IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Gallium Nitride Power Electronic Devices Modelling using Machine Learning

N. Hari[1], M. Ahsan[2], S. Ramasamy[3], P. Sanjeevikumar[4], Senior Member, IEEE, A. Albarbar[2], Frede Blaabjerg[5], Fellow, IEEE

[1] Centre for Photonics and Electronics, Department of Engineering, University of Cambridge, J J Thomson Avenue, CB30FA, U.K.
[2] Smart Infrastructure and Industry Research Group, Department of Engineering, Manchester Metropolitan University, John Dalton Building, Chester St, Manchester M1 5GD, U.K.
[3] Department of Electrical and Electronics Engineering, SRMIST, SRM Nagar, Kattankulathur, Tamil Nad, 603 203, India
[4] Department of Energy Technology, Aalborg University, Esbjerg, 86700, Denmark
[5] Center of Reliable Power Electronics (CORPE), Department of Energy Technology, Aalborg University, Aalborg, 9220, Denmark

Corresponding author: M. Ahsan (e-mail: M.Ahsan@mmu.ac.uk).

ABSTRACT A state-of-the-art Machine Learning (ML) based approach, by modeling the behavior of Gallium Nitride (GaN) power electronic devices, is presented in this paper. Switching voltage and current waveforms of these novel devices are accurately predicted using the developed supervised ML algorithm. This was utilised to build a more generic black-box model for these devices. Moreover, long short-term memory unit (LSTM) and gated recurrent unit (GRU) device models have been proposed to make the approach more user friendly. The performance of the developed approach is verified using a set of simulations and experimental tests under 450 V, 10 A test conditions. Model results demonstrate an error rate of 0.03 and convergence speed of 3s with excellent stability. Compared to the existing models, the developed ML-based model produces more accurate results, converges faster and has a better stability. Additionally, the developed ML-based GaN model offers the ability to select the best fit available GaN model (Panasonic, GaN Systems, Transphorm etc.). It automatically configures them into a system that would optimally yield the desired power conversion. This enables a shorter learning curve for the power electronics community, which would lead to acceptance and faster adoption of these devices by the power electronics industry.

INDEX TERMS Gallium Nitride, power electronics, modelling, machine learning, neural networks.

I. INTRODUCTION

GaN-based devices have superior performance and material properties compared to those made of Si. However, before wider adoption by the power electronics industry, the behaviour of GaN devices must be fully understood. The steep learning curve involved is acting as a roadblock to the adoption of these devices by the industry [1]. To solve this problem, an in-depth understanding of the switching performance of different types of GaN devices (which are based on different structures) is required. Conventional modelling methods are derived from semiconductor physics, the property of materials and structure of the device, which usually are not available for the device users, resulting in difficulties in modelling the device [2], [3]. The authors have explored RF-based parasitic extraction to develop a behavioural model, but, it was observed that this method is not highly accurate [4]. This is because it is dependent on the accuracy of the measurement circuitry.

Additionally, the RF model is developed, neglecting the effect of certain parasitic elements [5], [6]. Due to the complexity of the device structure, time involved in parasitic extraction and the analytical procedures involved, this model is not suitable for validating all applications [7], [8]. Thus, it cannot serve as a universal model for GaN. To solve this problem, GaN simulation models which are an accurate replica of the actual device is designed, built and demonstrated using ML techniques.

GaN-based RF devices have been widely used for microwave applications, and CAD-based modelling techniques are generally used for modelling these devices [9]. To perform statistical CAD with current approaches is not feasible as a single analysis of a component may require several hours or days and hundreds of analyses are required.

It is because these techniques make use of computer-intensive electromagnetic full-wave simulators. ML modelling algorithms, on the other hand, uses multi-dimensional non-linear approximator, which maps the input parameters to the output ones. Hence, Neural networks (NN) appear to be the perfect candidate to perform this process.

The rapidly evolving field of NN based modelling, especially in microwave CAD and optimization has led to several findings. With the increased proliferation of AI, researchers started investigating NN based modelling for microwave transistors. NN based RF transistor models can be developed through a computerized training process, and the models can be developed even if sophisticated device theory equations are unavailable. There are few papers in this regard to model microwave-based HEMT devices [10], [11]. But, there is not much progress for developing NN models that can reproduce their dynamic characteristics. While NN models have made inroads into wireless and communication areas, NN models for static and dynamic performance of power devices are still in its early stages of research [12], [13], [14].

Machine Learning techniques, particularly the Neural Networks, are recently starting to make an impact on power systems and motor drives. The underlying AI techniques such as fuzzy logic and genetic algorithms have been applied for elemental power electronic applications as shown in [15]. From all the different branches of AI, NN's barely penetrated the motor drives area that is evident by the publications in the literature, which are more than ten years old as listed in the above paper. Though there has been a lot of revolutionary strides in ML research and its application in many areas, there are only less than twenty-five literature /papers in the area of application of NN techniques for power electronics (PE). But some note-worthy papers are exploring neural network modelling for microwave devices as noted before. Similarly, few recent papers are currently exploring using NN for reliability assessment for improving the life of GaN power converters [16], [17]. Though reliability monitoring is out of the scope of this paper, the authors will be exploring this when the models are scaled up for commercialisation.

Main contributions of this research work can be summarised as follows:

1. ML models are used to predict the switching voltage and current waveforms; thus, making it possible to construct a black-box model of the GaN power device.
2. The predicted waveforms are verified using experimental results and found to be in good agreement. Moreover, this was achieved at a faster convergence rate of 3s and error rate of 0.03 compared to existing simulation models which converged at 68s and more.
3. This research demonstrates different types of GaN ML models. The developed voltage and current prediction models are based on long short-term memory unit (LSTM) and gated recurrent unit (GRU). Several parameters are quantified and compared for validating the models. They are the network architectures, parameters, training time, validation loss and error loss.

This paper is organised as follows: Section II describes the practical need for ML-based modelling for GaN power devices. Section III details the data collection set up and section IV introduces the GaN power device behaviour modelling using ML. In section V, RNN models are designed, developed and demonstrated. The models are then validated with existing manufacturer simulation parametric models. Section VI discusses the contribution of this research work. Section VII includes conclusions and future work.

II. PROBLEM DEFINITION: GAN HEMT BEHAVIOURAL MODELLING USING ML

This work uses both single and multi-recurrent neural networks (RNNs) to quicken the design process of GaN circuits and devices. It is done using supervised training to predict the switching voltage waveforms. Thus, a NN based GaN model is developed using ML techniques. This model has been compared to other conventional LT-Spice behavioural models to compare accuracy and convergence. The voltage between drain and source and device current at both conducting and switching states can be modelled by using the ML process. This is done using measurement data of these variables along with their corresponding gate voltage. The data required is acquired through recording a large number of switching events which are then used as the training and testing data.

Let x represent an Mx vector containing dynamic characteristics of the GaN device obtained from the double pulse test (DPT) circuit, like, input voltage, gate voltage, digital control signal and gate current. Let y represent a vector containing the output of the device switching behaviour under consideration such as device switching voltage and device switching current. The physical-mathematical relationship between y and x can be represented as $y = f(x)$. This relationship for GaN device is highly non-linear and multi-dimensional. GaN being a nearly ideal device, this relationship is influenced by the parasitic of the circuit, unlike its Si counterparts where such effects can be neglected. The effect of these on the device behaviour is challenging to measure. Additionally, the analytical physics-based model is computationally intensive for online implementation.

So, this research aims to develop a fast and accurate generic neural network model for GaN. This is done by training a neural network to learn the GaN-based switching circuit problem through a set of the measured and simulated sample set of data called training data were: $[(Xs, Ds)s \in Tr]$, where Ds represents the measured/simulated output y for the input Xs and Tr represents the overall set of training data. Now, the neural network model can be defined as $y = y(x, w)$, where w represents the parameters inside the neural network generally termed the weight vector.

In this modelling, to make sure that the neural network makes predictions that are close to the actual value of the output voltages, a loss function Mean Absolute Error (MAE) that will be able to reduce the distance between the predicted and real values and in effect increase the accuracy is used. The Mean Absolute Error (MAE) is the sum of the absolute differences between predictions and actual values. It gives an idea of by how much wrong the predictions are. It gives an idea of the magnitude of the error, but no idea of the direction (e.g. over or under predicting). It is defined as the average error over the test sample of the absolute differences between prediction and actual values, where all discrete differences have equal weight.

$$E(\omega) = M\ AE = \sum_{j=1}^{n} \left| y'\left(x_j, w_j\right) - d_{sj} \right|,$$ where yj is the prediction, and dsj is the measured value from the experimental results/simulation.

