



UNIVERSITÀ DI PARMA

ARCHIVIO DELLA RICERCA

University of Parma Research Repository

A Neural Network based Approach to Simulate Electro-thermal Device Interaction in SPICE Environment

This is the peer reviewed version of the following article:

Original

A Neural Network based Approach to Simulate Electro-thermal Device Interaction in SPICE Environment / Chiozzi, D.; Bernardoni, M.; Delmonte, N.; Cova, P.. - In: IEEE TRANSACTIONS ON POWER ELECTRONICS. - ISSN 0885-8993. - 34:5(2019), pp. 4703-4710. [10.1109/TPEL.2018.2863186]

Availability:

This version is available at: 11381/2850724 since: 2021-10-18T06:53:33Z

Publisher:

Institute of Electrical and Electronics Engineers Inc.

Published

DOI:10.1109/TPEL.2018.2863186

Terms of use:

Anyone can freely access the full text of works made available as "Open Access". Works made available

Publisher copyright

note finali coverpage

(Article begins on next page)

A Neural Network based Approach to Simulate Electro-thermal Device Interaction in SPICE Environment

Diego Chiozzi[†], Mirko Bernardoni^{*}, Nicola Delmonte[†], Paolo Cova[†]
^{*}Infinion Technologies Austria, Automotive Division
[†]University of Parma, Dipartimento di Ingegneria dell'Informazione

Abstract—An innovative modelling methodology for the simulation of electro-thermal interaction in power devices, based on Neural Networks (NNs), is shown. The suitability of NNs in modelling the complicated non-linear, temperature dependent characteristic that power electronics devices feature, is shown. The proposed methodology is particularly suited to be implemented in electrical simulators. The approach can be divided in two parallel steps: firstly, NNs are used to describe the complex, highly non-linear electro-thermal characteristic of the considered device; secondly, a non-linear RC-based thermal model is generated, with a method published in a previous work. These two sub-systems are coupled together in order to achieve a self-consistent electro-thermal model. The modelling results are validated against experiments with very satisfactory results. The technique is explained in detail; advantages and limitations of the method are then discussed.

Keywords: Electro-thermal modelling, Neural Networks, Simulation, Compact modelling

This work has not been submitted elsewhere.

I. INTRODUCTION

Electro-thermal simulation is well known to be the most appropriate way to simulate the dynamic of an electron device under operating conditions; it is usually implemented in an electrical simulator by coupling two subsystems, the first describing the thermal behaviour of the device (which in general depends on technology, packaging, and assembly), the second the electro-thermal behaviour of the device itself. Fig. 1 shows the system architecture of the simulation setup.

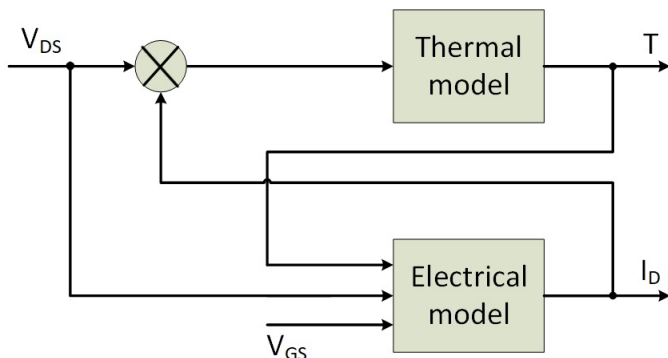


Fig. 1. Block diagram of a fully-coupled electro-thermal MOSFET model.

Thermal models are usually made by combinations of thermal resistors and capacitors which describe the heat propagation through the structure. It is possible to classify these compact models (also referred to as lumped since heat propagation is actually modelled by lumped elements), under two categories: *physical* and *empirical* ones.

Physical models tend to be complicated and more difficult to generate and handle, because they are strictly linked to the real structure and therefore carry physical sense. Empirical models aim only at replicating a certain response of the system, they are hence quicker to solve but no physical sense can be attributed to them. Examples of different approaches can be found in the work of Szekely [1]-[3] (about the RC transmission line heat propagation, the Network Identification Deconvolution and the concept of structure function) and by the DELPHI project work on the Boundary Conditions Independence (BCI) [4], [5]. Many compact model approaches are purely linear [6]-[10]; to overcome this limitation, the method developed in [15] is used in this work, which is based on Finite Element Method (FEM) models [11]-[14].

Despite the great availability of device electro-thermal models provided by manufacturers, it is always useful to dispose of methodologies to generate models tailored on the own application, for instance, to improve numerical stability, model flexibility of use, and so on.

This work deals with the generation of an electro-thermal simulation setup based entirely on empirical models. An electro-thermal analytical model of a MOSFET is developed by using Neural Networks (NNs); this solution is especially suited to improve convergence in electrical simulators.

Usage of NN in circuit modelling is nowadays a well established methodology, examples of which can be found in several works whose application focuses on different topics like behavioral modelling, electro-magnetic compatibility modelling, and device modelling [27]-[32]. In general, NNs are used to learn the behaviour of a circuit, so that the network is able to replicate its behaviour without implementing the actual circuit. This offers of course an advantage in terms of simulation speed.

Although modern electrical simulators allow the definition of models based on Look-Up Tables (LUTs), if too many models in the schematic are defined by LUTs, the simulator might experience difficulties in converging. The

NN must develop an analytical function which can describe the $I_D = f(V_{GS}, V_{DS}, T)$ measured characteristic. NNs are advantageous in such tasks since:

- the user does not need to devise an analytical formula which fits the characteristic of the device;
- the complexity of the analytical formula can (to a certain extent) be reduced to the minimum necessary;
- thanks to the availability of NN-orientated frameworks in free programming languages like python, automation of the process is straightforward to implement, and netlist writing can be easily performed.

