Thermal Models Derived from Raster Images for Printed Circuit Board Based Power Electronics



Ngoni Mugwisi

Trinity College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Hilary 2023

Abstract

Modern power electronics require reliable, durable and efficient thermal management systems to cope with stringent packaging, heating and cost constraints. Thermal modelling is a crucial tool used to inform these design decisions early in the development cycle. This thesis presents a general framework for thermal analysis of printed circuit boards (PCBs) power electronics using finite difference discretizations created from raster images for solving the steady state heat equation with a convective boundary condition.

The framework incorporates a model for vias which are typically challenging to model numerically due to their small lateral size. A pixel-counting method that uses a discretization correction factor to account for the inherent errors in discretizing circular features using square pixels is proposed for capturing the thermal resistance contribution of vias to the PCB. Furthermore, a 'detached' node heat source and heat sink model is introduced and shown to simulate the realistic thermal behaviour of heat sources and heat sinks on a PCB by providing a way to incorporate manufacturer-provided thermal parameters for these devices.

The models are validated against COMSOL for simplified PCB geometries that approximate the features of PCB power electronics. The results show that the model can capture the realistic thermal behaviour of PCBs with vias without increasing the computational cost significantly. Excellent agreement (less than 1% error) is observed between COMSOL and the proposed method with solution times in the order of 1 minute. A comparative case study against industry-standard Thermal Risk Management (TRM) software further confirms the accuracy of the model, with a Mean Percent Error of -3.31% across varying heat loads in a four-layer bridge rectifier PCB design.

Finally, this thesis proposes a method for generating directional thermal conductivities for a tile of pixels based on a downsampling technique that preserves the orthotropic thermal properties of the layers. The method enables pixel-derived bare-PCB thermal models constructed from tiled layouts to achieve accurate thermal results, with grayscale downsampling providing more accurate results at low resolutions than binary downsampling. A significant decrease in computation time is possible, with thermal profiles where the heat source temperature error within 5% is computed in less than 2 minutes using this approach, compared to over

60 minutes if the board is modelled using the same fidelity using conventional thermal analysis tools.

The proposed model is shown to overcome the limitations of other off-the-shelf numerical analysis tools, such as COMSOL, by modelling complicated full-featured four-layer modern power electronics PCBs with high fidelity on a standard desktop PC. The model can complete thermal analysis for simple boards at least five times faster than COMSOL and handle complicated boards to accuracies within acceptable engineering tolerances using numerical thermal modelling techniques introduced in this work.

Acknowledgements

I am immensely grateful to Prof Dan Rogers for his invaluable guidance, support, and encouragement throughout my DPhil. His vast knowledge, expertise, and technical depth, coupled with his dedication to research, have left a lasting impact on my outlook and inspired me to pursue excellence. I deeply admire his wit and effortless ability to question the essence of everything from first principles.

My thanks go to the staff and faculty of Trinity College for their unwavering support, including access to cutting-edge facilities and resources, support during the COVID pandemic and a vibrant research environment that has nurtured my intellectual growth.

I am grateful to the Power Electronics Group members for their support and companionship throughout my research. Their insightful comments, constructive feedback, and helpful suggestions significantly improved the quality of my work. Special thanks to Kawsar, Jack, Pengling, Derek, Aleks, Gwilym, Ed, and Matthew Deakin.

I extend my heartfelt appreciation to the Rhodes Trust for funding my work and granting me access to a lifelong network of shapers: colleagues who are making a meaningful impact in their respective fields.

My family deserves my sincerest thanks for their constant love, encouragement, and moral support during my DPhil. Their unwavering faith in me has been my source of strength and motivation. I am grateful to my mom, Marc, Cindy, and Faith.

Lastly, I would like to express my gratitude to all my friends who have enriched my DPhil experience and made it memorable. Their friendship and moral support have played an integral role in shaping my personal and professional growth. I thank Said, Philippe, A.J., Sam, Kavita, Ellena, Debi, Gloria, Innocent, Prof McCarville, and Prince.

Contents

List of Figures **x**

List of Tables **xiv**

List of Abbreviations **xv**

1 Introduction **1**
1.1 Overview of PCB technology . . . 3
1.1.1 The PCB structure . . . 3
1.1.2 A brief history of PCBs . . . 5
1.1.3 Drivers of thermal modelling requirements in PCBs . . . 7
1.2 Features of the PCB . . . 8
1.2.1 The PCB stackup . . . 9
1.2.2 Vias . . . 10
1.2.3 Thermal pads and component footprints . . . 12
1.2.4 Heat sinks and thermal interface materials . . . 12
1.2.5 Other auxiliary features of the PCB . . . 14
1.3 The PCB design process . . . 15
1.4 The PCB manufacturing process . . . 20
1.5 Considerations for PCB design - a balancing act . . . 21
1.6 Thermal modelling and thermal management of PCB systems . . . 22
1.6.1 Origin of heat in PCB-based electronic systems . . . 22
1.6.2 Passive components . . . 23
1.6.3 Active components . . . 24
1.6.4 Consequences of excess heat in electronic systems . . . 25
1.7 Motivation: Challenges and opportunities in PCB thermal modelling 27
1.8 Outline and Contributions of this work . . . 28

2 Background: 3D thermal modelling of electronic systems **30**
2.1 Modes of heat transfer . . . 30
2.1.1 Conduction . . . 31
2.1.2 Convection . . . 31
2.1.3 Radiation . . . 32

2.2 Heat transfer for a PCB . . . 33
2.3 The heat equation . . . 35
2.4 Methods for solving the heat equation for electronic thermal models 36
2.4.1 Analytical solutions . . . 37
2.4.2 Numerical solutions . . . 38
2.5 Comparison between FDM and FEM for PCB thermal modelling . 47
2.6 Thermal models for electronic components and PCBs . . . 48
2.6.1 Foster and Cauer thermal resistor and capacitor-based networks 48
2.6.2 Lumped models . . . 52
2.6.3 Image-based thermal models . . . 53
2.7 Conclusion and outlook . . . 55

3 Development of a pixel-derived bare-PCB thermal model 57
3.1 Introduction to the pixel-derived bare-PCB thermal model . . . 58
3.2 From Pixelated Image to Finite Difference Grid . . . 63
3.2.1 Thermal resistances of a cuboid structure . . . 63
3.2.2 2D FDM discretization for a grid of pixels . . . 65
3.2.3 Generalization to 3D system . . . 69
3.2.4 Boundary conditions for the 3D system . . . 72
3.3 Justification for using a linear model for PCB thermal analysis . . . 74
3.4 Choosing n_z . . . 77
3.5 Equivalence between resistor network and finite difference grid formulation . . . 78
3.6 Modelling supporting features . . . 81
3.6.1 Heat sources . . . 81
3.6.2 Heat Sinks . . . 83
3.7 Justification and limitations of the steady-state assumption in bare-PCB thermal models . . . 84
3.7.1 Limitations of the steady-state assumption in PCB thermal modeling . . . 86
3.8 Solving the system . . . 87
3.9 Implementation . . . 89
3.9.1 Image acquisition . . . 89
3.9.2 Image processing . . . 90
3.10 Model validation . . . 93
3.10.1 Choice of Test Case . . . 93
3.10.2 Results . . . 94
3.10.3 Impact of FR4 sublayer discretization scheme on the accuracy of the model . . . 96
3.11 Summary . . . 99

4 Pixel-derived thermal models for full-featured PCBs 100
4.1 Holes, cutouts and arbitrary PCB shapes . . . 101
4.2 Vias . . . 102
4.2.1 Thermal modelling of vias in the literature . . . 103
4.2.2 Benchmark analytical thermal resistance model of vias . . . 105
4.2.3 Pixel counting approach to approximate dimensions of circular features on a raster image . . . 107
4.2.4 An analysis of circle rasterization using the Bresenham algorithm 109
4.2.5 Applying the pixel-counting method to compute the thermal conductance of a layer in the vertical direction . . . 113
4.2.6 Validation and verification of the pixel-derived via model . . . 114
4.3 Heat Sources . . . 121
4.3.1 Motivation for an improved heat source model . . . 121
4.3.2 Overview of a detached node heat source thermal model . . . 123
4.3.3 Calculating the conductances from heat source to 'detached nodes' . . . 123
4.3.4 Adding the nodes to G . . . 125
4.4 Heat Sinks . . . 126
4.5 Assembling G . . . 127
4.6 Validation of the improved thermal models . . . 129
4.6.1 Description of Case Study PCB . . . 129
4.6.2 Obtaining the ground truth benchmark temperature . . . 131
4.6.3 Numerical experiments results . . . 132
4.7 Conclusions . . . 135

5 Improving the performance of pixel-derived thermal models through downsampling by tiling 136
5.1 Introduction . . . 136
5.2 Tiling PCBs in the literature . . . 138
5.3 Analytical computation of in-plane directional thermal conductivity 142
5.4 Computing through-plane directional thermal conductivity . . . 145
5.5 Analysis of theoretical computational performance . . . 145
5.6 Choice of n_z . . . 148
5.7 Modelling 'fuzzy' heat sources from grayscale downsampled images . 149
5.8 Model verification . . . 149
5.8.1 Choice of test case . . . 150
5.9 Verification of the full model . . . 156
5.10 Choosing thermal modelling parameters . . . 159
5.11 Case Study: Modelling a bridge rectifier to compare FDM against Thermal Risk Management (TRM) software . . . 159

5.11.1 Results and discussion 161
5.12 Conclusions 164

6 Conclusions and further research **166**
6.1 Contributions of this thesis 167
6.1.1 Rapid discretization generation technique for the FDM method 167
6.1.2 Framework for modelling complicated vias on the PCB from raster image data 168
6.1.3 Method for modelling lumped heat sources 168
6.1.4 Method for modelling heat sinks using a single node 169
6.1.5 Framework for computing anisotropic thermal conductivity for the PCB for improved thermal performance 169
6.2 Limitations of the research 170
6.3 Future work 171
6.3.1 Multigrid discretization for improved resolution in PCB modeling 171
6.3.2 Study the use of the modelling methodology for other planar electronic systems that resemble PCBs, for example, ICs . . 172
6.3.3 Investigate the potential of using other numerical methods to solve the problem, for example, FEM 172
6.3.4 Use of hardware acceleration to solve the problem faster . . 173
6.4 Software availability 173

List of Figures

1.1 Top and bottom side renderings of a PCB with all the thermally relevant features. Source: Cadence Orcard PCB Design Example . . 4
1.2 Evolution of PCBs and thermal modelling 7
1.3 Stacked planar PCB structure where copper and FR4 layers are arranged in an alternating fashion 9
1.4 Types of vias on the PCB. 11
1.5 Example of a typical heat sink 13
1.6 PCB design cyle. 18
1.7 PCB design cycle. 19
1.8 Operation of an electronic system 23

2.1 Heat transfer in a non-uniform solid 33
2.2 Heat transfer on a PCB showing conduction in red and convection in blue 34
2.3 Types of 3D elements 40
2.4 Example PCB mesh with tetrahedral elements 42
2.5 Grid of points for a generic finite difference discretization. The nodes contributing to the 5-point stencil for the Laplacian about the point (i, j) are shown in red 43
2.6 Cauer thermal circuit. 50
2.7 Foster thermal network. 50

3.1 Top and bottom renderings of a bare-PCB model. Only the traces, pads, copper pours and vias are visible. Images: Courtesy of Prof D.J. Rogers and the RELCON Project 60
3.2 Four PCB layers of a multiport converter PCB shown with the dielectric layer hidden. Notice the prevalence of large blocks of copper pours and the low trace count. The layers are: red - top, green - middle, blue - bottom. Images: Courtesy of Prof D.J. Rogers and the RELCON Project 61
3.3 Rasterized representation of a portion of a PCB layer using pixels . 61
3.4 Canonical pixel array 62
3.5 Resistor network derived from pixel array 62

3.6 A cuboid structure and its associated thermal resistances. $R_{i,j,k+1}$ and $R_{i,j,k-1}$ are the half-pixel thermal resistances for the top and bottom halves of the structure respectively. Similarly, $R_{i+1,j,k}$ and $R_{i-1,j,k}$ and $R_{i,j-1,k}$ and $R_{i,j-1,k}$ are the half-pixel thermal resistances in the positive or negative i or j halves of the structure respectively. 64
3.7 General 2D resistor network representing a single layer image 66
3.8 Representative node for a randomly selected pixel $p_{i,j}$. The surrounding pixels are superposed above the node in light blue, showing a 3×3 grid of pixels, and illustrating the idea that series half-pixel resistances have been combined. 66
3.9 5x4 2D FDM sparsity pattern 68
3.10 Generalized $n_x \times n_y \times n_z$ 3D resistor network showing the discretization of a bare PCB thermal model 70
3.11 Spy plot for 3D discretization 4×5 images and 3 layers 73
3.12 Temperature dependency of the thermal conductivity of copper . . 75
3.13 Different vertical discretization schemes in the PCB vertical plane. 5 divisions are shown for the FR4 layer placed between two copper layers 79
3.14 Node (i, j) showing that the neighboring temperatures are referred to the edges of the pixel instead of the centers of the neighboring pixels 80
3.15 Resistor network for an SMD heat source 82
3.16 Distributed heat source model showing all the pixels receiving an equal portion of the heat generated by the source 83
3.17 Summary of MATLAB implementation of the finite difference code. 92
3.18 Simple PCB test geometry 93
3.19 Percentage error of mean and peak temperatures of the source between COMSOL and FDM solver 95
3.20 Average source and peak board temperature comparison between COMSOL and FDM solver 95
3.21 Impact of FR4 sublayer discretization choice on the accuracy of the solution 97
3.22 Impact of the number of n_z layers on the accuracy of the model. A decaying symmetric discretization approach was used 98
3.23 Impact of the number of n_z layers on the computation time. A decaying symmetric discretization approach was used 98

4.1 (a) 3D rendering of a plated through hole via (b) Top view of a single via 104
4.2 (a) 3D rendering of a plated through hole via showing all the internal layers it is connected to. We consider this section of the PCB to be a 'via cell' (b) Top view of the single via cell 106

4.3 Resistor network for a representative via cell 107
4.4 Discrete models of a via perimeter and via filler 110
4.5 Error in the area and perimeter of a circle drawn using Bresenham's algorithm and the vector representation of the same circle 110
4.6 Correction factors needed for discretizing circles with pixels 112
4.7 Via filler distributed among 4 tiles. The tile boundaries are drawn in red for clarity. A 1 pixel perimeter drawn around the filler material can be used to compute the arc length of the via in each tile. In this example sf = 36 113
4.8 Via array used to validate the thermal model for vias. It shows a section of a two-layer PCB with vias filled using solder (black). The dielectric is shown in green 115
4.9 Input images to FDM for via verification 116
4.10 Input images to the FDM discretizer for verifying the via model PCB 116
4.11 FDM top and bottom layer thermal profiles with vias modelled under the source 117
4.12 COMSOL generated top and bottom layer thermal profiles with vias modelled under the source 118
4.13 Error in the average heat source temperature for two nearly identical FDM models (one with and another without vias) compared against temperature calculated using COMSOL 118
4.14 Plot of the difference between the top layer temperature profile computed using Comsol and FDM with via model 120
4.15 Plot of the difference between the bottom layer temperature profile computed using Comsol and FDM with via model 121
4.16 Lumped heat source model showing all the pixels occupied by the heat source having a parallel connection to an isothermal 'super-node' 122
4.17 Lumped heat sink model showing all the pixels occupied by the heat source having a parallel connection to an isothermal 'detached-node' 127
4.18 Input images of the layers for modelling the PCB. Courtesy of Prof. D Rogers 130
4.19 Input images of the heat sources present on the PCB 132
4.20 Top, middle and bottom layers thermal profiles for a four-layer power electronics board 133
4.21 Errors associated with sources 134

5.1 Pixel grid examples showing different patterns that lead to the same thermal resistance using Blackmore's calculations 141

5.2 Equivalent half-pixel resistor networks for a 3×3 pixel grid used to compute the directional thermal conductivity. Notice the additional node due to the current injection (a) Power flow path used in the computation of the equivalent $R_{\theta,x}$ resistance (b) Power flow path used in the computation of the equivalent $R_{\theta,y}$ resistance 142
5.3 Spy plot showing the sparsity pattern of the conductance matrix used in computing orthogonal thermal conductivities. Note that it maintains the banded nature of the general layer matrix starting from node $(1, 1)$ but with additional entries in the first row and column corresponding to all nodes that share a path with the added power injection node. This has the impact of increasing the bandwidth of the matrix 144
5.4 Scale factor vs number of operations for $n_x = 4096$ and $n_y = 4096$. 148
5.5 Input images used to validate the performance of grayscale downsampling. The slanted features and holes are included to ensure that the model can handle slanted features, tiles from which directional thermal paths are important 150
5.6 COMSOL geometry of the test case 151
5.7 (a) Thermal profile from tiling (i) Top (ii) Bottom 152
5.8 (a) Reference thermal profile computed in 167 seconds using COMSOL 152
5.9 Top: Average heat source error computed against scale factor. Bottom: Comparison of solution time vs scaling factor for FDM solutions using tiled and binary layouts 153
5.10 Plot of the difference between the top layer temperature profile computed using comsol and FDM with tiling 154
5.11 Plot of the difference between the bottom layer temperature profile computed using comsol and FDM with tiling 155
5.12 Comparison of temperature profiles obtained from a high-resolution binary downsampling (left) and tiled downsampling (right) 158
5.13 FDM input images for the comparative study, showing key layers and features. 160
5.14 Temperature profile for full rectifier computed using the FDM method 161
5.15 Temperature profile for full rectifier computed using TRM 162
5.16 Percentage error between TRM and FDM heat source temperatures. 163

List of Tables

1.1 Typical materials used in PCB design and their properties [6] [87] [89] 9
1.2 Features of a PCB [6], [87], [90] 16

2.1 Summary of selected works on PCB thermal modelling 56

3.1 COMSOL Model Parameters 94

4.1 Analytical model parameters 117
4.2 Parameters for testing the full-featured board 130
4.3 Results from modelling the thermal profiles at various scale factors 131
4.4 Average temperature rise of the heat source 133
4.5 Results of the full-featured PCB simulation at different scale factors 134

5.1 Parameters used for modelling the PCB using tiling 151
5.2 Comparison of FDM and COMSOL results for temperature profile layers 155
5.3 Results of the full-featured PCB simulation at different scale factors 157
5.4 Comparison of error in the average source temperature for binary and tiling downsampling at selected scale factors 157
5.5 PCB and environmental parameters for TRM validation study . . . 161

List of Abbreviations

1D, 2D, 3D	One-, Two- or Three-dimensional. Refers to spatial dimensions in this thesis.
BOM	Bill of Materials
CPU	Central Processing Unit
CFD	Computational Fluid Dynamics
dpi	dots per inch. A measure of the resolution of a rasterized image
EDA	Electronic Design Automation
EMC	Electromagnetic Compatability
EMI	Electromagnetic Interferance
FDM	Finite Difference Method
FEM	Finite Element Method
GPU	Graphics Processing Unit
HDI	High Density Interconnect
IC	Integrated Circuit
MHz	Megahertz
MOSFET	Metal Oxide Field Effect Transistor
PC	Personal Computer
PCB	Printed Circuit Board
PDE	Partial Differential Equation
PTH	Plated Through Hole
SMD	Surface Mount Device
SMT	Surface Mount Technology
SNR	Signal to Noise Ratio
RF	Radio Frequency
TIM	Thermal Interface Material

1 Introduction

Printed Circuit Boards (PCBs) are essential components of electronic devices and must operate reliably under a wide range of thermal conditions including variable ambient temperatures, thermal cycling and extreme heat generation by the components [109]. These thermal conditions can affect the electrical performance, reliability, and lifespan of both the PCB and its components. Thermal modelling is, therefore, an important tool in the design process for understanding and predicting the performance of PCB electronics: it can help identify potential thermal issues, and optimize the PCB layout, material selection, component placement, and cooling strategies [34].

Thermal modelling of PCBs is a process of predicting the temperature distribution and heat transfer across the PCB layers, components, and the surrounding environment. In order to successfully model the board's thermal behaviour, it is necessary to have detailed knowledge of the PCB's geometry, the thermal properties of the materials from which it is constructed, the dissipation of heat sources and the presence of heat sinks on its surfaces, as well as the boundary and initial conditions. Several idealizations and simplifying assumptions are often necessary to

model PCBs due to the practical limitations inherent in modelling these complex structures to high fidelity.

This thesis investigates the use of rasterized images of the layers of a PCB to drive thermal analysis using the finite difference method (FDM). Such images are readily available from modern electronic design automation (EDA) tools that are used to lay out the boards in the design stage. We study the trade-off between the computational cost and the accuracy of the temperature profiles computed from finite difference discretizations rapidly generated from pixel arrays of the conducting layers of the PCB. We propose a framework capable of efficiently generating FDM discretizations from raster images in an automated fashion to model the PCB, incorporating its complex multi-material construction and irregular features whose size and material properties often differ by orders of magnitude. We show how the system of equations representing steady-state heat transfer can be assembled efficiently, solved and post-processed in a reasonable time (*i.e.* in the order of a few seconds to a few minutes) to gain insight into the thermal characteristics of the board.

This chapter serves as an introduction to the PCB, focusing on its material properties, design constraints, and physical construction. We begin by providing an overview of PCB technology, including its history and impact on the electronics industry and other technology sectors. We then review the function and properties of key features of the board, including its electrical and thermal design process, and state-of-the-art tools in use. Throughout, we identify areas where the studies conducted in this thesis can add value.

Next, we identify the thermally relevant features of the board and discuss how components mounted on its surface contribute to heat transfer. We also explore the origins and impact of heat in electronics, and ways of managing its dissipation in the operation of PCB-based electronics.

While the thermal modelling ideas presented in this thesis can be applied more broadly to general PCB-based electronics, we focus specifically on power electronics

PCBs. This class of PCBs is a crucial enabling technology in energy conversion, and thermal management is often a primary concern in these designs. Power electronics PCBs are a good candidate for academic study because of the relatively simple layouts involved, which feature large blocks of copper pours on most layers and low counts of thin copper traces. Additionally, the thermal properties of the dominant heat-generating components in power electronics, such as MOSFETs, are well-understood and widely available, making it less challenging to validate numerical experiments on their thermal behaviour compared to other components, such as CPUs with BGA packaging.

1.1 Overview of PCB technology

1.1.1 The PCB structure

PCBs are an essential building block of electronic systems prevalent in industry, with applications in various sectors, such as telecommunications, computing, automotive, aerospace, and power systems. A PCB is a composite structure that provides mechanical support and a medium for electrical interconnections of electronic components. Its primary function is to contain and support an electronic circuit while operating within a set of design specifications that includes thermal limits, signal integrity, and power requirements. The PCB is typically made of a non-conductive substrate material, such as FR4 fibreglass, while conductive pathways, usually made of copper, are etched or printed onto the surface of the dielectric using photolithographic techniques.

Modern PCBs are complex, multilayer devices that are often densely populated by a variety of electronic components including switching semiconductor devices (for instance MOSFETs and diodes), passives (such as resistors and capacitors), magnetics (for example inductors and transformers) and integrated circuit packages.

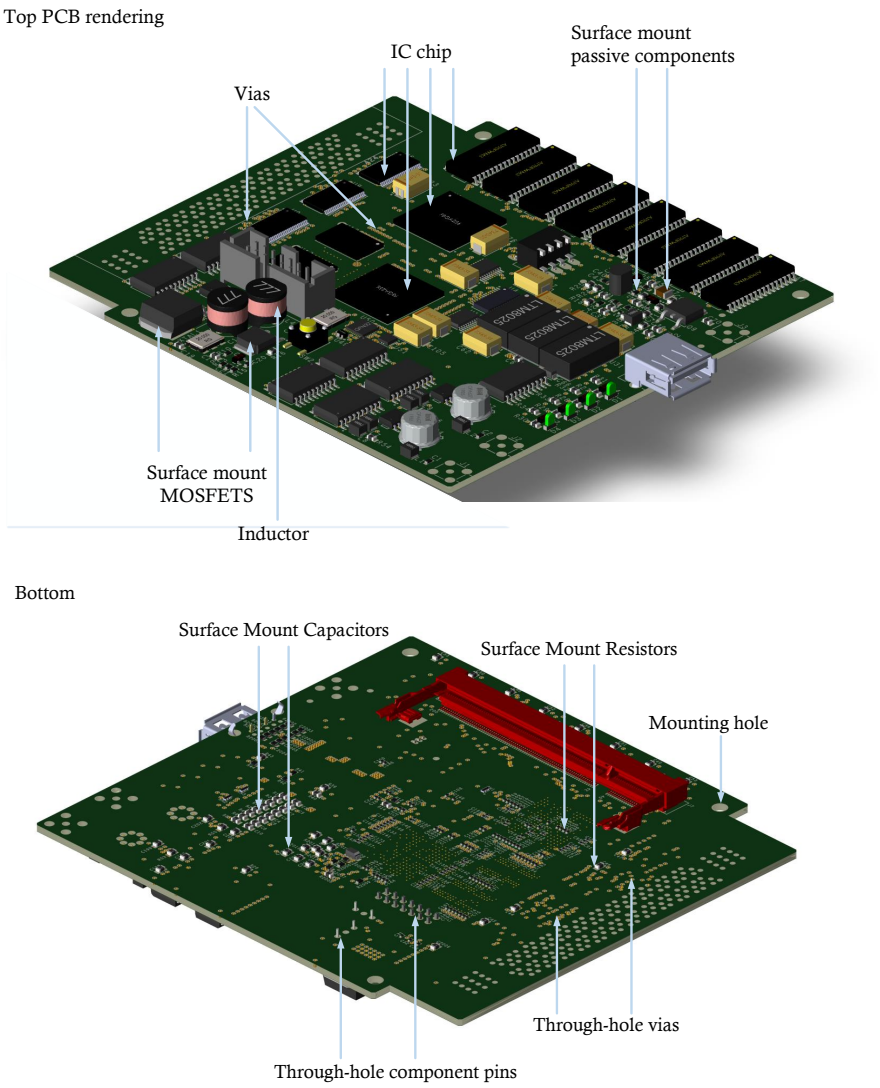


Figure 1.1: Top and bottom side renderings of a PCB with all the thermally relevant features. Source: Cadence Orcard PCB Design Example

Fig 1.1 illustrates an example of a double-sided PCB compactly packed with various components on either side.

Although single-sided PCBs with components mounted on either the top or bottom side are possible, modern circuits are increasingly complex and often require double-sided boards. These boards have gained popularity as they offer more space for routing traces and components. The specific dimensions of a PCB depend on the application it is designed for and the packaging requirements it must meet. They can range from small handheld devices in consumer electronics to large-scale

devices in high-power industrial machines. The thickness of a PCB is typically 1.6 mm, which is two orders of magnitude greater than the typical thickness of a copper layer (35.0 μm). The width of traces can range from as small as 0.1 mm, depending on the required current carrying capacity. Component size and spacing are also important design considerations. Surface-mount components can have surface areas less than 3 mm^2, while capacitors and inductors often have large physical profiles in power electronic applications, inversely related to the operating frequency of the converter. This wide range of possible dimensions presents a significant challenge in constructing high-fidelity thermal models of PCBs.

1.1.2 A brief history of PCBs

Fig 1.2 summarizes the major developments in PCB design and thermal modelling in the last century. The history of PCBs has followed a trend of miniaturization driven by demands for devices that are both powerful and small, *i.e.* highly power dense. Conceived in the 1900s, PCBs did not enjoy the widespread adoption that we see today until the 1950s when the development of the 'print and etch' process both simplified and sped up the PCB manufacturing process while accommodating increased circuit complexity and reducing errors. This process involved printing the circuit design onto a copper-clad board and then etching away the excess copper to leave behind the conductive pathways [6].

