Thermal Calibration of Electronic Components using Artificial Intelligence Predictions

Teodor G. Dinu[1,2], Ioana G. Ciuciu[1], Ioan C. Leordean[2] and Diana Zaharie-Butucel[2]

[1]Babes-Bolyai University, Str. Mihail Kogălniceanu 1, Cluj-Napoca 400084, România
[2]Bosch Engineering Center, Str. Constanța 30-34, Cluj-Napoca 400158, România

Abstract

This work presents a **novel hybrid methodology** combining **Finite Element Method** thermal simulations and **Physics-Informed Neural Networks** for improved **thermal property and resistance prediction** in electronic systems. By directly encoding **physical laws** into the learning procedure, the methodology facilitates **accurate real-time inference** of internal thermal parameters based on **sparse external measurements**.

Keywords

Thermal calibration, Physics-Informed Neural Networks, Finite Element Method, PyAnsys, Ansys

1. Introduction

As electronic systems continue to shrink and increase in functional density, managing heat becomes a **critical challenge**, especially in **embedded and power electronics**. Conventional thermal calibration approaches, including **standardized datasheet values** or **finite element simulations**, often rely heavily on **expert intuition** and **trial-and-error procedures**, making them **time-consuming** and dependent on **human experience** rather than offering **precise, automated solutions**. To address these limitations, this paper introduces a **hybrid calibration approach** using **Physics-Informed Neural Networks**, which embed thermal physics directly into the learning process. By automating simulation workflows through **PyAnsys** and training **lightweight, explainable models** suitable for deployment, the proposed method enables **accurate real-time prediction** of internal thermal properties from sparse external measurements—achieving a balance between **physical fidelity** and **computational efficiency**.

Despite the increasing academic interest in **Physics-Informed Neural Networks** [1] for thermal modeling, their **practical implementation** in real-world calibration scenarios, especially within systems, remains constrained. These limitations primarily stem from **significant computational overhead** and the **scarcity of publicly available datasets** suitable for such applications.

This paper addresses these limitations by introducing a **lightweight, explainable Physics-Informed Neural Network architecture** specifically engineered for integration into a **real-time calibration loop**. The proposed architecture is trained using a **novel, high-throughput Finite Element Method dataset**, generated through **PyAnsys**. In contrast to prior research, such as the work by Du and Lu [2], which focuses on complex, chip-to-system-level applications utilizing architectures like DeepONet or ThermPhysics-Informed Neural Network, the current methodology is specifically **tailored for MOSFET calibration on multilayer Printed Circuit Boards (PCBs)**. It accurately predicts crucial thermal resistance parameters, namely $R_{\mathbf{th(j\text{-}a)}}$ **(junction-to-ambient)** and $R_{\mathbf{th(j\text{-}b)}}$ **(junction-to-board)** which indicate how effectively heat is transferred from the electronic component's junction to its surrounding environment and to the PCB, respectively. A key innovation of the current approach lies in the **direct coupling of Physics-Informed Neural Network training with Finite Element Method simulation workflows**. This integration effectively accounts for **structural and material non-uniformities**, enhancing the model's fidelity. Consequently, the proposed Physics-Informed Neural Network framework is readily **deployable in real-time contexts**, thereby bridging the existing gap between **high-fidelity simulation** and **practical application** in thermal management.

RuleML+RR'25: Companion Proceedings of the 9th International Joint Conference on Rules and Reasoning, September 22–24, 2025, Istanbul, Turkiye
ioana.ciuciu@ubbcluj.ro (I. G. Ciuciu)
0000-0002-7126-0585 (I. G. Ciuciu)
© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

CEUR-WS.org/Vol-4083/paper53.pdf

2. Fundamentals of MOSFETs, PCBs, and Thermal Resistance

A **Metal-Oxide-Semiconductor Field-Effect Transistor (MOSFET)** shown in [Figure 1] is a **three-terminal device—gate**, **drain**, and **source**—widely used in both digital and analog circuits due to its **high efficiency**, **fast switching speed**, and **low power consumption**. It functions by applying an electric field at the gate to control the current between the drain and source, with a thin silicon dioxide (SiO_2) layer ensuring high input impedance. In the automotive industry, MOSFETs serve in various roles including **energy control** for hybrid and electric vehicles, **motor control** in electric power steering (EPS) and HVAC systems, **battery management systems (BMS)**, **switching in DC-DC converters and inverters**, and power regulation for LED lighting and infotainment systems.

