when the inductor operates in the roll-off region. In order to obtain a reliable core loss estimation, a correct K_2 value should be adopted, properly de-rated on the basis of the achieved inductance value. For example, if the inductor works with the average current in the middle of its saturation curve (at 50% of its nominal inductance), the adopted K_2 value should be half the nominal value – the obtained core losses would result four to eight times lower (Y values typically going from 2 to 3), compared to the losses obtained by using the nominal K_2 value. The results of classical formulas shown in Fig. 4.13,4.14 and 4.15 have been obtained by using the nominal $\{K_1, K_2, X, Y\}$ values, provided by the manufacturer.

The experimental validation tests prove that the GP-based model provides a much higher accuracy compared to classical Steinmetz models in the correct prediction of the impact of inductors saturation on power losses. Inductor ac loss models obtained by means of the GP-MOO technique provide compact and accurate behavioral models for ac loss estimation of the analyzed ferrite inductors.

The GP-MOO has huge potentiality in the power inductors manufacturing industry and in the SMPS design. Indeed, it can be used to better characterize all types of inductors, with any type of core materials and geometry, and to provide SMPS designers reliable information allowing to reduce the size of power converters.

Conclusions and future developments

In this Thesis a method to identify models that can be used to predict power losses in Power Electronics application has been described and successfully applied to identify both semiconductor devices and magnetic components.

The obtained models are simple, accurately fit the measured experimental data, and have input variables that can be easily measured and estimated, differently to other models that require the knowledge of physical parameters to be used, often not included in the data-sheets of the devices. Moreover the solutions are repeatable and the GP tree's structure adopted guarantees a physical meaning of the models. The fact that parametric functions returned comport that once a new device of the same family has been characterized measuring its training set, its model coefficients can be easily calculated with simple optimization techniques.

The fact that instruments to measure automatically inductor power losses already exist makes the possibility to characterize magnetic components very easy and fast, and consequentially this kind of approach in future can be easily applied to them, allowing also an easy investigation of partial saturation effects.

For semiconductor devices automatic instrumentation able to measure their power losses does not exist yet, and surely it is a more difficult task, due to the high sampling frequencies needed to measure the switching losses and to their position in critical paths, but this work can be an incentive to investigate accurate methods and instruments to measure their losses, that anyway can also been measured with the experimental setup described in Chapter 3, or with a difference between input and output power measured by watt-meters or using thermal methods.

This approach can have different practical applications: first it is possible to construct a database with all the devices suited for a specific applications, having each one a different set of coefficients describing its behavior, and this can be used to decide which is the best device once specific circuital functional parameters like switching frequency have been set; otherwise if a device has been chosen the best circuital functional parameters can be easily found in order to minimize the power losses; another possible application is to have a real time prediction of the device behavior and power losses, measuring input variables during the device normal working, in fact the code can be easily loaded on an embedded system calculating the power losses using a model and so implementing appropriate control techniques in order to avoid thermal damages.

Some considerations must be done about the thermal analysis. In this thesis, the temperatures have not been considered as input variables, but their effect has been included considering the power losses measured under thermal balance condition. By the way, the approach described in this thesis can be adopted to construct thermal models. In fact, the device case temperature (that is easily measurable) can be used as output variable instead of the power losses.

Finally, more accurate models could be constructed, considering the room temperature as input variable. In this case, the test should be repeated at different opportune room temperatures. Unfortunately, such approach would require more complicated and expensive experimental setups, able to regulate and maintain constant the room temperature.

Appendix A

Multi objective Optimization

Considering the case of two objectives to be minimized at the same time, any two models can be chosen from the feasible objective space and compared.

In some cases, given two models, it is possible to say that one solution is better than the other in both objective, for example a model more complicated and with a greater error than the other one is certainly worse. In this case it can be said that the first solution is *dominant* on the other one.

In other cases it is possible that one model is better than the other in one objective, but it is worse in the second objective. When this happens the two models are *non-dominated* solutions, and one cannot say which of the two models is better than the one.

