

# Surrogate Thermal Model for Power Electronic Modules using Artificial Neural Network

1<sup>st</sup> Zhigen Xu  
School of Electronic and  
Electrical  
Dongguan Polytechnic  
Dongguan, China  
zhigen.xu@hotmail.com

2<sup>nd</sup> Yuan Gao  
Power Electronics, Machi-  
nes and Control Group, F  
aculty of Engineering  
University of Nottingham  
Nottingham, UK  
yuan.gao@nottingham.ac.  
uk

3<sup>rd</sup> Xin Wang  
Institute for Aerospace Te-  
chnology  
University of Nottingham  
Nottingham, UK  
xin.wang2@nottingham.a  
c.uk

4<sup>th</sup> Xiaoyu Tao  
School of Sciences  
Wuhan University of  
Technology  
Wuhan, China  
Tizzy.Tao@hotmail.com

5<sup>th</sup> Qingui Xu  
School of Computer  
Sciences  
Dongguan University of  
Technology  
Dongguan, China  
xqg@dgut.edu.cn

**Abstract**—Virtual prototyping of power electronic modules aims to allow rapid evaluation of potential designs without building and testing physical prototypes. Among the interests in thermal models of the virtual modules, process of compact thermal models needs effective methodology to fast generate small models describing the thermal performance of a potential design. This study chooses the Generalized Minimized Residual (GMRES) Algorithm to process thermal models due to its efficiency. Based on that, a machine learning aided surrogate model is proposed for the prediction of thermal performance since existing approaches take much time to determine the thermal response to a particular input power. This surrogate model is created by training a dedicated artificial neural network (ANN) on simulation data, after that this model can quickly map the module temperature and the power input in time domain. In the training process, cross-validation method is introduced to determine which neuron structure should be selected for the practical data generated by thermal equations. The test group is noted in cross-validation to give the prediction performance of structure candidates. To verify the proposed method, the resulting data of trained surrogate models are compared with the accurate simulation data after the ANN based cross-validation.

**Keywords**— *Artificial Neural Network (ANN), cross-validation, power electronic device (PED), thermal model, Generalized Minimized Residual (GMRES)*

## I. INTRODUCTION

Power electronic device (PED) is the core component of electrical power system (EPS) and it is now widely applied in not only electrical power transmission but electrical transportation [1]. However, it is usually operated under a high input/output power and produces considerable amount of heat, especially for that in rapid switching circuits. Therefore, PED is easy to be damaged for long-time usage and overheating owing to the limitation of their junction temperature [2]. Chen et al. [3] pointed out that repetitive thermal cycling leads to problems of PED like cracks, voids and delamination due to different CTEs (coefficients of thermal expansions). On the other hand, the trend of PED design and manufacture is to make it smaller with higher efficiency and longer lifetime which

highlights the importance of effective thermal management [4]. Namely, it is crucial to analyze, design and optimize the thermal models of PED for the best balance between the reliability, performance and other metrics. Fig. 1 shows a typical application of power electronic device. Most of power loss in PED is transferred to thermal energy heating it, referred as  $P_{loss}$  in this figure.

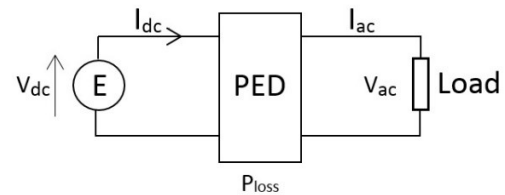


Fig.1. A typical application of PED

To address the thermal problems, virtual prototyping technique has been developed for predicting electrical and thermal effects of a power electronic module before constructing an actual prototype of PED [5]. This technique can help designers modify and re-simulate their thermal modules in a loop to finally get the best design. The design process mainly involves model generation, simulation and performance evaluation. Due to the high-frequency switching of PED and its application in the complex EPS, one of the challenges for this virtual technique is the ability to generate signals in a short time using limited computational resource. Compact thermal model is therefore utilized to build system simulation model from geometry data since it has high computational efficiency. Large system thermal equations can be modeled by finite-difference discretization and then reduced to smaller ones in order to get efficient computation [6].