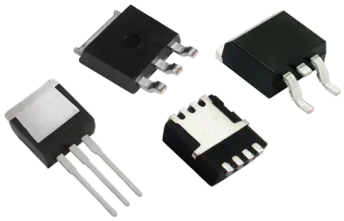

Figure 1: Example of a MOSFET (Metal-Oxide-Semiconductor Field-Effect Transistor)

These devices are typically mounted on **Printed Circuit Boards (PCBs)** as illustrated in [Figure 2], which are composed of insulating materials such as **FR4 fiberglass** and have **copper traces** for electrical connectivity. PCBs provide both **mechanical and electrical support** for components and come in **single-layer, double-layer, or multi-layer** configurations, making them essential in modern electronics due to their **reliability, compactness**, and **ease of mass production**.

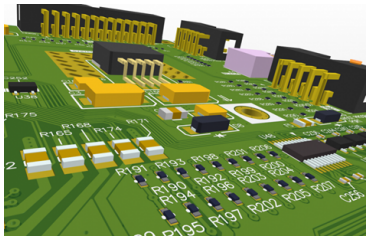

Figure 2: Example of a PCB (Printed Circuit Boards)

Thermal resistance, a **critical parameter** for MOSFET operation, is typically provided in datasheets as $R_{th(j-a)}$, representing the **junction-to-ambient thermal resistance**. This value is influenced by **PCB parameters** such as material, copper thickness, and layout. Derived from Fourier's Law and an analogous to Ohm's Law, thermal resistance is expressed as:

$$R_{th} = \frac{L}{kA} \tag{1}$$

where L is the material thickness, k is the thermal conductivity, and A is the heat conduction area.

The specific thermal resistance from **junction to ambient** is given by:

$$R_{th(j-a)} = \frac{T_j - T_a}{P} \tag{2}$$

while the **junction-to-board resistance** is:

$$R_{th(j-b)} = \frac{T_j - T_b}{P} \tag{3}$$

Here, T_j is the junction temperature, T_a is the ambient temperature, T_b is the board temperature, and P is the power dissipated. By optimizing **copper area, thermal via placement**, and **PCB design**, engineers can significantly reduce $R_{th(j-b)}$, improving **heat dissipation** and **device reliability**.

Despite the mathematical similarity between thermal and electrical resistance, there are **practical differences**. For example, thermal conductivity spans only about three orders of magnitude, while electrical conductivity varies by over twenty. Furthermore, complex phenomena such as **spreading resistance** and **multidimensional heat flow** make analytical solutions less accurate. Therefore, simulation tools like **Finite Element Analysis (FEA)** are often used to model thermal behavior and optimize designs.

Datasheets often present multiple $R_{th(j-a)}$ shown in [Figure 3] values depending on test setups—such as **JEDEC-standard footprint** [3] or a **1-inch square copper pad**—and include package outlines, which are also available in **CAD formats** for use in mechanical and thermal modeling. These resources aid in **precise component placement, thermal analysis**, and **system reliability optimization**.

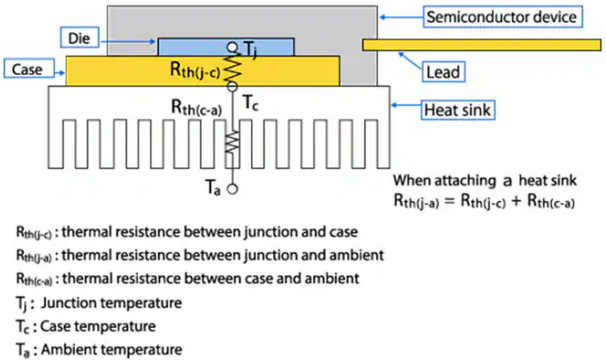

Figure 3: Example of lateral cross section of TO-220 package

Though the mathematical analogy between thermal and electrical resistance is sound, **practical differences** render this analogy of limited utility. As noted by Philips researcher Clemens J. M. Lasance in a 2008 review paper [4], electrical conductivity ranges over roughly 20 orders of magnitude between insulators and conductors, while thermal conductivity only covers a range of roughly three orders. Because of this, the behavior of heat is much more **complicated and less straightforward to predict** than the flow of electrical current. One complication in thermal analysis is the phenomenon of **spreading resistance**—an effect that cannot be accounted for in one-dimensional analytical models. Real heat propagation is affected by **material inhomogeneity, copper traces**, and **PCB layout**, resulting in **multidimensional and nonlinear heat flow** that resists simple calculation, making the problem a good candidate for machine learning solutions.