The main drawback to such approach relates to extrapolation; the model derived from the NN fits well the original data within the range of values used to train it, however, extrapolation has always to be carefully considered and it might lead to grossly wrong results.

The demonstration of such approach is carried out by using an *ad-hoc* developed board, where several devices are soldered and operated in conditions where strong self-heating is induced. This means, a strong electro-thermal feedback is expected, and a SPICE model of the board is to be built. The following sections will detail the followed procedure and recall the necessary concepts about neural networks.

II. ARTIFICIAL NEURAL NETWORK ARCHITECTURE

An Artificial Neural Network (ANN) is a powerful tool inspired by brain modelling studies [17]-[19], here a few basic concepts are recalled for sake of readability. ANNs are used thanks to their capability of learning a certain input-output relation (image classification, function fitting, and so on). A typical ANN is represented in Fig. 2.

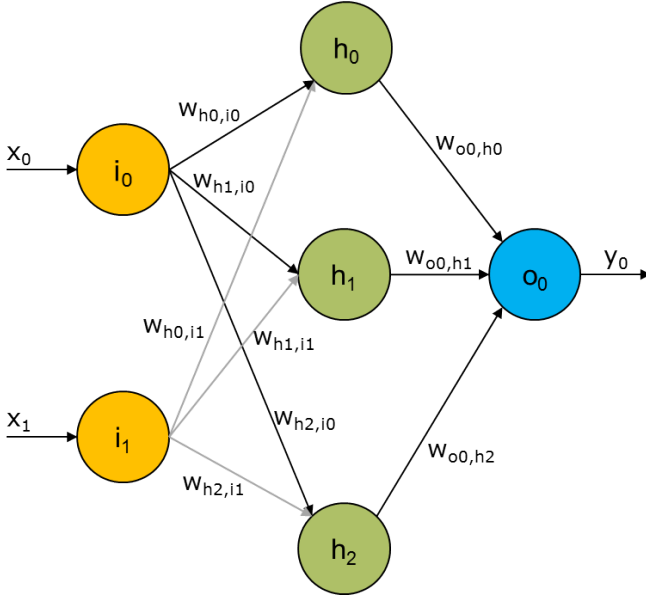


Fig. 2. Simple ANN with input layer, hidden layer and output layer.

Basic components

Any ANN is composed by *neurons* connected by *synapses*. A neuron implements the learning function by its activation

function $f_{AN}(x)$; commonly used functions are hyperbolic tangent or sigmoid, amongst many other. The simplest ANN architecture is composed of three neuron layers: input layer, hidden layer, and output layer, as shown in Fig. 2. Bias neurons can be added to improve the training process. Input neurons accept input variables, output neurons describe the output variables, while hidden neurons are those which are actually responsible for the training process.

Application and ANN training

A physical phenomenon can in general be expressed by an analytical formulation \mathcal{F} , a function which accepts n inputs and returns m outputs:

$$\mathbf{y} = \mathcal{F}(\mathbf{x}) \rightarrow [y_0, \dots, y_{m-1}] = \mathcal{F}([x_0, \dots, x_{n-1}]) \quad (1)$$

However, in many cases \mathcal{F} is not known, but only the observation of its response \mathbf{y}_i to a certain input vector \mathbf{x}_i can be observed (with i being the i -th observation which can be obtained during an experiment); only a set \mathcal{S} of (\mathbf{x}, \mathbf{y}) observed input-output points is therefore known. The task of an ANN is to learn the relation expressed by the set \mathcal{S} and end up with an approximated \mathcal{G} function which can capture such behaviour. Usually, $\mathcal{S} = \mathcal{S}_t \cup \mathcal{S}_v$, where \mathcal{S}_t is used as a training set, and \mathcal{S}_v as verification set, to check if \mathcal{G} can predict sufficiently well the behaviour described by the unknown \mathcal{F} . Eventually, the learning process is completed if the following error becomes smaller than a certain threshold ε_0 :

$$\begin{aligned} \varepsilon &= \sum_i \left\| \mathcal{G}(\mathbf{x}_i) - \mathcal{F}_i(\mathbf{x}_i) \right\|^2 = \\ &= \sum_i \left\| \mathcal{G}(\mathbf{x}_i) - \mathbf{y}_i \right\|^2 < \varepsilon_0 \quad (2) \end{aligned}$$

Clearly i ranges over all the points available in the training set \mathcal{S}_t . Each synapse features a *weight*, which varies during the ANNs' training process. This is actually what makes the ANN flexible: the values of the weights is modified during each backpropagation learning step until the convergence criterion is satisfied. At this point, the ANN is defined by its topology, that is, how many neurons on how many layers, their connections and the weights for each of these connections. In this work, a Multi-Layer Perceptron (MLP) is used as network topology.

ANN mathematical description

For the ANN to be implemented in a circuit simulator, its analytical formula must be retrieved. This step is explained for a MLP with 1 hidden layer, being the formulation more and more complex, the higher the number of inner layers used. It is also possible to demonstrate (universal approximation theorem, [20]) that a single-hidden layer MLP is sufficient to approximate any function to any accuracy (provided a sufficient number of hidden neurons are used). Referring to Fig. 2, where $I = 2$ input layer, $H = 3$ hidden layer, $O = 1$

output layers are present, inspection of the network returns the following analytical formulation:

$$y_0 = f_{AN} \left(\sum_{p=0}^{H-1} w_{o_0, h_p} \cdot h_p \right) = f_{AN} \left(\sum_{p=0}^{H-1} w_{o_0, h_p} \cdot f_{AN} \left(\sum_{q=0}^{I-1} w_{h_p, i_q} x_q \right) \right) \quad (3)$$

A single hidden layer with H between 6 and 10 neurons is typically used (more neurons allow a better level of approximation) [21], [22].

