Letters

An Efficient Electrothermal Coupling Simulation Method Based on Neural Network-Aided Power Loss Model for Power Module Thermal Optimization

Yayong Yang, *Student Member, IEEE*, Zhiqiang Wang, *Senior Member, IEEE*, Lingqi Tan, *Student Member, IEEE*, Guoqing Xin, *Member, IEEE*, Wubin Kong, and Xiaojie Shi, *Senior Member, IEEE*

***Abstract*—This article introduces a highly efficient electrothermal coupling simulation method that leverages a data-driven neural network power loss model to enhance the simulation efficiency in the thermal optimization of power modules. First, methods to build the neural network power loss model and to extract data automatically for neural network training are introduced. Then, an indirect coupling strategy is applied for the bidirectional coupling between the power loss model and COMSOL thermal model. To address the challenge of balancing simulation accuracy and efficiency in the proposed coupling strategy, an adaptive time step adjustment algorithm based on Lagrange interpolation functions is designed. Finally, quantitative comparisons of different simulation methods and experiments validate the high precision and efficiency of the proposed method. The results indicate that the proposed method achieves the same level of simulation accuracy as existing methods, while improving simulation efficiency by approximately five to seven times compared to the thermal network method and by about two to five times compared to the COMSOL-PSpice method across 100 cases.**

***Index Terms*—Electrothermal coupling simulation, neural network, power loss, power module, simulation efficiency.**

I. Introduction

THERMAL simulation plays a crucial role in the design and optimization of power semiconductor modules. With low thermal simulation accuracy, the design will likely be too conservative for reliability, which impedes the improvement of power density. To ensure sufficient simulation accuracy, electrothermal coupling simulations that account for the bidirectional coupling relationship between temperature and power losses are imperative [1].

The electrothermal coupling simulation methods that have been proposed can be categorized into two groups: reduced-order-based and nonreduced-order-based. In the reduced-order-based electrothermal simulation, reduced-order thermal network models are widely utilized due to their ease of implementation in the circuit simulator [2]. The 1-D thermal network model is mainly suited for single-chip applications, limiting its practicality. In contrast, 2-D and 3-D models consider thermal coupling between multiple chips, offering greater accuracy [3]. However, extracting parameters for 2-D or 3-D models requires numerous finite-element method (FEM) simulations or experiments, which scale significantly with the number of heat sources [4]. Furthermore, in simulation scenarios where thermal models differ in each case, as in thermal optimization, multiple FEM simulations are necessary for each case to generate a corresponding thermal network model tailored to that specific scenario. This method is ideal when the thermal network model stays constant while the circuit model changes, allowing the thermal network to be reused across multiple scenarios and saving simulation time. However, if the thermal model varies in each scenario, the approach becomes complex, cumbersome, and time-consuming. In [5], a nonreduced-order-based filed-circuit coupling method with a MATLAB script to coordinate data exchange between PSpice circuit simulation and COMSOL thermal simulation is proposed. While this approach can fully leverage the strengths of commercial software to enhance simulation accuracy, the data exchange between the two different software platforms is highly complex and challenging to implement. When the circuit simulation model remains unchanged, as in thermal optimization where the circuit model is mostly fixed, this approach requires rerunning circuit simulations for each electrothermal analysis. With a high volume of simulations, repeatedly conducting circuit simulations becomes time-consuming, reducing overall simulation efficiency.

To avoid repetitive circuit simulations in power module thermal optimization and improve simulation efficiency, this letter introduces a reduced-order-based electrothermal modeling method with a specialized interface tool that links a neural

Received 31 August 2024; revised 9 October 2024; accepted 13 November 2024. Date of publication 18 November 2024; date of current version 26 December 2024. This work was supported by the National Natural Science Foundation of China under Grant 52277179. *(Corresponding author: Zhiqiang Wang.)*

Yayong Yang is with the Institute of Artificial Intelligence, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: yangyayong@hust.edu.cn).

Lingqi Tan is with the Electric Power Research Institute of Guangdong Power Grid Company, Ltd., Guangzhou 510030, China (e-mail: tanlingqi@gddky.csg.cn).

Zhiqiang Wang, Guoqing Xin, Wubin Kong, and Xiaojie Shi are with the School of Electrical and Electronic Engineering, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: zhiqiangwang@hust.edu.cn; guoqingxin@hust.edu.cn; wbkong@hust.edu.cn; xiaojie_shi@hust.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TPEL.2024.3501584.

Digital Object Identifier 10.1109/TPEL.2024.3501584

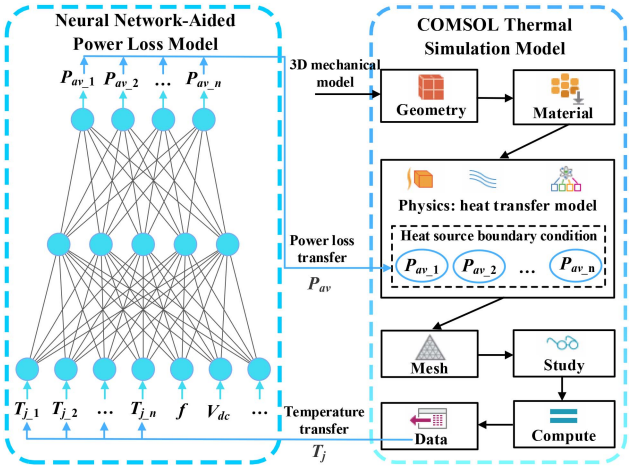


Fig. 1. Proposed thermal simulation method based on neural network-aided power loss model.