3. ANSYS and PyAnsys Simulation

ANSYS [5], a **comprehensive engineering simulation software**, employs the **Finite Element Method** to model, simulate, and analyze complex physics-based phenomena across diverse domains, including **structural mechanics, computational fluid dynamics, heat transfer**, and **electromagnetics**. This capability facilitates **virtual prototyping, design optimization**, and **cost-efficient analysis** by enabling engineers to predict system behavior under various loads and environmental conditions without the need for physical prototypes. The typical simulation workflow within ANSYS encompasses several critical stages: initial **geometry creation** or import from CAD software, precise **material property assignment**, **discretization** of the geometry into a finite element mesh, application of appropriate **boundary conditions** and external loads, execution of the **solver** to compute system responses, and finally, comprehensive **post-processing** for result interpretation and visualization. This iterative process often concludes with **validation** against experimental or theoretical data to refine and optimize designs.

Complementing ANSYS's robust graphical interface, **PyAnsys** [6], an **open-source Python library**, significantly extends these capabilities by enabling **scripting, automation**, and **seamless integration of simulations**. PyAnsys provides **direct programmatic access** to **ANSYS Parametric Design Language (APDL)** functionalities within a Python environment, allowing for **granular control** over simulation procedures such as **mesh generation, boundary condition definition**, and **post-processing**. This integration streamlines **multi-step tasks**, facilitates **efficient parametric studies**, enables **large-scale batch simulations**, and supports the implementation of **advanced optimization procedures**. Furthermore, PyAnsys enhances **data processing and reporting** through Python's extensive data libraries and fosters strong integration with **CAD packages, cloud-based computing resources**, and **ML frameworks**, optimizing **simulation-driven product development** and supporting **large-scale, cloud-enabled workflows** for both industrial and academic applications.

4. Physics-Informed Machine Learning

Our proposed approach, **Physics-Informed Machine Learning (PIML)** represents an **essential paradigm shift** in scientific simulation, overcoming the shortcomings of **standard machine learning (ML) algorithms** and **classical numerical methods**. PIML combines the **predictive capabilities of machine learning** with the **strict structure of physical laws**, usually by placing governing equations such as **partial differential equations (PDEs)** directly within the learning procedure. This **hybrid framework** provides **notable benefits** in scientific applications where data can be **limited, noisy, or costly to obtain**, facilitating **data-efficient** and **physically consistent models** with the ability to **generalize well beyond observed data**. The rationale for PIML is driven by the **computational expense, uncertainty quantification challenges**, and **inability to incorporate incomplete or noisy data** of traditional numerical simulations [7, 8, 9].

At the heart of PIML are **Physics-Informed Neural Networks**, neural Networks that are trained to minimize both observed data and the underlying physical laws represented as PDEs. This is done by introducing **PDE residuals** in the **loss function** of the network along with a data-driven term. Physics-Informed Neural Networks have multiple advantages, such as **mesh-free implementation**, **strong handling of noisy and/or incomplete data** [7, 2, 10], and a **common framework** for **forward, inverse**, as well as systems with **missing physics**. Physical information is also imparted through **observational, inductive** (e.g., architectural choices based on symmetries), and **learning biases** (soft constraints) [4, 1]. **Physics-Informed Neural Networks** have been used successfully in various areas of science ranging from **fluid dynamics**, **biophysics**, **plasma physics**, **quantum chemistry**, and **materials science**. Current research in the field is pursuing new avenues such as **scalable algorithms**, **probabilistic approaches** for **uncertainty quantification**, **hybrid modeling approaches**, and the establishment of **rigorous theoretical foundations** and **sophisticated software packages** [8].To realize the **theoretical benefits** of **Physics-Informed Neural Networks** in practice, we created a **fully integrated computational workflow** that integrates **high-throughput Finite Element Method simulations** and **machine learning-based thermal property estimation** for **multilayer semiconductor packages**.