The development of integrated circuits (ICs) in the 1960s made it possible to build more complex circuits on a smaller scale, which led to a rapid increase in the use of PCBs in electronic devices [89]. This decade coincided with early efforts at thermal modelling that were based on simple analytical models that used basic equations to calculate temperature distributions based on assumptions about the PCB's geometry, materials, and boundary conditions. These models were limited in their accuracy and usefulness due to their simplicity and inability to account for the complex geometries and heat transfer mechanisms that were becoming apparent in PCBs.

The introduction of automated PCB assembly in the 1970s further increased the speed and efficiency of PCB manufacturing. Powerful computers were also being developed using the ICs of the 1960s, which meant that numerical techniques that generally required significant computing resources, primarily the finite element method, gained popularity in modelling the thermal behaviour of PCBs. Although numerical approaches could account for complex geometries, materials and boundary conditions, they still relied on extreme simplifying assumptions to reduce the size of the model, leading to reduced order models that the computing power of the time could cope with [40].

In the 1980s and 1990s, the use of surface mount technology (SMT) became more common, allowing for even smaller and more densely packed electronic components to be mounted onto PCBs. This technology has continued to evolve, with the introduction of high-density interconnects (HDI) PCBs that can contain multiple layers of circuitry. In the 1990s, the development of Computational Fluid Dynamics (CFD) software made it possible to simulate the flow of air or other cooling fluids over the surface of a PCB. CFD simulations were especially important because they could provide insight into how the design of the PCB and its surrounding components affected the flow of cooling air and the resulting temperature distributions.

The use of PCBs in electronic designs has brought about several benefits, including reducing the weight and size of electronic gadgets, enhancing their reliability, and enabling mass manufacturing, thereby facilitating their widespread use. In recent times, further advances in computing power have renewed interest in thermal modelling approaches that were once out of reach due to the limitations of computing power in the last three decades. Multiphysics approaches that integrate finite element analysis (FEA) and CFD models are now possible, enabling the simulation of both heat transfer within the PCB and the flow of cooling fluids around it. Moreover, there has been an increase in the level of fidelity achievable in PCB thermal modelling. This thesis demonstrates that the extent

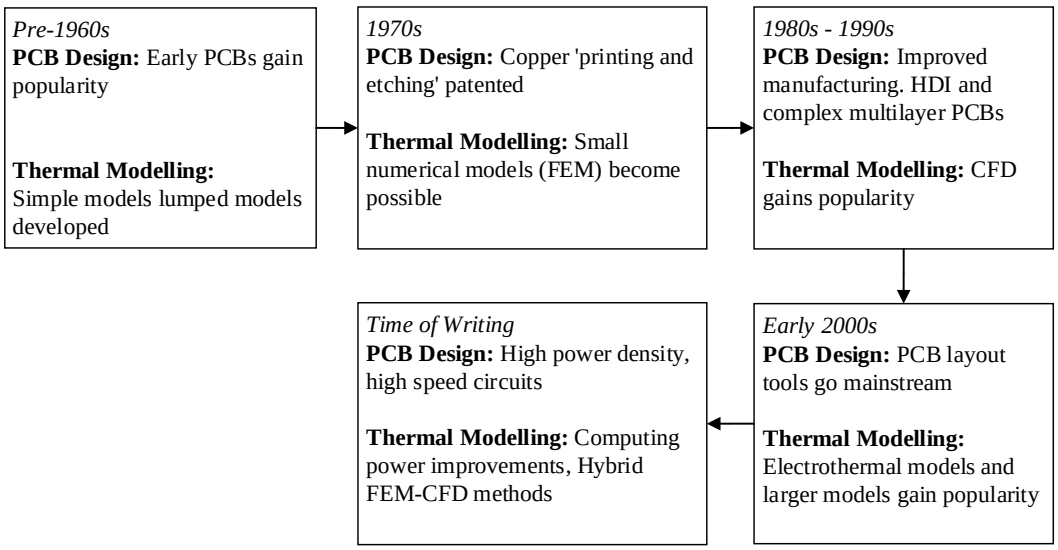


Figure 1.2: Evolution of PCBs and thermal modelling

of simplifying assumptions required to model a PCB has reduced and is expected to continue decreasing.

1.1.3 Drivers of thermal modelling requirements in PCBs

As the trend towards high-speed circuits (enabled by, for example, wide bandgap devices in power electronics), and compactly packaged ICs (a result of progress predicted by Moore's law) accelerates, the demand for both electrically and thermally robust PCBs is set to increase [10]. Thermal dissipation is of concern in PCBs and be from a few watts in small-scale converters to a few hundred watts in high-performance power supplies and motherboards, *e.g.* in a GPU or server, and known to exceed a few kilowatts in a modern inverter found in an electrical vehicle.

Three prevailing trends in power electronics have motivated the need for computationally efficient thermal modelling techniques, namely (a) the increasing power handling capabilities of modern circuits (b) the increasing frequency of operation of high-speed switching circuits (which, for instance, can be in the MHz range for converters based on wide bandgap transistors), and (c) the miniaturization of

many PCBs to increase power density while adhering to tight packaging constraints. Combined, these factors lead to a requirement for large amounts of heat energy to be dissipated from small footprints by the components of a PCB, with the board being a key thermal path for lateral and vertical heat transfer, making the thermal management problem harder.

The motivations for thermal modelling include gaining insights into a number of design goals including the predicted reliability and useful life of the system, its electrical and thermal performance while ensuring user safety and complying with manufacturing and operational standards.

1.2 Features of the PCB

Although PCBs have many features that work together to enable their functionality, in this work we limit our scope to features that are thermally relevant - *i.e.* those that have a non-negligible impact on the thermal behaviour of the structure.

The simplest and perhaps least common PCB consists of one very thin layer of conducting copper patterned to the appropriate circuit design (usually at least 35 μm thick) atop a slab of much thicker FR4. The uncommon use of single-layer PCBs is due to an inherent limitation: it is not possible to include intersecting copper tracks to connect components on a single layer without shorting the circuit. Obviously, this flexibility is an almost guaranteed requirement in practical circuit systems as they usually contain hundreds or even thousands of components and dense involute tracks.

The multilayer PCB addresses this problem by enabling intersecting tracks to be routed on different layers. Such a PCB is constructed by bonding alternating layers of conducting copper tracks and dielectric material together. The different copper layers are electrically connected using vias, which are small holes drilled to link two or more adjacent layers of copper and made conductive by electroplating. This design approach is attractive because it allows components to populate the

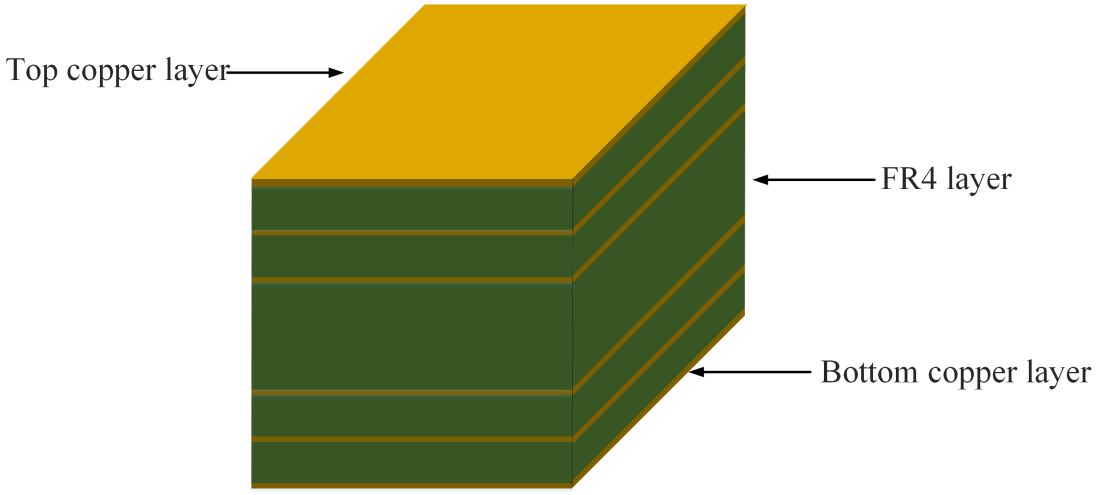


Figure 1.3: Stacked planar PCB structure where copper and FR4 layers are arranged in an alternating fashion

PCB on both the top and bottom layers, and in some cases, to be embedded in the internal layers [10] [68] .

Material	Purpose	σ (S/m)	ρ (Ωm)	κ (W/mK)	c_p (J/kgK)
Copper	Electrical and heat conduction	5.96×10^7	1.68×10^{-8}	400	385
Solder	Bonding of components to the PCB	1.6×10^7	7.5×10^{-8}	$50 - 60$	220
FR4	Dielectric and mechanical support	$10^{-22} - 10^{-6}$	1.6×10^6	$0.25 - 0.35$	1200
Plastic	Packaging and insulation	$10^{-14} - 10^{-10}$	$10^{10} - 10^{14}$	$0.1 - 0.3$	$1000 - 2000$
Prepeg	Bonding between FR4 and copper layers	10^{-6}	10^6	$0.25 - 0.50$	$1100 - 1300$
TIM	Heat conduction and electrical insulation	low	high	$1 - 10$	$1000 - 3000$

Table 1.1: Typical materials used in PCB design and their properties [6] [87] [89]

1.2.1 The PCB stackup

Figure 1.3 shows a small section of a six-layer PCB that illustrates the alternating arrangement of copper and FR4 layers used in multilayer PCBs. These two materials define the electrical and thermal properties of the board. The thermal conductivity of copper is 3 orders of magnitude greater than that of FR4 1.1. As a result,

copper provides the primary thermal conduction path even when thin traces of the metal are embedded in much larger regions of FR4. By a similar argument, copper also provides the primary current conduction path because its electrical conductivity (5.9×10^7 S/m at 20 °C) is at least 10 orders of magnitude greater than that of FR4. A summary of the electrical and thermal properties of materials used in PCBs is given in Table 1.1.

Consequently, copper traces provide a low resistance path for both heat and electric current flow, and should be designed to optimise both functions. In contrast, FR4 creates considerable thermal and electrical resistance. To address this material property mismatch, some researchers have proposed replacing FR4 as the substrate material with alternatives that boast higher thermal conductivities, for example, ceramics and even thick copper cores [49] [110] [118]. These approaches are advantageous because they facilitate better lateral heat spreading within the board, and minimize hotspots by enabling more uniform operating temperatures. However, they are expensive and not mass-produced to the same extent as PCBs that use FR4 substrates.

1.2.2 Vias

Vias are categorized by their size [50] and placement in the vertical plane of the PCB, as shown in Fig 1.4.

When categorized by placement, three basic types emerge. First, through-hole vias are drilled all the way through the board and may connect two or more layers. Secondly, buried vias are drilled such that they connect only the internal layers of the board and are usually required on densely routed PCBs. Lastly, blind vias connect either the top or bottom layers with one or more internal layers without going all the way through the board. Microvias are a specialized form of blind and buried vias that only connect two adjacent layers and are limited to a diameter smaller than 6 mils.

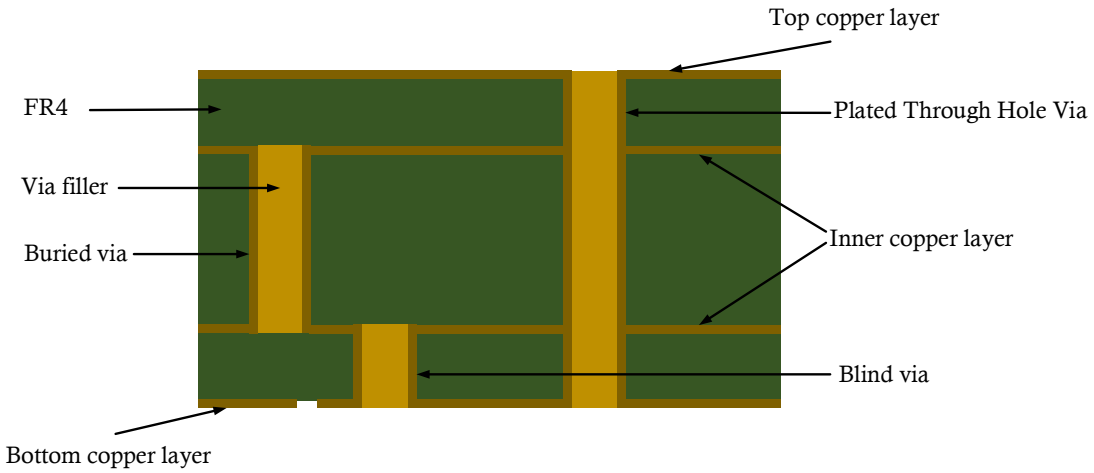


Figure 1.4: Vias for the PCB

In addition to their role in routing electrical signals between different layers of the circuit, vias also play an important role in thermal management. Vias designed for this purpose are called thermal vias, and they provide a low thermal resistance path for heat conduction from the heat-dissipating sources to other layers within the FR4 substrate. By conducting the heat away, thermal vias help to keep the devices within safe operating temperatures.

The number and configuration of thermal vias are important considerations for optimal thermal management [23]. In a design optimization study, the authors of [102] demonstrated that for a known PCB size, there exists an optimal spacing, diameter, and layout of vias that can minimize their thermal resistance. However, their study did not take into account the interaction of vias with other tracks on the PCB or investigate how vias behave as part of the whole system. Such comprehensive studies of via thermal models are necessary to understand the behaviour of vias as part of a full PCB system, and Chapter 4 of this thesis delves into the details of modelling vias in this context.

It is important to realize that there is a trade-off between the space available for trace routing and heat removal. In PCB modeling, there is a crucial trade-off between allocating space for trace routing and providing adequate pathways for heat

removal. Maximizing the area for electrical traces can limit the space available for thermal management features like thermal vias or copper pours, potentially leading to localized hotspots. The authors of [58] studied the relationships between physical design parameters and the thermal resistance of thermal-via clusters in PCBs and system-level packaging. These relationships were determined by simplifying via clusters into parallel networks based on the observation that heat transfer is much more dominant vertically through the thickness of the PCB rather than laterally from heat spreading. They found that as the size of thermal-via islands increases in multi-chip modules, better heat removal was achievable at the cost of less availability of space for routing, leading to large board designs.

1.2.3 Thermal pads and component footprints

Via islands are normally drilled within the boundaries of an exposed pad on the surface of a board onto which SMDs are then soldered. A pad serves a dual purpose by providing the location of the electrical connections of the component leads to the rest of the circuit in addition to improving the thermal contact between the electronic component and the board. This improved thermal contact allows heat to be dissipated more efficiently from the component into the PCB. Several reference thermal design manuals for thermal pads are available from manufacturers of high-power dissipating devices [1], [56]. The size and shape of the thermal pad must also be carefully considered to ensure proper coverage of the component and effective heat dissipation, a topic that has attracted both industrial and academic research interest [1], [18], [43], [88].

1.2.4 Heat sinks and thermal interface materials

Although heat sinks and thermal interface materials are add-on features that are not commonly integrated into a PCB, they are crucial to understanding and

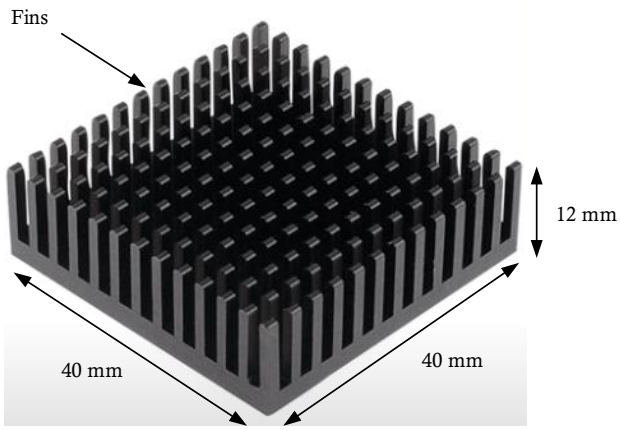


Figure 1.5: Example of a typical heat sink

holistically modelling the thermal behaviour of the board. Hence it is appropriate to discuss their overall properties here.

Heat sinks

A heat sink is a passive cooling device that works by absorbing and dissipating heat away from the component it is attached to. Conventional heat sinks work by increasing the surface area for heat loss by convection, usually achieved by using a finned design as shown in Fig 1.5 [117]. They are typically made of a thermally conductive material, such as aluminium and attached either directly to the dissipating device for top-cooled devices or indirectly through a cluster of vias for bottom-cooled devices [24]. In both cases thermal interface materials are placed between the device and the electrical component to ensure electric isolation.

The design and selection of a heat sink depend on several factors, including the power dissipation of the electronic component, the space available on the PCB, the desired operating temperature range, and the characteristics of the ambient environment where the PCB is designed to operate. Heat sinks come in various shapes and sizes, ranging from simple flat plate designs to complex finned structures,

that are designed to maximize the available surface area in contact with the external environment to provide efficient heat dissipation in different applications.

Thermal Interface Materials (TIM)

TIMs are used to facilitate the transfer of heat from electronic components to heat sinks or other cooling devices. TIMs are typically used to fill gaps between the component and the cooling device, where air pockets can cause poor heat transfer [51]. There are several types of thermal interface materials commonly used in PCB design, including thermal paste, thermal pads, and phase change materials. Thermal paste is a common TIM made of a silicone or ceramic-based compound that is applied to the surface of a component and heat sink to fill in microscopic gaps between the two surfaces, allowing for efficient heat transfer. Pre-cut pads made from a compressible material, such as silicone or graphite, and placed between the component and the heat sink are also a common TIM solution [55]. Phase change materials are a type of TIM that change state when exposed to heat. They are often used in applications where a higher thermal conductivity is required than what is achievable with traditional TIMs. Phase change materials can provide excellent thermal performance but can be more expensive and challenging to use than other TIMs [24].

The thermal conductivity/thermal resistance and compressibility of the TIM are important considerations in the selection of TIM. For our purposes, the TIM and heat sinks will be presented using the quoted thermal resistance values available in manufacturer data sheets.

1.2.5 Other auxiliary features of the PCB

The features discussed so far are considered core to the thermal functionality of a PCB. For completion's sake, we now mention other supporting features that

make the whole structure possible despite considering their influence on the thermal behaviour of the PCB negligible.

Mounting holes are included on PCBs to aid with packaging. They play an important role in ensuring the EMC shielding of the board in high-speed applications and should be plated and connected to ground in those situations. Holes can alter the thermal properties of the board, especially if they lead to a discontinuity in the copper pours of a layer.

The silkscreen is a thin inked layer that is included on the top and bottom surfaces of the board. It cosntains labelling information used to identify components, the polarity of devices, warnings, symbols, branding information, test points, and so on. It is useful for PCB assembly and debugging.

The solder mask, which is also included on the external layers of the PCB, protects the exposed metal traces and pads from damage, in addition to preventing oxidation and unintended short-circuiting that may arise during soldering. An additional important role of the solder mask is to protect the PCB from environmental factors by acting as a barrier between the PCB and the surrounding environment, thus preventing the PCB from being damaged by moisture, dust, and other contaminants. This helps to ensure the reliability and longevity of the PCB. Solder mask is, however, often not a candidate for thermal modelling of the board. Table 1.2 summarises the features of the PCB and their design constraints.

1.3 The PCB design process

The goal of this thesis is to investigate ways of improving the PCB design process by incorporating rapid thermal modelling in the design loop. To that end, a brief overview of a typical PCB design workflow is given, to provide the context of the design step we wish to improve.

Feature	Function	Thermal design considerations
Copper traces	Thin lines of copper that connect the various components of the circuit	Maximise the size of copper pours for heat spreading. Interleaved FR4-copper parts undesirable
FR4 dielectric layers	Mechanical support of traces, signal and power planes, electrical insulation	Minimise inclusion on heat conduction paths
Vias	Holes filled with conductive material that connect different layers of the PCB	Optimal configuration and number needed to improve thermal performance
Pins	Increase thermal conductivity between adjacent board layers	Filler material must be highly conductive
Solder mask	A layer of polymer material applied over the copper traces to prevent accidental solder bridges and shorts.	Negligible impact on thermal performance
Silkscreen	Printed markings on the board to identify components, their values, and their placement	Negligible impact on thermal performance
Copper pads	Copper-plated holes or flat areas on the board where components are soldered or inserted	Large pads for better heat dissipation, but can also interfere with signal routing and placement
Ground and power planes	Large areas of copper that provide low-impedance paths for ground and power connections	Maximise size to improve heat spreading
Heat sinks	Increase available surface area for convective heat dissipation	Proper sizing to maximize surface area, placement close to heat sources using thermally conductive and electrically inert TIM
TIM	Improve thermal contact between the heat sink and the component	Must be as thin as possible and have a high thermal conductivity

Table 1.2: Features of a PCB [6], [87], [90]

The PCB design process is a complex and multi-step process that requires careful attention to detail and the use of specialized tools and techniques. A plethora of commercial and free EDA tools exist to automate and streamline common tasks involved in PCB design such as schematic capture, routing and layout, and thermal design. These EDA tools are the source of CAD files that are used as input to FEA software for thermal modelling and signal integrity analysis. In this thesis, EDA tools are also the source of the high-resolution raster images of the layers of a PCB

that will be used as the input required to construct the PCB thermal model.

Fig 1.6 shows two possible thermally aware design cycles for a power electronics PCB, contrasting virtual and physical prototyping approaches to thermal verification.

Specification and Schematic Capture

The first step involves gathering the design requirements of the converter such as the switching frequency, the power rating, the input and output voltage, required operating temperature and so forth. Next, candidate topologies are considered, and one that meets the requirements is selected, and simulated to assess its electrical functionality, for example using software such as P-Spice and PLECS. The outcome of the simulation informs the selection of the components, especially the power semiconductors, whose electrical and thermal ratings are carefully considered. The next step, schematic capture, involves the creation of a circuit diagram, which serves as a graphical representation of the electrical connections and components. At the time of writing, some of the major EDA design tools include Eagle (a subsidiary of Autodesk at the time of writing), KiCad, an open-source alternative, Altium, and Cadence. More information about a possible software suite for PCB design at the time of writing is listed in Fig 1.7.

PCB Layout

The schematic is then used to develop a physical representation of the circuit in a PCB layout tool where the placement of components and vias as well as the routing of the traces is defined. Most modern EDA tools generate both vector and raster representations of the board, and, in some cases, highly detailed 3D models. The EDA tools also help the designer to ensure that the circuit is represented accurately and that it conforms to predefined electrical and design rules. They also support manual or automatic routing of the circuit paths, and placement of vias and copper

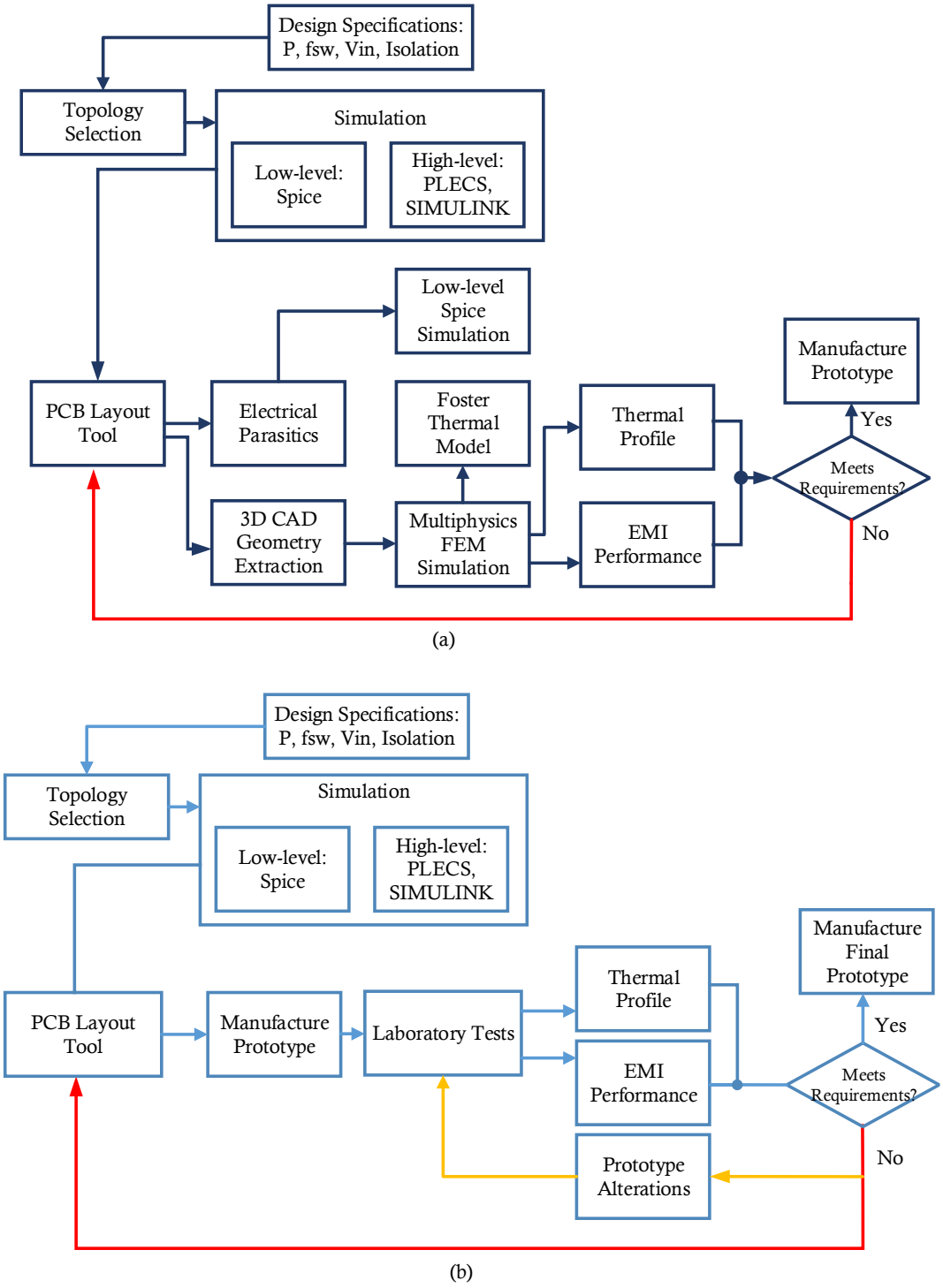


Figure 1.6: Framework for PCB-based converter design. (a) Virtual prototyping-based design (b) Physical prototyping design

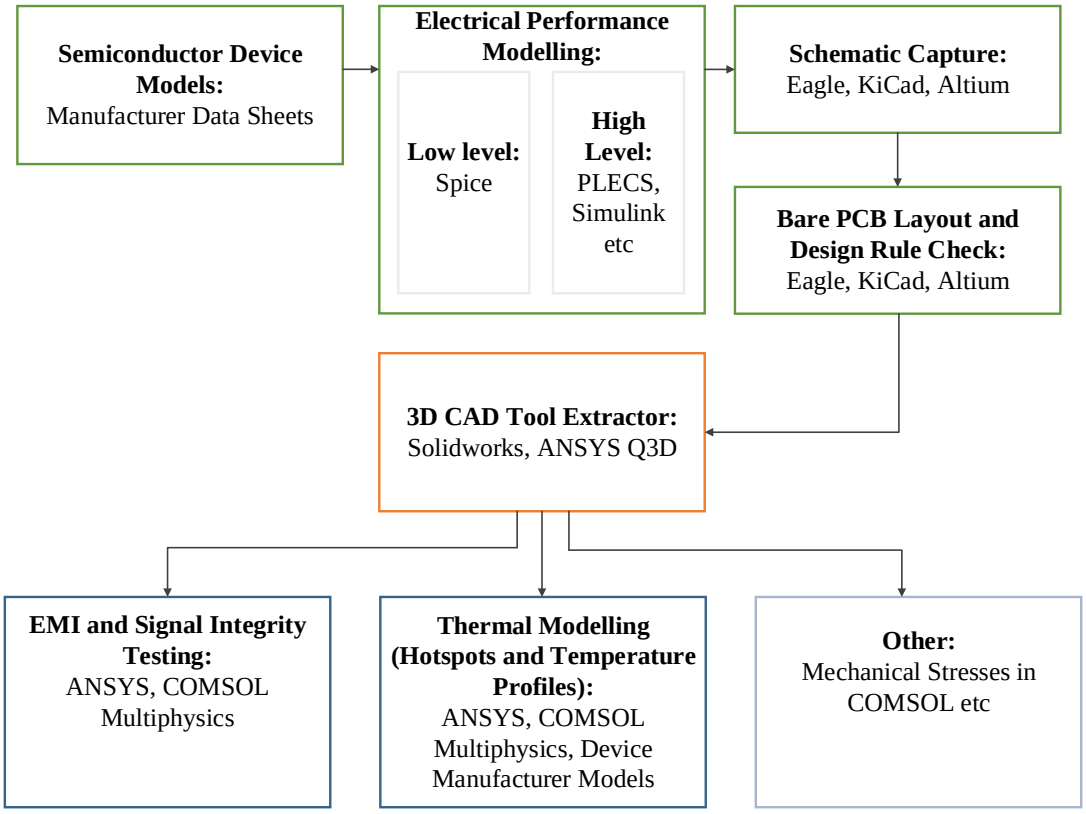


Figure 1.7: Example software suite for PCB design

pours on both power and ground planes. Naturally, this is an iterative process. The routing pattern is selected to minimize crosstalk and signal interference and to ensure that the board stays within acceptable passive impedance limits. While the experience and skill of the designer are important at this stage, fast and accurate tools that enable rapid prototyping are necessary to speed up the process.