Many researchers focus in the Design for Reliability (DfR) area have mostly been on investigating the thermal loading of power devices [7, 8]. Nevertheless, these investigations in the existing literature are made on PEDs whose design parameters are initially fixed. This means that every time a designer would

This study is supported by the Brand Subject Grant of Guangdong (Project Number:2016gzpp027)

like to check how different design parameters affect the temperature distribution, he would need to exercise the time-consuming simulations or experiments all over again. The aim of this paper is to bridge this research gap by building a dedicated surrogate thermal model based on the training of artificial neural network (ANN), a well-known method in artificial intelligence (AI). We take advantage of this surrogate model to estimate the temperature variation of any device with regards to input power and time. The estimation would be several orders of magnitude faster compared to running the detailed simulation model.

The rest of the paper is organized as follows: Section II describes the background and fundamentals of thermal model with applied simulation method. In Section III, basic ANN theory is briefly discussed and two different ANN designs are given for comparative studies. In addition, the cross-validation method of ANN model is introduced in Section IV, aimed at choosing the best structure according to the mean error of prediction. Lastly, the case study in Section V shows that the applied surrogate model can predict the temperature variation of a new input power with the error less than 0.1 °C after training the simulation data.

## II. THERMAL SIMULATION BASED ON COMPACT THERMAL MODEL

The studied thermal compact model for PED is based on heat equations. The relation among inner heat generation, material and thermal conduction is a conservation of energy problem which is presented in (1). This equation states that any internal heat generation of an object must keep balance with the sum of heat flux that responsible for object temperature changing and flowing through the object.

$$c\rho \frac{\partial T}{\partial t} + \nabla \cdot \vec{q} = q_{int} \quad (1)$$

where  $c$  is the object's material specific heat capacity, J/(kg×K);  $\rho$  is mass density, kg/m<sup>3</sup>;  $T$  is temperature, K;  $t$  is time, second;  $\vec{q}$  is heat flux vector, W/m<sup>2</sup>;  $q_{int}$  is object internal heat generation, W/m<sup>3</sup>.

According to Fourier's Law, heat flux can be expressed as (2), where  $k$  represents for the thermal conductivity of this object, unit W/(m×K), which can be assumed as a constant or piecewise constant in compact thermal model.

$$\vec{q} = -k\nabla T \quad (2)$$

Therefore,

$$c\rho \frac{\partial T}{\partial t} - k\nabla^2 T = q_{int} \quad (3)$$

For a 3-dementional Cartesian system, this formula is:

$$c\rho \frac{\partial T}{\partial t} - k \cdot \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} \right) = q_{int} \quad (4)$$

This formula discretizes thermal conduction problem spatially by separating it into smaller region and using the approximately adjacent regions substituting the spatial partial derivatives to simplify calculation. Finite difference method

(FDM) solves infinite continuous problem by converting it to finite point parameters. In this method, temperature distribution is divided into lots of discrete points and the central difference approximations are used to substitute the second order spatial derivatives. Assuming the distance between two adjacent discrete point (node) are  $\delta x$ ,  $\delta y$  and  $\delta z$  (for  $x$ ,  $y$ ,  $z$  direction respectively). Fig. 2 sketches the 2-dimensional structure of mesh near node  $i, j, k$  in  $Y$ - $Z$  plane, this node can be expressed by (5).

$$T(i, j, k) = (i\delta x, j\delta y, k\delta z) \quad (5)$$

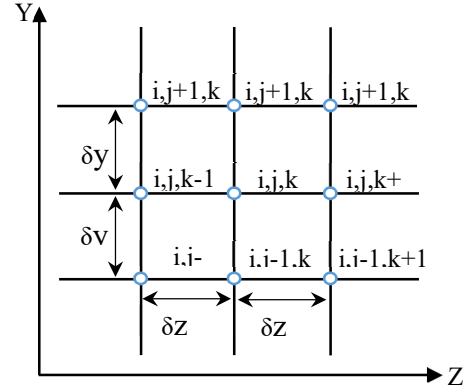


Fig.2. 2-dimensional node finite-difference mesh in  $Y$ - $Z$  Plane

For node  $i, j, k$ , the approximately first order temperature partial for  $z$ -derivatives are expressed by:

$$\frac{\partial T(i, j, k-1)}{\partial z} \approx \frac{T_{i, j, k} - T_{i, j, k-1}}{\delta z} \quad (6)$$

$$\frac{\partial T(i, j, k+1)}{\partial z} \approx \frac{T_{i, j, k+1} - T_{i, j, k}}{\delta z} \quad (7)$$