5. Data Generation and AI-Based Prediction Pipeline

This section details **the proposed fully integrated computational pipeline** for estimating the **thermal conductivity of materials** within semiconductor packages using **artificial intelligence**. The pipeline couples **automated high-fidelity simulations** with a **data-driven prediction model**. Specifically, we target a **six-layer semiconductor structure**, leveraging **finite element thermal simulations** to generate temperature responses under controlled thermal loading. These responses are then used to train a **neural network** capable of inferring thermal conductivity across individual layers. The overarching goal is to develop a **surrogate model** that predicts **internal material properties** from **Finite Element Method thermal measurements**, with potential application in **digital twins** and **real-time diagnostics**.

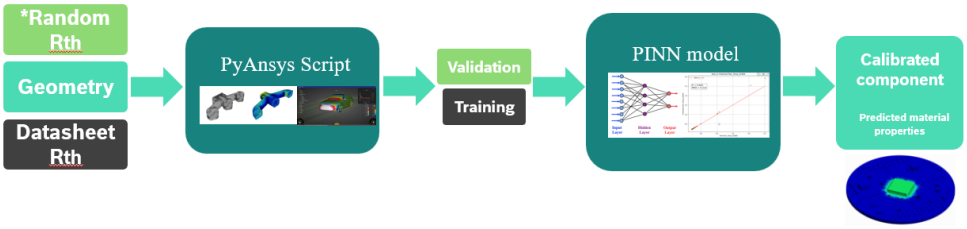

Figure 4: Workflow Diagram

6. Simulation Environment and Dataset Generation

Thermal behavior of semiconductor package was simulated by **ANSYS MAPDL**, invoked through Python under the **PyAnsys API** for **complete automation**. For every simulation run, **thermal conductivity** of the six inner layers had been perturbed randomly with values picked from a **uniformly bounded distribution** in $\pm 10\%$ of typical engineering baselines. This was done to provide a realistic account of **manufacturing tolerance** along with **environmental variations**.

The end-to-end simulation process included:

- Generating the **3D layered geometry**.

- Assigning **randomized conductivities** to each layer.
- Applying **thermal loads** and **boundary conditions**.
- **Meshing**, solving the **heat equation**, and extracting **temperature results**.

This induced **heterogeneity** was critical for training a **generalizable learning model**.

6.1. Boundary Conditions and Thermal Loading

Simulations imposed a **constant 0°C boundary condition** on the top and bottom surfaces of the structure, replicating the **heat sinks**. Internal **volumetric heat generation** replicated device-level power dissipation. The resulting **steady-state heat transfer equation** was solved numerically:

$$\nabla \cdot (\mathbf{k} \nabla \mathbf{T}) + \mathbf{Q} = 0$$

Where k is the space-dependent conductivity and Q the internal heat generation. The **boundary-driven temperature gradients** recorded the thermal response with sensitivity to the conductivity prescribed.

6.2. High-Throughput Execution

To meet dataset size requirements, a **multithreaded Python orchestration layer** coordinated concurrent **MAPDL session launches** with **dynamic port assignments**. This allowed for **near-linear scaling** with CPU core count. Over **50,000 simulations** were run and recorded successfully, with **temperature outcomes** and **conductivity vectors** written to a **structured CSV format**.

6.3. Machine Learning Pipeline

The goal of the **AI model** is to **regress the per-layer conductivity values** using only **high-level temperature metrics**—specifically, the **maximum observed temperatures** near the junction and the board interface. These inputs serve as **proxy measurements**, from which the model must infer the **internal material configuration**.

6.4. Data Curation and Normalization

The raw simulation output was **filtered to ensure data quality**. **Malformed or duplicated rows** were removed, and entries with **negligible heat flow** (Temperatures < 0.2°C) were discarded to avoid learning from physically unrealistic cases.

The final cleaned dataset was **standardized** using a `StandardScaler` to center and scale inputs for **improved optimization performance**. Output targets (conductivity values) were retained in raw form to preserve interpretability.

6.5. Evaluation and Results

Prediction accuracy was assessed using both **mean and maximum absolute percentage error (APE)** per volume:

$$\textbf{APE} = \left| \frac{\hat{y} - y}{y} \right| \times 100\%$$

The model performs exceptionally well for **Volumes 1, 4, 5, and 6**, with **mean errors near zero** and **maximum deviations under 2.5%**, as shown in **Table 1**. In this context, each **volume** corresponds to one of the **six internal layers** of the **semiconductor package**, whose **thermal conductivities** were individually varied during simulation. These layers exhibit **consistent** and **distinct temperature–conductivity relationships**, allowing the network to learn **smooth** and **reliable functional mappings** from **surface temperature data** to **internal material properties**.