The objective of the neural network training is to find 'w' such that E(w) is minimized. The structure/architecture of the NN is defined by the definition of w, the methodology by which yj is computed through x and w. Since the switching waveforms are a continuous function, it can be predicted with reasonable accuracy using ML. The 6-step ML-based GaN modelling process used is as follows:

1. Problem Definition: Building an accurate 600 V black-box model of GaN device using ML.
2. Analyse Data: Gate voltage and input voltage are used as inputs; device current and switching voltage used as outputs for training; test data is collected from the double pulse test measurements and simulations.
3. Prepare Data: Normalization is done to convert data for training the neural networks.
4. Choose Model: Regression-based feed-forward and recurrent models are used, and the process is as shown in Figure 1.
5. Training: Training data is used to incrementally improve the model's ability to predict the switching waveforms of GaN.
6. Present Results: The output of the device switching voltage and the switching current is predicted.

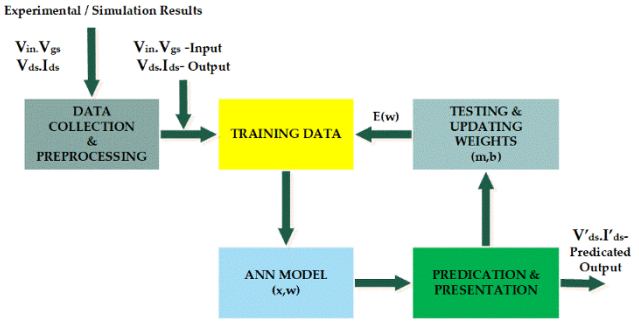


FIGURE 1. Block diagram of GaN device behavioural modelling using ML

III. DATA COLLECTION AND PREPROCESSING

The first step in NN model development is the identification of inputs and outputs. Once the inputs and outputs are identified, the device/ circuit/ experimental data needs to be gathered or generated depending on the problem definition. For PE-based applications, there can be two or three types of data generation: measurement, analytical calculation and software simulation. In the case of PE applications, experimental data is collected via appropriate measurement techniques; simulation results are generated and exported to compatible formats that can be processed by the NN model.

For this modelling, data is collected via experiments and simulation using double pulse test (DPT). Both switching experiments and simulation were done using the available GaN power devices to collect as much data as possible. Due to the ease of recording simulation data, more set of such data could be collected. In this work, approximately 70 per cent of training data is from simulation, and the remaining 30 per cent is experimental data.

Double pulse test - To be able to validate the model, it is necessary to compare the performance of the simulation using the proposed model with the performance of the actual device in the experimental rig, i.e. the double pulse test in this case. The prototype, as shown in Figure 2 is used for the double pulse testing and has been supplied by Sanken Inc as part of the team's collaborative work with them. The circuit can be customized to use TO, and other SMD packages and is thus used for accurate measurement, convenience and flexibility. The double-pulse switch test is set at 500 V DC with a switched load current of 15 A (half the device rated current). The driving current is set at around 800 mA. The supply voltages for gate drive are adjusted according to the specification of the device being tested.

The test set up and simulation system used is as per the following specifications:

a. 500Vdc-bus, 15A from the inductive load.
b. In-built and customised measurement set-up.
c. Agilent oscilloscope with double pulse signal from Agilent waveform generator.
d. Electrical power from benchtop power supplies.

The current device measurement was done using a current probe. Whereas, the voltage measurements are checked using a precision probe. The circuit was tested using GaN Systems, Transphorm, Panasonic and Sanken devices. Since the author did not have further access to the datasheet of the discrete Sanken devices, it is not investigated further in this work and is not used for the model design.

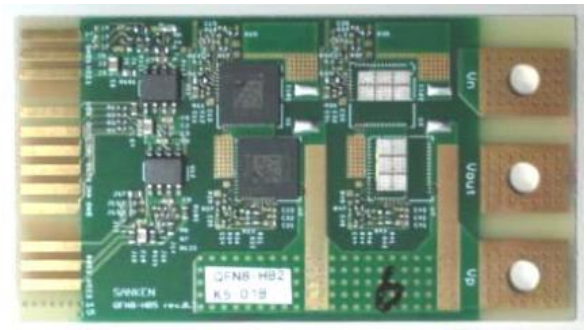


FIGURE 2. Double Pulse Test Circuit Prototype (Sanken Inc)

Generally, NN modelling requires the following sets of data:

1. Training data (TR),
2. Validation data (V),
3. Test data (TE)

TR is used to govern the training process, i.e. to update the NN weights during training. During training, validation data is used to track the error of the model and test data is used to evaluate the final accuracy/error of the developed model. There are no requirements for the sizes of the partitions, based on the practical methodology for data partitioning; the percentage depends on the available data size. In general, 50 per cent or more of the data is allocated to the training set, 25 per cent to the test set, and the remainder is set apart for the validation set. When the sample size is small like in this case, machine learning experts and literature point out that a good practice would be to leave out the validation data and use a 60 - 40 or 70 - 30 ratio. As can be seen, a 70 - 30 ratio is the most commonly used split. The authors have hence used the 70-30 ratio split between training and testing data for this work.

IV. GAN HEMT BASED MODELLING USING RNNs

One of the most popular ML algorithms is NNs [18], [19]. Neural Networks gained much popularity recently owing to their effectiveness in many difficult tasks like image classification and natural language processing [20], [21]. NNs are a connected system of computational units that can be trained from examples rather than being explicitly programmed. They are modelled loosely after biological neurons and can be used to solve a variety of tasks that are hard to solve using rule-based programming [22], [23], [24]. An NN consists of an input layer, hidden layers and output layers. Hence, each layer performs calculations based on its weights, inputs, biases and activations and gives an output. A combination of a different number of neurons and hidden layers forms an architecture. A simple feed-forward neural network works by multiplying the inputs to the neurons with the respective weights of the connections, adding bias and then applying a non-linearity like tanh. Simple neural networks like these have proved to be very useful in solving complicated problems like image classification and language generation.

NN is a consequence of inter-linkage of artificial neurons to mimic the operation of a human brain to solve scientific, engineering, industrial and many other real-life problems. The architecture of the biological neural network is not yet well-understood, and therefore, many NN models have been proposed till date and research is still ongoing [25], [26], [27], [28], [29].

Neural networks where the output from one layer is used as input to the next layer are called feed-forward neural networks. These networks define a mapping function y = f(x,w), the function y learns the value of the parameters w that result in the best function approximation. Conventional feed-forward neural networks are regarded for their learning and generalization capabilities. However, they can only map static input and output co-relation network; information is always fed forward, never fed back. To model a non-linear circuit, responses such as behavioural responses of devices in the time domain, a NN that can incorporate temporal information is necessary and is possible via feedback loops. Such models are called Recurrent Neural Networks (RNNs) [25] [26].

One of the significant drawbacks with traditional NN is that it cannot connect information from one instant to another past or present event. It only learns from a particular event. It is a massive problem while dealing with PE problems, especially with the dynamic behaviour of devices. Hence a relatively new NN model called Long short-term memory units (LSTM) and Gated Recurrent Unit (GRU) is first explored in this work which can learn from previous experience and can remember information for more extended periods, unlike RNNs. These are preferred in behavioural modelling due to their inherent capability to connect the output dependencies at previous instants to other instants by comparing the information stored over a more extended period of time.

LSTM unit: Due to the unstable gradient problem, early RNNs models were challenging to train [27]. Hochreiter and Schmidhuber introduced the LSTM units in 1997 with the explicit purpose of helping address the unstable gradient problem. The LSTM, as shown in Figure 2, can erase or augment information using ‘forget gate’ and ‘input gate’ to the cell state, coordinated by structures called gates. Using LSTMs when training RNNs makes it easier to get good results and is used in this work for building one of the GaN ML models.

GRU unit: Gated Recurrent Unit introduced by Cho [28] is a more powerful variation on the LSTM. It merges the ’forget’ and ’input’ gates into a single ’update gate’. It also fuses the cell state and hidden state and makes some other changes making the resulting model more understandable than standard LSTM models. Its performance is commensurate with LSTM but computationally more efficient (less complicated structure) and hence is beginning to be more widely used. Since its more comfortable to

generate one output for a NN model, inputs have been initially used to predict the output voltage.