For example in Figure A.1 the solution E is dominated both by B and C, that have all the two fitness lower than E. The solution A, B, C and D are not dominated by any other solution, and are called *Pareto Optimal Solutions*, and so they belong to the *Pareto Front*, represented in figure is by the curve joining all the Pareto Optimal models. In the presence of multiple Pareto-optimal solutions, it is difficult to prefer one solution over the other without any further information about the problem. Accordingly, all Pareto-optimal

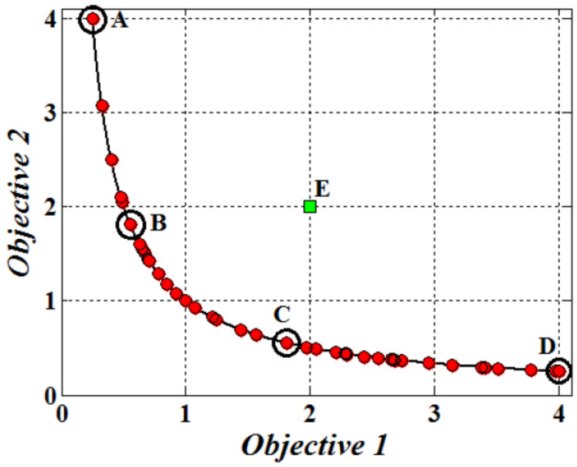


Figure A.1: Four Pareto-optimal solutions and one non-optimal solution [59]

solutions are equally important to the user. Hence, in the light of the ideal approach, it is important to find as many Pareto-optimal solutions as possible in a problem. Thus, it is possible to define two goals in a multi-objective optimization:

1. To find a set of solutions as close as possible to the Pareto-optimal front.

2. To find a set of solutions as different as possible.

In addition to being converged close to the Pareto-optimal front, they must also be sparsely spaced in the Pareto-optimal region. Only with a diverse set of solutions, can one be assured of having a good set of trade-off solutions among objectives.

Solutions A, B, C and D constitute a *non dominated set*, that is composed by all the elements that fulfill the following conditions :

1. Any two solutions of the *non dominated set* must be non-dominated with respect to each other.

2. Any solution not belonging to the *non dominated set* is dominated by at least one member of *non dominated set*

A.1 Non-Dominated Sorting of a Population

In the case of MOOP there is not a single fitness function, and so during the tournament it is sometimes impossible to say if a function is better than the other. To give a value to the models it is necessary to classify the entire population in different levels of non-domination levels, that means that to each model of the GP populations is given a level of non-domination. So if a model has a level of non-domination greater than one other it is surely better.

The best non-dominated solutions of the Pareto front are called *non-dominated solutions of level 1.* The next level of non-domination, that is *non-dominated solutions of level 2*, is obtained finding the non-dominated solution of the population temporarily disregarding the best non-dominated models from the population.

To find the *non-dominated solutions of level 3* the *non-dominated solutions of level 1 and 2* are temporally eliminated by the population and the best non-dominated element are found and so on, until a rank is given to all the element of the population.

For example in Fig. A.2 all the non-domination levels are shown.

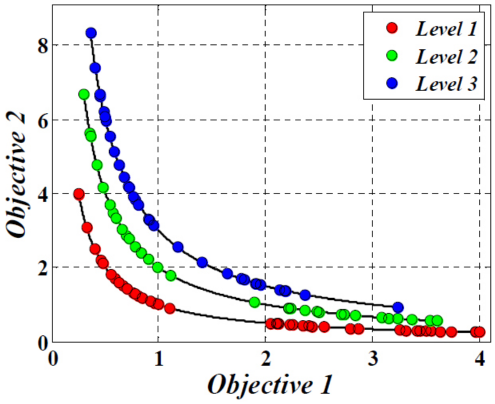


Figure A.2: Non-Dominated Sorting of a Population [59]

A.2 Elitist Non-Dominated Sorting Genetic Algorithm

Elitist Non-Dominated Sorting Genetic Algorithm (NSGA II) is the most used MOOP algorithm. Elitist means that the best elements of the population are preserved and directly carried over to the next generations,so that the fitness of the best elements can not get worse but can only improve during the generations, enhancing the probability of creating better offspring.