The next step is to validate the electrical performance of the PCB using simulation and analysis tools such as SPICE simulation, IBIS modelling, and signal integrity analysis, for example in HyperLynx. Once complete, the fabrication documentation of the PCB (Gerber files that contain the PCB design, assembly drawings and a bill of materials) is prepared, ready to be sent to the manufacturer.

A prototype of the PCB is then built and tested to ensure that it meets the design specifications. Physical prototypes Fig (1.6. b) are expensive to produce and carry a significant time penalty in an iterative design process. Due to the high

cost of physical prototyping, it is desirable for designers to complete important prototyping activities in a virtual environment using software tools Fig (1.6. a). The thermal models proposed in this thesis are useful in a virtual PCB prototyping loop and can enable PCB designers to quickly gather insight into the thermal behaviour of intermediate design candidates and layouts, and optimize the design.

1.4 The PCB manufacturing process

Once a satisfactory prototype has been validated, the fabrication process follows. This is performed by a PCB manufacturer using the Gerber files, drilling data, and BOM generated using the steps described in the preceding section. Although the technology is mature and covered extensively in a number of books and online articles, we will now present a brief overview [15]. Though sophisticated, the process is highly automated and precise, and enables the creation of high-density circuit boards with multiple conductive layers. The core material is typically FR-4, a glass-reinforced epoxy laminate, chosen for its electrical and thermal properties, though alternatives such as polyimide or PTFE are sometimes used. Thin copper layers are bonded to each side of the core. The circuit pattern for each layer is then created using photolithography, after which a photoresist is applied, exposed to UV light through a mask, and developed. Unwanted copper is then removed through chemical etching or mechanical milling.

Next, prepreg sheets (pre-impregnated epoxy resin) are used to separate and insulate the conductive layers. The prepared layers are stacked precisely, often using alignment pins, and laminated in a hydraulic press under high temperature (around 175°C) and pressure (up to 400 psi) for several hours. This process cures the prepreg, bonding the layers into a solid board.

Holes are then drilled using high-speed CNC machines with carbide bits, capable of creating holes as small as 0.1mm in diameter. These holes are metallized

through electroless copper deposition followed by electrolytic copper plating, creating electrical connections between layers. The outer layers undergo additional processing, including applying solder mask (typically through screen printing or photoimaging) and surface finishes such as HASL (Hot Air Solder Leveling), ENIG (Electroless Nickel Immersion Gold), or OSP (Organic Solderability Preservative).

Finally, electrical testing is performed using flying probe or bed-of-nails testers to check for shorts, opens, and correct impedance. The boards are then routed or punched to their final shape, often using X-ray alignment for critical geometries.

1.5 Considerations for PCB design - a balancing act

In practice PCB designs are generally optimized for three things: (i) proper electrical function which is determined by the power integrity and signal integrity (*i.e.* maximizing the SNR), (ii) minimizing EMI and improving EMC for RF systems and (iii) thermal management (which is aimed at minimizing hotspots, thermal cycling and enabling the integration of different cooling solutions). While all these considerations are receiving intense research interest individually, they are rarely considered together due to the difficulty of coupling the different physics and competing design priorities that must be resolved when developing electronic systems and designers tend to rely on heuristic guidelines.

In the conventional design of PCB power electronic systems, the electrical and thermal aspects are often addressed separately in a sequential manner. The electrical design is finalized and optimized first before the thermal behaviour is considered. Commonly, the electrical design may be optimized to minimize losses or to maximize the power density. This is followed by the design of a thermal management system that must meet the safe operating temperatures of all the electronic components and may additionally seek to optimize a relevant metric such as minimizing the weight of

the heat sink. This decoupled design optimization strategy may lead to sub-optimal system-level designs, and in some cases costly rework later in the design cycle.

Several authors have proposed techniques for simultaneously considering the thermal and electrical performance of the system. For example, in [84] and [106], the authors showed that considering both the thermal (heat dissipation of the MOSFETs) and electrical (parasitic loop inductance) constraints in the design of an onboard EV charger in parallel produced the best-performing design than a design optimized for one function before the other is considered.

Inherently, such optimization-based approaches are multi-objective and thus yield several locally optimal solutions and the designer's heuristics are leveraged to select the final design. Developing techniques for integrating electrical and thermal design (using so-called electro-thermal co-design) has thus attracted research and industrial development interest (for example from CFD tool vendors such as Cadence) [67].

1.6 Thermal modelling and thermal management of PCB systems

1.6.1 Origin of heat in PCB-based electronic systems

The energy balance during the operation of an electronic system is summarized in Fig 1.8. The most fundamental source of heat in electronic systems is Joule heating, also known as ohmic or resistive heating. This is a phenomenon that occurs when an electrical current flows through a conductor and encounters resistance. The resistance causes the conductor to dissipate heat, and the heat generated is directly proportional to the square of the electrical current and the resistance of the conductor and is given by the equation:

$$P_{\text{loss}} = I^2 R \tag{1.1}$$

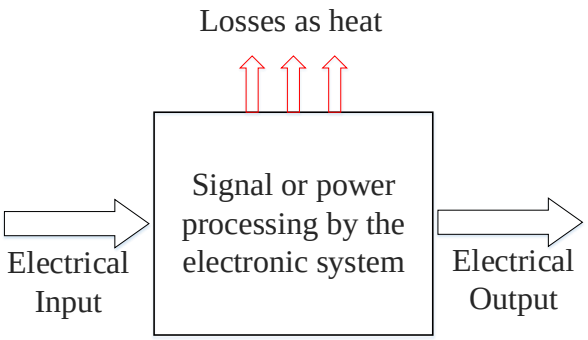


Figure 1.8: Thermal losses in an electronic system

where P_{loss} is the thermal power lost, I is the current and R is the resistance of the conductor [22]. At the atomic level, Joule heating results from the conversion of kinetic energy into thermal energy caused by the collisions between the moving electrons with the atoms of the conductor, causing it to heat up.

1.6.2 Passive components

Joule losses in passive devices are often modelled by assuming the device has an equivalent effective series resistance (ESR). This arises from the fact that all non-superconducting materials have finite electrical resistance defined by their physical dimensions (cross-sectional area and length), and material properties (resistivity). The ESR is responsible for dissipating heat when a current flows through the wires used in the construction of the device.

The ESR in capacitors is a result of the resistance of the metallic conductors used to construct the component, typically ranging from 0.01 Ω for ceramic capacitors to over 1.5 Ω for aluminium capacitors. The power losses in a capacitor are given by the Joule heating equation above.

The loss mechanisms in inductors can be split into core losses and copper losses (the Joule heating portion of the losses) [83], [74]. Many models of the copper losses of a power inductor are known and can be summarized by the expression:

$$P_{\text{loss}} = R_{\text{DC}} I_{\text{DC}}^2 + c_0 R_{\text{DC}} \frac{I_{\text{AC,pk}}^2}{2} \tag{1.2}$$

where I_{DC} is the DC current flowing through the inductor, I_{AC} is the AC current, and R_{DC} is the effective series resistance of the inductor obtained from its physical structure such as the number of turns, length of conductors, cross-sectional area of the window *etc* and c_0 is a constant that represents the skin and proximity effects, which become a concern in high-frequency operation [74], [83].

Other auxiliary passives such as copper traces, vias, and bus bars also experience Joule heating. Although in most cases the effect of such heating is negligible, it may sometimes play a major role in raising the overall board temperature and has been studied in research literature [16], [25], [101], [123].

The heating contribution of passive components is often ignored in engineering practice because they do not typically generate enough heat to cause thermal hotspots or require direct cooling, for example using a heat sink.

1.6.3 Active components

In most electronic systems, heat dissipation by active devices dominates the losses. For example, MOSFETs dissipate heat through a combination of conduction losses, which are analogous to Joule heating and account for the heat generated when the drain current flows through the on-state resistance of a MOSFET, as well as switching losses. The conduction losses in a MOSFET are given by:

$$P_{\text{cond}} = I_{\text{D,on}}^2 R_{\text{DS,on}} D \tag{1.3}$$

where $I_{\text{D,on}}$ is the drain to source current that flows when the device is on, $R_{\text{DS,on}}$ is the turn-on resistance of the device, and D is the duty cycle of the gate driving signal.

Switching losses result from the nonalignment of the voltage and current edges during the switching action of the MOSFET such that the current continues to flow in the device during turn-on and turn-off [39], [115]. Switching losses are given by:

$$P_{\text{sw}} = (E_{\text{on}} + E_{\text{off}}) f_{\text{sw}} \tag{1.4}$$

where f_{sw} is the switching frequency, E_{on} and E_{off} are the turn-on and turn-off energies and respectively approximated by:

$$E_{\text{on}} = I_{\text{D,on}} V \frac{t_{\text{on}}}{2} \tag{1.5}$$

$$E_{\text{off}} = I_{\text{D,on}} V \frac{t_{\text{off}}}{2} \tag{1.6}$$

A diode also dissipates due to its forward voltage drop and reverse recovery [70].

One or more of these heat generation mechanisms can be observed in complex IC packages, and processor units in motherboards and high-performance computing hardware, which are constructed from large-scale integration of the basic electronic components we have discussed so far.

The thermal models developed in this thesis (see Chapters 3 and 4) do not consider the specific heat generation mechanism. Instead, we focus on the size of the heat loss P_{loss}, and its location (i.e., area and footprint within the PCB), to be supplied by the thermal designer.

1.6.4 Consequences of excess heat in electronic systems

Many failures of electronic systems (for example damage to one or more of the components) are facilitated by high-temperature operation exceeding the maximum safe operating temperature and thus are more likely to occur due to poor thermal management. Common thermal failure modes in active devices include the breakdown of the oxide layer in the MOSFETs, a phenomenon that occurs when

safe operating temperatures are exceeded [64]. Additionally, current leakage can also occur and is worsened by high-temperature operation.

Thermal expansion accompanies the heating of all materials. The multi-material nature of PCBs and their constituent electronic components makes them prone to mechanical stresses induced by mismatches in the coefficients of thermal expansion, causing failure in the solder joints and internal wire bonds. Additionally, the electrical resistivity of materials typically increases with temperature,leading to diminished electrical performance, manifesting, for example as higher conduction losses in MOSFTETs at high temperatures (*i.e.* higher losses).

Delamination, another critical failure mode in PCBs and IC packages occurs when the adhesive bonds between different layers of the PCB or within the IC package weaken and separate due to thermal stress, potentially leading to electrical disconnections, increased thermal resistance, and compromised structural integrity [26] [47]. Delamination is particularly problematic in multi-layer PCBs and can be exacerbated by repeated thermal cycling. Once initiated, the process is prone to propagating, and leading to catastrophic failure.

Furthermore, bond wire lift off is a thermal-induced failure mechanism common in power electronic modules [19]. As components heat up, the difference in thermal expansion between the bond wire and the die or substrate creates mechanical stress at the interface. Over time this causes the bond wire to detach from the die, resulting in open circuits which are a precursor to device malfunctioning. This issue is particularly critical in high-power applications where large currents flow through the bond wires.

Thermal runaway in power electronics is a self-reinforcing process where an increase in temperature causes a device to draw more current, which generates additional heat and further elevates the temperature. This cycle arises in semiconductor components such as transistors and diodes when higher temperatures reduce their electrical resistance or alter their material properties, leading to increased current

flow. For instance, as operating temperature rises, the drain-to-source on-resistance of most power MOSFETS typically increases, leading to higher power dissipation. If the heat cannot be removed quickly enough, this cycle continues, until the device fails potentially destroying both the switch and surrounding components [62] [114].

If it is not mitigated, thermal runaway results in excessive heating, degradation of device performance, and ultimately lead to device failure or catastrophic breakdown.

The thermal failure modes discussed in this section underscore the critical importance of effective thermal modelling to better inform thermal management in electronic systems. Proper design considerations, including adequate heat dissipation mechanisms and careful material selection, are essential to mitigate these risks and ensure the reliability and longevity of electronic devices and systems.

1.7 Motivation: Challenges and opportunities in PCB thermal modelling

Thermal modelling enables problems to be diagnosed early and changes to be made in less-costly virtual environments, reducing the number of prototypes needed for verification. It can also help power electronics PCB designers to answer important design questions such as the level of derating required for the semiconductor devices, the airflow requirements to aid dissipation by convection and the selection of appropriate cooling solutions.

Thermal modelling at the component level is an active research area. Several research works have proposed FEA-based thermal models for inductors, high-speed transformers, MOSFETs, and thermal vias [58] [110] [102] [39] [116] [105]. While useful for understanding the thermal behaviour of individual components, many of these research works treat the thermal models of these components in isolation and do not address the challenge of integrating them in a PCB assembly.

Using FEA for thermal modelling remains a slow and computationally intensive process. Many commercially available simulators require extensive computing

resources to solve the large systems of equations characteristic of high-fidelity thermal models and can be extremely memory intensive. The upfront cost of acquiring such computing hardware can be prohibitive in some cases, thus making lightweight models that can be constructed, solved and post-processed quickly using desktop computing an attractive proposition.

Additionally, a lack of standardization among the tools involved in PCB electrical and thermal design (Fig 1.7) makes integrating the workflows challenging. Many vendors of CFD simulators have proprietary software packages, and file formats that do not easily integrate with other tools in the PCB design suite. Thus, neutral file formats that can be used to extend the functionality of EDA tools with thermal modelling tools are attractive. To be able to obtain near real-time thermal performance insight in the design cycle, designers need fast ways of obtaining accurate thermal profiles. While many CFD software supports the importation of standard PCB CAD files such as STEP, ODB++, and Gerber, they do so at an additional prohibitive cost and time penalty. It is often required for the imported PCB CAD geometry to be preprocessed, for example by performing lengthy defeaturing, and manual mesh refinements, before thermal analysis by FEM can be performed.

With this background, developing a fast and accurate numerical thermal modelling process that leverages neutral file formats in common use (such as rasterized images) is an attractive proposition. The raster image-based FDM thermal modelling of PCB-based electronics proposed in this thesis aims to address this issue.

1.8 Outline and Contributions of this work

The primary objective of this work is to enhance the speed of thermal modelling for power electronics PCBs by using raster images as a foundation for FDM discretization. We introduce image processing techniques capable of generating a

thermally accurate PCB model while maintaining essential features across a broad range of image resolutions. We then present a computationally efficient modelling approach to generate high-quality thermal profiles of the PCB.

Additionally, we develop comprehensive and mathematically rigorous thermal models of vias, heat sources and heat sinks from the combined analysis of raster images and the use of available datasheet thermal parameters. We also propose a method for reducing the number of nodes in a pixel-derived FDM mesh of a PCB by applying a downsampling technique that is based on analysing tiles of pixels to calculate the anisotropic thermal properties of the PCB precisely.

This chapter has presented a comprehensive overview of PCB technology and its thermal features and design aspects. Chapter 2 introduces the theories and concepts of heat transfer and numerical analysis of the heat equation using finite differences, which form the basis of the thermal models proposed in this thesis. A review of thermal modelling techniques covered in the literature is also provided. Chapter 3 develops a raster image-derived discretization scheme for a multilayer PCB and verifies its usefulness and accuracy for modelling the thermal behaviour of simple PCB structures. We extend this model in Chapter 4 to incorporate vias, heat sources and heat sinks, for which we propose a 'detached node' thermal model that incorporates data-sheet thermal parameters. Chapter 5 explores how a tiling downsampling approach can preserve the thermal properties and accuracy of the thermal model of a PCB at low-layer image resolutions. In Chapter 6, we summarise the contributions of this thesis to fast thermal analysis of PCBs and identify further research questions that arise from our work.

2

Background: 3D thermal modelling of electronic systems

This chapter presents a brief discussion of the fundamentals of heat transfer as well as the mathematical foundations of the finite difference method. In addition to being used throughout this thesis, these general concepts are the basis on which we develop the steady state raster-image-derived thermal models for PCBs.

Furthermore, a survey of the literature is conducted to summarize prevailing techniques for modelling thermal problems that arise in electronic systems, revealing their strengths and limitations when applied to PCBs. Potential gaps in the current literature are identified and a framework for addressing them is proposed.

2.1 Modes of heat transfer

When heat flows from a source to the ambient environment, it follows thermal paths provided by the medium in contact with the source. In general, the medium can be one of three options: a solid, a fluid (gas or liquid), or a vacuum. There are three fundamental laws that govern the modes of heat transport in such mediums, namely: Fourier's law of heat conduction in solids, Newton's law of cooling in fluids and Stefan-Boltzmann's law of radiative heat transfer in vacuums (Fig 2.1). Although a

combination of all these laws collectively governs heat transfer in all systems, it is often the case that one or more is negligible, enabling thermal designers to ignore the less dominant ones to simplify thermal analysis without meaningful loss of accuracy.

2.1.1 Conduction

Conduction is the transfer of heat energy in solids along a negative temperature gradient, *i.e.*, from a region of higher temperature to one with lower temperature. It arises due to the exchange of translational, rotational, and vibrational energy among the molecules comprising the solid. Fourier's law of heat conduction governs this phenomenon, and for a homogeneous solid can be written as:

$$\mathbf{q}(x, y, z) = -\kappa \nabla T(x, y, z) \tag{2.1}$$

where $\mathbf{q}(x, y, z)$ is the heat flux per unit area of the surface normal to the direction of heat flow in $\mathrm{W/m^2}$ and κ is the thermal conductivity of the solid in W/mK. More generally, for a non-homogeneous thermal domain, the thermal conductivity exhibits spatial variation, and Fourier's law is written in Cartesian coordinates as:

$$\mathbf{q}(x, y, z) = -\hat{\mathbf{i}}\kappa_x \frac{\partial T}{\partial x} - \hat{\mathbf{j}}\kappa_y \frac{\partial T}{\partial y} - \hat{\mathbf{k}}\kappa_z \frac{\partial T}{\partial z} \tag{2.2}$$

where $\hat{\mathbf{i}}$, $\hat{\mathbf{j}}$ and $\hat{\mathbf{k}}$ are unit vectors in the x, y and z directions respectively. This form of the equation captures the inherent anisotropic nature of PCBs which are of interest in this work.

2.1.2 Convection

Heat transfer by convection is facilitated by fluids in motion. Forced convection arises when such motion is driven by an external force, for example, a stream of air resulting from a fan or a liquid pumped through a region of high temperature, such that it carries the heat from a source to another region for dissipation. On the other hand, natural convection is inherent to most systems due to the expansion

of heated fluids. Such convection is a result of the buoyancy that results from temperature gradients within a fluid which cause the density to decrease in the hotter regions by an expansion of the volume.

Newton's law of cooling governs the rate of convective heat transfer and is written as:

$$q = h(T - T_\infty) \tag{2.3}$$

where T is the temperature of the surface, T_∞ is the reference temperature of the fluid, q is the heat flux leaving the surface in $\mathrm{W/m^2}$ and h is the heat transfer coefficient in $\mathrm{W/m^2K}$. In most thermal analysis of electronics, h is assumed to be known to a reasonable degree of accuracy. However, the actual value of h is difficult to obtain in practice because it depends on many factors such as the properties and flow characteristics of the fluid in contact with the surface. Experimental methods of measuring the constant for electronic systems exist. For the systems considered here, heat convection appears as a boundary condition for the heat conduction problem, assigned to established values obtained from the relevant literature.

2.1.3 Radiation

Radiation involves heat transfer in the absence of a facilitating medium, or more strictly, in a vacuum. Under these conditions, thermal radiation propagates as electromagnetic waves. Stefan-Boltzmann's law relates the maximum possible radiative heat flux that can be emitted from a black surface to its temperature and is written as:

$$q = \epsilon\sigma T_\infty^4. \tag{2.4}$$

Here the emissivity of the black material, ϵ, is unity and dimensionless and the Stefan-Boltzman constant, σ is 5.67×10^{-8} $\mathrm{W/(m^2K^4)}$. In practical systems $0 < \epsilon < 1$.

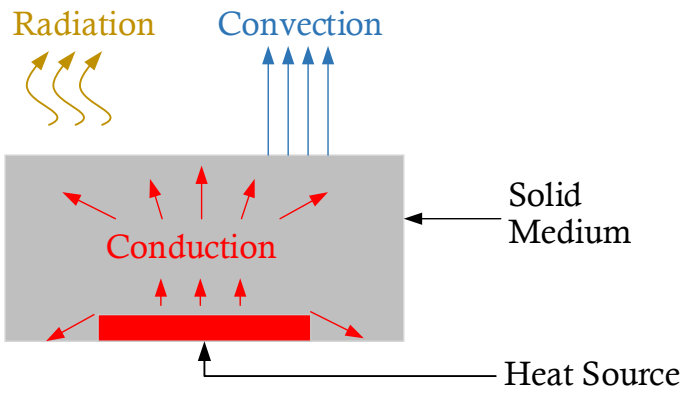


Figure 2.1: Heat transfer in a non-uniform solid showing the three processes: conduction, convection, and radiation

A generalized form of Stefan-Boltzman law captures the radiative heat flux from one surface of interest, Γ, at temperature T_Γ and a larger surface, ∞, at a temperature T_∞ as:

$$q = \epsilon\sigma(T_\Gamma^4 - T_\infty^4) \tag{2.5}$$

2.2 Heat transfer for a PCB

In electronic systems, the goal of thermal management and thermal design is to remove heat from dissipating sources on either the top or bottom layers of the board efficiently. Fig 2.2 summarizes the heat transfer processes that occur within and from a PCB.

Heat conduction is a key heat transfer process within a PCB. Copper pours in contact with heat sources provide high thermal conductance paths that spread heat within the plane of the PCB. Thermal via arrays are typically placed under heat sources to provide low thermal resistance paths along the vertical plane of the PCB through which heat is transported to the opposite layer for dissipation.

Convection plays an important role in rejecting wasted heat from the surface area of the PCB to the ambient environment. Because PCBs are typically very

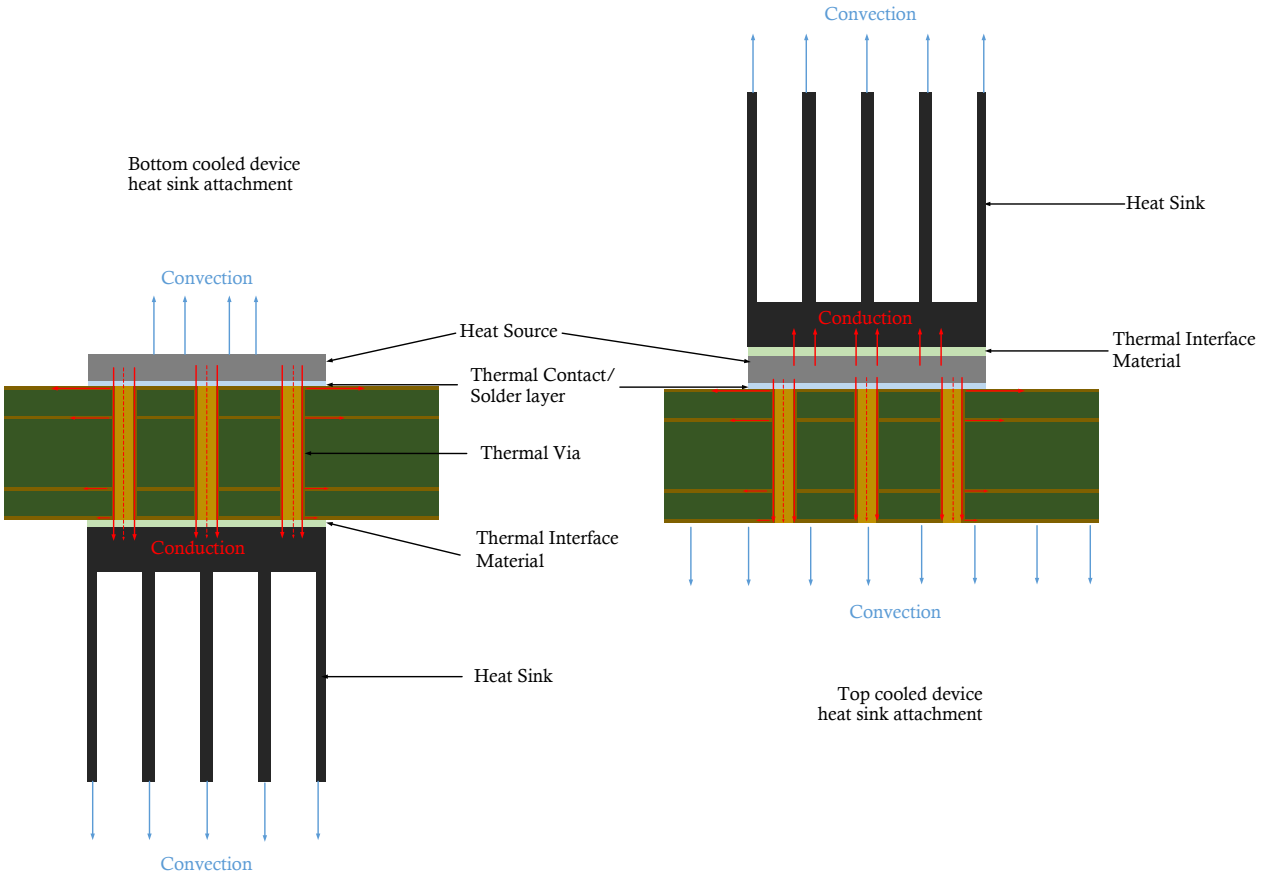


Figure 2.2: Heat transfer on a PCB showing conduction in red and convection in blue

thin (thicknesses of less than 2 mm are typical), the sides of the board provide a negligible surface area and play a small role in convective heat transfer. The surface area of a typical heat source on a PCB is usually too small to facilitate meaningful convection. As a result, heat sinks are attached to the heat source using thermally conductive interface materials to increase the available surface area for convective heat transfer to the ambient environment.