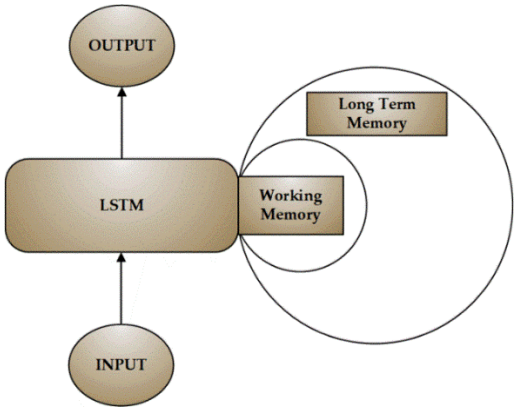


FIGURE 3. LSTM NN

Then a second model was trained using output voltage as another input to predict the output switching current. It is done to allow the model to learn the dependencies and co-relation of switching voltage and current on each other and with other inputs.

To understand the working of the NN modelling process and to start off with a much simpler and more manageable data processing, shallow NN (one hidden layer) based models are used at the start. The complete set of simulation results obtained from the double pulse test circuit is used for training. The DPT simulation is done using the manufacturer model for the following devices: GaN Systems (650 V, 30 A), Panasonic (600 V, 15 A) and Transphorm (600 V, 15 A).

1) Developing the NN Model using TensorFlow: One of the popular numerical platforms in Python that provide the basis for the deep learning research and development is the TensorFlow. This system has compelling libraries but can be difficult to use directly for creating deep learning models. For this research, Keras Python library is used. It provides a clean and convenient way to create a range of learning models on top of TensorFlow.

TensorFlow is the most famous library used in production for deep learning models. It has an extensive and active community. However, TensorFlow is not that easy to use. On the other hand, Keras is a high-level API built on TensorFlow (and can be used on top of Theano too, which has been recently shut-down).

Reasons for choosing Keras for this research work are the following:

(i) Rapid prototyping: In this work, there is a need to quickly build and test a neural network with minimal lines of code, and so Keras was the first choice. With Keras, one can build simple or very complex neural networks within a few minutes.

(ii) Modularity: Keras is very user-friendly and hence more pythonic. Everything in Keras can be represented as modules which can further be combined as per the user's requirements.

(iii) Simple coding: There is not much code required, and the steps involved are: 1) Data Loading 2) Data Pre-Processing 3) Data Preparation 4) Defining a Model

Models in Keras are defined as a sequence of layers. A Sequential model is created first, and layers are added one at a time until the right network topology. The number of layers and structure is difficult to decide from the beginning. There are some guidelines and rubric that can be used, but often the best network structure is found through a process of trial and error experimentation. Generally, we need a network large enough to capture the structure of the problem. In this work, a fully connected network structure with single and multiple layers are designed and demonstrated.

Once the model is defined, it can be compiled. Compiling the model uses the existing numerical libraries under the covers (called backend). In this work, TensorFlow is used as the backend. It automatically chooses the best way to represent the network for training and making predictions to run on the hardware. When compiling, there is a need to specify some additional properties required when training the network.

Training a network is to find the best set of weights to make predictions for the problem. So, there is a need to specify the loss function to evaluate a set of weights, the optimizer to search through different weights for the network and any optional metrics to collect and report during training. In this work, we have used mean absolute percentage error as the loss function, Adam as the optimizer and accuracy as the metrics of performance. These are best fit for this problem which has time-series data. Adam is used as it is best for handling sparse and noisy data. Additionally, it is easy to use and fast.

GRU model: For training the dynamic behaviour, the following inputs and outputs are selected.

1) Inputs: Gate voltage, Input voltage, Digital voltage (ON/OFF), Device switching current.
2) Output: Device switching voltage.
3) No. of data sets used: 30 (training: 25; testing: 5) [experimental data: simulation data split = 30:70].
4) Epochs:500.
5) Type of NN used: GRU.
6) 4 inputs, 1 output, 1 hidden layer and 65 nodes are used.
7) Gate voltage and device switching current is scaled by a factor of 10 while plotting.

Table. 1 shows the architecture that lists the trainable parameters and related info of the GRU NN used for developing the GRU based behavioural model for GaN power devices.

For begin with, in this model, the switching current is also used as an input. It is to generalize the model to be able to process both voltages and currents so that this can be used for current-controlled devices as well.

TABLE 1: GaN GRU model architecture

Layer (type)	Output Shape	Parameters
gru_5(GRU)	(None, 1002, 64)	12864
dense_5(Dense)	(None, 1002, 1)	65
Total params:12,929	Trainable Params: 12,929	Non-Trainable Params:0

The ability of the model to use voltages and currents to be able to predict the output voltage is a clear indication that this model can carefully map the inter-relationship between switching voltage, gate voltage and current. It is an essential improvement over the NN models for microwave devices which can only be voltage controlled. Firstly, the GRU model was trained using data from the DPT results and from the simulations done using manufacturer models. The data contained values for switch OFF and ON instants. After the initial data-processing was done, the data was normalized. After normalization, the values were squashed in the range of (0,1). After data pre-processing and normalization, the dataset is split into input-output pairs.

For example, plotting the prediction for a random set of training, the following waveforms are obtained as in Figure 3 and Figure 4. It can be noted that the GRU model closely follows the training data in terms of the waveform shape, but not during turn off. It is interesting to note that in Figure 3, in the ML model, the predicted voltage turns-off immediately after the gate voltage goes negative as should be the case. So, it is clear that in this case, our model is trying to predict the ideal case switching behaviour. It is possibly because the model has been fed with a lot of manufacturer model simulation waveforms while training which was more or less ideal waveforms.

TABLE II. GaN LSTM model architecture

Layer (type)	Output Shape	Parameters
lstm_11(LSTM)	(None, 1250, 32)	12700
dense_5(Dense)	(None, 1250, 1)	33
Total params:12,833	Trainable Params: 12,833	Non-Trainable Params:0

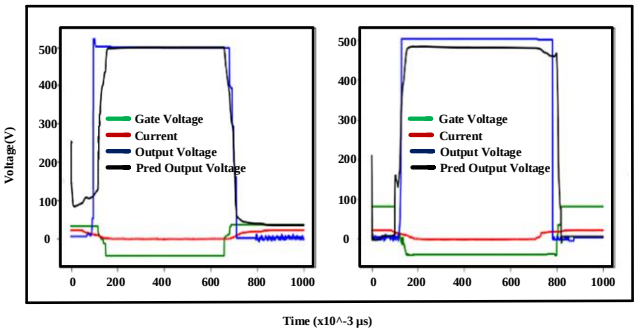


FIGURE 4. GaN GRU model training (Panasonic: left) (Transphorm: right)

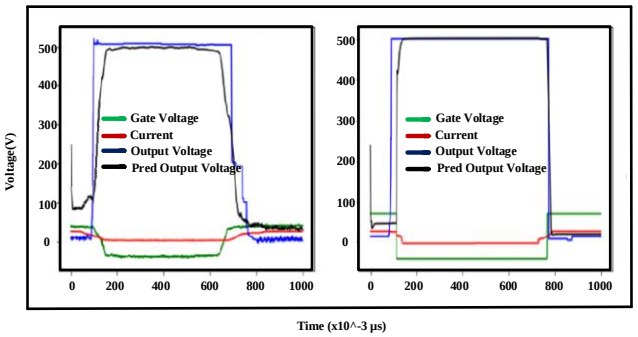


FIGURE 5: GaN GRU model prediction (Panasonic: left) (Transphorm: right)

LSTM Model: This testing was repeated using LSTM NN for the same set of data. The results obtained are very similar to the GRU model, with only minor differences in accuracy. The difference in accuracy is not much noticeable in the graphs due to the fact that we only have limited data for training and testing. For training this model, the following inputs and outputs are selected:

1) Inputs: Gate voltage, Input voltage, Digital voltage (ON/OFF), Device switching current
2) Output: Device switching voltage
3) No. of data sets used: 30 (training: 25, testing: 5) [consisting of both experimental and simulation data with a ratio of 30:70]
4) Epochs: 500
5) Type of NN used: LSTM
6) 4 inputs, 1 output, 1 hidden layer and 32 nodes are used
7) Gate voltage and device switching current is scaled by a factor of 10 while plotting