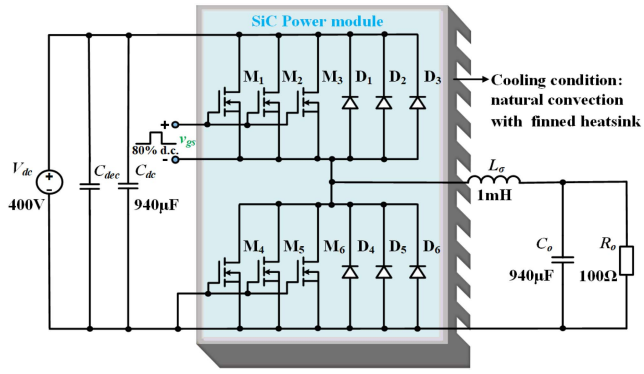


Fig. 2. Circuit schematic and cooling condition of the buck inverter.

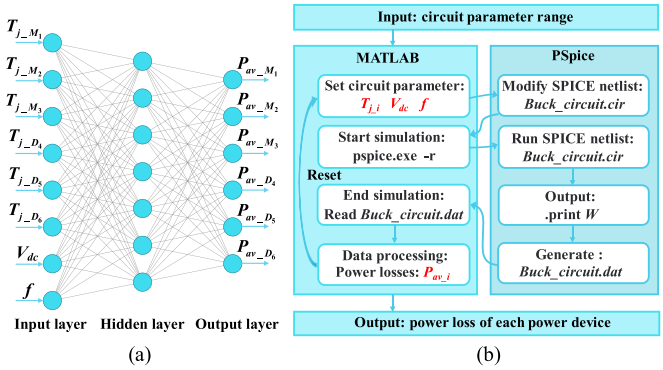


Fig. 3. Neural network power loss model and its automated training data extraction method. (a) Neural network-aided power loss model. (b) Automated training data extraction method.

network-aided power loss model to the COMSOL thermal simulation model. By utilizing this dedicated interface with an adaptive time step adjustment strategy, a balance between the simulation accuracy and speed can be achieved.

II. Proposed Methodology

An overview of the proposed coupling simulation method is illustrated in Fig. 1. In the proposed coupled simulation framework, the neural network-aided power loss model, with its powerful nonlinear fitting capability, automatic feature extraction, scalability, and flexibility, replaces the circuit model and directly integrates with the COMSOL thermal model. This framework eliminates the need for repeated circuit simulation and thermal network generation in the multicase simulations where the circuit model remains essentially unchanged but the thermal model varies. Despite the advantages of this simulation framework, the neural network loss model cannot be directly used as a boundary condition in thermal simulations, as neural networks are unable to directly exchange data with COMSOL. To address this issue, an indirect and bidirectional coupling strategy is proposed to achieve data exchange between the neural network-aided power loss model and COMSOL thermal model. The details are provided below combined with a practical study case.

A. Introduction of the Studied Case

A buck inverter with a customized multichip SiC phase-leg power module working in continuous operation is chosen as the studied case. The circuit schematic and cooling condition of the inverter are shown in Fig. 2. SiC MOSFET bare dies M_1-M_3 (CPM312000075A) serve as active switches and SiC Schottky barrier diodes D_4-D_6 (CPW41200S020B) act as freewheeling diodes. According to the schematic diagram, there are six heat sources, M_1-M_3 and D_4-D_6, which need to be embedded into the thermal simulation. In addition, in this case, the bus voltage V_{dc} and switching frequency f are variable, while all other circuit parameters remain fixed.

The cooling condition of the COMSOL model is natural convection cooling with finned heatsink. The geometric parameters of the finned heatsink, such as fin thickness, fin spacing, fin height, number of fins, and fin length, are the variables to be optimized. To accurately characterize the junction temperature of power module chips under different geometric parameter combinations, multiple electrothermal co-simulations are required. For each co-simulation case, the circuit model remains largely unchanged, while the thermal model varies due to differences in cooling structure parameters.

B. Neural Network-Aided Power Loss Model

In the studied case, a fully connected backpropagation neural network model illustrated in Fig. 3(a) is chosen to fit the complex nonlinear mapping relationship between inputs (junction temperature, voltage, and switch frequency) and outputs (power losses) in the buck inverter. The automated circuit simulation iteration process, illustrated in Fig. 3(b), provides ample data for neural network training. At the beginning of each cycle, input circuit parameters are randomly generated within a specified range. Based on the new input parameters, corresponding circuit parameters in the circuit SPICE netlist (Buck_circuit.cir) are modified to produce a new circuit netlist. MATLAB calls the PSpice A/D simulator to run the new circuit netlist SPICE

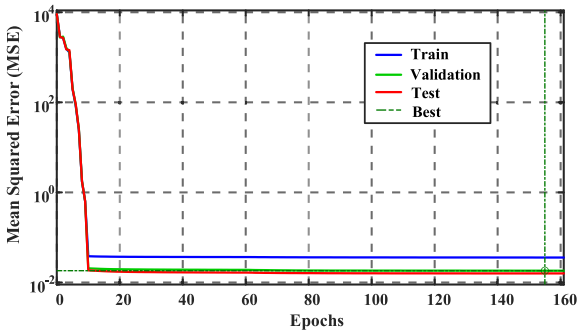


Fig. 4. Typical training performance of the neural network.