As shown in figure A.3, in NSGA-II the population, composed by N elements, is initially randomly created then it is sorted into different non-domination levels, giving a rank equal to its fitness level to each model, and an offspring of N element is created by recombination and mutation operators applied on the element of the population. Then the two population are merged to form a population of size 2N.

Then, a non-dominated sorting is used to classify the entire population, and the new population is filled by solutions of different non-dominated fronts, one at a time, starting by the best non-dominated front and continuing with models of the second non-dominated front, followed by the third non-dominated front, and so on, to form a new population of N elements. All the fronts that can not be copied in the new generation are deleted, while only some of the elements of the last allowed front can enter in the new generation.

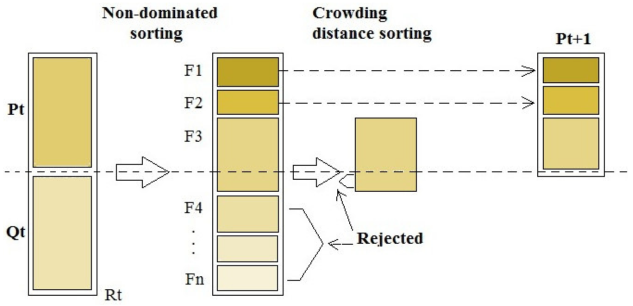


Figure A.3: Schematic of the NSGA-II procedure [59]

To choose which solutions enter in the new populations, considering that a good spread among the solutions has to be ensured , a *crowding factor* is used to determine which models go to the next generation, that is greater for the solutions in a crowded zone and lower for isolated solutions.

The procedure then is repeated with the new population of N elements until the maximum number of generations is reached.

In the last generations it is probable that the population in the first non-dominated front is grater than N, for this reason, adopting the *crowding factor* the continuation of this algorithm will ensure a better spread among the solutions.

The *crowding factor* represents an estimate of the density of solutions surrounding a particular model in the population, and is calculated as the average distance of two solutions on the either side of solution along each of the objectives. As shown Fig.A.4 it is the perimeter of the cuboid formed by using the nearest neighbors as the vertices.

In the models at the boundary (A and D in Fig. A.4), the crowding factor is assumed to be infinite.

In the tournament, to decide to which model apply the genetic operator to generate the offspring, a solution i is considered better than another solution j if any of the following conditions are true:

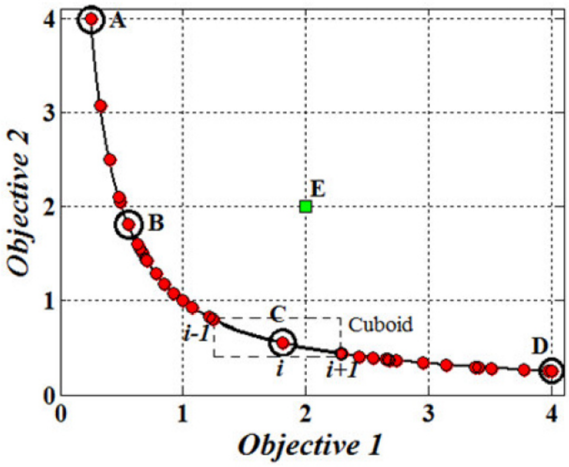


Figure A.4: The crowding distance calculation [59]

1. The solution i has a better rank than solution j
2. The two solutions have the same rank but the solution i has a better (greater) crowding distance than solution j

Bibliography

[1] R. Erickson and D. Maksimovic, *Fundamentals of Power Electronics.* Power electronics, Springer US, 2001.

[2] "STW45NM50 datasheet, ST."

[3] S. Davis, *Power Management.* Penton Media, Inc, 2016.

[4] "NGTB25N120IHLW-D datasheet, ON semiconductor."

[5] S. Maniktala, *Switching Power Supplies A - Z, Second Edition 2nd Edition.* Newnes; 2 edition (April 18, 2012), 2012.

[6] B. J. Baliga, "Power semiconductor device figure of merit for high-frequency applications," *IEEE Electron Device Letters*, vol. 10, pp. 455–457, Oct 1989.