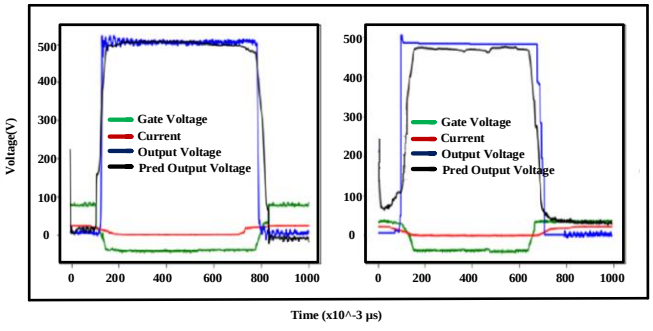


FIGURE 6. GaN LSTM model (Panasonic: left) (Transphorm: right)

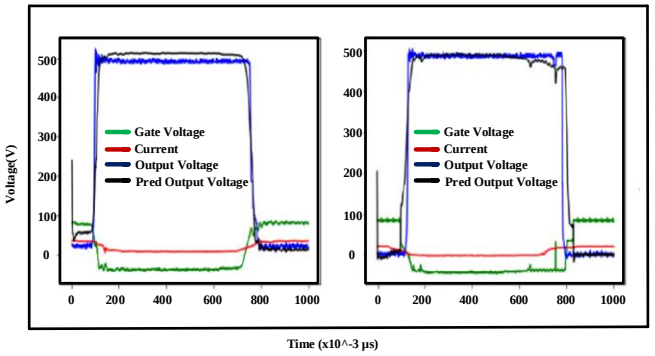


FIGURE 7. GaN LSTM model prediction (Panasonic: left) (Transphorm: right)

Plotting the predictions gives the following results, as shown in Figure 5,6. It can be seen that the ML model has very accurately predicted the oscillations, turn-on and turn- off time. There is only a small deviation concerning the magnitude.

It is seen from Table. II, this model has total trainable parameters of 12,833, which is less than the GRU model, all of which trains to learn how best to predict the GaN device switching. It can be noted that the ML model closely follows the training data in terms of the waveform shape, on and off timings, as seen in Figure 5 and Figure 6. For both model 1 and model 2, there are not many noticeable differences in the prediction voltages.

A. VALIDATION

There is a need to validate the demonstrated ML models. The logic used here is as follows:

1. The objective is to frame a model which is closer to actual test results than the ideal behaviour and with better accuracy than the proposed model behaviour.
2. In this work, the MSE is calculated against the DPT data for all the three devices. A comparison between the prediction error, the manufacturer model error and the proposed model error is made as seen in Table. III.
3. The lower the error, the better.

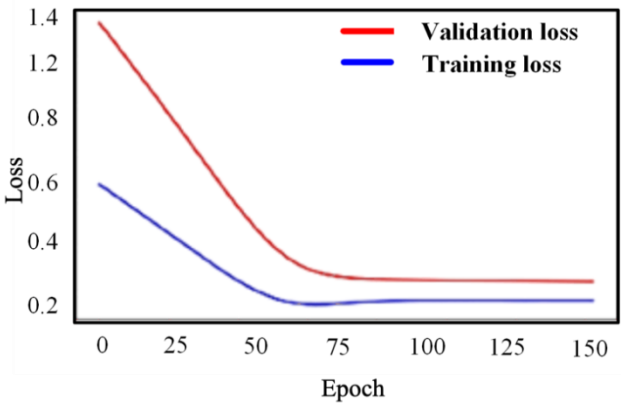
Table. III shows the comparison between the prediction error of GRU models with the proposed model. As is evident from Table III, the proposed model error is the lowest and is much closer to the actual experimental data, which is as expected. The ML model is not very close to the experimental data results as it is trained with data from multiple GaN devices and DPT tests. Its outputs values are discounting the effect of measurement and human error. ML model tries to predict the actual output of the GaN device for the given circuit without accounting for the measurement errors. Table. IV below shows the comparison between the prediction error of LSTM with other simulation models. Due to the lower error rate and lesser number of trainable parameters which leads to speedy simulation, the next sections will use RNN-LSTM based models for training. Figure 7 and Figure 8 graphically depicts the validation and training loss which is used for calculating prediction error.

Advantages of the shallow (one layer) model:

1) Simple
2) More comfortable to implement/run the simulation
3) Good performance

Disadvantages:

1) Memorization
2) Not good at generalizing



3) Non-scalable

FIGURE 8. Comparison of loss for RNN-GRU model

Deep/Multi NNs have more than one hidden layer. The advantage of numerous layers is that they can learn attributes at distinctive stages of abstraction. Based on the other layer's output, each layer of nodes trains on a distinct set of features/attributes. As we move deep into the neural net, they accumulate and re-join attributes from the previous layer and can recognize more complex attributes/features. This property termed as feature hierarchy makes deep-learning networks proficient of handling astronomical, high-dimensional data sets with zillions of parameters that pass through non-linear operations. Thus, these nets are adept at unearthing interconnections within unlabelled/unstructured data. Therefore, one of the issues deep learning resolves well is the processing and clustering of the world's raw data with insights into the similarities and variation in data in a relational database. For example, in this work, with each hidden layer, the model will learn specific features of the switching behaviour, in the next layer, it will learn about the DPT circuit, the next one about the parasitic etc. though not necessarily in this order.

TABLE III: Comparison of the prediction error of RNN-GRU model

Device Model	Prediction Error	Manufacturer Model Error	Proposed Model Error
Transphorm	0.1017385612225869	0.026574740757534	0.012041490771542818
Gansystems	0.05871962553303792	0.05913845983435681	4.989418096389726 e-6
Panasonic	0.2274280878218421	0.026096969917324058	0.011259930029712931

TABLE IV: Comparison of the prediction error of RNN-LSTM model

Device Model	Prediction Error	Manufacturer Model Error	Proposed Model Error
Transphorm	0.1017385612225869	0.026574740757534	0.012041490771542818
Gansystems	0.05871962553303792	0.05913845983435681	4.989418096389726 e-6
Panasonic	0.2274280878218421	0.026096969917324058	0.011259930029712931

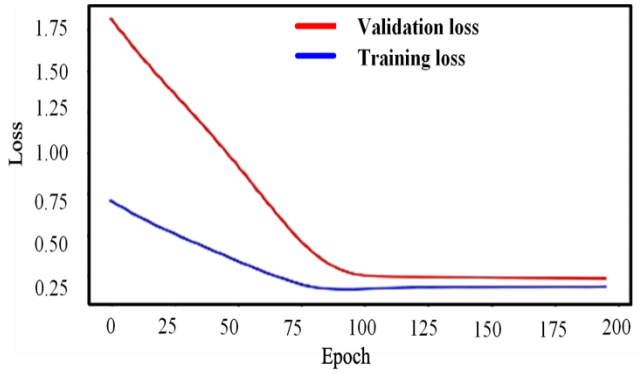


FIGURE 9. Comparison of loss for RNN-LSTM model

B. GAN HEMT BASED MODELLING USING MULTI NEURAL NETWORKS

Shallow networks are neural networks with one hidden layer, as shown in Figure 9 (left). A sufficiently broad shallow NN can approximate any function if provided with enough training data. Since we are dealing with PE-based applications, the data available is not very large, unlike classification and pattern recognition problems. But there are some complexities while using an extremely wide-shallow network such as the one used in this work. The first complication is that wide-shallow networks are high at memorization, but not that good at generalization. So, to ensure generalization and to reduce the number of parameters used, we explore multi NN models, as shown in Figure 9 (right).