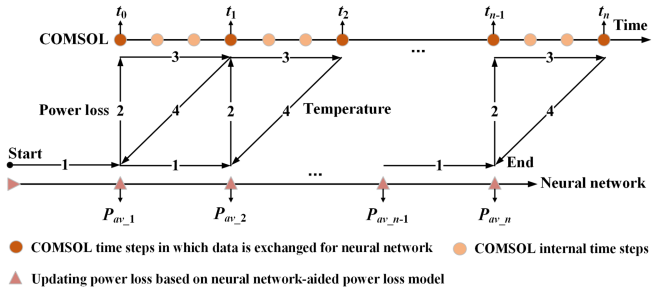


Fig. 5. Proposed indirect and bidirectional coupling strategy.

program for circuit simulation through system operation commands. When simulation ends, the transient power loss data (Buck_circuit.dat) of power devices are packed into the circuit output file. MATLAB reads and processes this data to obtain the average power loss. For other different power module configurations and circuit operational scenarios, automated parameter extraction can be performed using similar methods, as described previously.

Through iterations, 500 sets of labeled data are obtained and utilized for the neural network model training using the Levenberg–Marquardt algorithm. The 500 datasets do not cover all potential operational circumstances and failure scenarios. They only account for variations in input voltage and switching frequency in the buck circuit. When more input parameters related to circuit operating conditions are involved, it is necessary to consider expanding the magnitude and variety of the dataset to retrain. These datasets are randomly divided into a training set (70%), a test set (15%), and a validation set (15%). The typical training performance of the neural network is shown in Fig. 4, where MSE values on the training set, test set, and validation set are all very small after 160 epochs, indicating that the neural network power loss model can accurately capture the intricate and nonlinear mapping between inputs and outputs. This also indirectly proves that the randomly generated input parameters uniformly encompass all critical dimensions of the specific input parameter space within the specified range. It is worth noting that the neural network loss model does not account for the predictive performance under prolonged operational conditions, such as aging. Due to the numerous influencing factors and the complexity of the aging mechanism, it is challenging for all electrothermal co-simulation methods to achieve accurate modeling. Furthermore, although neural networks trained with more input parameters and a larger parameter space exhibit stronger generalization, the time required for data acquisition and model training is longer. To improve simulation efficiency, it is crucial to appropriately select the number and range of input parameters based on the specific research problem.

C. Indirect and Bidirectional Coupling Strategy

The proposed indirect and bidirectional coupling strategy is demonstrated in Fig. 5. This coupling approach involves four steps in each iteration cycle, as described in the following.

Step 1: The neural network power loss model in MATLAB calculates outputs (power losses of power devices) under given inputs (given temperature, specific voltage, specific switching frequency). The given temperature of the first iteration is the ambient temperature while that for the second and subsequent iterations is determined by the temperature delivered by COMSOL.

Step 2: The average power losses computed by the neural network are passed to COMSOL via MATLAB. The boundary condition (input power losses) for thermal simulations in COMSOL is updated.

Step 3: The transient thermal simulation with a given power loss is performed in COMSOL. The COMSOL transient solver will adopt an adaptive COMSOL internal time step (CITS) for efficient computation. It will pause when reaching the data exchange time ($t_n - t_{n-1}$), as defined as COMSOL time steps (CTS) in which data is exchanged for neural networks. The paused thermal simulation will resume from the state at the end of the previous adjacent cycle after the power loss boundary conditions are updated.

Step 4: MATLAB extracts the maximum temperature of heat sources at the last time point from the thermal simulation file. The maximum temperature is subsequently provided as a feedback to the neural network model, serving as the updated temperature condition.

Among four steps mentioned previously, only the third step requires noticeable simulation time, and that for other steps can be ignored. This is because the time required for neural network inference, i.e., calculating the power loss at a new junction temperature, is very short, and the automated data processing for updating power loss is similarly minimal. These steps have little impact on co-simulation efficiency, with the main bottleneck being the slow FEM thermal simulation in Step 3. In addition, to realize the dynamic and continuous thermal simulation in Step 3 during two consecutive iterations, states of each physical field at the end of the previous cycle will serve as the initial states of the next cycle (state inheritance), and the simulation continues under new boundary conditions. The inheritance of thermal simulation states is implemented by assigning the solution at the last time point as the initial value for the solution variable. With this configuration, the computational load for the COMSOL solver in the proposed bidirectional coupling method remains comparable to that required when directly inputting segmented power losses as heat sources in COMSOL.