It is reasonable and common practice to neglect radiative heat transfer from a PCB because of its negligible contribution to the overall thermal behaviour of the system. For example, the authors of [107] found that radiative heat transfer from a PCB surface contributed to less than 3% of the heat losses associated with a heat source. The common materials used to construct a PCB tend to have low

emissivity and thus emit and absorb radiation less effectively than black bodies, further reducing the impact of the process in heat transfer within the board.

2.3 The heat equation

The heat equation is a well-known partial differential equation that relates the rates of change of temperature within a medium with respect to both space and time. The heat equation can be derived in Cartesian coordinates by solving for the energy balance on a small cubic control volume. This is done by ensuring that the sum of the heat energy *entering* and *generated* by the system is equal to the sum of heat energy *leaving* and *stored* by it. The details of solving such an energy balance are well understood and treated in sufficient detail in many textbooks and are omitted here [81].

Instead, we start with an expression of the heat equation in Cartesian coordinates:

$$\frac{\partial}{\partial x}\left(\kappa_x \frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(\kappa_y \frac{\partial T}{\partial y}\right) + \frac{\partial}{\partial z}\left(\kappa_z \frac{\partial T}{\partial z}\right) + q = \rho c_p \frac{\partial T}{\partial t} \tag{2.6}$$

where κ_n is the thermal conductivity in the direction normal to surface n where $n \in \{x, y, z\}$, q is the volumetric heat generated by the system, c_p is the specific heat capacity and ρ is the density of the medium.

Here, we express the heat conduction equation in a form where the thermal conductivity, κ exhibits variations in the three Cartesian directions x, y, and z which is valid for anisotropic thermal domains. If the thermal conductivity is uniform in all directions within the domain, the domain is said to be thermally isotropic and $\kappa_x = \kappa_y = \kappa_z$.

The solution to (2.6) gives the temperature distribution within the medium subject to the prescribed boundary and initial conditions. The solution approaches steady state (i.e. thermal equilibrium) as time tends to infinity.

A simplified version of the equation can be obtained by treating the thermal conductivity as a spatially non-varying property - such that the material in question is treated as isotropic. We rewrite (2.6) as:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} + \frac{q}{\kappa} = \frac{1}{\alpha}\frac{\partial T}{\partial t} \tag{2.7}$$

In (2.7) we have defined a material property, α, called thermal diffusivity, which measures the rate at which heat propagates through the medium in terms of thermal conductivity, specific heat capacity c_p and density, ρ:

$$\alpha = \frac{\kappa}{\rho c_p} \tag{2.8}$$

For completeness, the boundary conditions must be defined and can be Dirichlet (where the boundary temperature is known) or Neumann (where the heat flux is predefined) boundary conditions, or a combination of both. In steady state thermal analysis, the right-hand side of (2.7) is set to zero. For heat transfer within a PCB, the boundary condition of concern is the convective heat flux normal to the surface of the PCB with a reference temperature of T_∞. We express this Neumann boundary condition as:

$$q = \kappa \frac{\partial T}{\partial y}\bigg|_{\text{surf}} = h(T - T_\infty) \tag{2.9}$$

2.4 Methods for solving the heat equation for electronic thermal models

Several methods of obtaining the solution to (2.6) for electronic thermal systems exist. It is usually not possible to obtain closed-form solutions to (2.6) due to the complicated geometries encountered in PCB-based electronics. The following is an overview of notable attempts in the literature to solve the problem using analytical and numerical approaches.

2.4.1 Analytical solutions

Analytical methods are based on applying mathematical equations to describe the thermal behaviour of electronics. They often rely on using simplifying assumptions and idealizations, and thus may not be able to capture the detailed and high fidelity features such as the non-uniformity of the PCB materials, the irregularity of the geometry and the interaction of the components. Among the most prominent analytical methods for solving the heat equation for electronic domains are Green's functions [35, 65, 80, 104, 112, 119] and separation of variables [52], [113].

When the steady state heat equation is solved by separation of variables, a general solution that is typically a truncated triple Fourier series summed up to some suitable limit, N, is obtained:

$$T(x,y,z) = \sum_{n=1}^{N}\sum_{m=1}^{N}\sum_{p=1}^{N} A_{n,m,p}\sin(\beta_n x)\sin(\eta_m y)\sin(\mu_p z) + q(x,y,z) \qquad (2.10)$$

where β_m, η_n and μ_p represent the set of eigenvalues, $q(x,y,z)$ is a function of the source term, and $A_{n,m,p}$ denotes the Fourier coefficient.

Ellison [29, 30] proposed the *TAMS* and *PTAMS* software packages that apply the solutions to three-dimensional heat conduction to four-layer rectangular structures with discrete sources on the surface. While these structures roughly approximate the most rudimentary PCBs, they are poor representations of realistic PCBs, and cannot account for more than a few components. Particularly, vias, and holes, all critical features to the thermal behaviour of the PCB, are not modelled. Also, the solutions were obtained for unpatterned copper layers under the assumption that the layers can be represented by equivalent thermal conductivities. This simplifying assumption has been shown to result in wildly erroneous results for PCB thermal modelling, with reported errors exceeding 50% [122].

In [112], the authors proposed an analytical approach that uses Green's functions to compute the temperature distribution in multilayer chips, solutions of which could

be used in an inner loop during the calculation of the overall thermal behaviour of the PCB system on which the chip resides. While attractive, this approach suffers from its lack of integration of the entire PCB by decoupling the components from other features on the board.

The authors of [122] proposed a coupled analytical-numerical solution to heat conduction in PCBs by observing that by neglecting the horizontal thermal contributions of vias, the FR4 layers could be treated as homogeneous domains for which Fourier-based analytical solutions can be obtained. While reasonable for simple boards with low via counts and no large holes, these assumptions would not hold for PCBs with dense via arrays, as is common in industrial applications, particularly power electronics. Other research focusing on solving the heat equation for multilayered rectangular structures with one dominant heat flow direction and fairly homogenous layers fail to generalise to realistic PCBs where heat flow is multi-directional, and the conducting layers have complex patterns while the insulating layers are populated by conducting vias, holes and cutouts.

Although these approaches have limited utility for PCB-based electronics, they have found applications in other electronic systems, for example, power modules, and discrete ICs, where both the geometries and the conducting copper patterns are simpler [31, 78].

2.4.2 Numerical solutions

To address the inherent limitations of analytical solutions when applied to complex geometries, numerical approaches that approximate the temperature profiles by discretizing the domain and computing the temperature at distinct locations have been proposed. While an analytic solution to the heat equation provides a continuous representation of temperature in space and in time, numerical approaches approximate the temperature values at predefined discrete points, referred to as nodes, when using numerical methods. Finite difference and finite element methods

are often used to solve thermal problems in this way and are discussed in the next two subsections for the steady state case.

The Finite Element Method

The finite element method (FEM) subdivides a problem geometry into a finite number of interconnected subregions or elements. The governing partial differential equation is then approximated for each of the elements using simpler linear or non-linear algebraic equations, and solved in a piece-wise fashion before being assembled into the solution of the larger system.

Key to the finite element method is the discretization step where the domain is broken into a mesh, a collection of simply-shaped elements that wholly approximate the complex domain. Many options exist for the shapes of the finite elements, ranging from simple lines for 1D problems to triangles and quadrilaterals for 2D problems and tetrahedrons, hexahedrons, pyramids, or triangular prisms for 3D domains (Fig 2.3). The FEM discretization has the advantage of combining several shapes of elements in ways that enable the mesh to resemble the original domain to a high degree of fidelity.

Other advantages of FEM include the ability to apply directional sizing without loss of accuracy and the possibility to apply adaptive meshing, a technique that increases the element count in select regions of the domain where large temperature gradients are expected by using smaller element sizes.

FEM Mathematical Background

The FEM for thermal problems can be expressed as a system of equations of the form [4]:

$$\mathbf{KT} = \mathbf{F} \tag{2.11}$$

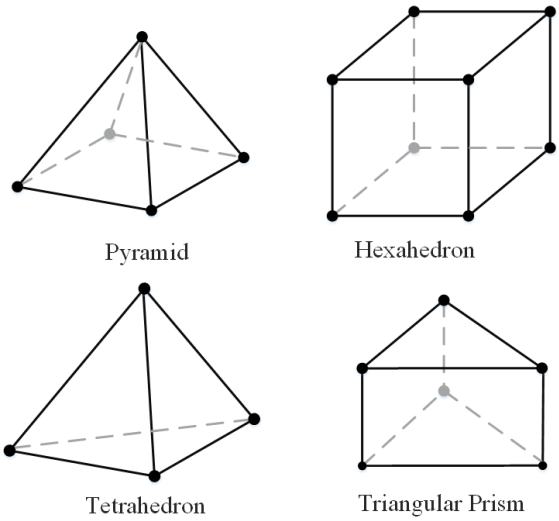


Figure 2.3: Types of 3 D elements

where $\mathbf{T}$ is the vector of unknown temperatures, and $\mathbf{K}$ is the global conductivity matrix:

$$\mathbf{K} = \int_{\Omega} \mathbf{B}^{\top}\mathbf{D}\mathbf{B}d\Omega + \int_{\Gamma} h\mathbf{N}^{\top}\mathbf{N}d\Gamma$$

with $\mathbf{F}$ representing the set of so-called forcing functions:

$$\mathbf{F} = \int_{\Omega} \mathbf{q}\mathbf{N}^{\top}d\Omega + \int_{\Gamma} q\mathbf{N}^{\top}d\Gamma + \int_{\Gamma} hT_{\infty}\mathbf{N}^{\top}d\Gamma$$

The remaining terms in the FEM equations above have their usual meanings where Ω is the entire region under consideration, Γ is the boundary of the domain, $\mathbf{B}$ is the matrix of shape function derivatives, $\mathbf{D}$ is the material property matrix containing thermal conductivity information, and $\mathbf{N}$ is the shape function vector. The heat transfer coefficient is represented by h, while q denotes the heat generation rate per unit volume. T_{∞} refers to the ambient temperature.

For a region, Ω discretized to contain many elements where we can represent a single element as Ω^e, the spatial variation of temperature in Ω^e in the x, y, and z directions can be approximated by the function:

$$T(x, y, z) = \sum_{i=1}^{n} T_i^e N_i(\zeta, \eta) = [\mathbf{N}]^\top T^e \tag{2.12}$$

where n is the number of nodes in the element, T_i^e is the temperature at node i and $N_i(\zeta, \eta)$ is the basis function (also called the shape function). T^e is the element nodal temperature vector, N is the shape function vector. Within the discretized domain, Ω^e represents a single element, with T_i^e being the temperature at node i of that element. $N_i(\zeta, \eta)$ is the basis or shape function for node i in local coordinates ζ and η.

This integration-based formulation makes FEM particularly well-suited for the thermal analysis of PCBs, which often involve complex geometries and multiple materials. It is important to note the fundamental difference between the Finite Element Method and the Finite Difference Method lie in their approach to solving partial differential equations. FEM is fundamentally based on integration, as evident in the formulation of $\mathbf{K}$ and $\mathbf{F}$, which are defined using integrals over the domain Ω and its boundary Γ. By using integration, FEM can better handle complex geometries more effectively and naturally incorporate boundary conditions.

In contrast, FDM is based on differentiation, approximating derivatives in the governing differential equations using finite difference approximations. While FDM can be simpler to implement for regular geometries, it may require special treatment for complex geometries or boundary conditions.

To solve a FEM problem, a mesh consisting of many thousands of elements, (or even some hundred million elements for complex PCB electronics) must be assembled, conforming as closely as possible to shapes of all the features in the domain. Localized calculations at the element level are made trivially based on the polynomial basis functions defined earlier, and then combined in large sparse matrices that are inverted to give the final result. Clearly, the number of elements has an important impact on the speed of attaining a solution. It is important to

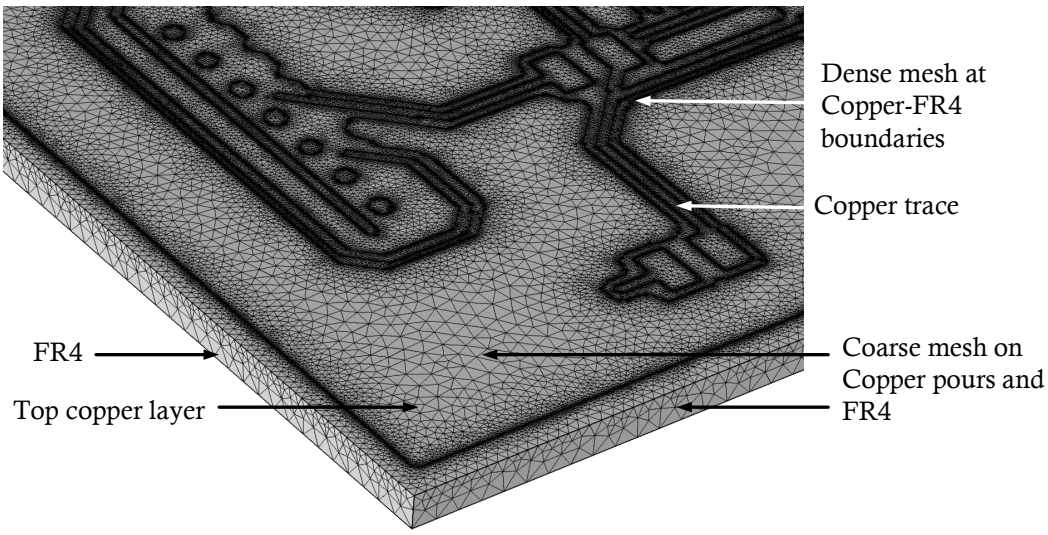


Figure 2.4: Example mesh of a section of a PCB using tetrahedral elements. Note the adaptive nature of the mesh with a refined grid in regions with thin copper and larger tetrahedrons within copper pours and FR4.

note that the unknowns are calculated at the nodes, and then interpolated to other locations in the domain that are not directly located at the nodes.

Although FEM is widely used in electronic thermal design, it is uncommon for electronic thermal designers to implement the method from scratch. Instead, commercially available software that implements the model, such as COMSOL, ANSYS, and Flotherm, are commonly used [45]. Fig 2.4 shows an example mesh for an FEM discretization of the PCB.

The Finite Difference Method

The finite difference method is another well-known and established numerical method for solving partial differential equations. Because this work focuses on solving PCB thermal models by the finite difference method, we now present a detailed mathematical background of the numerical procedure for thermal problems as it applies to electronic systems.

The finite difference method involves partitioning the geometry into rectilinear cells with connected grid points, also called nodes. The continuous PDE representing

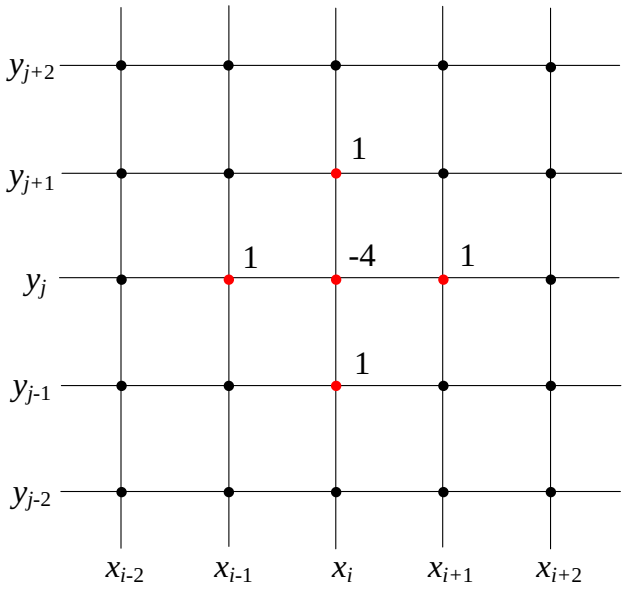


Figure 2.5: Grid of points for a generic finite difference discretization. The nodes contributing to the 5-point stencil for the Laplacian about the point (i, j) are shown in red (adapted from [60] [57])

the heat equation is then approximated at each of the nodes using simpler equations called finite differences. The unknown temperature values can be computed at these nodes based on the known initial values and boundary conditions.

To illustrate the finite difference discretization for the steady state heat equation, we consider a 2D problem which is a simpler version of (2.6) and written:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{q}{\kappa} = 0 \tag{2.13}$$

The formulation that follows is easily extendable to the 3D case.

Consider the grid of points shown in Fig 2.5, which form part of a larger $m \times n$ grid of points representing a rectangular domain.

Equation (2.13) consists of spatial second derivatives and can be approximated using the Taylor series expansion of the second derivative.

To do this, we focus on one grid point (i, j) whose temperature is T_{ij}. The truncated Taylor series expansion for the second derivative of x at this node is given by:

$$\left(\frac{\partial^2 T}{\partial x^2}\right)_{i,j} \approx \frac{1}{(\Delta x)^2}\left(T_{i-1,j} - 2T_{ij} + T_{i+1,j}\right) + \aleph \tag{2.14}$$

Here Δx is a small displacement from the point (i, j) to points $(i + 1, j)$ and $(i - 1, j)$ along the x-axis and $\aleph$ is the error term. A similar expression can be written for $\frac{\partial^2 T}{\partial y^2}$ as:

$$\left(\frac{\partial^2 T}{\partial y^2}\right)_{i,j} \approx \frac{1}{(\Delta y)^2}\left(T_{i-1,j} - 2T_{ij} + T_{i+1,j}\right) \tag{2.15}$$

Combining (2.14) and (2.15) allows (2.13) to be written in its discrete form as:

$$\frac{1}{(\Delta x)^2}\left(T_{i-1,j} - 2T_{ij} + T_{i+1,j}\right) + \frac{1}{(\Delta y)^2}\left(T_{i,j-1} - 2T_{ij} + T_{i,j+1}\right) + \frac{q_{i,j}}{\kappa} = 0 \tag{2.16}$$

where $q_{i,j}$ is the source term associated with the node i, j. Although the small displacements Δx and Δy need not be equal, we assume they are for simplicity and replace both with the variable w. Therefore we can rewrite the centred finite difference problem (2.16) as:

$$\frac{1}{w^2}\left(T_{i-1,j} - T_{i+1,j} - T_{i,j-1} - T_{i,j+1} + 4T_{ij}\right) = \frac{q_{i,j}}{\kappa} \tag{2.17}$$

The above formulation allows us to model discrete heat sources within the domain by applying the source term q to specific grid cells.

We can now write an algebraic equation of the form (2.17) at each of the grid points located at the indices (i, j) where $i = 1, 2, ..., m$ and $j = 1, 2, ..., n$. Each difference equation relates the unknown nodal temperature $T_{i,j}$ to known values of Δx, Δy, and $q_{i,j}$. Collecting these simultaneous equations yields a linear system of $m \times n$ unknowns and $m \times n$ equations. The difference equations at points near the boundary will involve the known boundary values such as the temperature or its rate of change and can be moved to the right-hand side of the equal sign.

If we arrange the equations into a matrix $\mathbf{L}$, the resulting system has the following coefficient matrix:

$$\mathbf{L} = \frac{1}{w^2} \begin{bmatrix} \mathbf{W} & \mathbf{I} & & & \\ \mathbf{I} & \mathbf{W} & \mathbf{I} & & \\ & \mathbf{I} & \mathbf{W} & \mathbf{I} & \\ & & \ddots & \ddots & \ddots \\ & & & \mathbf{I} & \mathbf{W} \end{bmatrix} \tag{2.18}$$

The diagonal entry of $\mathbf{L}$, $\mathbf{W}$, is an $m \times m$ block tridiagonal matrix given by:

$$\mathbf{W} = \begin{bmatrix} -4 & 1 & & & \\ 1 & -4 & 1 & & \\ & 1 & -4 & 1 & \\ & & \ddots & \ddots & \ddots \\ & & & 1 & -4 \end{bmatrix} \tag{2.19}$$

The off-diagonal matrix $\mathbf{I}$, is an $m \times m$ identity matrix.

The values of the ones in the $\mathbf{I}$ matrix are located $m-1$ points away from the main diagonal because these coefficients correspond to grid points lying above or below the central point in the stencil and hence are in the next. or previous row of unknowns. Following a systematic left-to-right and top-to-bottom numbering scheme for the nodes, we can write the unknown temperature vectors as:

$$\mathbf{T} = \begin{bmatrix} T^{[1]} \\ T^{[2]} \\ \vdots \\ T^{[i]} \end{bmatrix} \tag{2.20}$$

where

$$T^{[i]} = \begin{bmatrix} T_{i1} \\ T_{i2} \\ \vdots \\ T_{ij} \end{bmatrix} \tag{2.21}$$

$\mathbf{q}$ can be constructed in a similar fashion.

The final system can then be written in the form:

$$\mathbf{LT} = \mathbf{q} \tag{2.22}$$

This finite difference scheme is often represented by the 5-point stencil indicated by the red grid points in Fig 2.5. The 5-point stencil gives us insight into the

properties of **L** which is banded, with 5 diagonals and at most 5 nonzero elements per row. The stencil greatly informs the structure of the sparse matrix resulting from the system, and in the next chapter, we will show that this formulation is exactly equivalent to one derived from the perspective of equivalent circuits of resistors connected to nodes that constitute the grid. This later formulation perhaps makes intuitive sense to an electrical engineer. Henceforth, we then refer to the resistor network discretization as the finite difference grid.

For the 3D case, a 7-point stencil can be created by adding two additional nodes (one above and another below the node in question). As expected, this tells us that the resulting sparse matrix will be banded, with at most 7 non-zero entries per row [53].

Application of FDM to electronic thermal modelling

A 3D FDM discretization of a PCB hosting an LED driver is developed in [111]. The properties of the grid nodes are obtained from detailed analysis of the physical sizes of various portions of the board. What the paper boasts in mathematical rigour, it lacks in general applicability of the method. The FDM thermal model is validated for a simple and small 2-layer PCB (40×30 mm with 105 μm thick copper layers) without modelling key thermally relevant features of PCBs *i.e.* vias, holes, heat sinks and complicated copper patterns. They also use a simple uniform volumetric heat source model that is not representative of the complicated heat dissipation of components on PCBs (see section 4.3). While the method performed impressively fast (obtaining a satisfactory steady state temperature profile in about 190 seconds, compared to the 14 minutes that were required to produce the same results using ANSYS), the performance of the model is likely to degrade for larger boards and if more than two layers are considered.

A simple 2D finite difference method of performing thermal analysis for a PCB in an aerospace application is proposed in [91]. The authors model a PCB using 14

nodes and approximate the thermal conductivities between the paths connecting the nodes based on analysis of the copper coverage of the board. The authors then modelled three scenarios - conduction only, conduction and convection combined and conduction with convection and a heat sink attached. The authors showed that the method can predict temperatures within the safe operating ranges for all the components considered. However, they neither verify the results obtained from the analysis, nor comment on the accuracy of the method, which is perhaps hampered by the coarse grid chosen, and the application of lumped thermal conductivities.

An attempt is made to use a transient FDM scheme to model the heat dissipation characteristics of a PCB with SMDs in [32]. The authors base their model on a number of questionable assumptions, for example asserting that the thermal energy entering a PCB is evenly distributed within it.

2.5 Comparison between FDM and FEM for PCB thermal modelling

While both FEM and FDM are common approaches for thermal modelling of PCBs, there is no reliable literature contrasting the two methods for this application. For example, [97] claims that FDM is better than FEM without explaining why.

The FEM involves the discretization of a continuous domain into a set of discrete elements that can assume a number of arbitrary shapes, thus enabling the method to approximate complex geometries and boundary conditions. FEM can use triangular, quadrilateral, or higher-order elements. Moreover, FEM can provide accurate solutions with relatively coarse meshes through the use of adaptive mesh refinement techniques. In contrast, the FDM discretization is usually limited to rectilinear grids. This can make it difficult to use FDM to accurately approximate complex geometries such as circular shapes or slanted boundaries between materials in a multi-material domain such as a PCB. For example boundaries between copper and FR4 will often occur diagonal to the edges of the PCB. However, this is not a major

concern because the thermal problem can be considered 'blurry' and a stair-case type approximation of a boundary does not seem to affect the results much.

One advantage of the FDM is that it is easier to implement than FEM. FDM uses a fixed grid of points to approximate derivatives by finite differences. FEM, on the other hand, uses a mesh of elements with shape functions to approximate solutions by weighted averages. FEM requires more sophisticated mathematics for its formulation and meshing is computationally expensive.

Although PCBs are complex at a system level, they are simple enough when represented using pixels to be modelled satisfactorily using FDM. In fact, the vertical plane of a PCB tends to be well-defined into a natural FDM discretization. They are, after all, planar structures, constructed in a predictable pattern, one layer after another, where the discretised geometry may be approximated accurately layer by layer.

While the FEM is more general, flexible and suited for a wider range of PDEs, the FDM is well suited for thermal modelling planar structures like PCBs, and its simplicity of implementation is a major attraction for the method.

2.6 Thermal models for electronic components and PCBs

In addition to analytical and numerical solutions to the heat equation, research effort has gone into modelling the thermal behaviour of electronics by attempting to lump thermal paths and representing them using thermal resistors.

2.6.1 Foster and Cauer thermal resistor and capacitor-based networks

Cauer and Foster models rely on drawing an analogy between heat flow by conduction (Fourier's law) and current flow by conduction (Ohm's law). In this framework, the equation of heat conduction is often written as:

$$P_\theta = \frac{\Delta T}{R_\theta} \tag{2.23}$$

where P_θ is the thermal power, ΔT is the temperature drop between two points and R_θ is the thermal resistance. This is analogous to Ohms law, ($I = \frac{\Delta V}{R}$) where there is a direct equivalence between heat rate and current (P_θ and I); thermal resistance and electrical resistance (R_θ and R); as well as temperature change and the voltage change (ΔT and ΔV).

From this observation, thermal circuits that assign a lumped thermal resistance (R_θ) and thermal capacitance (C_θ) in addition to an associated temperature drop between two distinct points have been developed (in very much the same way basic electric circuits are defined by specifying voltages at chosen nodes, and impedances/resistances to link those nodes). Two flavors of RC thermal circuits, the Cauer and Foster networks, have been widely studied. They are currently used for simple thermal modeling and are usually included in datasheets for electronic components [108, 93, 72].

If a Cauer model [28] is used, then the thermal resistance and the thermal capacitance depend on both the geometry and material composition of the thermal path. It is complicated, if not impossible, to extract these parameters for a complex PCB whose layout inherently features irregular geometries and is often unavailable post-manufacture. Although R_θ and C_θ are distributed components, they are often modelled as lumped parameters in order to simplify the problem and reduce the computational complexity of the thermal problem. In other words, the models reduce the 3D thermal problem to a 2D one.