[7] H. Sarnago, O. Lucia, A. Medìano, and J. M. Burdío, "Analytical model of the half-bridge series resonant inverter for improved power conversion efficiency and performance," *IEEE Transactions on Power Electronics*, vol. 30, pp. 4128–4143, Aug 2015.

[8] K. Sheng, B. W. Williams, and S. J. Finney, "A review of igbt models," *IEEE Transactions on Power Electronics*, vol. 15, pp. 1250–1266, Nov 2000.

[9] SYNOPSYS, "HSPICE MOSFET Models Manual." `http://www.ece.tamu.edu/~spalermo/ecen474/hspice_mosfet.pdf`, 2005. [Online;].

[10] G. D. Capua and N. Femia, “A versatile method for mosfet commutation analysis in switching power converter design,” *IEEE Transactions on Power Electronics*, vol. 29, pp. 920–935, Feb 2014.

[11] I. Baraia, J. Galarza, J. A. Barrena, and J. M. Canales, “An igbt behavioural model based on curve fitting methods,” in *2008 IEEE Power Electronics Specialists Conference*, pp. 1971–1977, June 2008.

[12] C. P. Steinmetz, “On the Law of Hysteresis,” *Proceedings of the IEEE*, vol. 72, pp. 197–221, Feb 1984.

[13] Pulse Electronics, *P1172 datasheet*, 2007. online.

[14] A. M. Urling, V. A. Niemela, G. R. Skutt, T. G. Wilson, “Characterizing High-Frequency Effects in Transformer Windings-a Guide to Several Significant Articles,” in *Proc. of 1989 Fourth Annual IEEE Applied Power Electronics Conference and Exposition (APEC)*, pp. 373–385, Mar 1989.

[15] C. R. Sullivan, “Computationally Efficient Winding Loss Calculation with Multiple Windings, Arbitrary Waveforms, and Two-Dimensional or Three-Dimensional Field Geometry,” *IEEE Transactions on Power Electronics*, vol. 16, pp. 142–150, Jan 2001.

[16] Jiankun Hu, C. R. Sullivan, “AC Resistance of Planar Power Inductors and the Quasidistributed Gap Technique,” *IEEE Transactions on Power Electronics*, vol. 16, pp. 558–567, Jul 2001.

[17] M. Sippola, R. E. Sepponen, “Accurate Prediction of High-Frequency Power-Transformer Losses and Temperature Rise,” *IEEE Transactions on Power Electronics*, vol. 17, pp. 835–847, Sep 2002.

[18] W. A. Roshen, “A Practical, Accurate and Very General Core Loss Model for Nonsinusoidal Waveforms,” *IEEE Transactions on Power Electronics*, vol. 22, pp. 30–40, Jan 2007.

[19] T. Hatakeyama, K. Onda, “Core Loss Estimation of Various Materials Magnetized With the Symmetrical/Asymmetrical Rectangular Voltage,” *IEEE Transactions on Power Electronics*, vol. 29, pp. 6628–6635, Dec 2014.

[20] J. Reinert, A. Brockmeyer, R. W. A. A. De Doncker, “Calculation of Losses in Ferro- and Ferrimagnetic Materials Based on the Modified Steinmetz Equation,” *IEEE Transactions on Industry Applications*, vol. 37, pp. 1055–1061, Jul 2001.

[21] Jieli Li, T. Abdallah, C. R. Sullivan, “Improved Calculation of Core Loss with Nonsinusoidal Waveforms,” in *Conference Record of the 2001 IEEE Thirty-Sixth IAS Annual Meeting Industry Applications Conference*, vol. 4, pp. 2203–2210 vol.4, Sept 2001.

[22] K. Venkatachalam, C. R. Sullivan, T. Abdallah, H. Tacca, “Accurate Prediction of Ferrite Core Loss with Nonsinusoidal Waveforms Using Only Steinmetz Parameters,” in *Proc. of 2002 IEEE Workshop on Computers in Power Electronics*, pp. 36–41, June 2002.

[23] A. Van den Bossche, V. Cekov Vaichev, G. Bogomilov Georgiev, “Measurement and Loss Model of Ferrites with Non-Sinusoidal Waveforms,” in *2004 IEEE 35th Annual Power Electronics Specialists Conference (PESC)*, vol. 6, pp. 4814–4818 Vol.6, June 2004.