A. Final remarks about choosing and training the ANN

This section wants to provide some insights behind the rationale with which the ANN has been chosen and trained. It has to be mentioned that no exact rules exist in order to decide exactly how complex the ANN must be for a certain application. Usually, an ANN with *one* hidden layer is chosen, therefore its main structure is at least defined. This is also due to the fact that for such an ANN, the process of retrieving its analytical formulation (the combination of weights and activation functions) is pretty straightforward, while the complexity of the formulation increases with the number of hidden layers. This choice is supported by the Universal Approximation Theorem.

Once the topology is fixed, the number of hidden layers has to be chosen. This is a trial-and-error process, which is best achieved by training different ANNs at the same time and see which one provides the best approximation with the minimum amount of hidden neurons, and with and without bias. Usually, adding bias neurons (such as in Fig. 3, see next section for more details) adds flexibility in the ANN's response.

Regarding the training, it is generally implemented by the chosen tool and it is transparent to the user. In this work a GUI-based tool has been used, however, the process can be moved to programming tools such as Python, which offers many libraries to deal with ANN with various degrees of simplicity and performance. A parallel investigation, only mentioned here, has been done using Pybrain [26], a python module which offers a very user friendly Application Programming Interface (API) to the development and usage of ANNs.

III. APPLICATION OF ANNS TO MOSFET ELECTRO-THERMAL MODELLING

In this work, an MLP has been used to reconstruct an analytical formulation for the electro-thermal model of a given MOSFET. In particular, the following relation had to be reconstructed:

$$\mathbf{y} = \mathcal{F}(\mathbf{x}) \iff I_D = \mathcal{F}(V_{GS}, V_{DS}, T_j) \quad (4)$$

where T_j is the average channel temperature in the device. An ANN with 3 inputs and 1 output will be necessary. Its structure is shown in Fig. 3. In order to model the drain current of a power MOSFET, MLP with bias, which are networks with constants neurons [23], and sigmoid activation functions, were

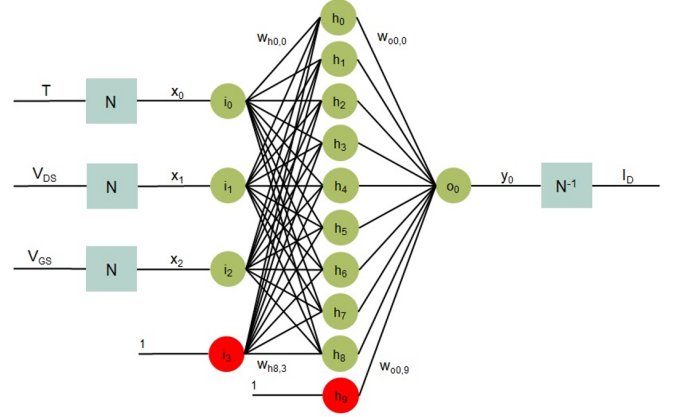


Fig. 3. MLP topology used to model the chosen MOSFET. In red the biasing neurons.

used. The MLP network has a fixed topology to simplify the extraction of the weights. Fig. 3 shows the MLP found to be suitable in this case, with normalization functions at the inputs and de-normalization function at the output (indicated by N and N^{-1} blocks, respectively). Quantities should always be normalized when operating with ANNs. In this work, the ANN is used to fit only the static I_D characteristic of the device; any switching behaviour is therefore neglected. This aspect is considered and discussed at the end of the paper, where a possibility to extend this approach to modelling of electric transients as well is proposed. In Fig. 3 the neurons named i_3 and h_9 are bias neurons for the input layer and the hidden layer, respectively. Bias neurons are not strictly necessary, but they add flexibility to the function implemented by the ANN. It is in general suggested to try to train two different ANNs, one without and one with bias neurons. Normally, only one bias neuron per layer has to be added. They do not add significant computational burden to the overall network, being just one neuron per layer needed. In this work, their presence was found to be beneficial for the fitting and therefore they have been included. They are basically introducing a degree of freedom in the fitting routine, which is otherwise performed with normalized functions which output values belong to $[0, 1]$.

A. Collection of the measurement data

Once set the MLP topology, it is necessary to train the network so that a set of weights w can be obtained. In this work, the training of the ANN was carried out via backpropagation and the software tool Neuroph [24]. This software allows to extract the trained network and to convert the data in SPICE format. The training set was collected by measurements made with a HP4142B semiconductor parameter analyzer. To set the desired temperature, the test board was put in a DY110 Angelantoni Temperature and Climatic Test Chamber. Pulsed sweep measurements has been set to avoid the device self-heating, triggering them at thermal stationary states, when the temperature of the chamber is stable and it is the same on the whole system under test. The test bench architecture to measure the training set is shown in Fig. 4.