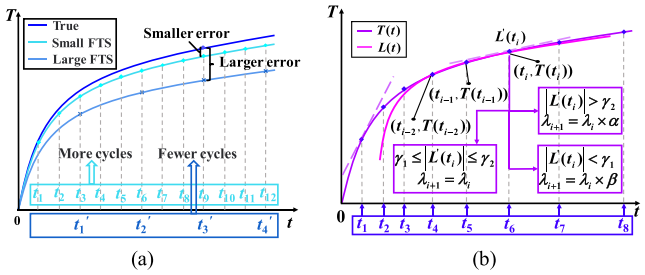


Fig. 6. Schematic diagrams of FTS and variable CTS. (a) FTS. ("True" represents the actual junction temperature). (b) Proposed adaptive CTS adjustment strategy.

By following the iterative process described previously, the loop will continue running until temperature field reaches an equilibrium state, resulting in a convergence solution. The convergence criterion varies depending on the research objective. For transient problems, the simulation ends once the target simulation time is achieved. For steady-state problems, this letter adopts an iterative termination criterion where the relative temperature decrease between two consecutive iterations falls below the set minimum ε. In this study, ε is set to 0.01. The above-mentioned process reveals that the essence of this strategy lies in the real-time update of power loss boundary conditions during the COMSOL thermal simulation, enabling bidirectional coupling between electro and thermal.

D. Adaptive Time Step Adjustment Strategy

CITS is the actual time step adopted by COMSOL solver when solving the transient partial differential equations that describe power module thermal systems. CTS determines the frequency of data exchange between the transient COMSOL thermal simulation and heterogeneous neural network model to update power losses. CTS has a significant impact on simulation accuracy and efficiency. As shown in Fig. 6(a), using a small FTS ("FTS" represents fixed CTS) in long-time simulations, such as second level simulations will induce more cycles and require longer simulation time. If a large FTS is adopted, the number of cycles will be greatly reduced, but with a higher transient simulation error. In short, FTS method is difficult to balance between simulation accuracy and efficiency.

As a solution, this article proposes an adaptive CTS adjustment strategy based on the Lagrange interpolation algorithm. As illustrated in Fig. 6(b), this algorithm constructs a Lagrange interpolation polynomial $L(t)$ to fit the variation of junction temperature, by extracting numerical solutions of the junction temperature for three consecutive iterative points (t_{i-2}, $t_{i-1} = t_{i-2} + \lambda_{i-1}$, $t_i = t_{i-2} + \lambda_{i-1} + \lambda_i$). The first derivative of $L(t)$ at the current time point t_i, $L'(t_i)$, reflects the change rate of junction temperature at t_i, and its expression is given as follows:

$$L'(t_i) = \frac{\lambda_i}{\lambda_{i-1}(\lambda_{i-1}+\lambda_i)} T(t_{i-2}) - \frac{\lambda_{i-1}+\lambda_i}{\lambda_{i-1}\lambda_i} T(t_{i-1}) + \frac{\lambda_{i-1}+2\lambda_i}{\lambda_i(\lambda_{i-1}+\lambda_i)} T(t_i). \quad (1)$$

The definition of the CTS adjustment threshold interval is $[\gamma_1, \gamma_2]$, and the CTS adjustment coefficients are $\alpha \in (0,1)$ and $\beta \in (1,+\infty)$. The values of the parameters α and β will affect the simulation accuracy and efficiency. The recommended range for α is [0.4, 0.7], and the recommended range for β is [1.5, 2.5]. The range of the two parameters is optimized for selection, making it suitable for power electronic power module electrothermal coupling simulation. In the studied case, the selected parameter values are $\alpha = 0.5$ and $\beta = 2$. These two specific parameter values have not been optimized to achieve optimal computational accuracy and speed. This is because parameter optimization would require multiple co-simulation tests, which will negatively affect simulation efficiency. For other co-simulation application scenarios, selecting these two parameter values within the recommended range is sufficient, without the need for special optimization. The value of CTS, apart from being equal to the initial small time step during the first two iterations, is entirely determined by the time step adjustment algorithm in subsequent iterations. Based on this definition and the absolute value of $L'(t_i)$, the corresponding CTS adjustment strategy is established.

1) If $|L'(t_i)| > \gamma_2$, then $\lambda_{i+1} = \alpha\lambda_i$. At this moment, the junction temperature changes rapidly. In order to ensure the accuracy of the simulation, CTS should be decreased.
2) If $|L'(t_i)| < \gamma_1$, then $\lambda_{i+1} = \beta\lambda_i$. In this situation, the junction temperature changes slowly. CTS can be appropriately increased to improve the simulation speed.
3) If $\gamma_1 \le |L'(t_i)| \le \gamma_2$, then $\lambda_{i+1} = \lambda_i$.

The adaptive adjustment strategy adjusts the CTS size dynamically based on the speed of junction temperature changes, which can improve computational efficiency. The range of each CTS λ_{i+1} can be determined as follows.