Volume 2, however, displays a **marked degradation in accuracy**. The root causes likely include:

- **Higher Sensitivity:** Volume 2 may lie near thermal bottlenecks where small conductivity shifts produce amplified effects on junction temperatures.
- **Nonlinearity:** The relation between conductivity and observed temperature might be highly nonlinear or multivariate for this layer.
- **Data Sparsity:** Volume 2 may have been underrepresented in the random sampling space, especially in edge cases.

Table 1
Mean and Maximum Absolute Percentage Errors by Volume

Volume	Mean Error (%)	Max Error (%)
1	**0.02**	**0.82**
2	**1.25**	**25.18**
4	**0.02**	**2.22**
5	**0.02**	**2.21**
6	**0.17**	**0.86**

Despite this, the model's **average performance remains suitable** for many practical applications. The **consistent accuracy** across the majority of layers confirms the **feasibility of indirect conductivity estimation** via thermal signatures. Improvements in Volume 2 accuracy could be obtained via **data augmentation, targeted oversampling**, or **physics-informed loss regularization**.

7. Discussion

In summary, this paper introduces a **computationally efficient hybrid framework** for coupling **Finite Element Method simulations** with **Physics-Informed Neural Networks** to facilitate precise, real-time inference of internal thermal properties in electronic systems based on sparse external measurements alone. By closing the gap between high-fidelity physical modeling and embedded, practical deployment, this method is particularly well-suited to the demands of **digital twin applications**, **predictive diagnostics**, and **automated thermal calibration workflows**.

Future work can be directed towards **extending the framework to a wider class of electronic packages**, enhancing **robustness of models with noisy or missing data**, and investigating the use of **transfer learning methods** to mitigate the requirement of large simulation datasets. Also, the incorporation of **real-time sensor feedback** and **edge AI deployment** can be considered to advance the system's applicability in industrial, automotive, and consumer electronics applications.

Declaration on Generative AI During the preparation of this work the authors used Generative AI tools in order to improve the clarity and quality of expression. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the content of the paper.

References

[1] W. Chen, Y. Feng, W. Chen, Thermpinn: Physics-informed neural networks for thermal analysis of electronic components, IEEE Transactions on Components, Packaging and Manufacturing Technology 11 (2021) 2101–2110. doi:10.1109/TCPMT.2021.3110269.

[2] Z. Du, R. Lu, Physics-informed neural networks for advanced thermal management in electronics and battery systems: A review of recent developments and future prospects, Batteries 11 (2025) 204. URL: https://www.mdpi.com/2313-0105/11/6/204. doi:10.3390/batteries11060204, open Access.

[3] JEDEC Solid State Technology Association, Jesd51-2a: Integrated circuits thermal test method environmental conditions – natural convection (still air), 2007. Standard document (downloadable from jedec.org).

[4] C. J. M. Lasance, Ten years of boundary-condition-independent compact thermal modeling of electronic parts: A review, Heat Transfer Engineering 29 (2008) 149–168. doi:10.1080/01457630701673188.

[5] ANSYS, Inc., ANSYS Mechanical User's Guide, 2024. Available at: https://ansyshelp.ansys.com/ (Accessed: 1 June 2025).

[6] P. D. Team, PyAnsys Documentation, 2025. Available at: https://docs.pyansys.com/ (Accessed: 1 June 2025).

[7] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, L. Yang, Physics-informed machine learning, Nature Reviews Physics 3 (2021) 422–440. doi:10.1038/s42254-021-00314-5.

[8] M. Raissi, P. Perdikaris, G. E. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, Journal of Computational Physics 378 (2019) 686–707. doi:10.1016/j.jcp.2018.10.045.

[9] X. Wu, D. Zhang, M. Zhang, C. Guo, S. Zhao, Y. Zhang, H. Wang, B. Yang, Autopinn: When automl meets physics-informed neural networks, arXiv preprint (2022). ArXiv:2212.04058.

[10] M. Oddiraju, Z. Hasnain, S. Bandyopadhyay, E. Sunada, S. Chowdhury, Physics-informed machine learning towards a real-time spacecraft thermal simulator, arXiv preprint (2024). ArXiv:2407.06099.