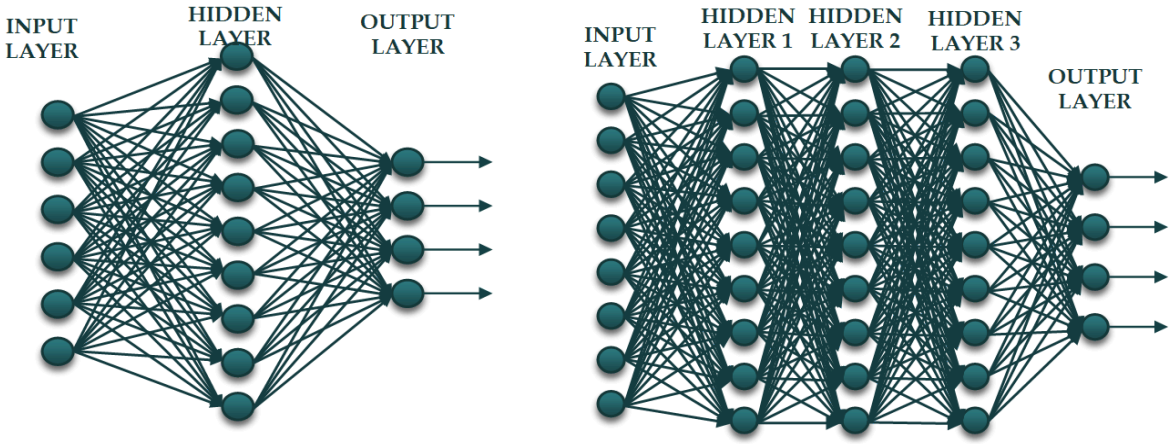


FIGURE 10. Shallow network (left) and multi feed-forward neural network (right)

In this part, training is done using RNN-LSTM network architecture to determine whether it can be used for predicting both the device voltages and currents. The input layer of all the models has 3 neurons, one for each feature. Since this is a regression problem, the output layer has one neuron with linear activation. All other layers have Rectified Linear Unit activations [30]. Adam optimizer was used during training [31], and the data were divided into batches of 500.

Predicting device switching voltage: For training this model, the following inputs and outputs are selected:

1) Inputs: Gate voltage, Input voltage, Device switching current

2) Output: Device switching voltage
3) No. of data sets used: 35 (training: 25; testing: 5) [consisting of both experimental and simulation data with a ratio of 30:70]
4) Epochs: 500
5) Type of NN used: LSTM
6) 3 inputs, 1 output, 2 hidden layers with 32 nodes each

Thus, here, the neural network model can be defined as:

Switching voltage = f (gate voltage, switching current)

So, the RNN-LSTM is trained to predict the switching voltage as a function of the gate voltage and the switching current. The ability of the model to use voltages and currents to be able to predict the output which could either be voltage/current is crucial. Unlike the NN models for microwave devices which can only act as voltage-controlled having only voltage as input and output, the ML models in this work can predict both voltages and currents and deals with both voltage and current inputs and outputs.

The RNN-LSTM model, with parameters shown in Table V, was trained using simulation and manufacturer test data. The data contained values for switch OFF and ON instants. The current measurements had noise issues, so, an extra set of 5 batches with improved current measurement was supplied for training. Besides, five batches of experimental data were set to part for validation.

TABLE V. GaN MULTI LSTM model architecture

Layer (type)	Output Shape	Parameters
lstm_11(LSTM)	(None, 1250, 32)	4480
lstm_2(LSTM)	(None, 1250, 32)	8320
dense_5(Dense)	(None, 1250, 1)	33
Total params:12,833	Trainable Params: 12,833	Non-Trainable Params:0

Plotting the voltage prediction for a set of training, the following waveforms, as shown in Figure 10, Figure 11 and Figure 12 are obtained for Transphorm, Panasonic and GaN Systems power devices.

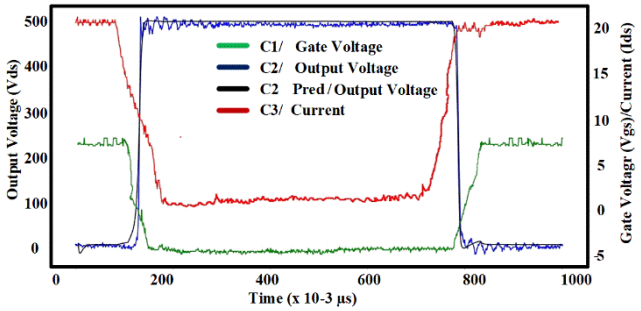


FIGURE 11. GaN ML model prediction for Transphorm Cascode HEMT

The predicted waveforms lack the oscillatory behaviour since it is fed with many manufacturer model waveforms while training which is more or less ideal waveforms. But unlike the previous models, this model very closely follows the experimental waveforms in terms of the on and off the rise, fall time and magnitude.

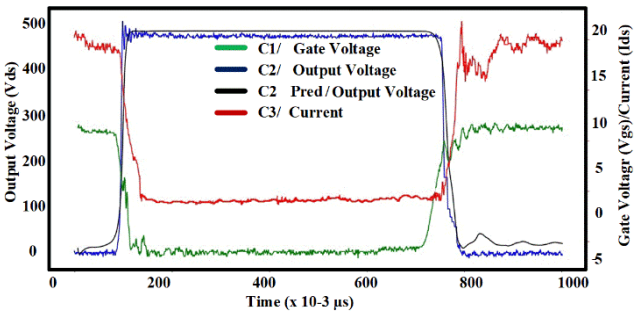


FIGURE 12. GaN ML model prediction for GaNSystems HEMT

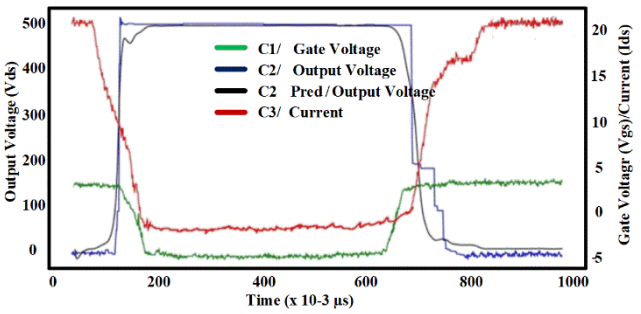


FIGURE 13. GaN ML model prediction for Panasonic GIT

Predicting device switching current: For training this model, the following inputs and outputs are selected:

1. Inputs: Gate voltage, Input voltage, Device switching voltage
2. Output: Device switching current
3. No. of data sets used: 35 (training: 25, testing: 5) [consisting of both experimental and simulation data with a ratio of 30:70]
4. Epochs: 1500
5. Type of NN used: LSTM
6. 3 inputs, 1 output, 2 hidden layers with 32 nodes each

Thus, here, the neural network model can be defined as switching current = f (gate voltage, switching voltage)

Here the RNN-LSTM model is trained to predict the switching current as a function of the gate voltage and the switching voltage. In the case of current, the noise in the DPT and with the waveform going negative, it was difficult to use the same logic of MSE used for validating the voltage prediction.

Plotting the current prediction for a set of training data, the following waveforms as seen in Figure. 13, Figure. 14 and Figure. 15 are obtained. It can be noted that the ML model closely follows the training data in terms of the waveform shape, but there is a deviation in the magnitude of the predicted current. The predicted waveforms lack the oscillatory behaviour for the same reason as in the case of

voltage prediction. Also, unlike the previous voltage prediction model, the current model does not carefully follow the experimental waveforms in terms of both the amplitude and shape. So, it is likely that the model was not able to learn the behaviour of current switching properly due to lack of noise-free training data.

Prediction of both Switching Voltage and Switching Current: For gaining familiarity with developing of NN based models for GaN, ease of programming and decreasing the training time involved, initially multiple-input, single-output RNN-LSTM and RNN-GRU based GaN models were developed. Now, to develop a complete black box/generic GaN-based behavioural model, it is necessary to output both switching voltages and currents at the same time. So, this section demonstrates the development of a generic behavioural model of a GaN HEMT that outputs switching voltage and switching current.

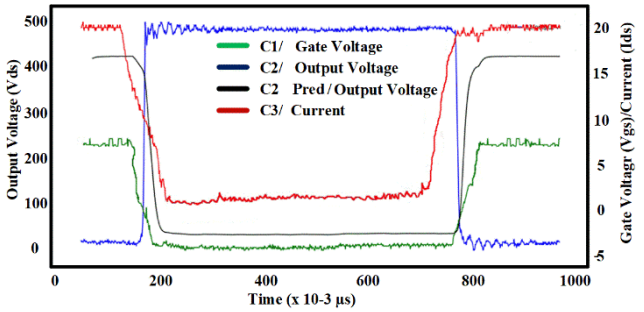


FIGURE 14. GaN ML model current prediction for Transphorm HEMT

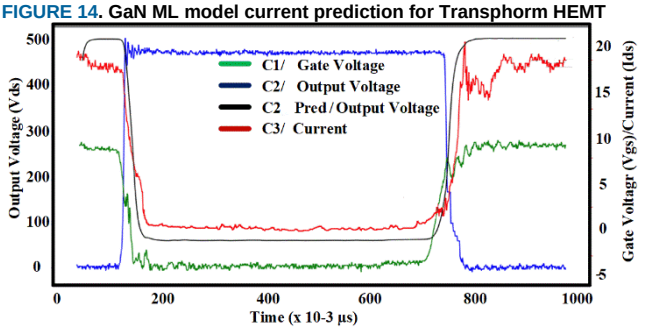


FIGURE 15. GaN ML model current prediction for GaNSystems HEMT

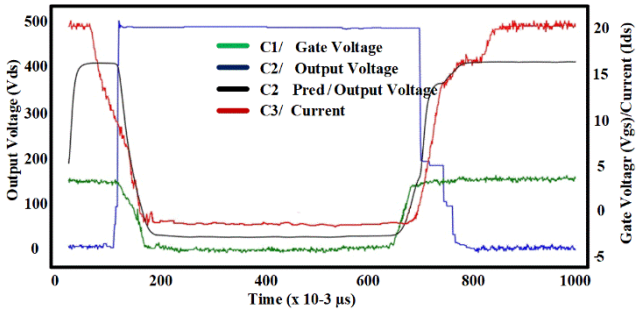


FIGURE 16. GaN ML model current prediction for Panasonic GIT

Since the current measurements obtained from DPT are slightly noisy and inaccurate, more accurate measurements of currents were taken and fed to this complete model for better training.