Then, second order  $z$ -derivative of central difference approximation can be given by:

$$\begin{aligned} \frac{\partial^2 T(i, j, k)}{\partial z^2} &\approx \frac{\partial T_{i, j, k+1}}{\partial z} - \frac{\partial T_{i, j, k-1}}{\partial z} \\ &= \frac{T_{i, j, k+1} + T_{i, j, k-1} - 2T_{i, j, k}}{(\delta z)^2} \end{aligned} \quad (8)$$

Similar equation can be found for  $x$ -derivative and  $y$ -derivative. Then (4) can be expressed as:

$$\begin{aligned} \delta v c \rho \frac{\partial T_{i, j, k}}{\partial t} - k \cdot \left( \frac{\delta y \delta z}{\delta x} (T_{i+1, j, k} + T_{i-1, j, k} \right. \\ \left. - 2T_{i, j, k}) + \frac{\delta x \delta z}{\delta y} (T_{i, j+1, k} \right. \\ \left. + T_{i, j-1, k} - 2T_{i, j, k}) \right. \\ \left. + \frac{\delta x \delta y}{\delta z} (T_{i, j, k+1} + T_{i, j, k-1} \right. \\ \left. - 2T_{i, j, k}) \right) = \delta v q_{i, j, k} \end{aligned} \quad (9)$$

In this equation  $\delta v$  represents for a small volume ( $\delta v = \delta x \delta y \delta z$ ). In a 3-d equivalent mesh, there is an electrical equivalent circuit existing in every node of the finite difference mesh, this equivalent circuit is the basic unit of the mesh. Meshing details can be found in Section 3.2.3 of [9]. For each

node there is a set of thermal resistances only depends of the material property.

Ultimately, the nodal temperatures can be described by a matrix equation, (10), in which  $M$ ,  $A$  and  $B$  are matrices representing for nodal thermal capacities, internode thermal conductivities and heat source respectively.  $\vec{U}$  is a vector which involves two elements, ambient temperature  $T_a$  and input power  $P_{in}$ .  $T$  is the instantaneous temperature which can be solved by utilising time-stepping integration algorithm in time-domain.

$$[M] \frac{d\vec{T}}{dt} - [A]\vec{T} = [B] \vec{U} \quad (10)$$

For steady state,  $dT/dt=0$ , thus the initial temperature  $T_0$  can be obtained. In time dependent situation,  $dT=T_n-T_{n-1}$ , (10) can be rewritten as (11), in which  $h$  is the time step between two frames,  $T_n$  is instantaneous temperature,  $T_{n-1}$  is the previous temperature.

$$([M] - h[A])T_n = h[A]T_{n-1} + h[B]\vec{U} \quad (11)$$

The time dependent temperature distribution of PED model can be found by solving (11). Since the matrices are very large, this process can be time-consuming. The matrices  $A$  and  $M$  in (11) are very sparse because there are only several elements per row are nonzero elements in  $[A]$ , while for  $[M]$ , only those elements on diagonal are non-zero. Therefore, Compressed Sparse Row (CSR) format is applied to reduce most of computational time and resources spent on zero multiplying other values. After using CSR, Generalized Minimal Residual Method (GMRES) is utilised to solve the matrix equation and generate instantaneous temperature distribution of this model. This method does not solve the equation directly but guessing the temperature  $T_n$  and multiplying the estimate value and matrix until the results meet the accuracy requirements. In summary, time consumption of GMRES depends on the sparse of matrices, the initial value of input vector and the tolerance requirement, thus it is a fast solution. More details can be found in [6].

In time domain, the temperature effect of PED will increase gradually from an initial temperature to the steady state temperature with regards to a certain input power. In this procedure, the average temperature of nodes in the studied module is used to plot the temperature variation under each power condition. Even applying GMRES in compact thermal model, time cost of simulation will probably be a little bit long because for every input power tens of thousand times of matrix operation need to be finished before the temperature reaching to the steady state, especially for the module which is divided into large number of small cubes.