The remainder term of Lagrange interpolation is used to quantify the error between the Lagrange interpolation polynomial and the actual function $T(t)$, and is commonly employed to assess the interpolation accuracy. By taking the first derivative of the remainder term for the Lagrange interpolation polynomial, the following expression is derived:

$$R'(t) = T'(t) - L'(t) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega'(t) = \frac{T^{(3)}(\xi)}{6}\omega'(t) \quad (2)$$

where $\omega(t) = (t-t_{i-2})(t-t_{i-1})(t-t_i)$, $\xi \in [t_{i-2}, t_i]$.

Substituting adjacent iterative nodes into (2), the following expression can be obtained:

$$R'(t_{i+1}) = \frac{\lambda_{i+1}(\lambda_{i+1}+\lambda_i)}{6} T^{(3)}(\xi). \quad (3)$$

Equation (3) represents the truncation error of the numerical differentiation formula for the temperature coupling variables. The truncation error tolerance of (1) is denoted as ξ_σ, and the following inequality can be derived from (3):

$$\frac{\lambda_{i+1}(\lambda_{i+1}+\lambda_i)}{6} T^{(3)}(\xi) \le \xi_\sigma. \quad (4)$$

If $\lambda_{i+1} = \alpha\lambda_i$, then

$$\lambda_i \le \sqrt{\frac{6\xi_\sigma}{\alpha(1+\alpha)T^{(3)}(\xi)}}. \tag{5}$$

If $\lambda_{i+1} = \lambda_i$, then

$$\lambda_i \le \sqrt{\frac{3\xi_\sigma}{T^{(3)}(\xi)}}. \tag{6}$$

If $\lambda_{i+1} = \beta\lambda_i$, then

$$\lambda_i \le \sqrt{\frac{6\xi_\sigma}{\beta(1+\beta)T^{(3)}(\xi)}}. \tag{7}$$

Since $\alpha \in (0,1)$ and $\beta \in (1,+\infty)$, therefore

$$\sqrt{\frac{6\xi_\sigma}{\beta(1+\beta)T^{(3)}(\xi)}} < \sqrt{\frac{3\xi_\sigma}{T^{(3)}(\xi)}} < \sqrt{\frac{6\xi_\sigma}{\alpha(1+\alpha)T^{(3)}(\xi)}}. \tag{8}$$

The smallest value among these three is taken as the upper limit of each CTS, denoted as $\lambda_{i\max}$. The function $T(t)$ represents the actual relationship that the true junction temperature variation follows, and it is unknown. Considering that the junction temperature variation curve approximately follows an exponential pattern, the exponential function shown below is used as $T(t)$

$$T(t) = a_i\left(1 - e^{-b_i t}\right) + c_i. \tag{9}$$

The values of a_i, b_i, and c_i are determined by the three points $(t_{i-2}, T(t_{i-2}))$, $(t_{i-1}, T(t_{i-1}))$, and $(t_i, T(t_i))$, which are the same points used in the Lagrange interpolation function. When the current node changes, both functions need to be refitted to more accurately capture the real-time variation pattern of the junction temperature.$T^{(3)}(\xi)$ can be approximately solved to obtain its maximum value according to (9), which can then be substituted to calculate $\lambda_{i\max}$.

After CTS is adjusted adaptively, the range of λ_{i+1} needs to be verified before the next simulation cycle. If $\lambda_{i+1} > \lambda_{i\max}$, then $\lambda_{i+1} = \lambda_{i\max}$. The aforementioned method enables control of the truncation error in the temperature change rate, limiting the maximum value that CTS can take in each iteration. This ensures that the error does not exceed the specified tolerance. Even in the event of misclassification at the current node, the CTS will not be excessively large due to the max time step limit, thereby maintaining simulation accuracy.

The above-mentioned is the detailed implementation method of adaptive CTS adjustment. It is worth noting that the adaptive CTS and CITS adjustment methods are fundamentally different. Adaptive CITS uses explicit equations, whereas adaptive CTS lacks such explicit formulations. Their goals also diverge: adaptive CITS aims for rapid and efficient equation solving, while adaptive CTS focuses on efficient data interaction in heterogeneous systems.

The proposed adaptive CTS adjustment method is tailored to address junction temperature variations in power modules. Utilizing three-point Lagrange interpolation ensures accurate and efficient computation, bypassing the complexity, and errors of higher order methods. The method accommodates smooth temperature changes by incorporating adjustment coefficients and thresholds suitable for power module applications. Furthermore, compared to other heterogeneous system co-simulation methods, this approach analyzes the truncation error in calculating the rate of temperature change. An error control method is devised to limit the maximum time step adjustment. This approach is simple, resource-efficient, and improves co-simulation efficiency without sacrificing precision when compared to other methods. Another noteworthy point is that the proposed adaptive CTS adjustment strategy is applicable to various power module configurations and operational scenarios.