Although this is a reasonable simplification to reduce the computational complexity, Davidson et al [20] found that a higher-order Cauer model (i.e. one with many R_θ C_θ elements) is required to accurately model a structure in an experimental study of the transfer impedance between the active region of a TO220-packaged

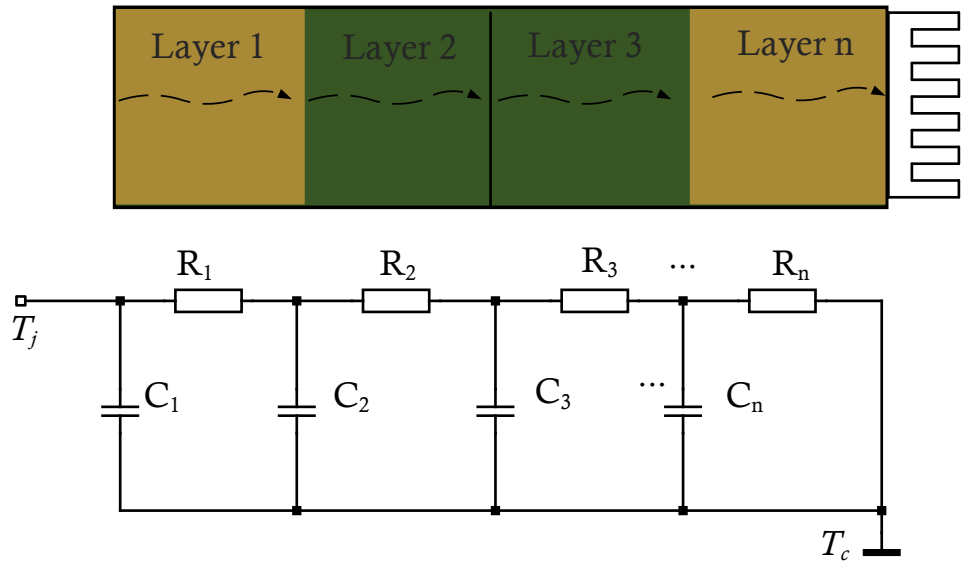


Figure 2.6: Cauer Network

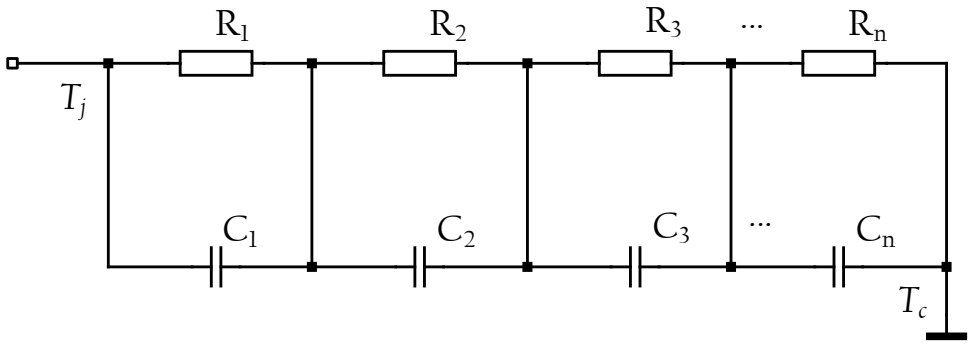


Figure 2.7: Foster Network

transistor and a thermocouple glued to its tab. Contrary to the common practice of representing a single path or material using a single R_θ C_θ element, at least four are required to guarantee the basic accuracy of the model.

There are a few important differences between the Cauer and Foster network representations of heat conduction paths. In the Cauer model, as shown in Fig 2.6 all the devices share a common thermal ground enforcing that the heat flow from node T_j to T_c has a delay corresponding to the time constants given by the various R_θ C_θ products in the circuit. This closely resembles the real nature of

heat transfer: temperature rises are not instantaneous. In this model, the physical contact between the layers which transfer heat from one to the next in a sequential manner is captured. The nodes in the Cauer network have physical meaning and are conveniently placed at specific points in the device stack up such that each thermal resistance only represents a portion of the domain made up of only one material. Consequently, the Cauer network models the internal temperatures at the layer boundaries, despite failing to accurately capture thermal spreading effects.

In the Foster model (Fig 2.7), the thermal capacitances no longer share a common ground, and therefore this arrangement models near instantaneous heat transfer from the node T_j to T_c, a significant departure from reality. The thermal impedance used in the Foster network is obtained from fitted parameters of the cooling curve and is given by:

$$Z_\theta = \sum_{n=1}^{n} R_i(1 - e^{-t/(R_i C_i)}) \tag{2.24}$$

The nodes in the Foster network do not have physical meaning (they do not correspond to geometrical locations of the device in question), and can be used to predict the temperature at any desired point on the thermal path using suitable approximations such as curve fitting [12], [28]. A drawback of this feature is that, unlike Cauer networks, Foster networks for different devices do not lend themselves to being superposed in order to create a larger thermal system model from smaller device models.

A Cauer network, however, is an ideal candidate for such an application. For example, consider a PCB-based power electronics converter consisting of two semiconductor switches, an inductor, a capacitor, and other peripherals. A Cauer thermal network may be derived for each of the components - an easier task than deriving one for the entire PCB system. These discrete RC thermal networks may be combined to form a larger network that accounts for cross-coupling effects

among the devices on the PCB, allowing for a simple compact thermal model to be realised. In fact, as will be studied in chapter 5, this insight will be used to model heat sources more accurately.

The problem with this approach is that it becomes intractable to solve for the large number of parameters in the thermal impedance matrix that need to be extracted which increases rapidly with the number of heat sources (components) on the PCB, thus increasing the computational cost and effort.

Lastly, because equivalent circuit models for heat conduction do not take into account the convective heat flux (convection is normally neglected as a dominant heat transfer mechanism favoring heat flow vertically into the structure to be rejected from the heat sink connected on the opposite side), they are not suitable for describing the thermal problem for PCBs where convection plays an important role.

2.6.2 Lumped models

This class of models is based on attempts to compute the so-called effective thermal conductivity of the entire PCB structure or portions of it. In practice, the thermal conductivity of multi-material domains such as a PCB is difficult to obtain. Significant research effort has sought ways to represent the layers of a PCB as a fictitious material with known anisotropic thermal conductivity. Simple methods of calculating the effective thermal conductivity of the PCB have been proposed since the 1990s. Several authors proposed modeling the PCB as a lumped object with an in-plane effective thermal conductivity (applied parallel to the board) and a normal thermal conductivity (modelled perpendicular to the board) [3] [59] [100]. While this is a simple and computationally low-cost approach, it produces massive errors in predicting the thermal profile of PCBs because it computes the thermal conductivity based on the ratios of copper to FR4, but fails to account for their orientation and location on the board. Approaches that lump the effective thermal

conductivity for each layer in the PCB stack up (such as the model proposed by [100]), also perform poorly for the same reason.

The authors of [25] proposed a 'trace PCB model' based on a conduction-only heat transfer approach for determining the spatial variation of temperature on a PCB. The researchers used `.brd` and `Gerber` files obtained from a PCB layout editor to construct the PCB geometry and to compute the thermal conductivity throughout the PCB from analysis of the local trace and via densities on each layer. After importing the geometry into their thermal solver, the authors mapped the data onto a 2D grid which was then segmented into cells based on the Cartesian coordinate system and used to calculate directional metal and dielectric fractions based on a fifth-order polynomial relation. The polynomial had an asymptote to the dielectric or metal thermal conductivities as the metal fraction reached zero or unity, respectively. The curve-fitted polynomial is not provided, and thus the method does not generalize and is also hard to reproduce.

Research is also available showing the impact of the size of the heat source relative to the entire board on the accuracy of effective thermal conductivity [77] [99]. The authors show that the errors produced are much greater for smaller sources and larger sources may be satisfactorily modelled using lumped properties.

2.6.3 Image-based thermal models

Using pixelated images to drive the thermal modelling of PCBs has received some research interest and can broadly be categorized into direct image-derived thermal modelling and inverse image-derived thermal modelling, where in the former, layer images are used to create thermal models/discretizations while in the latter approach, thermal images of temperature profiles are instead used to derive the heat dissipation causing them. The former are the subject of this thesis.

One approach proposed in [7] used image processing to compute a tessellated thermal conductivity map of a PCB layer using black and white images from

electronic design automation (EDA) tools. The thermal conductivity map was then used to define the material properties of the PCB geometry, and the problem was solved using the finite volume method. However, the method does not generalise well to complex PCBs, and the authors make an incorrect assumption that all pixels in a row represent thermal resistances in series and that all rows of pixels are thermally in parallel.

In [69] the authors used images of the via and conducting layers of a PCB to build a 2D mesh of triangular elements for which equivalent resistor values were computed depending on how the triangular elements overlapped the mathematical descriptions of the traces generated from an EDA tool. The 2D meshes were then extruded into 3D geometries suitable for thermal analysis using finite element analysis. However, the authors do not validate the model, and neither the accuracy nor the computational cost is reported.

Yabin *et al* proposed a coupled Fourier series analytical and finite volume method to model the thermal behaviour of PCBs [121]. The finite volume mesh was generated using pixelated images and then altered using a multigrid approach to reduce the element count. Unfortunately, although the results were validated for a simple one-layer PCB with a single track on it, they still failed to agree with COMSOL to a satisfactory degree.

The authors in [123] built a 2D PCB layout analysis that considered temperature-dependent Joule heating in traces as well as thermal conduction in traces and vias. They analysed the board by loading 2D layout bitmaps of each metal layer into a custom program, and meshed heat sources and traces using square elements. The authors were able to model the behaviour of standalone traces that were not integrated into a functional PCB design and conceded that the method was not implemented efficiently.

Igor and colleagues, take a separate approach: they use post-manufacture thermal images of the surfaces of a PCB to predict the power density of the board [53]. This

approach to thermal modelling, while considered the inverse of the goals of this thesis, is still an important approach to using image data for PCB thermal modelling, also using finite differences. In fact, other authors have attempted to use image processing of PCBs in this way. For example, the authors of [48] constructed a finite difference-based thermal model for a PCB-based flyback converter based on analysis of the surface temperature distribution of the board by solving the inverse heat problem. However, the authors did not report on the accuracy of their approach.

2.7 Conclusion and outlook

Prevailing literature reviewed in this chapter shows the promise of using images to drive thermal analysis. The limitations have been identified and the next chapters will build on the ideas developed in this background chapter to develop a steady state thermal model for PCBs based on rasterized images.

First Author	Year	Approach	Comments
G. Ellison [29] [30]	1992, 1996	Analytical 3D, separation of variables	Assumes unpatterned copper, thus cannot handle copper traces
G. Wan [112]	2007	Analytical, Green's functions	Only models discrete chips without integrating them with the PCB system
Y. Zhang [122]	2007	Coupled analytical-numerical	Simple PCB tracks considered not realistic
G. Wan [111]	2022	FDM discretization of PCB for LEDs	Good results but without a model for vias or heat sources
K.A. Pervaiz [91]	2005	FDM for PCB	Uses four nodes and lumps material properties in unrealistic ways
N. Evstatieva [32]	2021	Transient FDM	Assumes that thermal energy entering a PCB is evenly distributed in the board
M. B. Dogruoz [25]	2012	FVM with thermal properties calculated from analysis of Gerber and .brd files	-
B. Blackmore [7]	2009	FVM with thermal properties from tessellated thermal conductivity of the board	Uses image analysis to compute thermal conductivity of PCB
A. M. Lush [69]	2004	Image based 2D triangular mesh	No verification attempt
Y. Zhang [121]	2021	Image based multigrid mesh, solved by FVM	Numerical verification only performed for an unrealistic single track PCB
Y. Zhang [123]	2012	Square element mesh generation with semi-analytical solution	Good results obtained for simple PCB traces

Table 2.1: Summary of selected works on PCB thermal modelling

3

Development of a pixel-derived bare-PCB thermal model

In this chapter, we develop a steady-state bare-PCB thermal model derived from rasterized images of PCB layers. Using high-resolution pixelated images obtained from a PCB design editor, we develop a resistor network-based finite difference discretization that accurately models the PCB geometry for steady-state thermal analysis. We show that with appropriate pre-processing using common image processing techniques, the thermal behaviour of complex PCBs can be modelled to a high degree of accuracy at a low computational cost.

The techniques described here leverage the regularity of the 2D structure of pixelated images to generate a mesh that closely resembles the geometry of the board. We show that by representing each pixel with a node at its centre, connected to two pairs of orthogonally connected resistors in the 2D case, (or three pairs in the 3D case), termed 'half-pixel' resistors in this work, it is possible to create a fully representative model of the PCB geometry that naturally captures properties of all materials present in a layer.

We begin by showing a systematic method for generating large thermal resistor networks from a grid of pixels. We then show that by performing nodal analysis

at each node in the network, we can assemble a system of equations that is equivalent to the finite difference discretization of the heat equation presented in the previous chapter.

PDE discretizations often lead to systems of simultaneous equations, which can be arranged into the form $\mathbf{Ax} = \mathbf{b}$, and readily solved using established linear algebra techniques. We discuss the properties of the system of equations that result from our discretization and comment on how such a system can be solved in reasonable time on a modern desktop PC, with solution times fast enough for the approach to be useful inside an iterative PCB electrothermal co-design loop.

First, we introduce the bare PCB thermal model that forms the basis of this work. Secondly, general expressions for convective and conductive thermal resistances for an arbitrary cuboid structure are derived. Then the process of creating the conductance matrix for a single layer with a pixelated image as input is described. A description of a systematic method of combining an arbitrary number of such 2D FDM discretizations to construct the sparse system for a 3D discretization follows. Simple numerical examples are given in order to study the structure of the problem.

3.1 Introduction to the pixel-derived bare-PCB thermal model

In this work, a bare-PCB model refers to a representation of a PCB that contains no attached components on either side and includes only features that are relevant to the thermal behaviour of the board. Such a model can be rapidly constructed from rasterized images of the various layers of the board that are arranged in a systematic manner. The thermally relevant features that we study are thermal vias, holes (filled or empty), pads, copper traces, and dielectric layers.

The model is derived from pixelated images of the conductive layers in a PCB stack-up extracted from a PCB design editor. A pixel is the smallest discrete unit of a pixelated image and such an image is composed of tens of thousands to

millions of pixels arranged into a grid. The colour of each individual pixel identifies the material it represents. Additionally, by assigning a specific number of pixels per unit length, for example using a measure such as dots-per-inch (dpi), we can also store information about the dimensions of the features on the layer. From this perspective, a pixelated image of the layer of a PCB represents a natural 2D discretization of that layer, complete with details about the dimensions, layout, and identity of the features. In fact, as part of a thermal model, each pixel is a small thermal path that we can represent using thermal resistors.

Because a pixelated layer image represents a 3D physical domain, such as copper traces, it is necessary to extrude the layer into the third dimension. Following an extrusion, a homogeneous and isotropic solid results composed of small connected cuboids whose thickness corresponds to the thickness of that layer (typically 35 μm for 1 oz PCBs). We place a node at the centre of the pixel-derived cuboid and connect three pairs of orthogonal resistances connected to it. Fig. 3.6 shows an example of such a cuboid and its associated thermal resistances. Details about how to construct the thermal model from this information and also compute the half-pixel resistances follow later. Fig 3.4 shows a generic black and red pixel array with $n_x \times n_y$ pixels and its equivalent representation as a network of resistors is shown in Fig. 3.5.

The exclusion of components from the bare-PCB model is reasonable because these can be attached as external resistor networks to the relevant pixels without altering the internal thermal behaviour of the bare-PCB model. In fact, as will be discussed later, this is the approach we take to model heat sinks and heat sources within the pixelated image framework. Another reason to exclude components is that they vary in their shapes, sizes, and material content in hard-to-predict ways. They do not fit the planar and regular grid-like structure inherent to PCBs, on which our rapid finite difference discretization relies. To the benefit of PCB electrical and thermal designers, device manufacturers go to great lengths to model the thermal behaviour of individual components in great detail and provide relevant

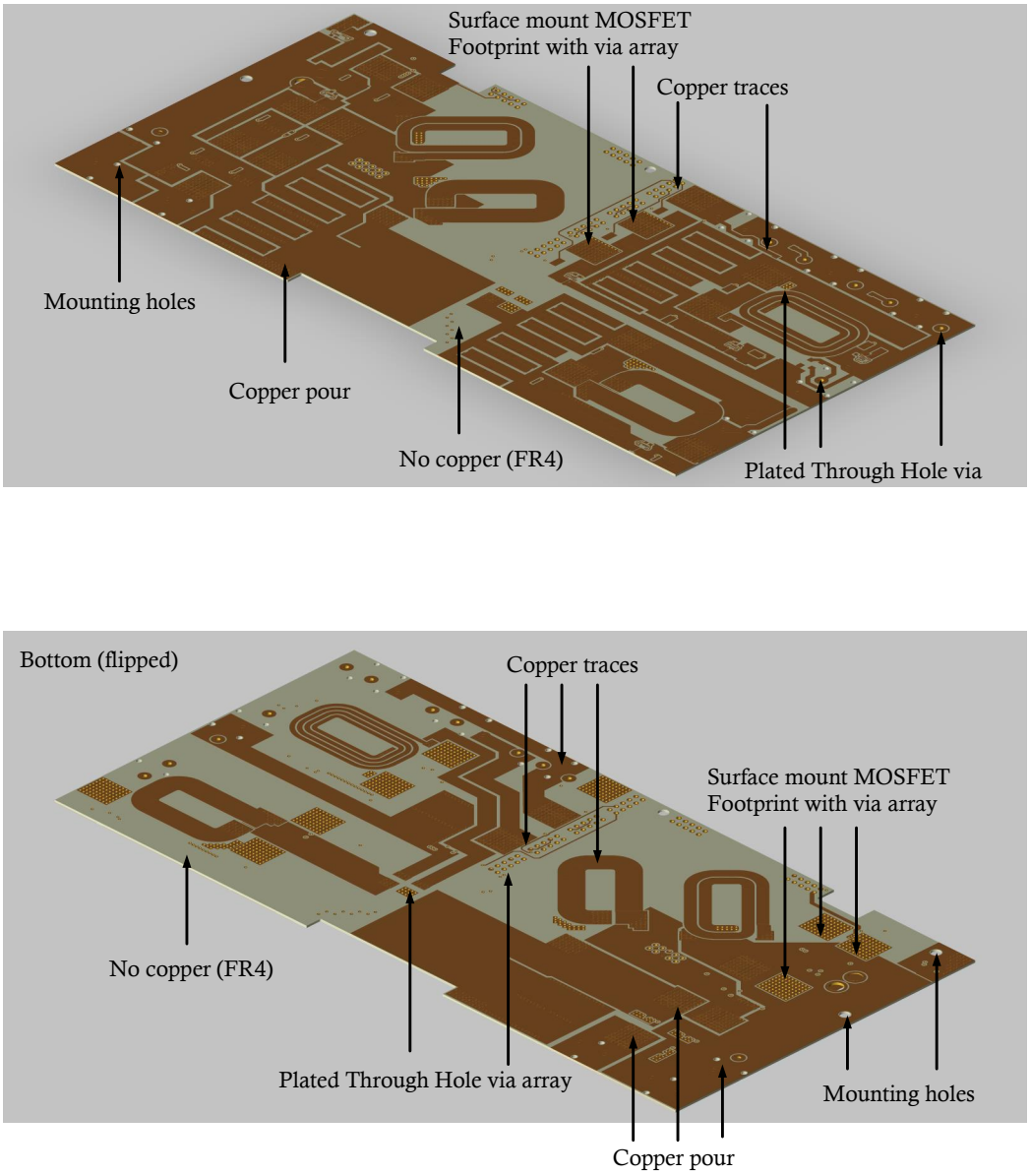


Figure 3.1: Top and bottom renderings of a bare-PCB model. Only the traces, pads, copper pours and vias are visible. Images: Courtesy of Prof D.J. Rogers and the RELCON Project

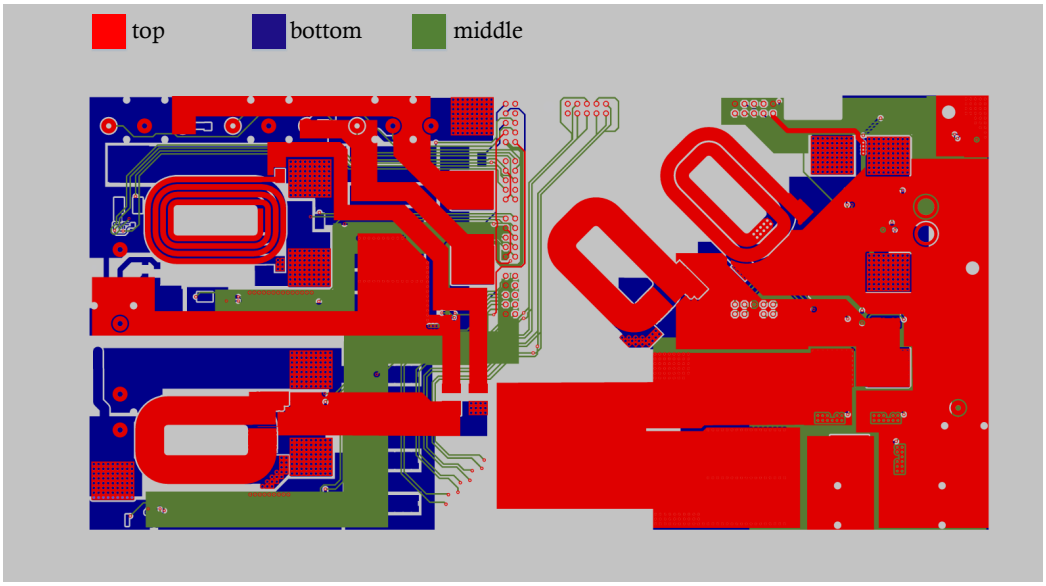


Figure 3.2: Four PCB layers of a multiport converter PCB shown with the dielectric layer hidden. Notice the prevalence of large blocks of copper pours and the low trace count. The layers are: red - top, green - middle, blue - bottom. Images: Courtesy of Prof D.J. Rogers and the RELCON Project



Figure 3.3: Rasterized representation of a portion of a PCB layer using pixels

parameters such as junction-to-case thermal resistances for MOSFETs, and bulk thermal resistances under various ambient conditions for heat sinks, as noted in [102]. Incorporating such discrete device thermal models in PCB thermal modellings is ripe research in its own right that has been studied by several research works [28] [27] [95].

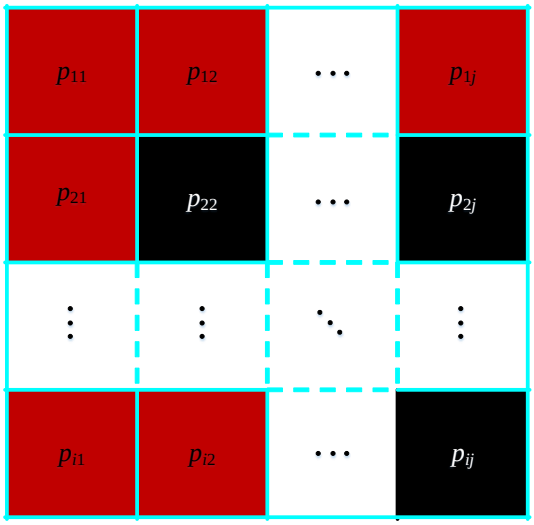


Figure 3.4: Canonical pixel array with $n_x \times n_y$ pixels

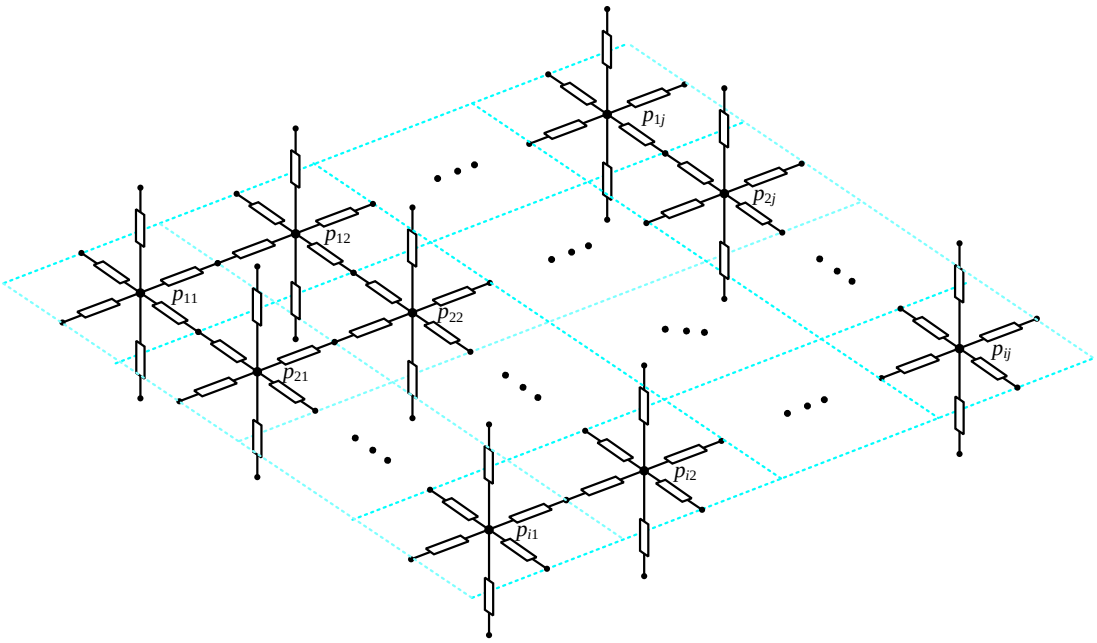


Figure 3.5: Resistor network for $n_x \times n_y$ pixel array

3.2 From Pixelated Image to Finite Difference Grid

3.2.1 Thermal resistances of a cuboid structure

Suppose a single pixel of size $l_x \times l_y$ is extended vertically by some length l_z. A simple $l_x \times l_y \times l_z$ cuboid shown in Fig. 3.6 results. We assume this structure forms part of a larger three-dimensional rectilinear solid composed of a combination of a large number of similar discrete cuboids. Additionally, we assume it provides a segment of a thermal path through which heat can freely flow in the x, y, or z Cartesian directions. We also define a node, $p_{i,j,k}$, at the center of the cuboid.

It is now possible to assign three pairs of mutually orthogonal 'half-pixel' thermal resistances, that constitute the thermal paths (Fig 3.6). The values of these resistances can be computed from the material properties and dimensions of the structure. Assuming that the dominant heat transfer processes within the cuboid are conduction and convection, it is possible to derive the basic expressions for the thermal resistances associated with the heat transfer processes.

Conductive thermal resistance

In general, thermal resistance, R_θ, is the ratio of a temperature change, ΔT between any two points and the heat flux, q, flowing between them and can be written as:

$$R_\theta = \frac{\Delta T}{q} \tag{3.1}$$

From Fourier's law of heat conduction, we can express the denominator in (3.1) as:

$$q = \kappa \frac{A\Delta T}{l} \tag{3.2}$$

where A is the cross-sectional area normal to the direction of heat flow, l is the length of the thermal path and κ is the thermal conductivity of the solid. This leads to the following general expression for conductive thermal resistance, $R_{\theta c}$:

$$R_{\theta c} = \frac{\Delta T}{\kappa \frac{A\Delta T}{l}} = \frac{l}{\kappa A} \tag{3.3}$$

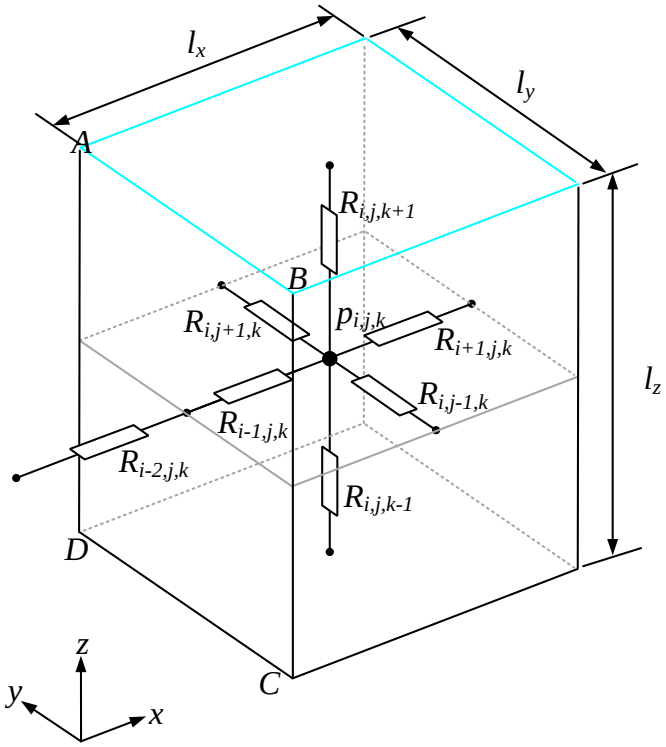


Figure 3.6: A cuboid structure and its associated thermal resistances. $R_{i,j,k+1}$ and $R_{i,j,k-1}$ are the half-pixel thermal resistances for the top and bottom halves of the structure respectively. Similarly, $R_{i+1,j,k}$ and $R_{i-1,j,k}$ and $R_{i,j-1,k}$ and $R_{i,j-1,k}$ are the half-pixel thermal resistances in the positive or negative i or j halves of the structure respectively. $R_{i-2,j,k}$ is an example of the convective thermal resistance associated with the face $ABCD$.