### III. ANN BASED MODELING

As explained above, it is essential to establish a simple thermal model of power electronic module that would be able to translate the power and time data into temperature variation. The state of art approaches deal with this task by simulating the detailed model of electronic modules. However, the detailed

model is very complex thus the simulation usually suffers from the inefficient use of memory space. In order to come around this difficulty, this paper proposes the usage of a forward artificial neural network (ANN) to serve as a fast, dedicated and flexible surrogate model of electronic module.

ANN is based on a nonparametric regression model which is a technique for supervised learning. User does not need to specify the relationship between the predictors (input) and responses (output) with a predetermined regression function since ANN will learn them automatically by using only several training parameters (i.e. weights and bias).

#### A. ANN fundamentals

This study selects forward ANN as the surrogate thermal model. Although feedforward ANN is the simplest type of ANN devised, it has been applied already to various electrical engineering problems, from predicting the voltage distortion in electrical distribution networks [10], to reliability study of power electronic systems [11].

A basic forward ANN comprises an input layer, one or more hidden layers, and an output layer. To calculate the output of a certain neuron  $n_i^l$  in layer  $l$  ( $l > 1$ ), the outputs of all the neurons  $p_j^{l-1}$  ( $j = [1..N_{l-1}]$ ,  $N_{l-1}$  is the neuron number of Layer  $l-1$ ) are multiplied with given weights  $\omega_{ij}^l$  and then the bias  $b_i^l$  is added. The result is processed through an activation function  $f_\sigma$  that usually takes the form of a sigmoid function, i.e.  $f_\sigma(z) = 1/(1 + e^{-z})$ , to generate the output  $p_i^l$ . This output then becomes one of the inputs for the next layer,  $l+1$ , and the same procedure is repeated to calculate the output of other neurons in layer  $l$ .

In Layer 1 (input layer),  $p_i^1$  takes the form of inputs through the neuron  $n_i^1$ , no bias in this layer. On the other hand, Layer  $L$  (output layer) typically uses the linear activation function to integrate signal(s) of Layer  $L-1$  for the desired output data  $p_i^L$ . In summary, the complete signal flow of ANN can be described as follows:

- Layer 1:

$$p_i^1 = x_i, i = 1, \dots, N_1 \quad (12)$$

where  $x_i$  are the inputs.

- Layers  $l = 2, \dots, L-1$  (hidden):

$$p_i^l = f_\sigma \left( \sum_{j=1}^{N_{l-1}} \omega_{ij}^l p_j^{l-1} + b_i^l \right), i = 1, \dots, N_l \quad (13)$$

- Layer  $L$  (output):

$$y_i = \omega_i^L p_i^L, i = 1, \dots, N_L \quad (14)$$

where  $y_i$  are the outputs.

#### B. Deployment of ANN for the thermal model

This section elaborates the deployment of two dedicated ANNs, both of them serve as a surrogate model of the studied electronic module and aim to predict the accurate temperature variation with regard to input power and time data; however, in

order to obtain a better prediction performance, this study proposes ANN<sub>2</sub> by using temperature gradient value as another output based on ANN<sub>1</sub> whose output layer has only one neuron. It should be noted that the step of input time data is small and stays unchanged for all power inputs thus the difference between the temperatures of current and next sampling time can be directly regarded as the gradient at each moment.

#### 1) ANN<sub>1</sub>:

The first neural network (labeled as ANN<sub>1</sub>) follows the basic thinking of power electronic module regarding input-output data. The purpose of this network is to map the operating condition (input power,  $P_{in}$ ) and time variable ( $t_{sim}$ ) into the junction temperature ( $T_0$ ). The data of three variables required to train this network is collected by running a detailed simulation model of the PED module for many times to cover some specific range of input parameter variations.

Concerning the structure of ANN<sub>1</sub>, it comprises an input layer, two hidden layers, and an output layer, as shown in Fig. 3(a). The numbers of neurons in the input layer and output layer are 2 and 1 ( $N_1=2, N_4=1$ ); however, the neuron numbers of the hidden layers are not given at the beginning since they should be decided by training the specific practical data via cross-validation. This will be discussed in the Section IV.