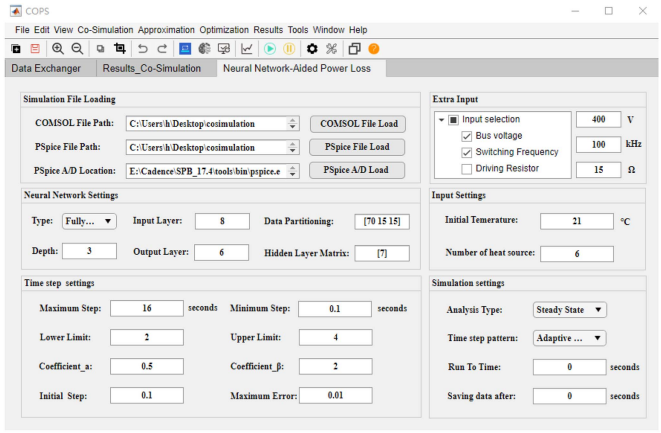
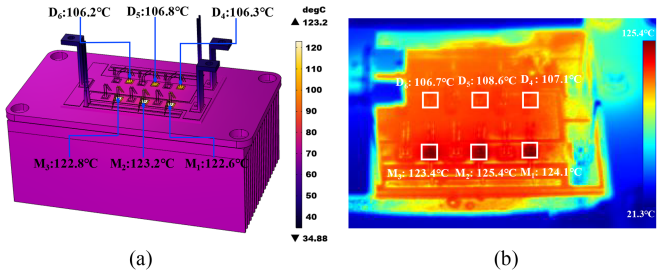


Fig. 7. Self-developed software interface.



(a) (b)

Fig. 8. Steady-state junction temperature in Case 1. (a) Simulation with the proposed method. (b) Experiment.

III. Experimental Verification

A. Validation of the Studied Case

To enhance usability, an interface software that incorporates the proposed method in Section II has been developed utilizing the App Designer software development tool, as depicted in Fig. 7. With the help of this software, electrothermal co-simulation in the studied case can be easily and efficiently achieved. The corresponding steady-state SiC chip temperature distribution, with simulation settings in Figs. 2 and 7, is presented in Fig. 8(a). Meanwhile, Fig. 8(b) displays the maximum steady-state temperature of chips measured by the PS610 thermal camera under same operating conditions. The discrepancy between temperature results obtained from the

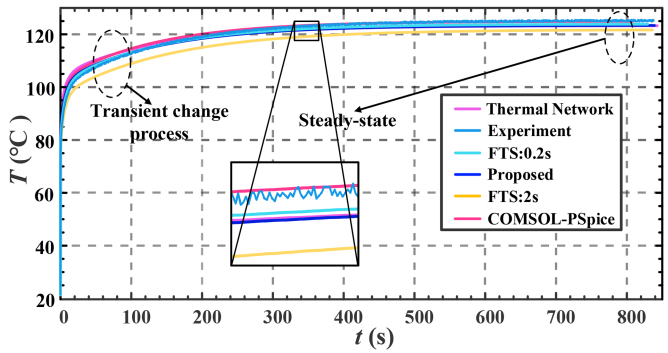


Fig. 9. Transient junction temperature change curve of the chip M_2 in Case 1. (Case 1: 400 V DC input voltage, 100 kHz switching frequency.).

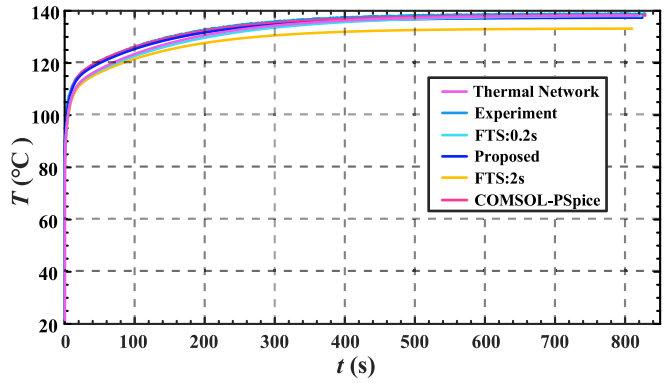


Fig. 10. Transient junction temperature change curve of the chip M_2 in Case 2. (Case 2: 450 V DC input voltage, 100 kHz switching frequency.).

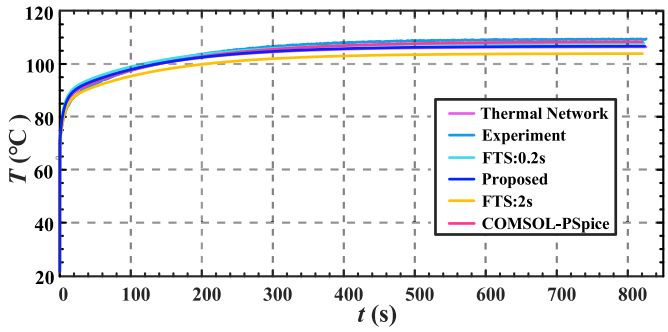


Fig. 11. Transient junction temperature change curve of the chip M_2 in Case 3. (Case 3: 400 V DC input voltage, 80 kHz switching frequency.).

simulation and experiment is less than 2.5 °C, representing an error under 3%.The corresponding transient junction temperature change curves of the chip M_2 (the chip with the highest temperature) in three different cases, obtained from the proposed simulation method, FTS simulation method, 2-D thermal network electrothermal co-simulation achieved in PLECS [3], COMSOL-PSpice coupling simulation method proposed in [5], and experiment, are depicted in Figs. 9–11. The maximum junction temperature error compared to the experiment in the three different cases, and computation time costs for the single co-simulation in Case 1 and 100 co-simulation cases under different design structure parameters combination are list in