Convective thermal resistance

Convective thermal resistance applies to heat transfer that originates from the surface of a body to the external environment. This process occurs from the faces of the cuboid that are in contact with the ambient environment, where a coefficient of convection h is assumed. Using the definition of thermal resistance (3.1) and Newton's law of cooling ($q = hA\Delta T$), we can express the convective thermal resistance, $R_{\theta h}$ as:

$$R_{\theta h} = \frac{\Delta T}{hA\Delta T} = \frac{1}{hA} \tag{3.4}$$

Assuming the cuboid is composed of a homogeneous material with thermal conductivity k, and possibly subjected to a convective heat flux h on one or more

faces, we can express thermal resistances on the conductive heat paths as:

$$R_{i-1,j,k} = R_{i+1,j,k} = \frac{\frac{l_x}{2}}{l_y l_z\ \kappa}, \quad R_{i,j-1,k} = R_{i,j+1,k} = \frac{\frac{l_y}{2}}{l_x l_z \kappa}, \quad R_{i,j,k-1} = R_{i,j,k+1} = \frac{\frac{l_z}{2}}{l_x l_y \kappa} \tag{3.5}$$

The surface-to-ambient thermal resistances on convective heat paths are expressed as:

$$R_{i-2,j,k} = R_{i+2,j,k} = \frac{1}{h\ l_y l_z}, \quad R_{i,j-2,k} = \frac{1}{h\ l_x l_z}, \quad R_{i,j,k-2} = R_{i,j,k+2} = \frac{1}{h l_x l_y} \tag{3.6}$$

In (3.6), the indices (i, j, k) are incremented or decremented by 2 to indicate that the reference point for computing the convective thermal resistance is at the center of the respective face of the cuboid, and not at $p_{i,j,k}$.

3.2.2 2D FDM discretization for a grid of pixels

We now build on the resistor calculations for a single extruded pixel presented in the preceding section to construct a bare-PCB model built from pixelated images of the layers of the PCB.

Consider the general pixel grid shown in Fig. 3.4 with size $n_x \times n_y$. We identify any pixel-centered node by $p_{i,j}$ where the indices $i = 1, 2, \ldots, n_x$, and $j = 1, 2, \ldots, n_y$.

By assigning infinite thermal resistance to all the resistors in the z dimension of a single layer image (Fig. 3.5), there is no thermal path along the z axis. We also assume that the edges of the image are at a commonly known temperature. Therefore, we can consider the 2D resistor array in the xy plane only. A simpler 2D discretization of the image results and is shown in Fig 3.7. Notice that the series half-pixel resistances connecting two adjacent pixels have been combined into a single series equivalent resistance.

To simplify the analysis, we focus on a single node $(p_{i,j})$, centered on one pixel, as shown in Fig. 3.8. We assume that thermal power, $P_{i,j}$ is being injected into $p_{i,j}$.

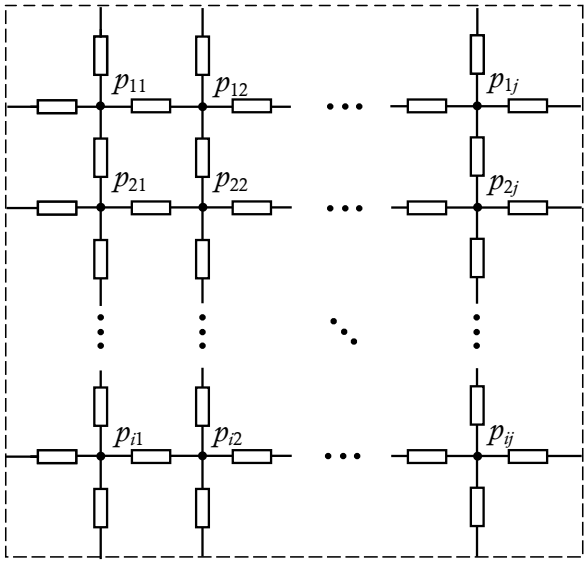


Figure 3.7: General 2D resistor network representing a single layer image

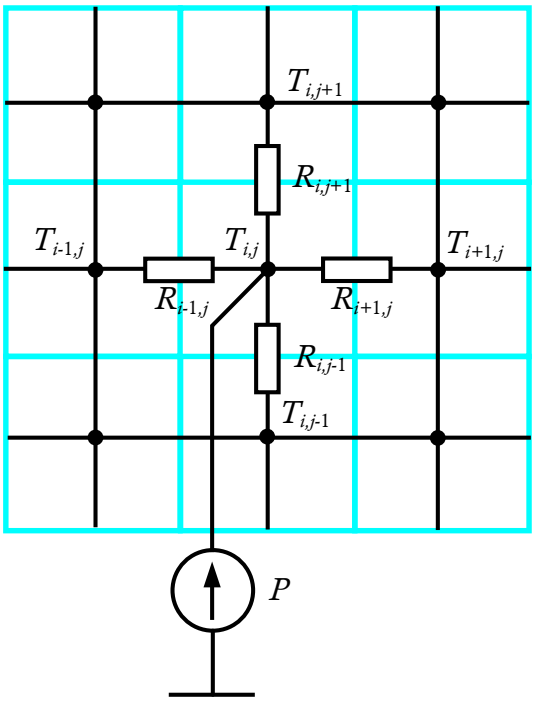


Figure 3.8: Representative node for a randomly selected pixel i, j. The surrounding pixels are superposed above the node in light blue, showing a 3×3 grid of pixels, and illustrating the idea that series half-pixel resistances have been combined.

Treating this thermal system as analogous to a simple electrical one, we use Kirchhoff's voltage law, in this case with temperature equivalent to the voltage, to write the temperature nodal equation as:

$$(T_{i,j} - T_{i-1,j})\frac{1}{R_{i-1,j}} + (T_{i,j} - T_{i+1,j})\frac{1}{R_{i+1,j}} + (T_{i,j} - T_{i,j+1})\frac{1}{R_{i,j+1}} + (T_{i,j} - T_{i,j-1})\frac{1}{R_{i,j-1}} = P_{i,j} \quad (3.7)$$

The resistance values used in (3.7) are computed using (3.5), with the index k set to zero. This formulation of the nodal temperature equation based on the resistor network equivalent circuit will be shown to be mathematically equivalent to the discretization introduced using the 5-point stencil in Chapter 2.

If a current or in this case, thermal power, is neither sourced nor sunk from node $p_{i,j}$, then the right-hand side of (3.7) is assigned to zero. Otherwise, it takes the value of the power with the appropriate sign.

Applying the above analysis to every single node yields a system of simultaneous equations that relate the unknown temperatures to the thermal conductances and powers in the system. Such a system can be arranged into a matrix equation of the form $\mathbf{LT} = \mathbf{q}$ where $\mathbf{L}$ is the layer thermal conductance matrix associated with a PCB layer represented by the pixelated image, $\mathbf{T}$ is the vector of unknown nodal temperatures, and $\mathbf{q}$ is the vector of powers sourced or sunk at each node.

$\mathbf{L}$ is a sparse matrix of size $n_x n_y \times n_x n_y$, with the number of nonzeros entries given by:

$$n_{\text{nz}} = n_x n_y + 2n_x (n_y - 1) + 2n_y (n_x - 1) \quad (3.8)$$

The shape of $\mathbf{L}$ (and thus the solution time) strongly depends on the ordering of the nodal equations or rows in the matrix. The characteristic properties of this matrix are:

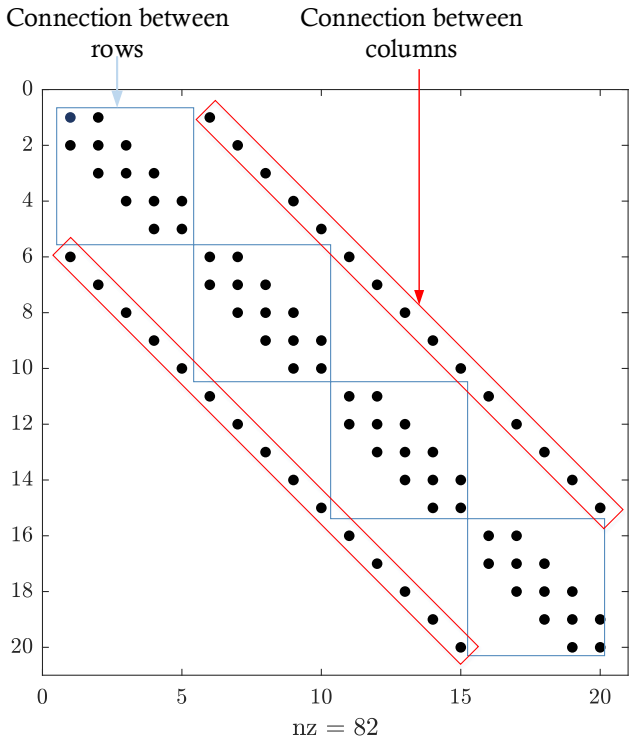


Figure 3.9: Sparsity pattern of the conductance matrix generated from a 4×5 pixel grid

1. it is banded and contains five diagonal bands

2. it is symmetric about the main diagonal

3. it is square and positive semi-definite

Fig. 3.9 shows the sparsity pattern of $\mathbf{L}$ resulting from a resistor network derived from a 5×4 grid of pixels. As can be seen in Fig. 3.9, there are 4 subgroups of tridiagonal sub-matrices (each with a blue square around it in the figure) which correspond to the number of pixel-centered nodes in the y dimension of the original image. The number of nonzero entries in the main diagonal of each inner tridiagonal matrix (in this case 5) corresponds to the number of pixel-centered nodes in the x direction.

A different way of interpreting the structure of the conductance matrix is to consider that each node in the network is represented by a row in $\mathbf{L}$. The number of

nonzero entries in any row corresponds to the number of nodes directly connected to the node $p_{i,j}$, inclusive, and the column occupied by the nonzero entry corresponds to the node that shares a thermal path with the row in question. In this framework, the pixel-centered nodes in the original image are numbered from left to right and in a top-down routine in the range 1 to $n_x n_y$.

For example, from the sparsity pattern shown in Fig. 3.9, one can quickly deduce that node 1, corresponding to the first row of pixels, is connected to nodes 2 and 6 (the columns with nonzeros in the sparse matrix), and so forth.

3.2.3 Generalization to 3D system

A 3-D rectilinear geometry composed of n_z layers is discretized by assembling individual conductance matrices representing each layer into a larger system. Therefore, a good starting point for a domain with k layers is to compute k $\mathbf{L}$ matrices for all the layers by following the steps in the preceding section.

Consider the 3D bare-PCB model represented by the $n_x \times n_y \times n_z$ network in Fig. 3.10. In this setup, we have defined $k = 1, 2, \ldots, n_z$. For the 3D problem, we now have to take into account the inter-layer thermal resistor matrices between adjacent layers of the pcb, aligned in the z direction.

We begin the process of constructing the overall combined conductance matrix $\mathbf{G}$ by placing each of the k layer matrices, $\mathbf{L}_k$ as diagonal entries, $\mathbf{G}_{k,k}$ of $\mathbf{G}$. This arrangement represents an unconnected system with k layers.

To complete the system, interconnection matrices that link $\mathbf{L}_k$ to the adjacent layer $\mathbf{L}_{k-1}$ below it and $\mathbf{L}_{k+1}$ above it if one exists are constructed. Each interconnection matrix denoted $\mathbf{M}_{k,k\pm1}$, is a diagonal matrix of the same size as $\mathbf{L}_k$ (i.e $n_x \times n_y$) whose entries are the transconductances corresponding to the conductive thermal resistances normal to the layer. $\mathbf{M}_{k,k\pm1}$ is filled using the same left-right/top-bottom numbering of nodes used for $\mathbf{L}_k$.

In general, this leads to a full conductance matrix with the form:

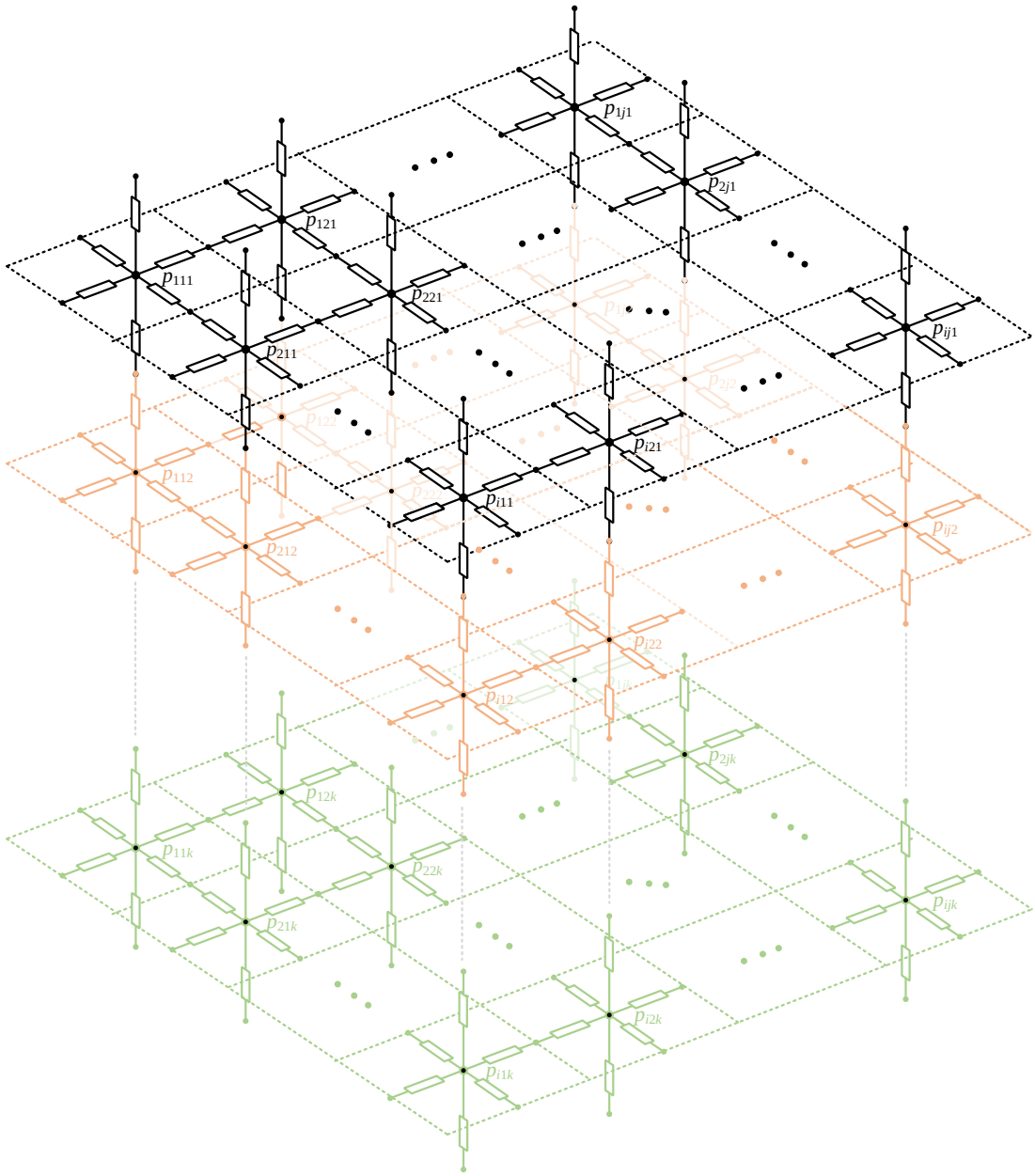


Figure 3.10: Generalized $n_x \times n_y \times n_z$ 3D resistor network showing the discretization of a bare PCB thermal model

$$\mathbf{G} = \begin{bmatrix} \mathbf{L}_1 + \mathbf{M}_{1,2} & -\mathbf{M}_{1,2} & & & & \\ -\mathbf{M}_{1,2} & \mathbf{L}_2 + \mathbf{M}_{1,2} + \mathbf{M}_{2,3} & -\mathbf{M}_{2,3} & & & \\ & -\mathbf{M}_{2,3} & \mathbf{L}_3 + \mathbf{M}_{2,3} + \mathbf{M}_{3,4} & -\mathbf{M}_{3,4} & & \\ & & \ddots & \ddots & \ddots & \\ & & & -\mathbf{M}_{k-1,k} & \mathbf{L}_k + \mathbf{M}_{k-1,k} \end{bmatrix} \tag{3.9}$$

The diagonal matrices in $\mathbf{G}$, $\mathbf{G}_{k,k}$, is given by:

$$\mathbf{G}_{k,k} = \begin{cases} \mathbf{L}_k + \mathbf{M}_{k,k+1} & k = 1 \\ \mathbf{L}_k + \mathbf{M}_{k-1,k} + \mathbf{M}_{k,k+1} & 2 \leq k < n_z - 1 \\ \mathbf{L}_k + \mathbf{M}_{k-1,k} & k = n_z \end{cases} \tag{3.10}$$

where $\mathbf{M}_{k,k-1}$ and $\mathbf{M}_{k,k+1}$ represent the conductances connecting layer $\mathbf{L}_k$ to layers $\mathbf{L}_{k-1}$ and $\mathbf{L}_{k+1}$ respectively. The top and bottom layers are each connected to one layer below or above respectively, and these connections are represented in the diagonal entries in $\mathbf{G}$ as $\mathbf{L}_k + \mathbf{M}_{k,k+1}$ for the top layer and $\mathbf{L}_k + \mathbf{M}_{k-1,k}$ for the bottom one.

The off-diagonal entries in $\mathbf{G}$ are given by:

$$\mathbf{G}_{k,k\pm1} = \mathbf{G}_{k\pm1,k} = -\mathbf{M}_{k,k\pm1} \tag{3.11}$$

The properties of $\mathbf{G}$ are that it is:

1. sparse which allows for efficient storage and fast solution methods by either direct or iterative methods.

2. banded with 7 bands as shown in Fig 3.11, confirming that the system is consistent with the general form of a 3D finite difference discretization derived using a 7-point stencil.

3. symmetric, square and positive definite which make the system a good candidate for a solution by efficient iterative methods that tend to be memory efficient.

4. built in a predictable repeating pattern, where the k^{th} 5 banded sub-matrix along the diagonal correspond to layer k. The number of pixels per layer can be extracted by observation as was done for $\mathbf{L}_k$.

3.2.4 Boundary conditions for the 3D system

For the PCB thermal problem, the only boundary condition we consider is heat loss by convection from the surface of the board, with a convection coefficient h. In this formulation, we only consider convective heat losses occurring from the top and bottom surfaces to be adequately representative and ignore convective losses from the sides of the PCB. This is a reasonable assumption because, due to the typical thinness of modern PCBs relative to their area in the xy plane, the approximation $A_{\text{PCB}} \approx A_{\text{top}} + A_{\text{bot}}$ holds.

This boundary condition is included by modifying $\mathbf{G}$, such that the entries $\mathbf{G}_{1,1}$ and $\mathbf{G}_{k,k}$, respectively become:

$$\mathbf{G}_{1,1} = \mathbf{L}_1 + \mathbf{M}_{1,2} + \mathbf{H}_{1,\infty} \tag{3.12}$$

and

$$\mathbf{G}_{k,k} = \mathbf{L}_k + \mathbf{M}_{k-1,k} + \mathbf{H}_{k,\infty} \tag{3.13}$$

where the matrices $\mathbf{H}_{1,\infty}$ and $\mathbf{H}_{k,\infty}$ contain the transconductances corresponding to the convective thermal resistances from the top and bottom surfaces to ambient respectively.

Fig. 3.11 shows an illustrative sparsity pattern obtained by extending the 5×4 pixel array presented before to include a third dimension with 3 layers. The different submatrix entries in the system are labeled, with the diagonal layer and interconnection matrices circled in blue, and the off-diagonal matrices $\mathbf{M}$ circled in green.

The resulting system is represented by:

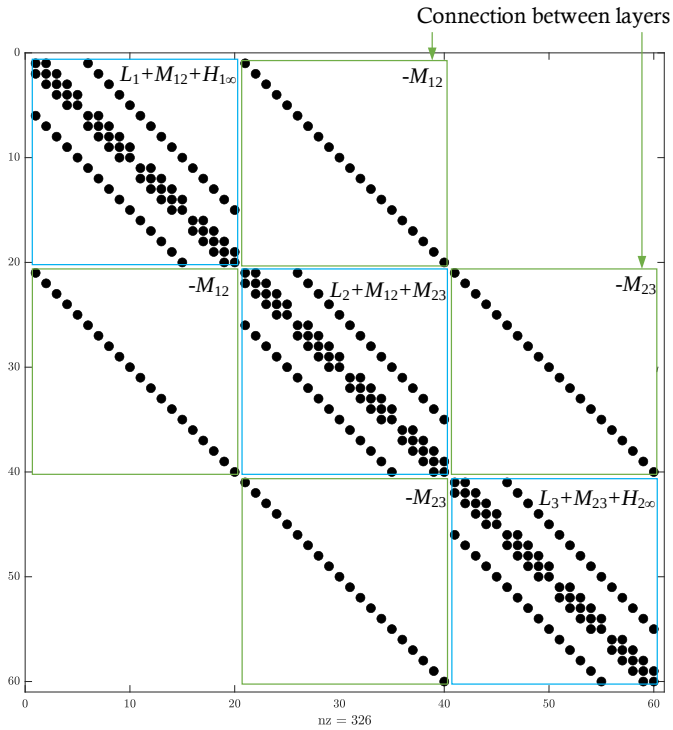


Figure 3.11: Spy plot for 3D discretization 4×5 images and 3 layers

$$\mathbf{GT} = \mathbf{q} \tag{3.14}$$

Note that $\mathbf{T}$ is assembled by following the left-right/top-down numbering scheme described previously. For the multilayer system, the nodes in the top layer are thus numbered $1, 2, \ldots, n_x n_y$, and the following layer $n_x n_y + 1, n_x n_y + 2, \ldots, 2n_x n_y$ and so forth. In general, the node numbers for the k^{th} layer are given by

$$(k-1)n_x n_y + 1, (k-1)n_x n_y + 2, \ldots, k n_x n_y$$

Therefore

$$\mathbf{T} = \begin{bmatrix} \mathbf{T}^{[1]} \\ \mathbf{T}^{[2]} \\ \vdots \\ \mathbf{T}^{[k]} \end{bmatrix} \quad \text{where} \quad \mathbf{T}^{[k]} = \begin{bmatrix} \mathbf{T}^{[1]} \\ \mathbf{T}^{[2]} \\ \vdots \\ \mathbf{T}^{[i]} \end{bmatrix} \quad \text{and} \quad \mathbf{T}^{[i]} = \begin{bmatrix} T_{i1} \\ T_{i2} \\ \vdots \\ T_{ij} \end{bmatrix} \tag{3.15}$$

The notation $\mathbf{T}^{[k]}$ is used to represent the vector of unknowns for layer k, $\mathbf{T}^{[i]}$ to represent the vector of unknowns for row i in the pixel grid and T_{ij} to represent the unknown at node (i, j) in the pixel-centered grid of nodes. The RHS value $\mathbf{q}$ is assembled in a similar fashion. $\mathbf{q}$ is largely sparse and all nodes associated with a heat source or boundary condition have nonzero entries. This is a commonly occurring linear system whose solution methods have been studied extensively and are well established.

3.3 Justification for using a linear model for PCB thermal analysis

The derivation of the thermal model so far has relied on an implicit assumption of linearity. In PCB thermal analysis, nonlinearity refers to the temperature-dependent variation of material properties and heat transfer characteristics. This concept is an important modelling consideration as it can significantly impact the accuracy of thermal models.

The choice between linear and non-linear models for PCB thermal analysis is primarily dictated by the temperature-dependent properties of the constituent materials. A non-linear model is required where the thermal resistance of a material changes with temperature meaningfully, such that if a constant value is used in the calculations, it would lead to errors.

Non-linear thermal models are typically implemented by defining thermal conductivity as a function of temperature, enabling the model to update the values as high temperature values are computed by the iterative solver. In the case of PCBs, a linear model proves to be sufficiently accurate for several compelling reasons:

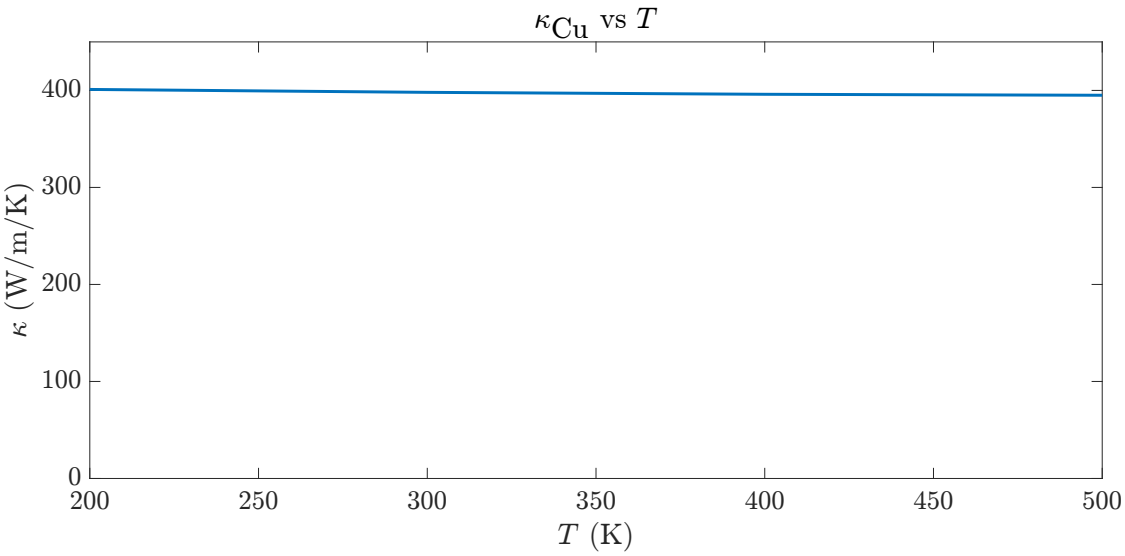


Figure 3.12: Temperature dependence of copper thermal conductivity (κ_{Cu}) in the temperature range of 200–500K. [46]

Stability of material properties

The thermal behavior of a PCB is predominantly influenced by the thermal properties of copper, the primary electrical and thermal conductor, and FR4, the glass-reinforced epoxy laminate substrate that serves as a dielectric. These thermal properties exhibit remarkable stability across the operational temperature range of most PCBs (typically -40°C to 150°C). Copper's thermal conductivity changes by less than 3% over the mentioned temperature range (Fig. 3.12) [46]. Consequently, a linear model employing constant thermal conductivity values is fundamentally justified by this inherent stability.

Potential sources of nonlinearity and how to handle them

Nonlinear effects can, however, arise from other sources. For instance, while copper's thermal conductivity remains relatively stable, its electrical resistance does increase with temperature, potentially affecting heat generation in transistors and other dissipating components.

When temperature-dependent nonlinearities are considered in power electronics PCB thermal modeling, they are typically focused on specific, critical areas, particularly modelling heat sources. Conduction losses in high-power semiconductor devices are often modeled nonlinearly, to account for the dependency of $R_{\mathrm{ds,on}}$ on temperature, which increases at high temperatures [14]. While these nonlinearities, which originate in the electrical domain, should generally be taken into account, they do not directly affect the accuracy of the thermal models presented in this work. This is because, as will be explained in Chapter 4, we rely on manufacturer data sheets for modeling supporting components that are external to the PCB substrate.