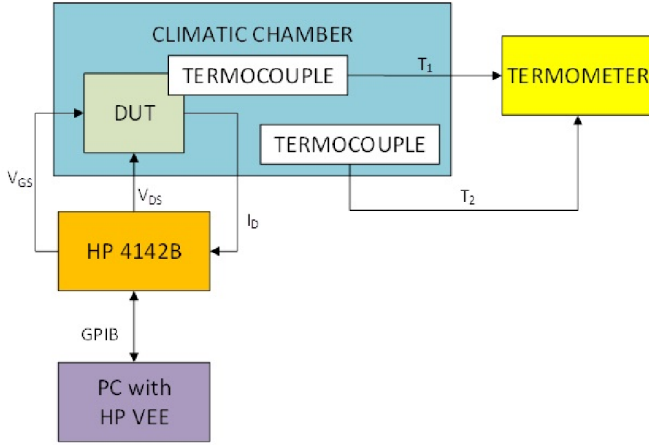


Fig. 4. Block diagram of the test bench used to measure the drain current $I_D = f(V_{GS}, V_{DS}, T)$ for the MOSFET as Device Under Test (DUT in the schematic).

The MOSFET drain [33] was connected to the power supply by $R_D = 4.7\Omega$. In addition to the resistance R_D , in the model, it was important to consider the parasitic resistance R_S due to copper tracks, solder joints and cables between board, chamber, and power supply. Parasitic resistances at the source might influence V_{GS} especially in cases such as this where the device was biased with V_{GS} values just slightly above the gate threshold voltage. In this case, a value of $72\text{m}\Omega$ was measured, and then the actual circuit to consider is the one in Fig. 5.

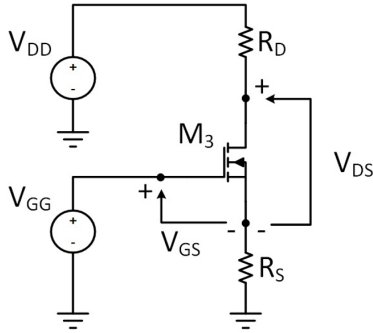


Fig. 5. Schematic of the circuit made to validate the electro-thermal model.

Results of the learning process

The device was then characterized and the family of output curves were saved, thus generating the training set \mathcal{S} . A complete set of data $I_D = f(V_{GS}, V_{DS}, T)$ was obtained by extracting the device's output characteristic for different temperatures. Temperature was swept between $[20, 120]^\circ\text{C}$ in steps of 20°C , V_{DS} between $[0, 3]\text{V}$ in 0.05V steps, and V_{GS} between $[2, 2.5]\text{V}$ with steps of 0.05V . Half of the dataset was used for training, while the remaining half for verification. Every second point in V_{DS} was taken to create the training set. For instance, if V_{DS} were swept between $[0, 3]$ with steps

of 0.1V , the points at $V_{DS} = 0, 0.2, 0.4, \dots, 3.0$ would be used for training, while $V_{DS} = 0.1, 0.3, 0.5, \dots, 2.9$ would be used for verification. Scrambling the dataset can be also recommendable, because it somehow forces the ANN to learn over a less structured pattern and therefore reaching a more stable set of weights. This set is then used to train the ANN as described and the comparison between measured and learned curves is shown in Fig. 6. A note regarding the limited range of voltages used to characterize the devices. The work wants to focus on the modeling of the self-heating in the device and this can be achieved in an easier way by keeping the device in a high-ohmic biasing condition, which means keeping the gate overdrive $V_{GS} - V_{th}$ low, where V_{th} is the MOSFET's threshold voltage. The limitation in V_{DS} , instead, comes from the fact the used Source-Measurement-Unit (SMU) is limited to a maximum current of 10A . This means, SMU compliance is reached pretty soon when $V_{GS} > 2.3\text{V}$ and $V_{DS} > 3 \div 4\text{V}$, where the device enters fully R_{on} mode. In any case, the part of the characteristic which is more demanding to be fitted is indeed for low V_{GS} and V_{DS} , therefore the admittedly reduced range does prove anyway the flexibility of the ANN for such task. As shown, the agreement between simulations

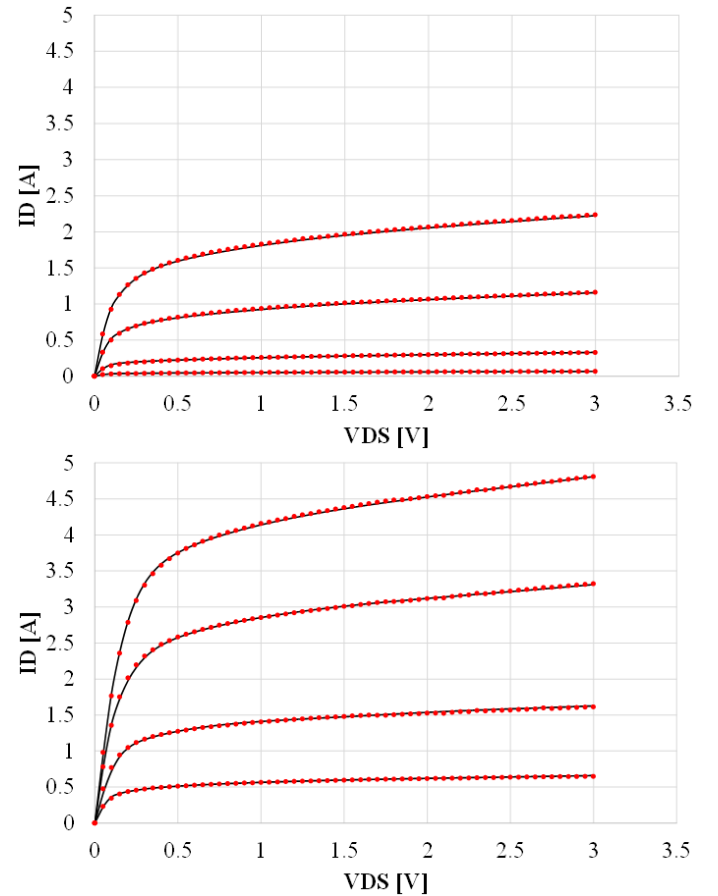


Fig. 6. Comparison between measurement (red dots) and ANN-learned model (black lines) for two different ambient temperatures (20°C , top and 120°C , bottom) for different gate-source voltages, $V_{GS} = (2.1, 2.25, 2.4, 2.5)\text{V}$.