TABLE I
SIMULATION SOLUTION TIME AND MAXIMUM ERROR

Method	Time costs (hours)		Maximum error (°C)	
	Single	100 cases	Transient	Steady-state
Thermal network	1.52	108.5	4.2	3.3
FTS:0.2s	7.56	597.4	3.6	1.9
Proposed	1.78	19.6	2.7	3.4
FTS:2s	2.30	71.9	5.5	6.2
COMSOL-PSpice	1.13	113.0	2.5	1.8

Table I. It is worth noting that the computation time includes both the time required for the generation of the reduced-order model and the time required for the co-simulation process. It can be seen that, except for the slightly lower accuracy of the FTS method with large time step, all other co-simulation can guarantee good consistence with experimental results throughout the entire simulation process under three different cases. This indicates that the proposed method maintains high accuracy in junction temperature estimation under different cases. Compared with the FTS method, the proposed approach offers efficiency benefits in both single simulations and multiple case simulations due to the adaptive time step adjustment strategy. Compared with the co-simulation methods based on thermal networks and COMSOL-PSpice, the proposed method shows a 5.5-fold and 5.8-fold improvement in efficiency, respectively, in the simulation of 100 cases. This is because the thermal network method requires multiple transient and steady-state FEM simulations to generate the thermal network for each case, while the COMSOL-PSpice method requires repeated circuit simulations for each case. The proposed method avoids the aforementioned issues, thereby enhancing simulation efficiency. The augmented efficiency stems from the combined effect of the neural network power loss model and the adaptive time step adjustment algorithm.

B. Expanded Validation

To further validate the efficiency advantages and general applicability of the proposed co-simulation method, a new circuit consisting of a direct water-cooled three-phase half-bridge SiC power module is introduced as a new case study, as shown in Fig. 12. Each switch position consists of four 1.2 kV SiC MOSFET chips (Rohm S4108), without any antiparallel diode chips. The gate and source of the SiC MOSFET chips are short-circuited, allowing reverse current flow through the body diodes of the chips. To verify the co-simulation accuracy of the new case, a liquid cooling performance testing platform for the SiC power module is established, as illustrated in Fig. 13.

In the new studied case, a new fully connected neural network is employed to model the nonlinear relationship between 25 inputs (24 chip junction temperatures and input current) and 24 outputs (chips power losses). It features five hidden layers, each containing 128 neurons. In the new case, the number of weight parameters is notably higher than in the original study case. By employing the automated training data extraction method, 1000 datasets are iteratively acquired.

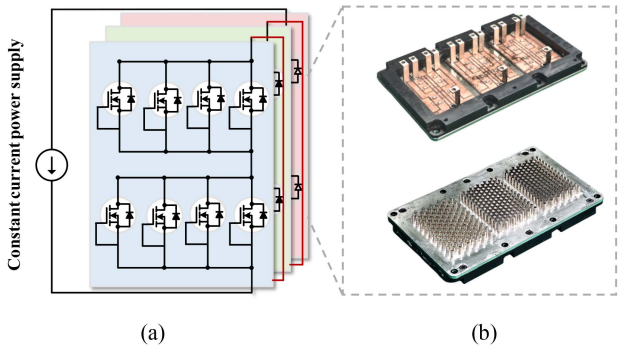


Fig. 12. Circuit schematic and the fabricated power module. (a) Circuit schematic. (b) Fabricated SiC power module.

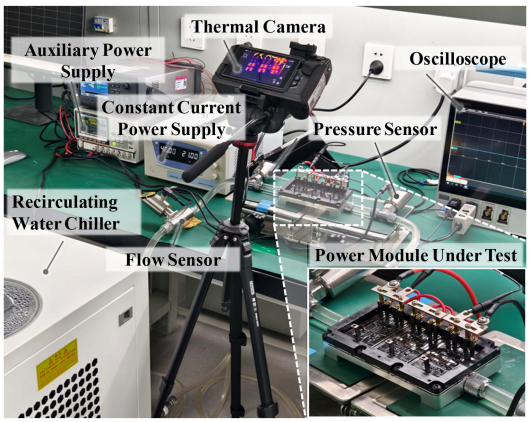


Fig. 13. Power module thermal performance testing platform.

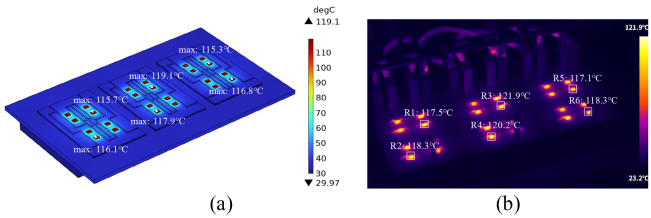


Fig. 14. Steady-state junction temperature of the new study case. (a) Simulation with the proposed method. (b) Experiment.