For training this model, the following inputs and outputs are selected:

1) Inputs: Gate voltage and Input voltage
2) Output: Device switching current and switching voltage
3) No. of data sets used: 35 (training: 25, testing: 5) [consisting of both experimental and simulation data with a ratio of 30:70]
4) Epochs: 1500
5) Type of NN used: Long short-term network (LSTM)
6) 2 inputs, 2 output, 2 hidden layers with 32 nodes each

Thus, here, the neural network model can be defined as:

(switching voltage, switching current) = f (gate voltage, input voltage)

TABLE VI. GaN RNN-LSTM model architecture

Layer (type)	Output Shape	Parameters
lstm_5(LSTM)	(None, 1002, 32)	4736
lstm_2(LSTM)	(None, 1002, 32)	8320
dense_3(Dense)	(None, 1002, 2)	66
Total params:13,122	Trainable Params: 13,122	Non-Trainable Params:0

RNN-LSTM model is trained to predict the switching voltage and current as a function of the gate voltage and input voltage. The architecture of the model employed is as shown in Table. VI. The number of trainable parameters is 13,122 and is slightly higher due to the extra node present for the output layer. From the predicted waveforms from Figure. 16, Figure. 17 and Figure. 18, it is evident that there is a tendency to predict idealized waveforms which, as explained before, is due to the large number of simulation waveforms fed during training.

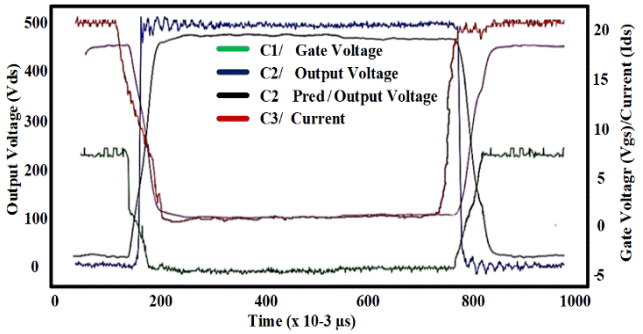


FIGURE 17. GaN ML model current and voltage prediction for Transphorm HEMT

As seen in Figure. 17, the prediction for current in case of GaN Systems HEMT is way below the measured magnitude. This is because of the volume of noisy current measurement

fed to the model as training data. Both of these limitations can be overcome if better DPT/experimental waveforms are fed during training. There is a limitation to gathering such a high volume of DPT results from a lab setting. Hence, the results are limited to the available data set for training and testing.

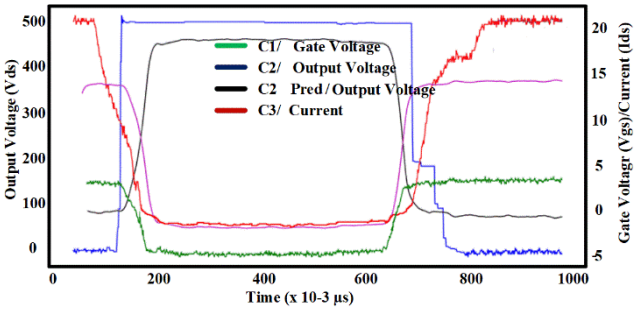


FIGURE 18. GaN ML model current and voltage prediction for GaNSystems HEMT

C. VALIDATION

Table. VII below shows the comparison between the prediction error of GaN ML models with the manufacturer models. As is evident from Table VII and Figure. 19, the proposed model error is the lowest. It is much closer to the actual experimental data, which is as expected. The ML model will not be very close to the experimental data results as it is trained with data from multiple GaN devices and DPT tests. So, it learned to negate the effect of measurement and human error. ML model has tried to predict the actual output of the GaN device for the given circuit without accounting for the measurement errors.

As is evident from Figure. 20 and Table. VIII, the training loss is high for current prediction. The prediction values were also not close enough to the expected values as detailed in the earlier section. It can be avoided by training the model with accurate and less variant current waveforms.

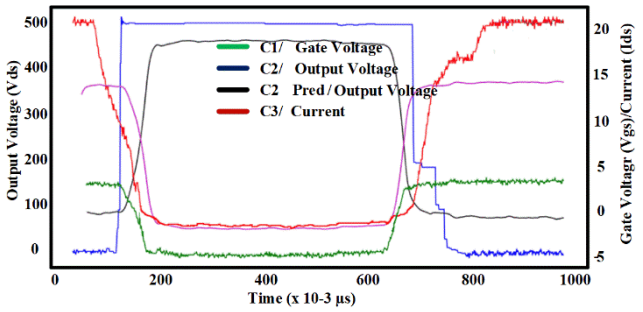


FIGURE 19. GaN ML model current and voltage prediction for Panasonic GIT

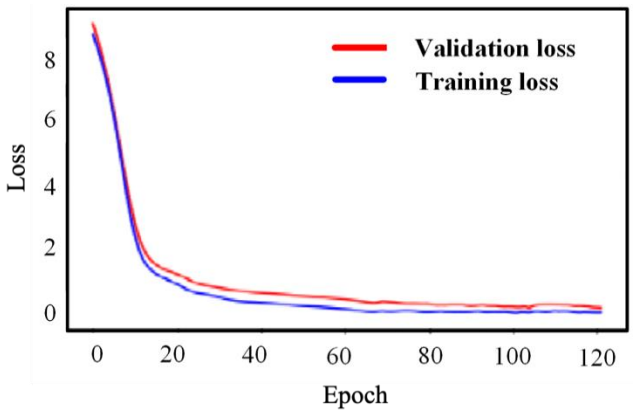


FIGURE 20. GaN Comparison of loss for LSTM voltage prediction model

The black box GaN ML model has a small loss. Training and validation loss are very close, and the model is fast and accurate. Thus, it is apparent that with a large volume of data, this model can be scaled up efficiently and made highly accurate and fast for speedy simulation and convergence time.

TABLE VII. GaN RNN-LSTM model architecture

Device Model	Prediction Error	Manufacturer Model Error	Proposed Model Error
Transphorm	0.17824583887744233	0.03329356763051398	0.040253270403163184
Gansystems	0.08003233255423499	0.019392490852331388	0.005827762329102095
Panasonic	0.07988175598975399	0.002823714782894578	0.0031721715965811028

TABLE VIII. Comparison of the current prediction error of LSTM model

Device Model	Prediction Error	Manufacturer Model Error	Proposed Model Error
Transphorm	0.16167914107258552	0.03329356763051398	0.040253270403163184
Gansystems	0.03633400042308652	0.019392490852331388	0.005827762329102095
Panasonic	0.07466878545118938	0.002823714782894578	0.0031721715965811028

IV. DISCUSSIONS – CONTRIBUTION

1) The proposed modelling using machine learning techniques are accurate, fast and more practical for power design engineers.
2) The total training computation time for LSTM took around 120 minutes with 4s for each epoch. For GRU it took 75 minutes with 3s for each epoch. The simulation running time for validation data took around (238- 240) ms/step for each sample input. This computation time is for an Intel(R), Core (TM), i5-6600 CPU at 3.30GHz with 48GB RAM.
3) ML modelling does not require detailed knowledge of the physics nor geometry of the device and is independent of any intrinsic device measurement errors.
4) It is noted that the predicted voltages are tending towards ideal behaviour prediction. It is due to the presence of a large number of simulated waveforms from manufacturer models which don't capture the parasitic of the circuit.
5) The variation in current is due to the inaccuracy associated with the measurement circuitry.
6) The demonstrated model has been explored using recurrent neural network models such as LSTM and GRU. It is found that LSTM models are accurate, but GRU models are faster.
7) Verification of the proposed models is performed by checking the ability of the NN model to generalize, i.e. to output targeted responses to values not used during training.
8) Shallow and multi-layer NNs are both used to model GaN to find the best fit.
9) Single output and multi-output models are demonstrated and validated.
10) The ability of the model to use both voltages and currents to be able to predict the outputs map the interrelationship between switching voltage, gate voltage and current. Its significant achievement compared to the existing NN models for microwave devices which are exclusively voltage prediction models.
11) This paper designs develops and demonstrates a generic universal black box behavioural model for different GaN devices using ML. The benefits include simplicity, accuracy and speedy simulation with fast convergence time.
12) The observed variation of the proposed model from the actual device is due to the lack of a considerable volume of data that is generally required for ML training. Nevertheless, this model is the best approximation for an accurate generic GaN behavioural model. These models can be scaled up and accuracy improved with training compared to currently available models.