Our method accounts for nonlinear effects by incorporating manufacturer-provided thermal parameters of the switching devices and heat sinks, typically generated from comprehensive multiphysics thermal modeling. Using this approach, we are able to capture device-level nonlinearities external to the PCB without significantly increasing the complexity of the linear bare-PCB thermal model.

For example, transistor manufacturers typically quote the maximum power dissipation in data sheets, which corresponds to the worst case thermal behaviour a converter using these components can be expected to encounter. By using these values as parameter inputs in the models presented in this thesis, we are able to characterize the PCB using a linear model while indirectly considering nonlinearities originating in the electrical domain. This modeling methodology differs from fully linear approaches by integrating nonlinear device behavior through datasheet parameters, while maintaining the computational efficiency of a linear PCB model. By anchoring our boundary conditions on these worst-case device parameters, the PCB-level model remains linear while still capturing the true thermal response.

Focus of nonlinear thermal models in research

The dominant materials used in PCB construction exhibit linear behaviour in the temperature ranges that most power electronics converters are likely to experience. As a result, research effort on non-linear thermal models has tended not to focus on PCBs themselves, but rather on heat generating devices such as wide-bandgap transistors and IGBTs, (see [14], [41], [42] for examples), and power modules, (see [120]). These components typically operate at higher temperatures, generate heat, and are constructed from materials whose thermal conductivity exhibit marked temperature dependency. For example SiC, used in wide bandgap MOSFETS, has a thermal conductivity that decreases by almost 50 % in the same temperature range of Fig. 3.12.

3.4 Choosing n_z

The minimum number of layers required to discretize a PCB with n_{Cu} copper layers is

$$n_z^{\min} = 2n_{\mathrm{Cu}} - 1 \tag{3.16}$$

This represents n_{Cu} conducting layers each separated by one dielectric layer. However, in practice, n_z must be chosen to be greater than the limit (3.16). As will be shown in subsequent sections, it is often required to discretize the FR4 layer using multiple divisions whose length is shorter the closer they are to the boundary between copper and the dielectric. This increases the resolution of the mesh in regions close to the conducting layers where large thermal gradients are typical and require finer meshes to capture accurately.

In this section, we propose three vertical discretization schemes for the PCB that determine the number and distribution of the FR4 segments. Due to the typical thinness of copper layers, we do not vertically segment the layer, but instead, represent the vertical copper cell with two half-pixel resistances. We can

now consider the spacing of the FR4 layers by either using *linear-spacing*, or two opposite versions of *symmetric-spacing*.

$$\text{Rule 1 (linear spacing):} \quad t_{\text{FR4},i,j} = t_{\text{FR4},i,j\pm1} \tag{3.17}$$

$$\text{Rule 2 (growing symmetric spacing):} \quad t_{\text{FR4},i,j} \geq t_{\text{FR4},i,j+1} \tag{3.18}$$

$$\text{Rule 3 (decaying symmetric spacing):} \quad t_{\text{FR4},i,j} \leq t_{\text{FR4},i,j+1} \tag{3.19}$$

where $t_{\text{FR4},i,j}$ is the thickness of the FR4 segment at the centre of the sublayer. The subscript i refers to the FR4 enclosed by two copper layers on either side and and j refers to a sublayer segment of the former. Using uniform linear spacing to segment the FR4 sublayer means that the minimum number of layers in the z direction is given by (3.16). If the thickness of the FR4 sublayer is $t_{\text{FR4},i,j}$, then the thickness of each division within the FR4 sublayer given by Rule 1.

Symmetric spacing can be implemented either as 'decaying', such that the middle FR4 segment is the largest and every successive segment is less than the previous one as we move away from the center of the sublayer (Rule 3). This means that the grid is finest directly underneath or above the copper layer. An alternative approach is to assign the smallest segment at the center and then use Rule 2 to assign the lengths of the FR4 segments away from it.

The three segmentation methods are shown diagrammatically in Fig. 3.13 and an analysis of the impact of these segmentation choices on the performance of the thermal model is given in section 3.10.3.

3.5 Equivalence between resistor network and finite difference grid formulation

It can be shown that the linear system formulated by approximating the steady-state heat conduction problem with central difference Taylor series approximations, as was done for the 2D case in Chapter 2, is equivalent to the discretization obtained using resistor networks followed in this chapter.

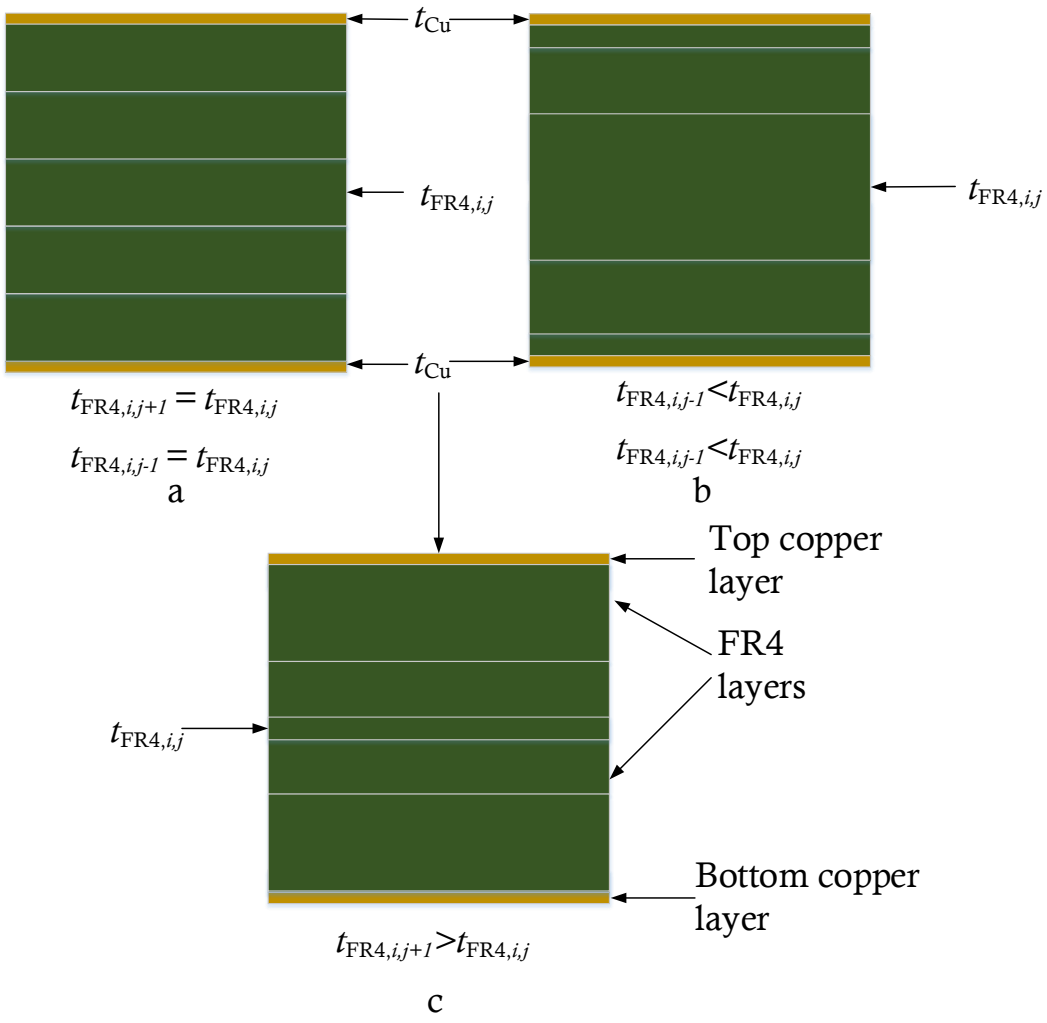


Figure 3.13: Different vertical discretization schemes in the PCB vertical plane. 5 divisions are shown for the FR4 layer placed between two copper layers. The schemes are (a) Rule 1: linear spacing. (b) Rule 2: growing symmetric spacing. (c)Rule 3: decaying symmetric spacing.

To illustrate this process, the nodal temperature equation (2.17) can be rewritten as:

$$T_{i-1,j} - T_{i+1,j} - T_{i,j-1} - T_{i,j+1} + 4T_{ij} = P_{i,j}\frac{A_{i,j}}{V_{i,j}\kappa} \tag{3.20}$$

By replacing the RHS value, $Q_{i,j}$, which represents volumetric power dissipation by $\frac{P_{i,j}}{V_{i,j}}$ [W/m^3] and multiplying both sides by the w^2 term which is equivalent

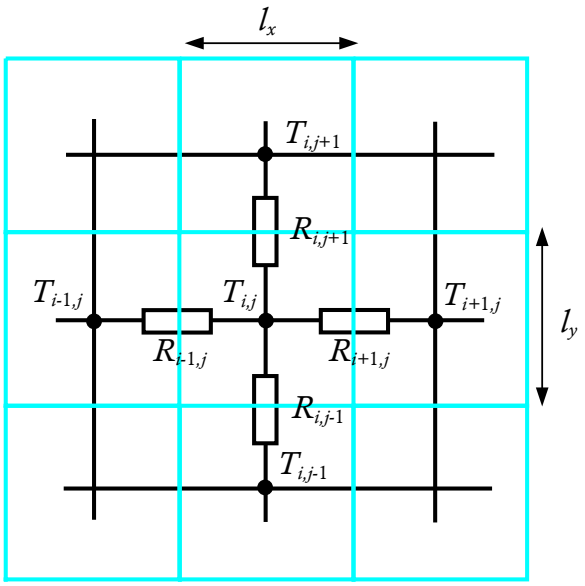


Figure 3.14: Node (i, j) showing that the neighboring temperatures are referred to the edges of the pixel instead of the centers of the neighboring pixels

to an area $A_{i,j}$. (3.20) simplifies to:

$$T_{i-1,j} - T_{i+1,j} - T_{i,j-1} - T_{i,j+1} + 4T_{ij} = P_{i,j}\frac{l}{A_{i,j}\kappa} \tag{3.21}$$

where the right hand side term $\frac{l}{A_{i,j}\kappa}$ is the definition of thermal resistance given by (3.3). We assign this thermal resistance to be $R_{i,j}$, and the RHS of (3.21) becomes $P_{i,j}R_{i,j}$.

Revisiting the nodal temperature equation derived from the perspective of an equivalent thermal circuit consisting of thermal resistances and given by (3.7), we can make some minor modifications to the derivation without changing the analysis. We use the same node (i, j) of Fig. 3.8, with a minor difference that the temperatures $T_{i+1,j}$, $T_{i-1,j}$, $T_{i,j+1}$, and $T_{i,j-1}$ are referred to the edge of the pixel (i, j) instead of at the centres of the respective neighbouring pixels. Fig. 3.14 depicts this change for clarity.

To be consistent with the assumption that $\Delta x = \Delta y = w$ made in Chapter 2, we assume that $l_x = l_y = \Delta x$. Therefore, all the resistors in (3.7) are equal and we

replace them with the resistance $R_{i,j}$. Modifying (3.7) by replacing all resistances by $R_{i,j}$, then multiplying throughout and collecting terms yields:

$$T_{i-1,j} - T_{i+1,j} - T_{i,j-1} - T_{i,j+1} + 4T_{ij} = P_{i,j}R_{i,j} \tag{3.22}$$

Thus (3.22) is mathematically equivalent to (2.17) and (3.21) showing that the resistor network is a valid basis for formulating the finite difference discretization. A similar argument and derivation can be made easily for the 3D case.

3.6 Modelling supporting features

3.6.1 Heat sources

If we consider the source of heat on the PCB to be a surface mount device (SMD) soldered onto the top layer, we can use the simple resistor network shown in Fig. 3.15 to model all the thermal paths present. The heat originates from the junction of the device, and roughly follows two parallel paths to the ambient environment (see Fig. 3.15):

1. From the junction to the top case then to the external environment (greyed out in Fig. 3.15)

2. From the junction to the bottom case, to the solder layer underneath the device, to the PCB to the heat sink (if it exists) which then rejects the heat to ambient.

With some modifications, we can model the SMD heat source within the pixelated image framework. Fig. 3.16 shows a proposed model for representing an SMD heat source atop a PCB. Although a typical SMD source is not readily available from a PCB editor as a pixelated image, a pixelated image that maps its footprint to its location on the board can still be obtained. As a first approximation, the heat source is thus defined to be the pad on the PCB onto which the SMD is soldered.

The model of Fig. 3.16 is based on the following simplifying assumptions:

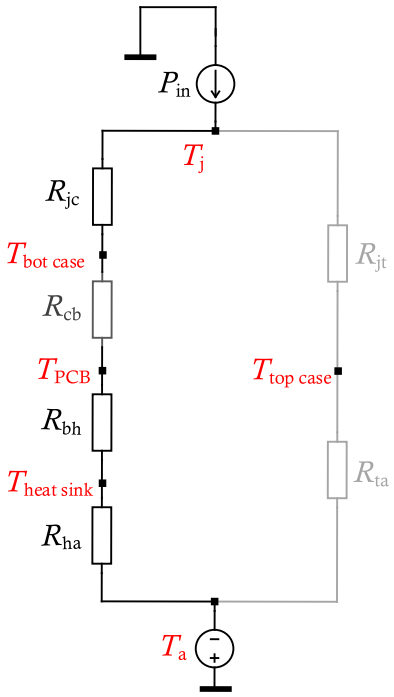


Figure 3.15: Resistor network for an SMD heat source. The greyed-out path is considered less dominant.

- there is perfect alignment and perfect contact between the device pad and its footprint on the PCB. This means that all the thermal power dissipated by the junction goes into the pad.

- the solder thermal resistance (R_{cb}) is very small and can be neglected without significantly altering the second thermal path, a reasonable assumption that is normally followed for modelling pads [102].

- the thermal path that goes from the junction to the ambient through the PCB dominates the heat transport from the source. This assumption is a reasonable approximation given that the material that forms the conductive path from the junction to the top of the device is not a good thermal conductor, and also the surface area of the heat source in contact with the ambient environment is very small.

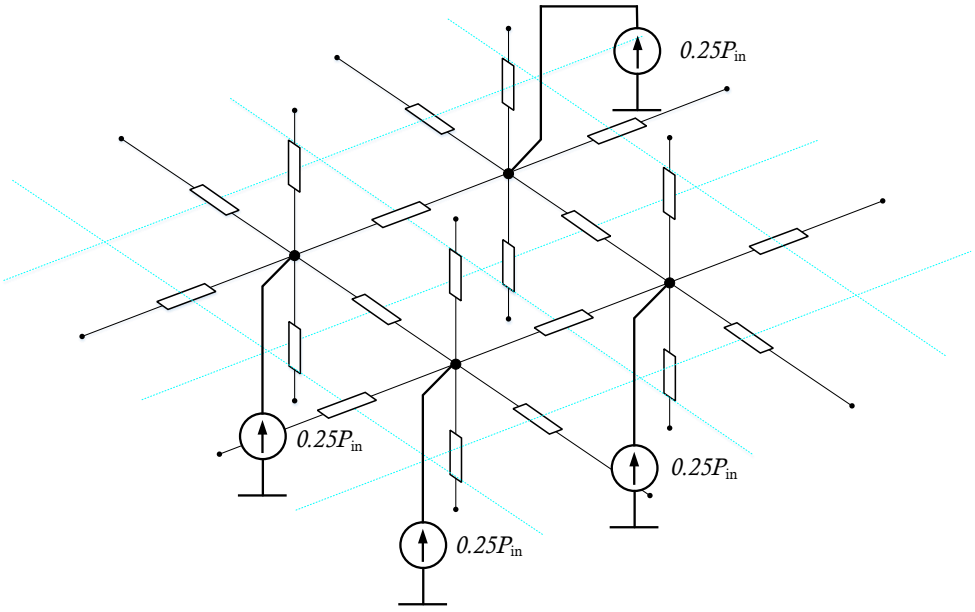


Figure 3.16: Distributed heat source model showing all the pixels receiving an equal portion of the heat generated by the source

With the above assumptions, the heat source is then represented by a thermal power injected from the device junction into the PCB footprint.

Each pixel covered by the heat source footprint is assigned a portion of the injected thermal power injection, P_{in}. The power is distributed equally among all p_s pixels covered by the heat source footprint, and the power per pixel is given by

$$P_{i,j} = \frac{P_{\text{in}}}{p_s} \tag{3.23}$$

3.6.2 Heat Sinks

The effect of a heat sink is to reduce the thermal resistance of the path from the PCB to the ambient environment by increasing both the convection coefficient of the region where improved heat removal is desired and the available surface area for dissipation.

From the inverse relationship between the convective thermal resistance $R_{\theta h}$ and both A and h given in (3.4), we can compute the convection coefficient associated with a heat sink h_{sink} if we know R_{sink} and A_{sink}. R_{sink} is commonly provided in data

sheets and A_{sink} can be deduced from the heat sink construction details provided in datasheets. Therefore, a route to compute h_{sink} exists.

With h_{sink}, known, an approximate pixelated-image-based model of the heat sink is thus derived from an image of the footprint of the heat sink, which can be extracted from a PCB design editor. The convective thermal resistance per pixel connected to the heat sink is then computed using h_{sink}. This means that on the layer occupied by the heat sink, it is possible to use different convection coefficients in the computations.

3.7 Justification and limitations of the steady-state assumption in bare-PCB thermal models

The thermal behavior of a PCB can be modeled in either the steady-state or transient state. A steady-state thermal model predicts the temperature distribution of a PCB when it has reached thermal equilibrium, which is observed when the rate of change of temperature falls below a predefined criterion where further temperature changes become negligible for practical purposes. While the threshold depends on the thermal constraints the system is subjected to, it typically falls in the range $0.1°\text{C/s} - 1°\text{C/s}$ for most power electronics applications. On the other hand, transient thermal modeling gives the temperature variation experienced by a PCB during the process of rapidly heating up or cooling down, for example during startup, in applications that encounter high levels of power cycling, in pulsed operation or when a fault occurs.

So far in this work, we have developed a steady-state thermal model for power electronics PCBs assuming static boundary conditions and thermal loading. This approach is justified for scenarios where thermal conditions remain relatively stable, such as in server environments where PCB converter temperature typically doesn't deviate by more than 10 °C over time. The steady-state model can effectively

verify that components won't exceed maximum temperature limits under normal operating conditions. For example, if the ambient conditions in the server and the expected worst case heat dissipation by the semiconductor switches on the converter are known, these can be used as parameters to the bare-PCB thermal model to verify that none of the components will exceed maximum allowed temperature limits. If they do, the design can then be modified to eliminate the hotspots.

To quantify the validity of the steady-state assumption, results must be compared with transient simulations. We consider the steady-state assumption valid when the difference in maximum component temperatures between the steady-state and transient models is less than 5% after a period of at least 4.4τ (the time for the thermal response to reach steady-state in most power electronics according to [71]). This typically occurs in the order of seconds for most high speed power electronics PCB designs [92].

Furthermore, power electronics converters often require heat sinks and exhibit significant thermal inertia, characterized by thermal time constants in the range of seconds. This property results in attenuated temperature fluctuations of the devices under nominal operating conditions [8] [92].

Steady-state analysis can also capture thermal behavior under extreme conditions by using peak heat dissipation values and boundary conditions encountered during dynamic events as its initial conditions. This approach transforms "worst-case" transient scenarios into steady-state problems, providing conservative estimates without the complexity and incurring high computational cost of time-dependent simulations.

The steady-state model is particularly valuable during the PCB design stage when design specifications and typical operating conditions are known in advance. It allows for informed design decisions, such as optimizing component placement and ensuring compliance with thermal requirements.

To further validate steady-state results, they should be compared with detailed transient analysis for various operating conditions. Transient thermal simulations

with fine time steps (typically 0.1s) run until system stabilization can provide a benchmark for steady-state predictions. In power electronics, steady-state models often predict maximum component temperatures within ±3°C of the stabilized transient model, provided it reaches steady-state.

3.7.1 Limitations of the steady-state assumption in PCB thermal modeling

The steady-state bare-PCB thermal presented herein is not universally applicable to all power electronics PCB operating scenarios, and may not accurately represent the system's thermal behavior under certain specific conditions. For example, it doesn't capture temperature dynamics during transient periods or temporary temperature peaks encountered during a transient event before settling. The accuracy of its predictions depends on the accuracy of the operating parameters (power loss, ambient temperature, and coefficient of heat convection) and sufficient time passing for thermal equilibrium to be established. The limitations of the method become apparent when:

- Load changes occur faster than the system's dominant thermal time constant, typically with sub-second frequency for power electronics applications. In this case, the power electronics PCB, whose thermal time constants occur in the order of a few seconds, will tend not to be able to reach thermal steady-state under the new operating conditions.

- The power dissipation exceeds design specifications, indicating that the system has critically failed.

- Dynamic environmental factors impact the model's accuracy, particularly in situations with rapidly changing ambient conditions, such as variable fluid flow rates in the cooling system or temperature changes exceeding 5°C, the engineering tolerance we adopt, in under 5 seconds.

- The system employs active cooling mechanisms where the convection boundary condition becomes variable. Specifically, in systems with variable fan speeds or liquid cooling systems, the transient interaction between cooling power and heat dissipation would be difficult to model accurately using a steady-state approach.

In all these cases, transient thermal analysis may be more fit for purpose, and other approaches available in the literature must be considered, for example from [82].

3.8 Solving the system

Several methods of solving the system of equations developed in this chapter have been studied and are in mainstream use in major numerical software computing software. In order to justify the approach used in this work, we briefly review approaches one can use to solve the problem. There are two classes of algorithms available for solving problems of the form $\mathbf{Ax} = \mathbf{b}$: namely iterative and direct methods.

Direct methods are variants of Gaussian elimination. These methods use the individual matrix elements directly, through matrix operations such as LU, QR, or Cholesky factorization. For example, when LU factorization is used the Gaussian elimination algorithm works by transforming the $\mathbf{A}$ matrix into an upper-triangular matrix $\mathbf{U}$, a lower-triangular matrix $\mathbf{L}$, and a permutation matrix $\mathbf{P}$ using elementary row operations. Once the transformation is complete, the algorithm solves for the variables in $\mathbf{x}$ by back-substitution or forward substitution, which is a simple and efficient method for solving triangular systems [21].

Direct methods can obtain an exact solution in a finite number of operations and are known to be robust and efficient. A major drawback of direct methods is that they are memory bandwidth constrained because the entire system must be stored in RAM for efficient computation. This becomes a bottleneck for large systems such as the ones considered here. Additionally, direct methods may also be

sensitive to round-off errors and ill-conditioning of the **A** matrix, a problem that does not apply to the simple systems considered in this thesis.

Iterative methods are based on generating a sequence of approximations that converge to the exact solution. These approaches start with an initial guess of the solution that gets progressively improved at each iteration step until the solution is within a predefined tolerance or the maximum number of iterations is reached. Iterative methods can be useful when the system of equations is large and sparse, and direct methods are too costly or impractical.

In our context, an iterative approach is suitable due to the extensive memory requirements of solving the large systems generated in the discretization. The built-in preconditioned conjugate gradient (PCG) algorithm in MATLAB is chosen for this purpose. At each iteration, the algorithm computes a search direction vector and then updates the solution vector by moving along the selected direction. The search direction vector is chosen to be conjugate to the previous search directions, which means that it is orthogonal to all previous search directions and can therefore explore new parts of the solution space [96].

We also use a preconditioner to increase the efficiency of the algorithm, which is a matrix that approximates the inverse of the coefficient matrix **G**. The preconditioner is applied to the residual vector, $\mathbf{r} = \mathbf{q} - \mathbf{GT}$ to produce a new residual vector which is easier to solve than the original one. This step has been found to significantly reduce the number of iterations required for the algorithm to converge to a solution. More information about the working principles of these algorithms is available in the literature and beyond the scope of this chapter [21, 38, 61, 96].

3.9 Implementation

3.9.1 Image acquisition

To build the 3D FDM discretization for the bare-PCB model, binary images (*i.e.* images with only two colours - one indicating the presence of the feature and another in its absence) are extracted from a PCB design software. The PCB geometry can be fully constructed from the six images listed below.

- Patterned copper layers: these are images of the top, bottom, and middle copper layers of the PCB. They contain the copper traces present in the layout.
- Outline: this image consists of a 1-pixel thick boundary of the PCB. As a result, arbitrarily shaped PCBs can be modelled provided the boundaries are represented correctly. Furthermore, regions within the boundaries of the board, such as cutouts, holes and vias, are also drawn using a 1-pixel thick boundary.
- Vias: the image for vias contains information about the location of the plated through hole vias, and the pixels that define the inner edge of the via. The via images are not discussed in detail in this chapter and will be revisited in Chapter 4, where we propose a pixel-derived model for vias.
- Heat sources: this set of images shows the footprints of the power-dissipating components on the PCB. Sources placed on both the top and bottom layers are represented separately, with one image for each source.
- Pins: this image contains information about the locations of filled holes on the PCB.
- Heat Sink: This collection of images represents the footprints of all the heat sinks on the PCB, with each heat sink represented by one image

If a layer is absent from the PCB model, the corresponding image is replaced by a matrix of zeros or an all-black image. The dpi chosen for each image determines its resolution and the maximum dpi supported by the PCB EDA tool must be chosen.

3.9.2 Image processing

The input images must be preprocessed before the FDM discretization can be constructed. The goal of the preprocessing is to perform any necessary resizing of the images in order to reduce the number of pixels and store the downsampled input images using efficient associative data structures before they are combined into the large resistor network. It is assumed that each layer image contains all the information required to construct the model, namely, the resolution, provided as a dpi constant, and the scale factor, sf, which is used to resize the image to different resolutions. The length of the pixel is determined from the scale factor sf, the dots per inch, and is given by:

$$l_x = \frac{2.54 \times 10^{-2} \text{ sf}}{\text{dpi}} \tag{3.24}$$

t is necessary to downsample all the images in order to reduce the number of pixels without distorting the distribution of copper and the physical construction of features on the layers in a way that creates thermal short circuits - which arise when copper-FR4-copper regions are bridged. Binary downsampling using a carefully chosen scale factor ensures that the PCB layers are aligned correctly at lower resolutions than the ones used for extracting the images from the EDA tool. In order to avoid pour-to-pour bridges a pixel in a downsampled image is assigned to be copper, if all subpixels in the original image in the region of interest are also copper. This tends to avoid bridging, but makes tracks skinny.

The outline of the PCB, which is obtained from the outline layer image, and the perimeters of the plated through holes and other holes and cutouts on the board, are first combined to form a binary mask that shows the location of the

PCB and excludes areas where the PCB is missing. More details on the processing of vias are provided in the next chapter.