and measurements is excellent, thanks to the number of fitting parameters made available by the ANN. Summarizing, the

procedure to build the temperature dependent electrical model using MLP is the following:

- 1) collect accurate raw data that describe the electrical behaviour of the device by measurements at well-known temperatures;
- 2) normalize the raw data to create a training set for the MLP;
- 3) train the MLP: if the error does not decrease to an acceptable value, the topology will be changed adding a neuron in the hidden layer;
- 4) build the MLP SPICE model using the extracted weights; **this means using a behavioural voltage-controlled-current-source (VCCS) which implements the analytical formula that the ANN describes.**

IV. THERMAL MODELLING

The process of determining a suitable thermal model for the considered structure is fully described in [15]; here, just a short summary of the methodology is shown. Basically, the thermal model to be coupled to the electro-thermal MOSFET model previously described must model the device junction temperature.

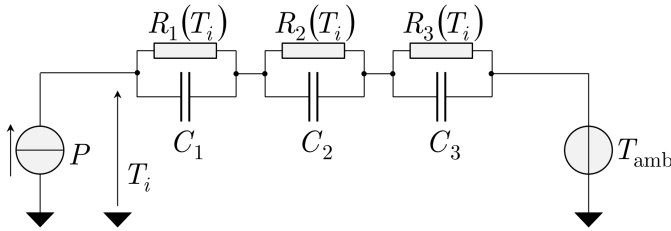


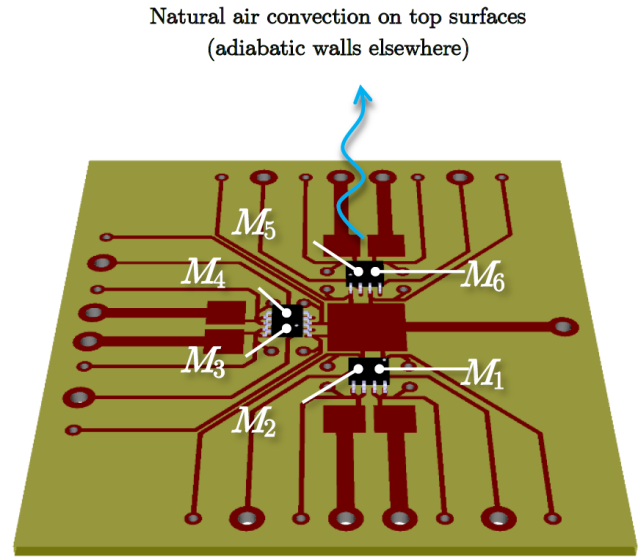
Fig. 7. Non-linear Foster thermal network.

To do so, a combined approach is developed, where infrared thermal measurements are used to tune a three-dimensional Finite Element Model, from which the thermal transient responses are extracted. **The test fixture used in this work is shown in Fig. 8. For a complete description of the setup, the experimental results and the thermal modelling process, the reader should refer to [15] which describes the full work in detail.**

The non linearities of the system are captured by a single RC Foster network, whose resistive terms are temperature-dependent. The temperature dependence of the terms is obtained by a set of thermal impedances, each of which extracted at different power dissipation levels. The topology of the obtained network is that depicted in Fig. 7. For further details about the method, the reader is invited to refer to the mentioned work.

V. COUPLED ELECTRO-THERMAL SIMULATION

Once the electro-thermal and the thermal model are built, they can be coupled in order to obtain a self-consistent SPICE electro-thermal model able to reproduce the fully-coupled electro-thermal behaviour of a heat source as a power MOSFET on a board, as done here to demonstrate the validity of the new proposed method. The main advantage of the



M_i silicon die = i -th heat source

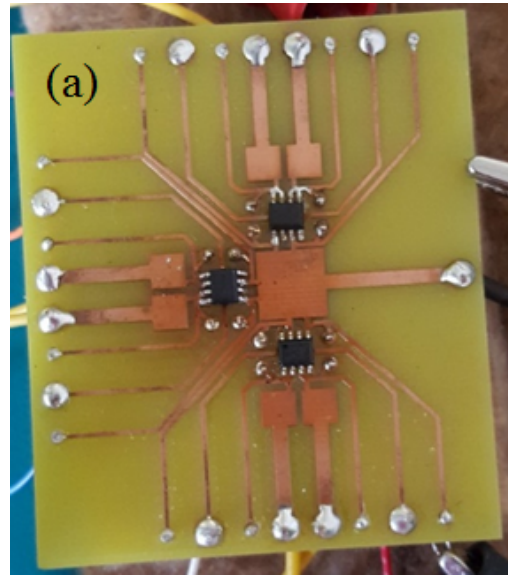


Fig. 8. Manufactured board to investigate self-heating and mutual-heating effects between the different transistors. Reprinted from [15].

proposed methodology is the capability to produce extremely accurate models, even without a deep knowledge of the device structure. The self-consistent [16] electro-thermal model built with this methodology can be directly simulated through a SPICE simulator [25]. A SPICE-based approach is desirable in case mission profiles have to be simulated [16]. The implementation of the fully-coupled simulation setup consists of the following two blocks:

- 1) a non-linear Foster network that simulating the non-linear thermal behavior;
- 2) an MLP network simulating the electro-thermal MOSFET behavior.