#### 2) ANN<sub>2</sub>:

Inspired by ANN<sub>1</sub> and temperature variation, another network (labeled as ANN<sub>2</sub>) is proposed to pursue better performance of network prediction. As shown in Fig. 3(b), the difference between these two networks locates in the output layer ( $N_4$ ) where there are two neurons in this layer of ANN<sub>2</sub> because it is of interest to find out two variables that characterize the junction temperature: real-time temperature value ( $T_0$ ) and the variation gradient ( $\Delta T$ ). After simulation, the original data should be processed for giving the gradient with regards to the time variable. Based on that, the structure of hidden layers in ANN<sub>2</sub> can be determined by training the processed data via cross-validation.

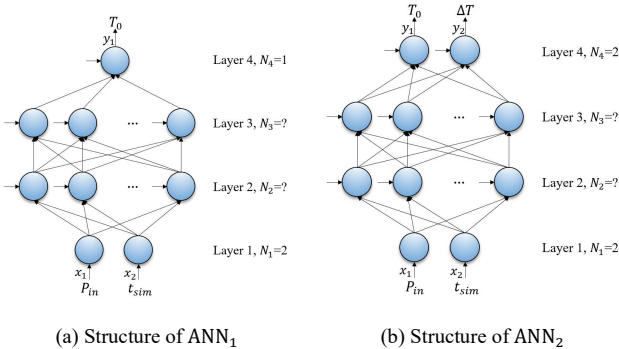


Fig. 3. Structures of two studied ANNs. Layer 1 is the input layer, Layer 2 and Layer 3 are hidden layers (neuron numbers to be determined by cross-validation), Layer 4 is the output layer. For simplicity, weights and bias terms are omitted from the figure.

## IV. PROPOSED SURROGATE MODLE BASED ON CROSS-VALIDATION

Regarding the structure of the network, if too few neurons are used, the strong nonlinear relationships may not be captured. On the other hand, overfitting may occur in ANNs with too many neurons. ANN structures in many studies are selected using trial-and-error method [11]. However, cross-validation can give a relatively objective criterion for the structure selection. That will be discussed as follows:

### A. Cross-validation

Cross-validation based ANN training is the way to find that using which structure the prediction performance is the best, namely the structure is most suitable for the given input-output data. The applied procedure of cross-validation is shown in Fig. 4. The overall procedure comprises 4 steps:

a). The first step is to process simulation results for ANN training following the structure design. This step can be divided into two aspects: input data and output data. Regarding the former, both ANN<sub>1</sub> and ANN<sub>2</sub> have two variables,  $P_{in}$  and  $t_{sim}$ ; however, there are two variables ( $T_0, \Delta T$ ) for the output layer of ANN<sub>2</sub> while only  $T_0$  for ANN<sub>1</sub>. Therefore, original simulation data should be collected and processed properly for these two networks.

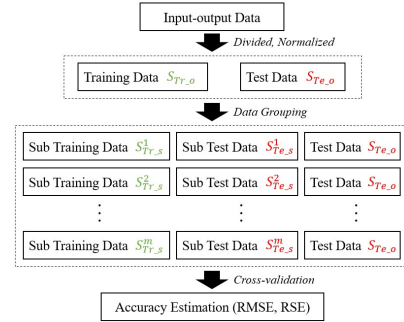


Fig. 4. Cross-validation procedure for choosing the best ANN structure

b). Normalization and division of input-output data are located in the second step. Firstly, the data of each variable should be normalized separately for the following cross-validation work. Furthermore, in order to better evaluate the prediction performance of the ANN candidates, the input-output data should be divided into two original groups: training group ( $S_{Tr_o}$ ) and test group ( $S_{Te_o}$ ), in which the test group will not be used for training during the whole process.

c). Concerning the cross-validation, further grouping of original training data needs to be done by dividing the training data  $S_{Tr_o}$  into 2 different subgroups for  $m$  times. Assume that there are  $N_s$  samples in above two original groups ( $S_{Tr_o}$  and  $S_{Te_o}$ ), divide  $S_{Tr_o}$  into  $m$  sub groups followed by setting each sub group as the sub test data ( $S_{Te_s}^i, 1 \leq i \leq m$ ) one by one. After each  $S_{Te_s}^i$  has been confirmed, the remaining data in  $S_{Tr_o}$  constitute the related sub training data ( $S_{Tr_s}^i, 1 \leq i \leq m$ ) with  $N_s(1 - 1/m)$  samples. Therefore, the grouping work in

this step is only for  $S_{Tr,o}$  because the other group ( $S_{Te,o}$ ) is only responsible for performance test.