[24] S. Barg, K. Ammous, M. Hanen, A. Ammous, “An Improved Empirical Formulation for Magnetic Core Losses Estimation Under Non-Sinusoidal Induction,” *IEEE Transactions on Power Electronics*, vol. PP, no. 99, pp. 1–1, 2016.

[25] J. Muhlethaler, J. Biela, J. W. Kolar, A. Ecklebe, “Improved Core-Loss Calculation for Magnetic Components Employed in Power Electronic Systems,” *IEEE Transactions on Power Electronics*, vol. 27, pp. 964–973, Feb 2012.

[26] J. W. Kolar, J. Muhlethaler, J. Biela, A. Ecklebe, "Core Losses Under the DC Bias Condition Based on Steinmetz Parameters," *IEEE Transactions on Power Electronics*, vol. 27, pp. 953–963, Feb 2012.

[27] K. Terashima, K. Wada, T. Shimizu, T. Nakazawa, K. Ishii, Y. Hayashi, "Evaluation of the Iron Loss of an Inductor Based on Dynamic Minor Characteristics," in *2007 European Conference on Power Electronics and Applications*, pp. 1–8, Sept 2007.

[28] H. Kosai, Z. Turgut, J. Scofield, "Experimental Investigation of DC-Bias Related Core Losses in a Boost Inductor," *IEEE Transactions on Magnetics*, vol. 49, pp. 4168–4171, July 2013.

[29] M. Mu, F. Zheng, Q. Li, F. C. Lee, "Finite Element Analysis of Inductor Core Loss Under DC Bias Conditions," *IEEE Transactions on Power Electronics*, vol. 28, pp. 4414–4421, Sept 2013.

[30] A. Ruszczyk, K. Sokalski, J. Szczyglows, "Scaling in Modeling of Core Losses in Soft Magnetic Materials Exposed to Nonsinusoidal Flux Waveforms and DC Bias," in *21st Soft Magnetic Materials Conference*, Aug 2013.

[31] O.Nells, *Nonlinear System Identification.* Springer Berlin Heidelberg, 2001.

[32] L. Ljung, ed., *System Identification (2Nd Ed.): Theory for the User.* Upper Saddle River, NJ, USA: Prentice Hall PTR, 1999.

[33] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, no. 5, pp. 359 – 366, 1989.

[34] J. Park and I. W. Sandberg, "Universal approximation using radial-basis-function networks," *Neural Computation*, vol. 3, pp. 246–257, June 1991.

[35] L. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338 – 353, 1965.

[36] B. Kosko, "Fuzzy systems as universal approximators," *IEEE Transactions on Computers*, vol. 43, pp. 1329–1333, Nov 1994.

[37] J. R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection.* Cambridge, MA, USA: MIT Press, 1992.

[38] R. Poli, W. B. Langdon, N. F. Mcphee, and J. R. Koza, "Genetic programming an introductory tutorial and a survey of techniques and applications," tech. rep., 2007.

[39] J. Nocedal and S. J. Wright, *Numerical optimization.* Springer Series in Operations Research and Financial Engineering, Berlin: Springer, 2006. NEOS guide http://www-fp.mcs.anl.gov/otc/Guide/.

[40] J. Madár, J. Abonyi, and F. Szeifert, "Genetic programming for the identification of nonlinear input–output models," *Industrial & Engineering Chemistry Research*, vol. 44, no. 9, pp. 3178–3186, 2005.

[41] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C (2Nd Ed.): The Art of Scientific Computing.* New York, NY, USA: Cambridge University Press, 1992.

[42] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, pp. 182–197, Apr 2002.

[43] Y.-S. Kwon, S.-B. Yoo, and D.-S. Hyun, "Half-bridge series resonant inverter for induction heating applications with load-adaptive pfm control strategy," in *Applied Power Electronics*

Conference and Exposition, 1999. APEC '99. Fourteenth Annual, vol. 1, pp. 575–581 vol.1, Mar 1999.