The non-linear Foster network is built using standard SPICE capacitors and SPICE behavioral models (B-Models), in order to model the temperature dependent thermal resistances. Differently, the electrical model based on MLP is modeled

through a unique B-Model (which implements the analytical function to which the ANN converged). The final circuit which is simulated is shown in Fig. 9.

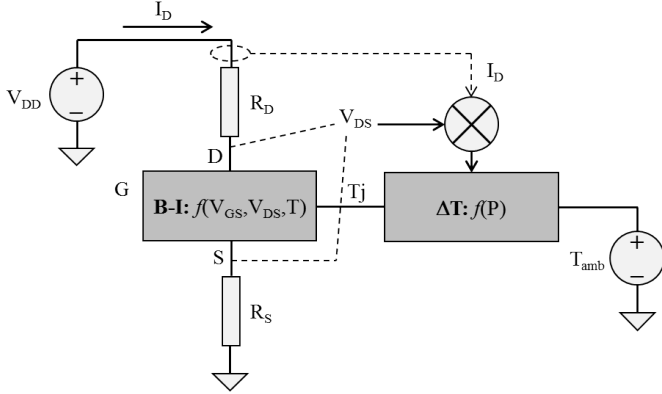


Fig. 9. Schematic of the simulated circuit where the behavioural model of the MOSFET is defined in terms of the analytical function that the ANN implements.

VI. VALIDATION AND RESULTS REVIEW

Some tests on the analysed device were devised to a steady-state comparison between measurements and SPICE simulations. FEM simulations were used to reconstruct the channel temperature in the device, being inaccessible from the measurement point of view [15]. The error was evaluated for V_{DS} , I_D and the channel temperature increase ΔT , respectively, according to the following formulas:

$$|E_V\%| = \frac{|V_{DS,meas} - V_{DS,sim}|}{V_{DS,meas}} \times 100 \quad (5)$$

$$|E_I\%| = \frac{|I_{D,meas} - I_{D,sim}|}{I_{D,meas}} \times 100 \quad (6)$$

$$|E_{T,FEM}\%| = \frac{|\Delta T_{FEM} - \Delta T_{sim}|}{\Delta T_{FEM}} \times 100 \quad (7)$$

$$|E_{T,meas}\%| = \frac{|\Delta T_{meas} - \Delta T_{sim}|}{\Delta T_{meas}} \times 100 \quad (8)$$

The thermal error is calculated with respect to a reference value which is either measured (when available) or FEM-simulated, depending on the fact if this reference temperature was either accessible by measurements (e.g. infrared thermography on the surface), or necessarily extracted by tuned FEM simulations (in case of an internal temperature, accessible only via FEM simulation tuned on thermal measurements). The simulations performed with the proposed method showed good results, because for each comparison, the error never exceeds 10%, as can be seen in TABLE I. Considering that no fitting parameters are used in this approach, and that a fully coupled electro-thermal dynamic is solved, an error below 10% is acceptable. It is also necessary to note that the highest error occurs at low V_{DD} , for low power dissipation conditions: this means that in this case, the absolute level of the quantities involved is anyway small. To show the capability of the model in a transient situation, the test case $V_{GG} = 2.3$ V and $V_{DD} = 3$ V is simulated, with the results shown in

TABLE I
ERRORS ON V_{DS} , I_D AND THE CHANNEL TEMPERATURE INCREASE IN THE SELECTED CASE STUDIES.

V_{DD}	V_{GG}	$ E_V\% $	$ E_I\% $	$ E_T\% $
4	2.15	5.2	2.1	2.5
3	2.15	2.8	2	0.2
2	2.15	2.4	2.8	1.4
4	2.3	8	1.2	4.8
3	2.3	7.8	1.3	4.5
2	2.3	8.2	1.1	4.5
4	2.45	6.9	0.2	3.9
3	2.45	1.4	0	2.1
2	2.45	10.4	0.2	8.3

Fig. 10. Simulation of such pulse takes very few minutes on a typical simulation workstation, compared to the few hours it would take to solve the thermal model only by Finite Element simulations.

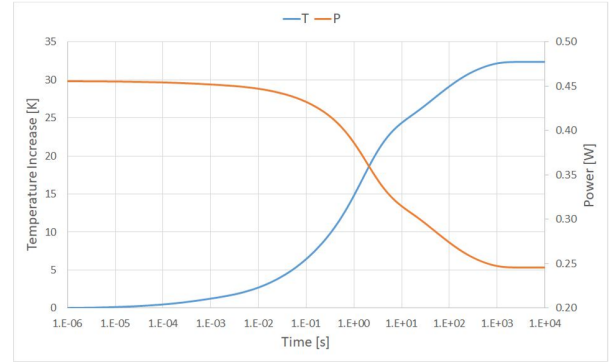


Fig. 10. Transient simulated dissipated power and transient channel temperature increase of the MOSFET M3 applying a biasing step with $V_{GG} = 2.3$ V and $V_{DD} = 3$ V.

Here the self-heating of the device starts to be visible already after 1 ms, where the power dissipated by the device starts to drop. Since the electrical bias remains unchanged during time, the device will progressively decrease its power dissipation, since channel conductivity will decrease with temperature.