d). This step is to train the sub training data from  $S_{Tr,s}^1$  to  $S_{Tr,s}^m$ . ANNs are trained using the train command, which is a part of Matlab's Deeping Learning Toolbox. It is noted that  $S_{Tr,s}^i$  is randomly divided into three data sets (training set, validation set and testing set) by train command in Matlab for confirming the termination condition. After the training stops, the trained ANNs will be tested using the corresponding  $S_{Te,s}^i$  and the original test data  $S_{Te,o}$ . The used index for the evaluation of prediction performance is Root Mean Square Error (RMSE) which is a popular formula to measure the error rate of a regression model.

### B. ANN selection for surrogate model

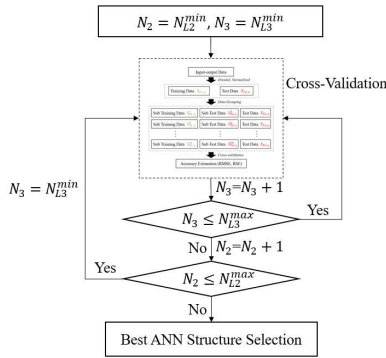


Fig. 5. ANN structure selection

As discussed in Section III.B, both ANN<sub>1</sub> and ANN<sub>2</sub> need to select proper numbers of neuron for Layer 2 and Layer 3 via the cross-validation. For the selection pool, we can just try different structures evenly within a certain range and record the related prediction performance (RMSE). The best structure can finally be chosen according to the minimum principle. The basic process of ANN structure selection is shown in Fig. 5.

Firstly, give ranges  $[N_{L2}^{min}, N_{L2}^{max}]$  and  $[N_{L3}^{min}, N_{L3}^{max}]$  to  $N_2$  and  $N_3$  separately. Then, try all pairs (structure candidates) within these two ranges one by one; for every candidate, set corresponding number values to ANN training model. After that, train and test ANN using the proposed cross-validation which results in the RMSE value for each network candidate. Lastly, choose the best structure whose RMSE is the minimum among the selection pool.

It should be noted that this selection process needs to be exercised twice as there are two ANNs in this study and their output data have difference architectures (see Fig. 3). The final ANN<sub>1</sub> and ANN<sub>2</sub> selected by training the simulation data can both serve as the overall representation of the thermal model for the targeted PED module.

## V. CASE STUDY

The PED studied here is a power diode whose model was divided into 724 meshes in simulation. The input power is from

1 W to 7 W, and the ambient temperature is 0°C. The time consumption for a PC equipped with a i5-6200U CPU at dual core 2.3Ghz is about 22 minutes for each power input.

The original data were given by simulating the detailed model of the PED module with associated thermal network for several selected  $P_{in}$  and extracting the corresponding  $T_0$  with regards to time variable ( $t_{sim}$ ). Concerning the cross-validation work, the data used for ANN training ( $S_{Tr,o}$ ) is shown in Fig. 6 and the data of 3 W is chosen as  $S_{Te,o}$ , which means it was not involved in ANN training but only for testing the trained ANN.  $S_{Tr,o}$  was then divided into 10 (value of  $m$ ) sub testing groups. Setting each of them as  $S_{Te,s}^i$  ( $1 \leq i \leq 10$ ) one by one while the remaining data in  $S_{Tr,o}$  must be  $S_{Tr,s}^i$ . After the grouping work, for each ANN structure design,  $S_{Tr,s}^i$  is utilized for training ANN while  $S_{Te,s}^i$  and data of 3 W ( $S_{Te,o}$ ) are for testing. Lastly, RMSE was recorded for each corresponding ANN structure.

For every input power, time variable was swept from 0.8 to 80000 s with a step of 0.8 s. As there are 7 different power conditions, the total number of sample points in this case is  $100000 \times 7$ .