V. CONCLUSIONS

This research demonstrates ML-based modelling for GaN power electronics. Different types of GaN ML models are derived, and their performance is demonstrated using state of the art neural network architectures. The developed voltage and current prediction models are based on long short-term memory unit (LSTM) and gated recurrent unit (GRU) models. Several parameters are quantified and compared for validating the models. They are the network architectures, parameters, training time, validation loss and error loss. The ML models are also compared with existing LT-Spice manufacturer models. Results show that a faster GaN ML model with an error rate of 0.03, and convergence at 3s with excellent stability can be developed.

The proposed ML models can be trained and scaled up for better accuracy using a larger volume of switching data. This research work is limited by the use of output voltage at 400 V, 200V and 100V and loads current at10-15A for GaN Systems, Transphorm and Panasonic GaN devices. However, this can be expanded by using a range of input voltages, load voltages and output voltages which can be recorded in steps and fed in for training. It helps the model better understand the device switching behaviour and increase prediction accuracy.

Having ML-based manufacturer models help speed up the learning curve, device simulation time and enable faster adoption of these novel devices by the power electronics engineers. Additionally, the ML-based GaN circuit models can also be scaled up by feeding data from different types of GaN power circuits used for different applications. Having accurate GaN device and circuit models help identify the suitability of a GaN device structure for a particular application. This would be highly beneficial for power designers in reducing the circuit simulation and prototyping time frames.

ACKNOWLEDGEMENT

This work is carried out as doctoral research funded by Google Europe, Cambridge University Nehru Trust and British Federation of Women Graduates. The authors would like to thank Edward Shelton (Senior Research Associate, Dept of Engineering, University of Cambridge), GaN Systems, Panasonic, Transphorm and Sanken for providing with the GaN device samples, circuits and for the technical support.

REFERENCES

[1] Lidow, A., GaN transistors Giving new life to Moore's Law, *Power Semiconductor Devices and IC's (ISPSD), IEEE 27th International Symposium*, vol., no., pp.1-6, 10-14 May 2015.
[2] H. A. Mantooth, K. Peng, E. Santi, and J. L. Hudgins, "Modeling of wide band gap power semiconductor devices part i. *IEEE T Electron Dev*,pp. 423433, Feb 2015.
[3] E. Santi, K. Peng, H. A. Mantooth, and J. L. Hudgins, Modeling of wide-band gap power semiconductor devices part ii *IEEE T Electron Dev*, pp. 434442, Feb 2015.
[4] Hari, N. (2019). Gallium Nitride Power Electronics using Machine Learning (Doctoral thesis).
[5] A.Lemmon,R.Graves. Parasitic extraction procedure for siliconcarbide power modules. *IEEE International Workshop on Integrated Power Packaging(IWIPP)*, pages 9194, May 2015.

[6] J. R. Shealy, J. Wang, and R. Brown. Methodology for small-signal model extraction of alganhemts. *IEEE T Electron Dev*, vol. 55 no. 7:16031613, July 2008.
[7] M. Mazzola, M. Rahmani, J. Gafford, A. Lemmon, and R. Graves. Behavioral modeling for stability in multi-chip power modules. In *2015 IEEE International Workshop on Integrated Power Packaging (IWIPP)*, pages 8790, May 2015.
[8] A. Lemmon, T. J. Freeborn, and A. Shahabi. Fixturing impacts on high- frequency low-resistance, low-inductance impedance measurements. *Electronics Letters*, vol. 52(21):17721774, 2016.
[9] Fang Wang, V.K. Devabhaktuni, Qi-Jun Zhang, "A hierarchical neural network approach to the development of a library of neural models for microwave design", *IEEE Trans Microw Theory Tech*, vol. 46, pp. 2391-2403, 1998.
[10] P.Sen,W.H.Woods,S.Sarkar,R.J.Pratap,B.M.Dufrene,R.Mukhopad-hyay, Chang-Ho Lee, E.F. Mina, J. Laskar, "Neural-network-based parasitic modeling and extraction verification for RF/millimeter-wave integrated circuit design", *IEEE Trans Microw Theory Tech*, vol. 54, pp. 2604-2614, 2006.
[11] Yunshen Long, Zheng Zhong, Yong-Xin Guo, A Novel 4-D Artificial- Neural-Network-Based Hybrid Large-Signal Model of GaAs pHEMTs", *IEEE Trans Microw Theory Tech*, vol. 64, pp. 1752-1762, 2016.
[12] A. Jarndal, "Genetic algorithm-based neural-network modeling approach applied to AlGaN/GaN devices", *Int. J. RF Microw. Comput.-Aided Eng.*, vol. 23, no. 2, pp. 149-156, 2013.
[13] Q. J. Zhang and K. C. Gupta, Neural Networks for RF and Microwave Design, Norwood, MA, USA:Artech House, 2000.
[14] M. Li, O. İrsoy, C. Cardie and H. G. Xing, "Physics-Inspired Neural Networks for Efficient Device Compact Modeling," in *IEEE Journal on Exploratory Solid-State Computational Devices and Circuits*, vol. 2, pp. 44-49, Dec. 2016
[15] B. K. Bose, Neural Network Applications in Power Electronics and Motor Drives: Introduction and Perspective, *IEEE Trans. Ind. Electron*., vol. 54, no. 1, pp. 14-33, Feb. 2007.
[16] M. Biglarbegian, M. Baharani, N. Kim, H. Tabkhi and B. Parkhideh, "Scalable Reliability Monitoring of GaN Power Converter Through Recurrent Neural Networks," 2018 IEEE Energy Conversion Congress and Exposition (ECCE), Portland, OR, 2018, pp. 7271-7277, doi: 10.1109/ECCE.2018.8557565.
[17] S. Khomfoi and L. M. Tolbert, "Fault Diagnosis and Reconfiguration for Multilevel Inverter Drive Using AI-Based Techniques," in IEEETransactions on Industrial Electronics, vol. 54, no. 6, pp. 2954-2968, Dec. 2007, doi: 10.1109/TIE.2007.906994.
[18] Michael A. Nielsen, Neural Networks and Deep Learning, in Deter-mination Press, 2015.
[19] K. Bakshi and K. Bakshi. Considerations for artificial intelligence and machine learning: Approaches and use cases. *2018 IEEE Aerospace Conference*, pp- 19, March 2018.
[20] C. E. Shannon. Computers and automata. *Proceedings of the IRE*, vol. 41, no.10 :12341241, Oct 1953.
[21] R. M. Friedberg. A learning machine: Part i.IBM *Journal of Research and Development*, vol. 2, no. 1:213, Jan 1958.
[22] M.Maron.Design principles for an intelligent machine. *IEEE Trans. Inform. Theory*, 8(5):179185, September 1962
[23] M. Minsky. Steps toward artificial intelligence.Proceedings of the IRE, 49(1):830,Jan 1961.
[24] C. Apte, L. Morgenstern, and Se June Hong. Ai at ibm research. *IEEE Intelligent Systems and their Applications*, 15(6):5157, Nov 2000
[25] D. B. Fogel. Imagining machines with imagination. Proceedings of the IEEE,88(2): 284288, Feb 2000.
[26] R. M. Friedberg, B. Dunham, and J. H. North. A learning machine: Part ii. *IBM Journal of Research and Development*, 3(3):282287, July 1959.
[27] H. L. Gelernter and N. Rochester. Intelligent behavior in problem-solving machines. *IBM Journal of Research and Development*, 2(4): 336345, Oct 1958.
[28] A. K. Jain, R. P. W. Duin, and Jianchang Mao. Statistical pattern recognition: a review, *IEEE Trans. Pattern Anal. Mach. Intell*., 22(1):437, Jan 2000.
[29] D. B. Cooper and J. H. Freeman. On the asymptotic improvement in the out-come of supervised learning provided by additional non supervised learning. *IEEE Trans. Comput.*, C-19(11): 10551063, Nov 1970
[30] B. Widrow and M. A. Lehr, 30 years of adaptive neural networks: perceptron,madaline, and backpropagation *Proceedings of the IEEE*, vol. 78, no.9, pp. 14151442, Sep. 1990.
[31] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," Proc. 27th International Conference on Machine Learning, 2010.
[32] S Hochreiter and Jurgen Schmidhuber, Long short-term memory, *Neural Comput*.,vol. 9, no. 8, pp. 17351780, Nov 1997.
[33] K Cho, et al, On the properties of neural machine translation: encoder- decoder approaches, in *arXiv preprint* arXiv, pp-1409.1259, 2014.
[34] G. Bebis and M. Georgiopoulos, Feed-forward neural networks, *IEEE Potentials*, vol, 13, no. 4, pp. 2731, Oct 1994
[35] G. E. Dahl, T. N. Sainath, and G. E. Hinton, "Improving deep neural networks for lvcsrusing rectified linear units and dropout", *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 86098613, May 2013.