[44] D. Puyal, C. Bernal, J. M. Burdio, I. Millan, and J. Acero, "A new dynamic electrical model of domestic induction heating loads," in *2008 Twenty-Third Annual IEEE Applied Power Electronics Conference and Exposition*, pp. 409–414, Feb 2008.

[45] J. Acero, J. M. Burdio, L. A. Barragan, D. Navarro, R. Alonso, J. R. Garcia, F. Monterde, P. Hernandez, S. Llorente, and I. Garde, "The domestic induction heating appliance: An overview of recent research," in *2008 Twenty-Third Annual IEEE Applied Power Electronics Conference and Exposition*, pp. 651–657, Feb 2008.

[46] J. Acero, R. Alonso, L. A. Barragan, C. Carretero, O. Lucia, I. Millan, and J. M. Burdio, "Domestic induction heating impedance modeling including windings, load, and ferrite substrate," in *2009 13th European Conference on Power Electronics and Applications*, pp. 1–10, Sept 2009.

[47] C. Carretero, J. Acero, R. Alonso, J. M. Burdio, and F. Monterde, "Temperature influence on equivalent impedance and efficiency of inductor systems for domestic induction heating appliances," in *APEC 07 - Twenty-Second Annual IEEE Applied Power Electronics Conference and Exposition*, pp. 1233–1239, Feb 2007.

[48] Richard Dunipace, *Test Saturation Voltage to Achieve High Efficiency.* Power Electronics Technology, 2008. online.

[49] EPC, *Device Models.* http://epc-co.com/epc/DesignSupport/DeviceModels.aspx,online.

[50] L. Milner, G.A. Rincon-Mora, "Small Saturating Inductors for More Compact Switching Power Supplies," *IEEJ Trans. on Electrical and Electronic Engineering*, vol. 7, pp. 69–73, Jan 2012.

[51] C. Wilhelm, D. Kranzer, B. Burger, "Development of a Highly Compact and Efficient Solar Inverter with Silicon Carbide Transistors," in *2010 6th International Conference on Integrated Power Electronics Systems (CIPS)*, pp. 1–6, March 2010.

[52] A. Stadler, T. Stolzke, C. Gulden, "Nonlinear Power Inductors for Large Current Crest Factors," in *proc. of 2012 European Power Conversion Intelligent Motion Conference (PCIM 2012)*, pp. 1548–1553, May 2012.

[53] A. P. Van den Bossche, D. M. Van de Sype, V. C. Valchev, "Ferrite Loss Measurement and Models in Half Bridge and Full Bridge Waveforms," in *2005 IEEE 36th Power Electronics Specialists Conference*, pp. 1535–1539, June 2005.

[54] A. Brockmeyer and J. Paulus-Neues, "Frequency dependence of the ferrite-loss increase caused by premagnetization," in *Applied Power Electronics Conference and Exposition, 1997. APEC '97 Conference Proceedings 1997., Twelfth Annual*, vol. 1, pp. 375–380 vol.1, Feb 1997.

[55] V. C. Valchev, A. P. Van den Bossche, D. M. Van de Sype, "Ferrite Losses of Cores with Square Wave Voltage and DC Bias," in *31st Annual Conference of IEEE Industrial Electronics Society, (IECON)*, pp. 5 pp.–, Nov 2005.

[56] Yunyu Tang, Fan Zhu, Jiong Ma, Hao Ma, "A Practical Core Loss Calculation Method of Filter Inductors in PWM Inverters Based on the Modified Steinmetz Equation," in *2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*, pp. 386–391, June 2014.

[57] J. Cale, S. D. Sudhoff, R. R. Chan, "A Field-Extrema Hysteresis Loss Model for High-Frequency Ferrimagnetic Materials," *IEEE Transactions on Magnetics*, vol. 44, pp. 1728–1736, July 2008.

[58] M. Wens, J. Thone, “MADMIX: The Standard for Measuring SMPS Inductors,” *Bodo’s Power Systems*, pp. 52–54, April 2015.

[59] K. Deb and D. Kalyanmoy, *Multi-Objective Optimization Using Evolutionary Algorithms.* New York, NY, USA: John Wiley & Sons, Inc., 2001.