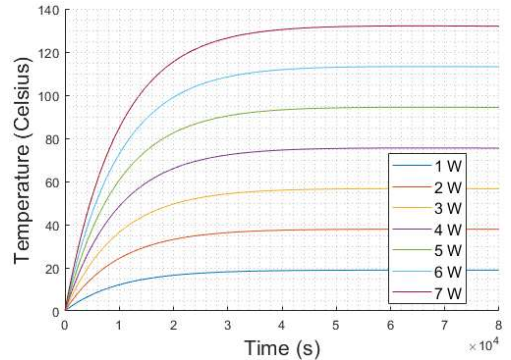


Fig. 6. Original training data given by simulation

TABLE I. NO. OF STRUCTURE CANDIDATES FOR ANN<sub>1</sub> AND ANN<sub>2</sub>

Table Head	Serial number of ANN structure candidates with regards to neuron number								
No.	1	2	3	4	5	6	7	8	9
$N_2$	2	2	2	2	2	3	3	3	3
$N_3$	2	3	4	5	6	2	3	4	5
No.	10	11	12	13	14	15	16	17	18
$N_2$	3	4	4	4	4	4	5	5	5
$N_3$	6	2	3	4	5	6	2	3	4
No.	19	20	21	22	23	24	25		
$N_2$	5	5	6	6	6	6	6		
$N_3$	5	6	2	3	4	5	6		

Regarding the best structure selection for ANN<sub>1</sub> and ANN<sub>2</sub>, the rang [2, 6] were given to both networks. The reason of not setting  $N_{L2}^{min}$  and  $N_{L3}^{min}$  as 1 is that there are two neurons in the input layer which determines that the nonlinear relationships between input and output should not be captured by that



structure. Therefore, there are 25 structure candidates in the selection pool, as shown in Table 1.

The next step is to explore all pairs of  $N_2, N_3$  and calculate RMSE for each structure following the process discussed in Section IV.B. As mentioned above, the structure selection should be exercised for ANN<sub>1</sub> and ANN<sub>2</sub> separately as they have different network architectures. The RMSE results of all structure candidates for ANN<sub>1</sub> and ANN<sub>2</sub> are shown in Fig. 7. In this figure, if the value of ANN RMSE is larger than 6 °C, the structure was distributed to an Up Group; if not, distributed to the Down Group. Therefore, the blue points of Fig. 7 are in the Up Group while the red points are in the Down Group.

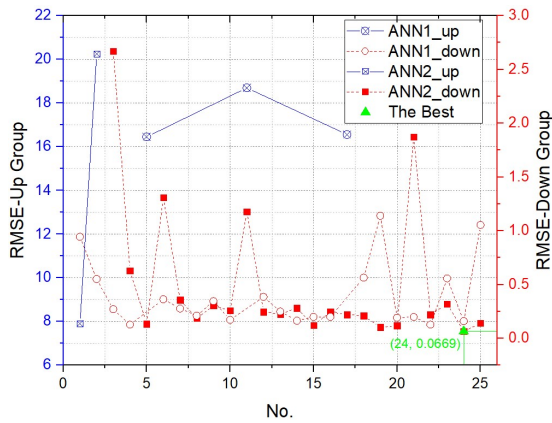


Fig. 7. RMSE results of 25 structure candidates for ANN<sub>1</sub> and ANN<sub>2</sub>

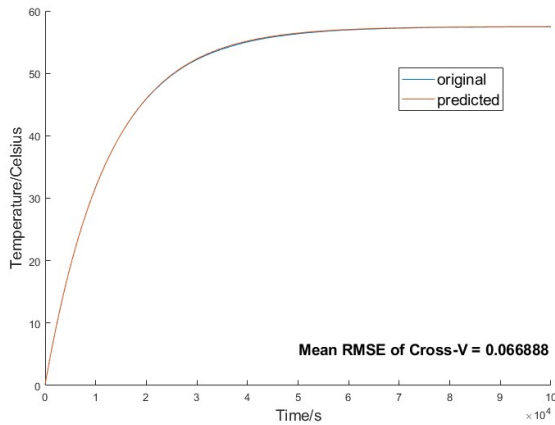


Fig. 8. Prediction result of 3 W data using the best structure. There are two neurons in the output layer,  $N_2 = 6, N_3 = 5$ .