Similarly, the developed software enables electrothermal co-simulation between the newly trained neural network model and the thermal model. Fig. 14(a) shows the steady-state simulated temperature distribution of the new SiC power module in one of the cases, while Fig. 14(b) presents the corresponding experimental temperature distribution. The boundary conditions corresponding to the case under study are an inlet water temperature of 30 °C, a flow rate of 10 L/min, and a power output of 1500 W from the current source. The error between the simulation and experimental temperature results is less than 3 °C. The corresponding transient junction temperature change curves of the chip with the highest temperature are depicted in Fig. 15. The maximum junction temperature error, and computation time costs for the single co-simulation and 100 co-simulation cases under different boundary condition parameters combination are

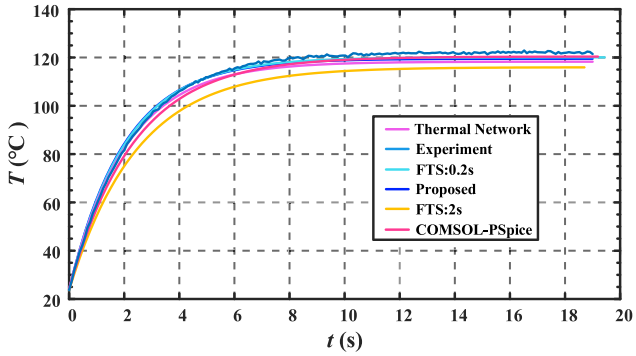


Fig. 15. Transient junction temperature change curve of the chip with the highest temperature.

TABLE II
SIMULATION SOLUTION TIME AND MAXIMUM ERROR IN NEW CASE

Method	Time costs (hours)		Maximum error (°C)	
	Single	100 cases	Transient	Steady-state
Thermal network	21.2	1874	4.1	3.9
FTS:0.2s	30.3	2789	2.2	1.7
Proposed	5.3	255	3.6	2.8
FTS:2s	7.5	473	7.4	5.3
COMSOL-PSpice	4.68	468	3.3	1.2

list in Table II. Apart from the slightly lower accuracy of the FTS method under large time steps, all other co-simulation methods maintain good consistency with experimental results during the simulation process. This also demonstrates that the proposed method can maintain high accuracy in junction temperature estimation under different power module configurations and circuit operating conditions. Compared to the co-simulation methods based on thermal networks, the proposed method shows a 7.3-fold improvement in efficiency, in the simulation of 100 cases. The efficiency improvement is greater than 5.5 times that of the original case. This is because in scenarios where the circuit remains largely unchanged but the thermal model varies, as the number of power module heat source chips increases, the traditional thermal network model approach requires more transient FEM simulations for each case to extract thermal network parameters. The increased number of FEM simulations significantly increases the simulation time. In contrast, the additional simulation time required due to the increased complexity of the neural network power loss model is much smaller. The comparison indicates that the proposed method exhibits a notable efficiency advantage over thermal network methods, particularly in scenarios involving more complex circuit structures and a greater number of chip heat sources.

IV. CONCLUSION

In this letter, an efficient coupling thermal simulation method based on the neural network-aided power loss model is proposed. This method establishes an innovative combination of the neural network and 3-D thermal simulation through a self-developed

interface software. Compared with the existing classical thermal network modeling method and the COMSOL-PSpice co-simulation method, the proposed method achieves comparable simulation accuracy while offering significant efficiency advantages in multicase simulations where the thermal model varies for each case while the circuit model remains largely unchanged. In the study of 100 cases, the simulation speed is approximately five to seven times faster compared to the thermal network method and about two to five times faster compared to the COMSOL-PSpice method. The proposed simulation method is also applicable to electrothermal co-simulation of other power electronic systems.

REFERENCES

[1] Z. Zeng, X. Zhang, F. Blaabjerg, H. Chen, and T. Sun, "Stepwise design methodology and heterogeneous integration routine of air-cooled SiC inverter for electric vehicle," *IEEE Trans. Power Electron.*, vol. 35, no. 4, pp. 3973–3988, Apr. 2020.

[2] A. Raciti, D. Cristaldi, G. Greco, G. Vinci, and G. Bazzano, "Electrothermal PSpice modeling and simulation of power modules," *IEEE Trans. Ind. Electron.*, vol. 62, no. 10, pp. 6260–6271, Oct. 2015.

[3] H. Li et al., "Improved thermal couple impedance model and thermal analysis of multi-chip paralleled IGBT module," in *Proc. IEEE Energy Convers. Congr. Expo.*, 2015, pp. 3748–3753.

[4] Y. Zhang, Z. Wang, H. Wang, and F. Blaabjerg, "Artificial intelligence-aided thermal model considering cross-coupling effects," *IEEE Trans. Power Electron.*, vol. 35, no. 10, pp. 9998–10002, Oct. 2020.

[5] Y. Yang, Z. Wang, Y. Ge, G. Xin, and X. Shi, "An automated field-circuit coupling simulation method based on PSpice-MATLAB-COMSOL for SiC power module design," *IEEE Trans. Power Electron.*, vol. 38, no. 10, pp. 12634–12647, Oct. 2023.