VII. LIMITATIONS AND FURTHER DEVELOPMENT

The proposed approach shows an alternative way to generate a device model based on measurements. The reasons to follow such path can be the unavailability of accurate models, need of tuning a model on a specific device, solving convergence problems, just to mention some. In this section however the limitations of such approach are discussed.

Firstly, the developed electro-thermal model of the MOSFET is a static one, meaning that no electrical-transient effects are considered. In the overall model, the thermal transients are considered to be the most relevant ones, and the electrical ones are neglected. This means, that in a typical MOSFET model such as in Fig. 11, the ANN-description is limited to the transconductance generator, that is, its DC characteristic (such model can be used also for large-signal simulations). The effect of internal parasitic capacitances is not accounted by the ANN.

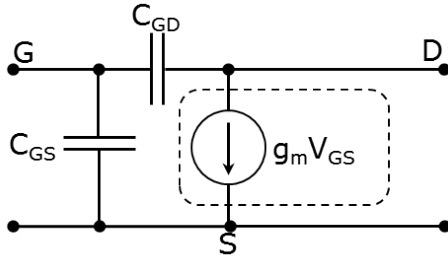


Fig. 11. Simple transient model of a MOSFET with the part modelled by ANN enclosed by dashed line.

The limitation in missing the parasitic capacitances C_{GS} and C_{GD} will result in a model unable to model the transient dynamic at the gate or the Miller effect, this might be of concern if the device is used in a high frequency DC/DC converter or in a linear amplifier, for instance.

Provided that a gate input model is present, the MOSFET model can also be used in switching applications and switching losses would be accounted (at the moment, the ANN model does not include such parameters). Without a gate model, switching losses cannot be accounted for, because the device would respond immediately to any change in V_{GS} . However, the addition of a suitable gate model will enable the accounting of switching losses, since the electro-thermal feedback is always active, also when V_{DS} is falling due to the rise of V_{GS} and viceversa.

Concluding, the lack of a gate model which accounts for the dynamic of V_{GS} does not allow the model in its current form to account for switching losses. The addition of input capacitances would fill this gap and enable the model to account for temperature-dependent switching losses. This is a topic that will be addressed in future extensions of this work.

VIII. CONCLUSIONS

In this paper, a methodology to implement MOSFET fully coupled, electro-thermal simulations based on compact, empirical models has been shown. The novelty of this approach resides in the usage of Artificial Neural Networks as a tool to obtain an analytical description of a complicated device, when only a LUT is available; LUT are in general unsuited for electrical simulators, leading to convergence problems. The proposed method solves this issue. The proposed methodology has shown very good results. The electrical model developed via MLP demonstrated to be an alternative to SPICE MOSFET model, with a good agreement with measurements. In fact, the MLP SPICE implementation demonstrated to be an easy alternative to complex models. Moreover, this kind of approach can be applied to any other electronic device. Finally, the combination of such MLP-based model with a non-linear thermal model, returned very good results once compared to measurements, without showing any convergence problem. The model at the moment is limited to slow electrical transients due to the lack of an input gate model. The addition of such input gate model, to be addressed in future extensions

of this work, will enable the model to account for temperature-dependent switching losses as well.

IX. ACKNOWLEDGEMENTS

This work was jointly funded by the Austrian Research Promotion Agency (FFG, Project No. 846579) and the Carinthian Economic Promotion Fund (KWF, contract KWF-1521/26876/38867).

REFERENCES

- [1] V. Szekely, "On the representation of infinite - length distributed RC one-ports," IEEE Transactions on Circuits and Systems, vol. 38, no. 7, pp. 711-719, 1991.
- [2] V. Szekely, "Identification of RC networks by deconvolution: Chances and limits," IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, vol. 45, no. 3, pp. 244-258, 1998.
- [3] V. Szekely, M. Rencz, A. Poppe, and B. Courtois, "New way for thermal transient testing [IC packaging]," in 15th Annual IEEE Semiconductor Thermal Measurement and Management Symposium, San Diego, USA, Mar. 1999, pp. 182-188.
- [4] H. I. Rosten, C. J. M. Lasance, and J. D. Parry, "The world of thermal characterization according to DELPHI-part I: background to DELPHI," IEEE Transactions on Components, Packaging, and Manufacturing Technology: Part A, vol. 20, no. 4, pp. 384-391, 1997.
- [5] C. J. M. Lasance, H. I. Rosten, and J. D. Parry, "The world of thermal characterization according to DELPHI - part II: experimental and numerical methods," IEEE Transactions on Components, Packaging, and Manufacturing Technology: Part A, vol. 20, no. 4, pp. 392-398, 1997.
- [6] P. E. Bagnoli, C. Casarosa, M. Ciampi, and E. Dallago, "Thermal Resistance Analysis by Induced Transient (TRAIT) method for power electronic devices thermal characterization - Part I: Fundamentals and theory," IEEE Transaction on Power Electronics, vol. 13, no. 6, pp. 1208-1219, 1998.
- [7] P. E. Bagnoli, C. Casarosa, E. Dallago, and M. Nardoni, "Thermal Resistance Analysis by Induced Transient (TRAIT) method for power electronic devices thermal characterization - Part II: practice and experiments," IEEE Transaction on Power Electronics, vol. 13, no. 6, pp. 1220-1228, 1998.
- [8] P. L. Evans, A. Castellazzi, and C. M. Johnson, "Automated fast extraction of compact thermal models for power electronic models," IEEE Transaction on Power Electronics, vol. 28, no. 10, pp. 4791-4802, 2013.
- [9] Y. C. Gerstenmaier, A. Castellazzi, and G. Wachutka, "Electrothermal simulation of multichip-modules with novel transient thermal model and time dependent boundary conditions," IEEE Transactions on Power Electronics, vol. 21, no. 1, pp. 45-55, 2006.
- [10] A. Castellazzi, "Comprehensive compact models for the circuit simulation of multichip power modules," IEEE Transactions on Power Electronics, vol. 25, no. 5, pp. 1251-1264, 2010.
- [11] D. Schweitzer, "Thermal transient multisource simulation using cubic spline interpolation of zth functions," in Proceedings of the 12th International Workshop on Thermal investigations of ICs (THERMINIC2006), Nice, France, Sep. 2006, pp. 182-188.
- [12] M. Bernardoni, N. Delmonte, G. Sozzi, and R. Menozzi, "Large-signal GaN HEMTelectro-thermal model with 3D dynamic description of self-heating," in Proceedings of the European Solid-State Device Research Conference (ESSDERC 2011), Helsinki, Finland, Sep. 2011, pp. 171-174.
- [13] M. Bernardoni, N. Delmonte, P. Cova, and R. Menozzi, "Self-consistent compact electrical and thermal modeling of power devices including package and heat-sink," in International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM), Pisa, Italy, Jun. 2010, pp. 556- 561.
- [14] D. Chiozzi, M. Bernardoni, P. Cova, and N. Delmonte, "A simple 1-D finite elements approach to model the effect of PCB in electronic assemblies," Microelectronics Reliability, vol. 58, no. 1, pp. 126- 132, 2016.
- [15] M. Bernardoni, D. Chiozzi, N. Delmonte, P. Cova, "Non-linear Thermal Simulation at System Level: Compact Modelling and Experimental Validation", Microelectronics reliability, vol. 80, pp. 223-229, 2018.