The best structure for ANN<sub>1</sub> is No. 22 ( $N_2 = 6, N_3 = 3$ ) whose RMSE value is 0.1273; Regarding 25 structure candidates for ANN<sub>2</sub>, the best structure is No. 24 ( $N_2 = 6, N_3 = 5$ ). The RMSE value is 0.06689 °C and it is the only candidate whose RMSE is smaller than 0.1. On the other hand, average RMSE of ANN<sub>2</sub> structures is 1.5836 °C while average RMSE of ANN<sub>1</sub> is 2.4070 °C. Therefore, the RMSE results demonstrate that the overall prediction performance of ANN<sub>2</sub> is better than ANN<sub>1</sub>, and the final surrogate thermal model

selected for the studied diode model is the No. 24 structure of ANN<sub>2</sub>. Fig. 8 shows the predicted temperature variation of 3 W input power using the best structure. Compared with the original data, the prediction performance is excellent.

## VI. CONCLUSIONS AND FUTURE WORK

This paper proposes a method of ANN aided thermal prediction for power electronic devices and gives a case study on a power diode. In addition, two different architectures of ANN are designed for the surrogate thermal model and cross-validation is noted to choose the best ANN structure for the thermal data given by the simulation system.

The future study will focus on the data processing for ANN training and the application of other methods in AI, e.g. support vector machine (SVM). Moreover, sensitivity analysis of training parameters in ANN for better prediction performance is also worth of studying.

## REFERENCES

- [1] F. Blaabjerg, M. Liserre, and K. Ma, "Power electronics converters for wind turbine systems," IEEE Trans. Ind. Appl., vol. 48, pp. 708–719, March 2012.
- [2] Xiang, Y., Zhou, Z., Mo, L., Hui, L. and Wu, F., "Simulation of the temperature field for bonding IGBT chip and DBC substrate using Al/Ni self-propagating foil". In Electronic Packaging Technology (ICEPT), 2017 18th International Conference on (pp. 1016-1020). IEEE. August, 2017.
- [3] Chen, H., Ji, B., Pickert, V. and Cao, W., "Real-time temperature estimation for power MOSFETs considering thermal aging effects". IEEE Transactions on Device and Materials Reliability, 14(1), pp.220-228. Mar, 2014.
- [4] Davidson, J.N., Stone, D.A. and Foster, M.P., "Real-time temperature monitoring and control for power electronic systems under variable active cooling by characterisation of device thermal transfer impedance", 2014
- [5] Fan, X. J., J. Zhou, and G. Q. Zhang. "Multi-physics modeling in virtual prototyping of electronic packages – combined thermal, thermo-mechanical and vapor pressure modeling." Microelectronics Reliability 44.12 (2004): 1967-1976.
- [6] Evans, P.L., Castellazzi, A. and Johnson, C.M., "Automated fast extraction of compact thermal models for power electronic modules". IEEE Transactions on Power Electronics, 28(10), pp.4791-4802. Oct, 2013.
- [7] A. Anurag, Y. Yang, and F. Blaabjerg, "Thermal performance and reliability analysis of single-phase pv inverters with reactive power injection outside feed-in operating hours," IEEE Jour. Emerg. Sel. Topics Power Electron., vol. 3, pp. 870–880, Dec 2015.
- [8] M. Andresen, G. Buticchi, and M. Liserre, "Thermal stress analysis and mppt optimization of photovoltaic systems," IEEE Trans. on Ind. Electron., vol. 63, pp. 4889–4898, Aug 2016.
- [9] Evans, P.L., "Fast multidisciplinary analysis and virtual prototyping of multi-chip power module based systems". PhD thesis, University of Nottingham, Dec 2011.
- [10] B. Singh, V. Verma, and J. Solanki, "Neural network-based selective compensation of current quality problems in distribution system," IEEE Trans. on Ind. Electron., vol. 54, pp. 53–60, Feb 2007.
- [11] T. Dragicevic, P. Wheeler and F. Blaabjerg, "Artificial Intelligence Aided Automated Design for Reliability of Power Electronic Systems," in IEEE Transactions on Power Electronics. doi: 10.1109/TPEL.2018.2883947