Nikita Hari is currently a Postdoctoral Research Fellow in the Department of Engineering Sciences, University of Oxford. She received her PhD in Electrical Engineering from the Department of Engineering at the University of Cambridge in 2019. She is the cofounder of Wudi an AI based EdTech start-up and Favalley, a social EdTech initiative. She was a Gold Medalist for both her masters and undergraduate degree in Engineering. She has eight years of teaching and research experience with 20 peer-reviewed research papers and 42 tech keynotes to her credit. Nikita received 3 best research paper awards and 2 best poster presentation prizes for her doctoral research at EPSRC Annual Power Conference'16 & 17, GaN Power Challenge'18 and UK Power Electronics Summer School'15. Listed in the 'Top 50 Women in Engineering in the UK' in 2017, she is an invited Global Ambassador for the Queen Elizabeth Prize for Engineering. She was the first Chairperson of EPSRC UK Power Electronics Forum, IEEE Cambridge Secretary and is an IEEE and IET member for the past ten years. Her current multidisciplinary research interests include AI for power electronics, Gallium Nitride power electronics, ML for electrical engineering applications and AI interventions for digital education.

Mominul Ahsan is a Postdoctoral Researcher in the Department of Engineering, Manchester Metropolitan University. He completed his PhD in 2019 from the School of Computing and Mathematical Sciences at the University of Greenwich, London, UK. Mr. Ahsan has obtained his MEng degree by Research from the Faculty of Engineering and Computing at Dublin City University, Dublin, Ireland in 2014 and Bachelor Degree from the Department of Computer Science and Engineering at State University of Bangladesh, Dhaka, Bangladesh in 2008. Mominul's research interests include prognostics, data analytics, machine learning, reliability, power electronics and wireless communication. He is currently a member of Institution of Engineering and Technology (IET), associate member of Bangladesh Computer Society (BCS), a recipient of a PhD scholarship at the University of Greenwich in 2014, and Excellent Poster Award in the International Spring Seminar on Electronics Technology in 2017.

Sridhar Ramasamy is currently working an Associate Professor in Department of Electrical and Electronics Engineering, SRM Institute of Science and Technology, Chennai, India. He received his Bachelor of Engineering degree in Electrical and Electronics Engineering from A.K College of engineering (now Kalasalingam University), Srivilluputhur, Tamil Nadu, India and Master of Engineering degree in Power Electronics and Drives from SASTRA University, Thanjavur, India, in 2002 and 2004 respectively. He has extensive expertise in the field of power electronics and its interface with renewable energy systems. He received his PhD degree in the year 2017. His research interests include the application of artificial intelligence for power electronic converters and energy system applications. He has published 40 research papers in refereed journals and supervised several research projects.

Sanjeevikumar Padmanaban (M'12–SM'15) received the bachelor's degree in electrical engineering from the University of Madras, Chennai, India, in 2002, the master's degree (Hons.) in electrical engineering from Pondicherry University, Puducherry, India, in 2006, and the PhD degree in electrical engineering from the University of Bologna, Bologna, Italy, in 2012. He was an Associate Professor with VIT University from 2012 to 2013. In 2013, he joined the National Institute of Technology, India, as a Faculty Member. In 2014, he was invited as a Visiting Researcher at the Department of Electrical Engineering, Qatar University, Doha, Qatar, funded by the Qatar National Research Foundation (Government of Qatar). He continued his research activities with Dublin Institute of Technology, Dublin, Ireland, in 2014.

He was an Associate Professor with the Department of Electrical and Electronics Engineering, University of Johannesburg, Johannesburg, South Africa, from 2016 to 2018. Since 2018, he has been a Faculty Member with the Department of Energy Technology, Aalborg University, Esbjerg, Denmark. He has authored more than 300 scientific papers. S. Padmanaban was the recipient of the Best Paper cum Most Excellence Research Paper Award from IET-SEISCON'13, IET-CEAT'16, IEEE-EECSI'19, IEEE-CENCON'19 and five best paper awards from ETAEERE'16 sponsored Lecture Notes in Electrical Engineering, Springer book. He is a Fellow of the Institution of Engineers, India, the Institution of Electronics and Telecommunication Engineers, India, and the Institution of Engineering and Technology, U.K. He is an Editor/Associate Editor/Editorial Board for refereed journals, in particular the IEEE SYSTEMS JOURNAL, IEEE Transactions on Industry Applications, IEEE ACCESS, *IET Power Electronics*, and *International Transaction on Electrical Energy Systems, Wiley*, and the Subject Editor for the *IET Renewable Power Generation*, *IET Generation, Transmission and Distribution*, and *FACTS* journal (Canada).

Alhussein Albarbar is a Reader at the Department of Engineering, Manchester Met University. He has well over 27 years of industrial working experience and as an active academic researcher. Alhussein led and participated in over $7M of significant projects and supervised over 21 research degrees, including 15 doctoral studies. He has published 3 books, 5-book chapters, over 100 technical papers in refereed journals and international conference proceedings. His current research activities include Industry 4.0 applications, renewable power systems, smart sensing, intelligent control and monitoring algorithms used for electromechanical power plants.

Frede Blaabjerg (S'86–M'88–SM'97–F'03) was with ABB-Scandia, Randers, Denmark, from 1987 to 1988. From 1988 to 1992, he got the PhD degree in Electrical Engineering at Aalborg University in 1995. He became an Assistant Professor in 1992, an Associate Professor in 1996, and a Full Professor of power electronics and drives in 1998. From 2017 he became a Villum Investigator. He is honoris causa at University Politehnica Timisoara (UPT), Romania and Tallinn Technical University (TTU) in Estonia.

His current research interests include power electronics and its applications such as in wind turbines, PV systems, reliability, harmonics and adjustable speed drives. He has published more than 600 journal papers in the fields of power electronics and its applications. He is the co-author of four monographs and editor of ten books in power electronics and its applications.

He has received 32 IEEE Prize Paper Awards, the IEEE PELS Distinguished Service Award in 2009, the EPE-PEMC Council Award in 2010, the IEEE William E. Newell Power Electronics Award 2014, the Villum Kann Rasmussen Research Award 2014, the Global Energy Prize in 2019, and the 2020 IEEE Edison Medal. He was the Editor-in-Chief of the IEEE Transactions on Power Electronics from 2006 to 2012. He has been Distinguished Lecturer for the IEEE Power Electronics Society from 2005 to 2007 and for the IEEE Industry Applications Society from 2010 to 2011 as well as 2017 to 2018. In 2019-2020 he serves as President of IEEE Power Electronics Society. He is Vice-President of the Danish Academy of Technical Sciences too. He is nominated in 2014-2019 by Thomson Reuters to be between the most 250 cited researchers in Engineering in the world. In 2017 he became Honoris Causa at University Politehnica Timisoara (UPT), Romania.