- [16] P.R. Wilson, J.N. Ross, and A.D. Brown, "Simulation of magnetic component models in electric circuits including thermal effects," *IEEE Transactions on Power Electronics*, vol. 17, no. 1, pp. 55-65, 2002.
- [17] M. Minsky and S. Papert, "Artificial intelligence progress report," Massachusetts Institute of Technology, Cambridge, USA, Tech. Rep. AIM-252, Jan. 1971.
- [18] A.P. Engelbrecht, *Computational intelligence an Introduction*, 2nd ed. Hoboken, USA: Wiley, 2007.
- [19] S. Russel and P. Norvig, *Artificial intelligence: a modern approach*, 3rd ed. London, UK: Pearson Education, 2016.
- [20] G. Cybenko, "Approximations by superpositions of sigmoidal functions", *Mathematics of Control, Signals, and Systems*, vol. 2, no. 4, pp. 303-314, 1989.
- [21] V. Kurkova, "Kolmogorovs theorem and multilayer neural networks," *Neural Networks*, vol. 5, no. 3, pp. 501-506, 1992.
- [22] K. Hornik, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, no. 5, pp. 359-366, 1989.
- [23] S. Geman, E. Bienenstock, and R. Doursat, "Neural networks and the bias/variance dilemma," *Neural Computation*, vol. 4, no. 1, pp. 1-58, 1992.
- [24] Neuroph. Java neural network framework neuroph. [Online]. Available: <http://neuroph.sourceforge.net/>
- [25] H. B. Hammouda, M. Mhiri, Z. Gafsi, and K. Besbes, "Neural-based models of semiconductor devices for SPICE simulator," *American Journal of Applied Sciences*, vol. 5, no. 4, pp. 785-791, 2008.
- [26] T. Schaul, J. Bayer, D. Wierstra, Y. Sun, M. Felder, F. Sehnke, Frank, T. Rückstieß, J. Schmidhuber, "Pybrain", *Journal of Machine Learning Research*, Vol. 11, pp. 743-746, 2010. Webpage: <http://pybrain.org/>
- [27] X. Han and M. Saeedifard, "Junction temperature estimation of SiC MOSFETs based on Extended Kalman Filtering," 2018 IEEE Applied Power Electronics Conference and Exposition (APEC), San Antonio, TX, USA, 2018, pp. 1687-1694.
- [28] L. Wu and M. Saeedifard, "A Simple Behavioral Electro-Thermal Model of GaN FETs for SPICE Circuit Simulation," in *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 4, no. 3, pp. 730-737, Sept. 2016.
- [29] R. M. Hasani, D. Haerle and R. Grosu, "Efficient modeling of complex Analog integrated circuits using neural networks," 2016 12th Conference on Ph.D. Research in Microelectronics and Electronics (PRIME), pp. 1-4, 2016.
- [30] R. M. Hasani, D. Haerle, C. F. Baumgartner, A. R. Lomuscio and R. Grosu, "Compositional neural-network modeling of complex analog circuits," 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, 2017, pp. 2235-2242.
- [31] M. Kraemer, D. Dragomirescu, and R. Plana, "Nonlinear behavioral modeling of oscillators in VHDL-AMS using artificial neural networks," *IEEE Radio Frequency Integrated Circuits Symposium 2008 (RFIC 2008)*, Jun 2008, Atlanta, United States.
- [32] M. Magerl, C. Stockreiter, O. Eisenberger, R. Minixhofer and A. Baric, "Building interchangeable black-box models of integrated circuits for EMC simulations," 2015 10th International Workshop on the Electromagnetic Compatibility of Integrated Circuits (EMC Compo), Edinburgh, 2015, pp. 258-263.
- [33] Datasheet Infineon BSO150N03MD.