The model presented in the preceding section is implemented in MATLAB, and organized as shown in Fig. 3.17

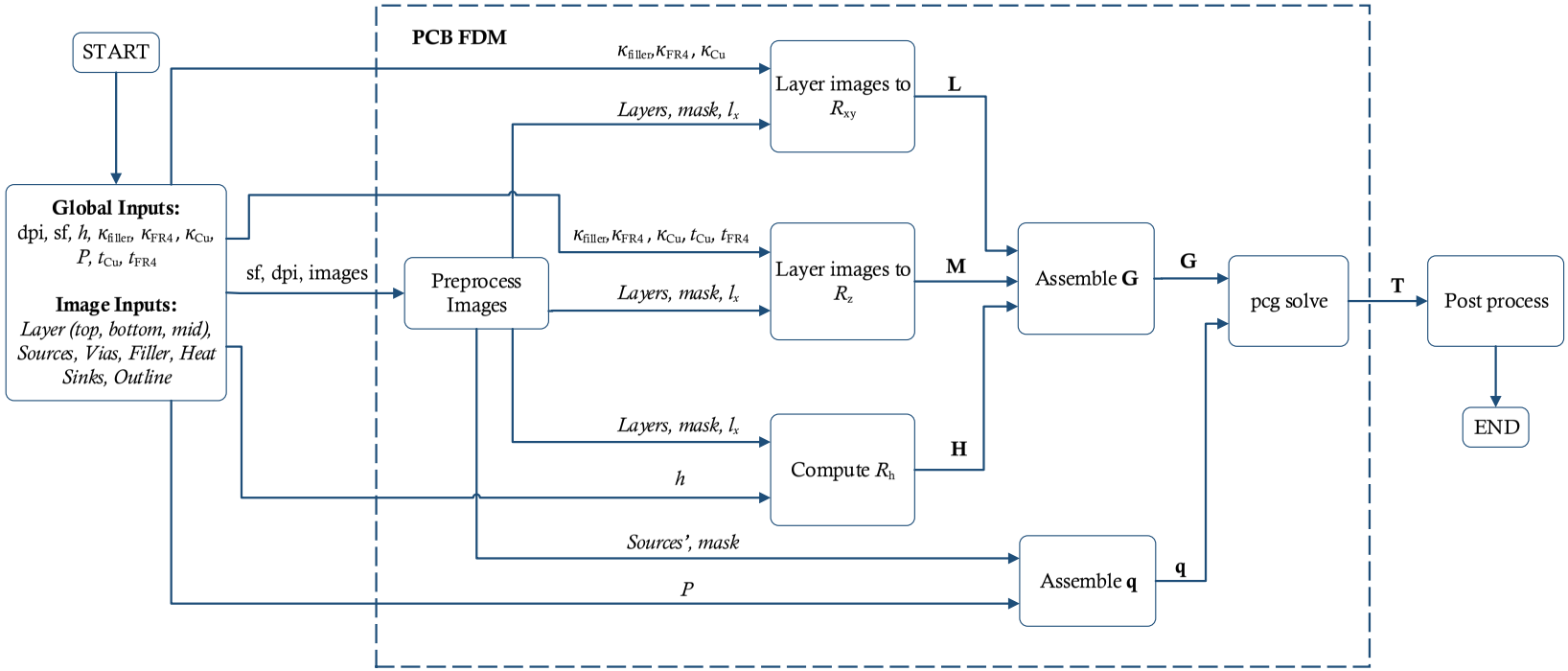


Figure 3.17: Summary of MATLAB implementation of the finite difference code.

1

[1]Acknowledgements: Dr. Dan Rogers, Professor of Engineering Science, for his contributions in implementing the MATLAB code used to construct the system matrices referenced in this section.

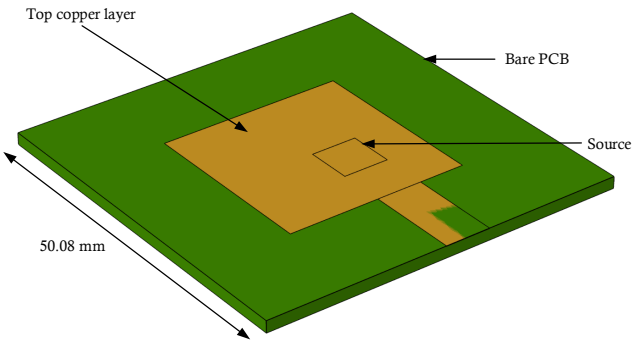


Figure 3.18: COMSOL simple geometry for testing and validating the FDM problem

3.10 Model validation

We now turn to COMSOL multi-physics to validate the thermal models developed in this chapter. Accurate FEM simulation performed in COMSOL on geometries built from CAD geometry imports that have been defeatured appropriately are used as the basis for comparison.

In the first numerical experiment, we validate the correctness of the finite difference method implementation using half-pixel thermal resistances.

3.10.1 Choice of Test Case

The following geometry is modelled in COMSOL and becomes the basis of our test. This is a simple two-layer representation of a PCB. Although the geometries involved are much simpler, the board provides a good approximation of how a power electronics PCB might look like by having large copper pours and no traces which tend not to have a significant contribution to the thermal behaviour of the board.

The key parameters used for the test case are given in Table 3.1. The dimensions of the board and the heat source are selected deliberately to enforce spatial equivalence between the COMSOL and image-derived thermal models at different scale factors. The finite difference model uses a uniform grid whereas COMSOL uses adaptive meshes which conform to the physical geometry of the problem more

accurately. From observation in numerical experiments, the uniform grids used in the image-derived mesh will tend to move edges around at different scale factors, and thus not keep the sizes of the heat source, copper pours and PCB consistent. As a result, for verification purposes, it is important to ensure that the models compared are strictly identical at different downsampling scales. This problem is addressed by grayscale downsampling which is discussed in Chapter 5.

Parameter	Value
k_{Cu}	400.0 W/m^{2}K
k_{FR4}	0.30 W/m^{2}K
h	10 W/mK
l_{PCB}	5.08 mm
P	5 W
T_{amb}	0 K

Table 3.1: COMSOL Model Parameters

Although the simple models of the heat source and the heat sink presented above are simplistic and not strictly representative of reality (improving them is the subject of Chapter 4), they are still instructive and reproducible/comparable in COMSOL as part of a validation effort. We choose a fairly simple and uncomplicated 2-layer geometry that can be directly reconstructed in COMSOL with minimal effort enabling us to compare the validity and performance of the FDM discretization developed in this chapter against the 'ground truth' thermal profiles generated in COMSOL. We also choose the model such that the PCB maintains its dimensions (heat source size, copper-FR4-copper gaps *etc*) at various scale factors in order to get a precise comparison of the model behaviour.

3.10.2 Results

The overall length of the PCB is shown in Fig. 3.15 (a), confirming that the dimensions of the board and its features did not shift at the different scale factors used. This edge-shifting phenomenon can lead to inaccurate results for example by

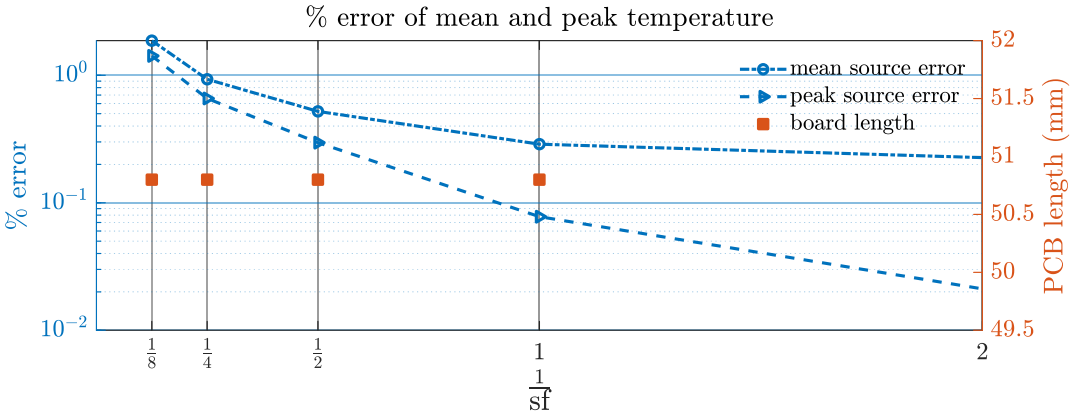


Figure 3.19: Percentage error of mean and peak temperatures of the source between COMSOL and FDM solver

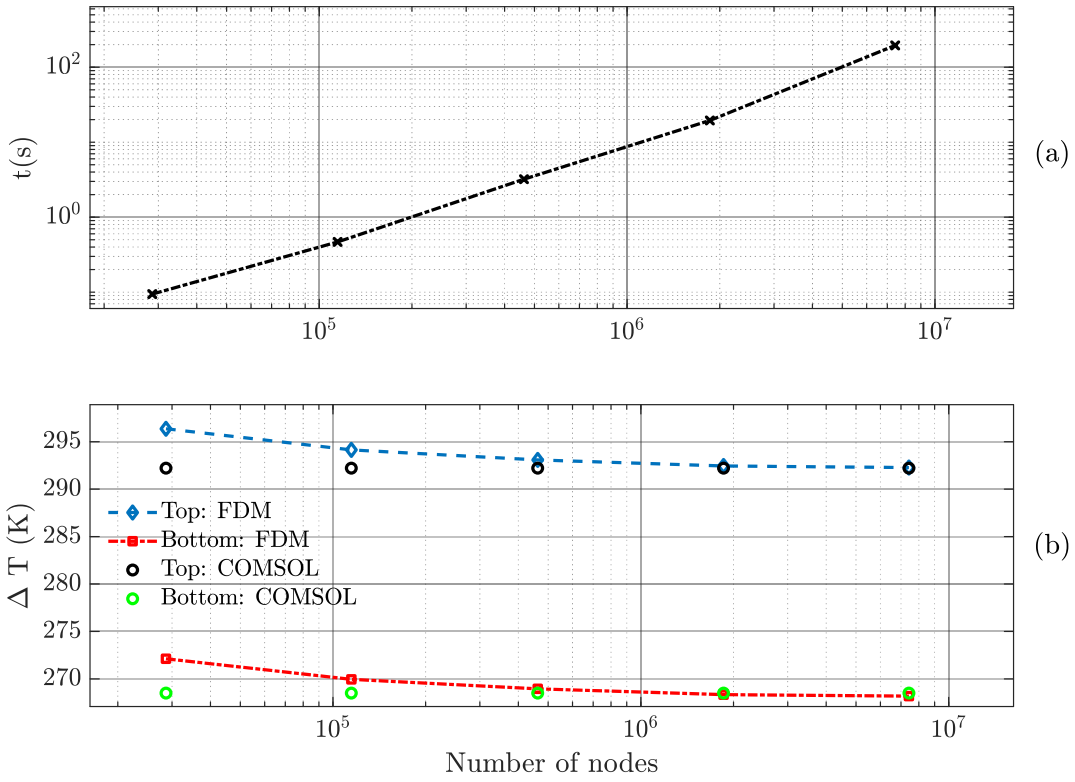


Figure 3.20: a. Solution time vs number of nodes. b. Comparison of average source and peak board temperature between COMSOL and FDM solver

altering the surface area available for the convection coefficient to be applied. Here we have chosen a length of the PCB and resolution that allows us to maintain the exact dimensions of the PCB at all tests scale factors by ensuring that the dpi used places the edges of the pixels in the same vector space as it gets scaled by 2.

As can be seen from the third plot, the solution time scales quadratically with the number of elements in the mesh. Consequently, the element count must be kept low in order to achieve short solution times.

The heat source absolute temperature and the associated error, when compared against COMSOL, is shown in Fig 3.15 (a) and (b). The error is computed using:

$$\epsilon_{\mathrm{T}} = \left| \frac{T_{\mathrm{FDM}} - T_{\mathrm{C}}}{T_{\mathrm{C}}} \right| \times 100 \tag{3.25}$$

where T_{FDM} is the average temperature of the heat source computed using the proposed thermal model and T_{C} is the temperature of the heat source computed using COMSOL. The temperature of the heat source in COMSOL is extracted at a length of $\frac{t_{\mathrm{Cu}}}{2}$ because that is the location of the nodes in the FDM discretization for the top copper layer. This allows us to obtain a precise comparison.

The error decreases as the number of elements increases. A worst-case error of 26% is observed for a solution time below 1 second, contrasted against a best-case error of 0.13%, obtained in 210 seconds. The COMSOL model solves in 187 seconds at a 'fine' mesh and 306 seconds when using an 'extra-fine mesh'.

3.10.3 Impact of FR4 sublayer discretization scheme on the accuracy of the model

The impact of the rule used to segment the vertical dimension of the PCB is shown in Fig. 3.21. This plot was generated by solving the simple thermal problem presented earlier using different discretization schemes for 5 sections of the FR4 sublayer. The error in the mean source temperatures between the FDM solution and COMSOL is compared.

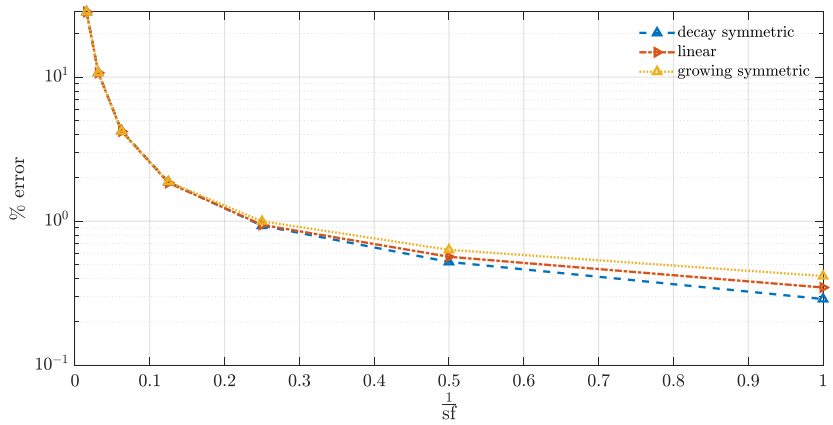


Figure 3.21: Error in peak and mean source temperatures as a function of scale factor for different segmentation rules. Comparison of average source and peak board temperature comparison between COMSOL and FDM solver

As the figure shows, for a given choice of FR4 sublayers, decaying symmetric spacing has the best numerical performance, followed by linear spacing, and lastly growing symmetric spacing. This confirms that the accuracy of the solution is not only determined by the number of segments in the z direction, but also by their distribution. This result also shows an inverse relationship between the scale factor and the measured error *i.e.* the improvement in the measured error increases as the scale factor decreases (in other words, as n_x and n_y increase). The results in Fig 3.21 indicate that decaying symmetric spacing is the best choice for segmenting the vertical dimension of the PCB for thermal analysis. This is because it results in the most accurate solution with the fewest number of segments. The improvement in accuracy is particularly pronounced for small scale factors, where the error in the solution can be significantly reduced by using decaying symmetric spacing.

Additionally, it is noteworthy that the better performance of the decaying symmetric segmenting scheme is related to the placement of the nodes at the center of the each thermal cuboid, and not exactly at the interface between the two materials. By segmenting the vertical dimension in this way, the distance between each node adjacent to an interface is minimized, allowing the thermal

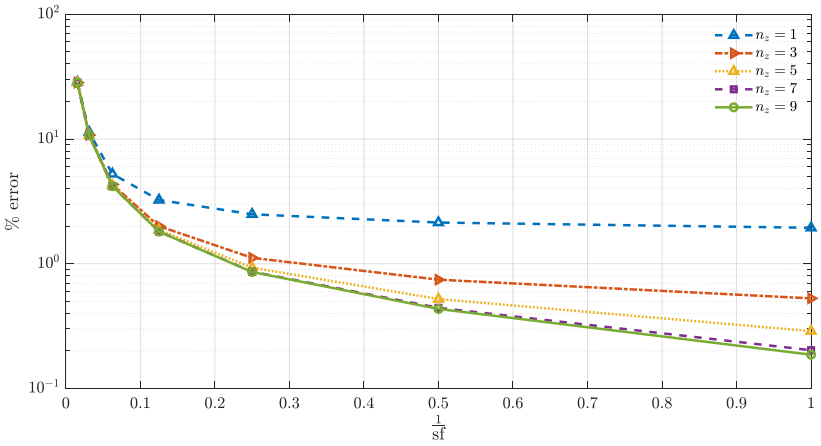


Figure 3.22: Impact of the number of n_z layers on the accuracy of the model. A decaying symmetric discretization approach was used. At high scale factors, all tested segment counts from 1 to 9 lead to equally inaccurate results. The accuracy improves for high segment counts as the scale factor reduces. Increasing the number of segments beyond 5 leads to minimal improvements in accuracy, with 7 and 9 segments being nearly identical in performance for all scale factors considered.

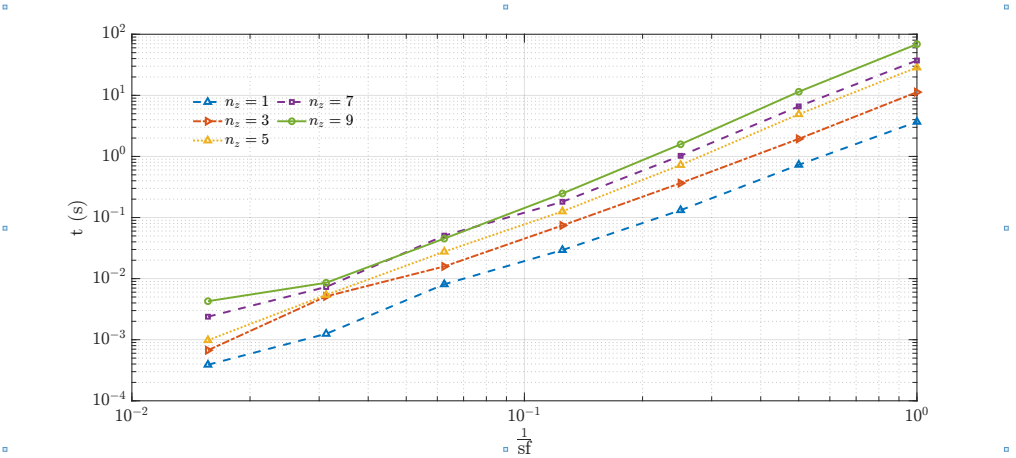


Figure 3.23: Impact of the number of n_z layers on the computation time of the model. A decaying symmetric discretization approach was used. At high scale factors, all tested segment counts from 1 to 9 lead to equally inaccurate results. The accuracy improves for large segment counts as the scale factor reduces albeit with a tradeoff of taking longer to compute.

path to capture the large thermal gradients that are dorminant in regions where there is a transition from one material to another.

Fig. 3.22 shows the variation of computational time with the scale factor as well as the relationship between the error and the number of divisions in the FR4

segmentation. We used the decay symmetric spacing segmentation method to subdivide the FR4 layer (Rule 3, Fig. 3.13 (b)). As expected, the error is lower for large numbers of divisions. However, the error tends to a limit, and $n_z > 7$ incurs a time penalty in solving the problem without meaningfully increasing the accuracy of the solution. We found that $n_z = 5$ is optimal for our problem, as it achieves a low error with a reasonable computational time - thus it is the segmentation choice used in all numerical experiments in the remainder of this thesis. Also, from the aforementioned figures, we notice that the choice of n_z has a negligible impact on the accuracy of the solution at high scale factors. This is because the resolution of the downsampled image (*i.e.* $n_{xd} \times n_{yd}$) is so low that the accuracy of the solution is equally bad for any number of FR4 segments.

3.11 Summary

This chapter presented a numerical method for solving the thermal problem of bare-PCBs using resistor networks and a convective boundary condition. Rationale for using a steady state model, as well as it's scope of application was given. Similarly, we also discussed why a linear thermal model is appropriate for studying power electronics PCB thermal behaviour. The method was verified against COMSOL using a simple heat source model with carefully selected dimensions to prevent grid mapping misalignments. However, this model was too simple to capture the realistic thermal behaviour of PCBs. The next chapter will improve the heat source and heat sink model and introduce a model for vias.

We also showed that a 'decaying' symmetric discretization scheme with at least 5 divisions is required to segment the FR4 layer in the vertical direction.

4

Pixel-derived thermal models for full-featured PCBs

The aim of this chapter is to extend the simple bare-PCB thermal model derived from rasterized images of PCB layers presented in the preceding chapter and generalize the framework to incorporate *full-featured* PCBs. In this context, a full-featured PCB consists of all the thermally important features presented in the first chapter: namely vias, heat sinks, multiple heat sources, intricate copper patterns on each conducting layer, arbitrary shapes, holes and cutouts. We propose techniques for deriving FDM discretizations for the aforementioned features from rasterized images and update the 3D FDM mathematical system derived previously to a generalized form.

Lastly, we propose improved models for heat sources and heat sinks by incorporating thermal parameters commonly provided by manufacturers in data sheets. This includes junction-to-case and junction-to-ambient thermal resistances for heat sources and bulk resistances for heat sinks. Component manufacturers obtain these device parameters from detailed thermal modelling of each device using FEM, a process they are uniquely positioned to perform due to their knowledge of the internal design details that are not normally publicly available. With this in mind, we consider these device thermal models to be satisfactorily accurate, and

ready to be incorporated into the bare-PCB thermal model without modification. Methods of attaching standalone compact thermal models of discrete components have been studied and demonstrated by several authors and we propose an intuitive approach for achieving this [33], [44] [94]. After integrating available component thermal parameters into the bare-PCB thermal model, we show that the improved model leads to more realistic thermal insights for PCB designers in the iterative pre-fabrication design stage. A representative full-featured power electronics PCB is presented and studied to benchmark the performance of the improved model and validate its utility.

4.1 Holes, cutouts and arbitrary PCB shapes

Although rectangular-shaped PCBs are the most commonly occurring in power electronics applications, other arbitrary shapes are also available. Packaging constraints are the main driver for non-standard PCB shapes. For example, PCB-based electronics featuring complex rectilinear and curvilinear shapes are common in automotive applications to conform to fitting requirements while circular and semi-circular ones are often used in motor drive applications.

Arbitrary PCB shapes can be modelled accurately by extracting an image of the board outline from the PCB design tool. Using a one-pixel thick line located at the boundaries of the PCB, a binary mask that marks the presence or absence of the board in various regions can be obtained. Pixels that fall outside the boundary are not processed in the computation of the thermal resistances and are replaced by zeros instead.

The $\mathbf{G}$ matrix can then be assembled as described in Chapter 3, with zero entries where the PCB is not present. The mask matrix can be used as a final preprocessing step before the system is solved to overwrite the thermal resistances of nodes that coincide with regions not covered by the PCB substrate such as holes, vias, and

other cutouts. Each row in **G** corresponds to a node in the PCB domain, therefore, we can simply delete non-existent nodes. Such a step does not change the desirable properties of the system including positive semi-definiteness, symmetry, and sparsity.

Practical PCBs often have cutouts serving various purposes, such as integrating planar magnetics or providing fitting clearances. In addition, holes are usually included for mounting purposes. These features can easily be included in the model by incorporating them into the outline images using a one-pixel thick line, and image processing to identify their locations on all the images used to construct the bare-PCB thermal model.

As before, the conductance matrix, **G**, is constructed as if the voids were part of the domain, and then modified by deleting entries that correspond to non-existent nodes in the network, before the system is solved.

4.2 Vias

A via provides a low thermal resistance path for heat dissipated on one PCB layer to be transferred to another layer before being rejected to the ambient environment. Vias are a key thermal design feature of PCB power electronics and are typically placed in a 2D array configuration under a power semiconductor device with a heat sink attached on the opposite side (for bottom-cooled devices). Optimising the configuration, count and thermal resistance of via arrays has received significant research interest for power electronics applications, see for example [36] [37] [102].

Modelling vias using thermal resistor networks has also been widely studied by several authors, and we briefly review some of the major works treating this topic for PCB-based electronics.

4.2.1 Thermal modelling of vias in the literature

The authors of [66] develop a compact analytical thermal resistance model for vias that gets incorporated into a larger 3D finite difference thermal model. The compact via model treats each via cell separately, with a focus on computing the lateral thermal resistance. The authors show that there is a slight improvement in the computation time and accuracy of the temperature profile of the vias but do not study how this model would behave as part of a full PCB thermal analysis.

In [107] the authors proposed a perhaps oversimplified model of vias model where the via cell was considered to be a combination of lateral and vertical thermal resistances, from which a fictitious material was derived and assigned to have thermal properties based on the orthotropic thermal conductivity combinations of the cell. They claim that this approach sped up the meshing process of the PCB in an FEA modelling scheme, and thus reduced the solution time due to the low number of elements used to represent the board. This simplistic method suffers from the fact that so-called effective thermal conductivities obtained in this way do not represent the properties of the materials in the domain accurately. Thus, the method is not attractive, as shown by the large errors they observed even for simple geometries.

In [75] , the authors proposed a 3D equivalent thermal circuit for transient thermal analysis of via arrays, considering both vertical and lateral heat flow. In this scheme, the via array was segmented such that each via was represented by a 3D cuboid cell whose thermal parameters could be analytically determined independently and then integrated into an RC thermal network, a common approach also followed in [76], [11]. Using a modified nodal analysis solver, and representing the heat sources as distributed across the via array, the authors' model was found to be accurate for small numbers of vias arranged in a uniform pattern. The accuracy of the model deteriorated when the number of vias was increased, leading to errors of around 8% when benchmarked against equivalent thermal analysis

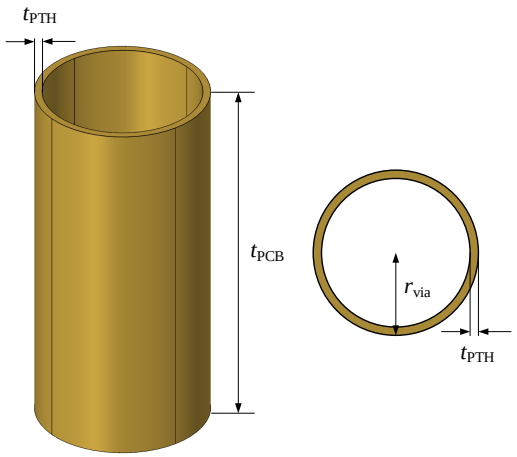


Figure 4.1: (a) 3D rendering of a plated through hole via (b) Top view of a single via

using COMSOL for arrays of 10×10 vias. Unfortunately, the authors do not provide further details about how the heat sources are incorporated into the model, or how the model integrates with an overall thermal model that incorporates other thermally relevant features.

Other research suggests that lateral heat spreading in thermal via clusters is a negligible concern in engineering practice especially if the ratio of the thermal conductivity of the copper plating and the filler material is large [9] [98]. This assumption is the basis of the via model developed in [63] where the authors posit that the thermal resistance of a via is a function of various design parameters such as the thickness of both the PCB and through-hole plating, the pitch, and diameter of the via as well as the properties of the filler material. Interestingly, this model behaves better for a large number of vias (36) which is opposite to the effect observed in [75].

A via is difficult to represent on a raster image because of its small plating thickness, typically 35 μm. Due to this small size, there is a tradeoff between exporting images of vias that accurately preserve the physical dimensions of the structure from most PCB design editors at a reasonable resolution. Prohibitively high-resolution images where the length of each pixel is only a few μm long or

at least 5 orders of magnitude smaller than the typical length of a PCB (which tends to be a few cm) would be required.

Although there is limited research work on pixel-derived thermal models of vias, a few efforts have appeared in the literature. In an attempt to model vias from pixels, the authors of [9] resorted to using extremely high-resolution images such that the plating thickness was represented by more than 4 pixels. In order to do this, they modelled the vias separately from the entire board, which departs from the full PCB thermal model studied in this thesis. The image-based thermal model presented in [7] does not provide details about how via thermal parameters were calculated, instead resorting to processing them in a general-purpose scheme that is also used for traces and copper layers.

4.2.2 Benchmark analytical thermal resistance model of vias

Computing the thermal resistance of a via is a trivial undertaking given knowledge of the materials from which it is made, its geometry, and its dimensions. The physical structure of a plated through hole via is shown in Fig 4.1. A via, structurally speaking, is a simple metallic tube that is surrounded by a complicated interleaved structure of copper and FR4, and filled with air or another material in its centre. Here we derive the equivalent thermal resistance of a single via, which becomes the basis on which we derive the thermal resistance of an array of vias that forms part of a multilayer PCB. We then treat this as the reference value that the pixel-derived thermal resistance of vias must match at all reasonable downsampled image resolutions in our bare-PCB pixel-derived thermal model. We propose ways of incorporating the vertical thermal resistance of the via into the bare-PCB model presented in the previous chapter.

Consider the via shown in Fig 4.1. The plating thickness, length, and radius of the via are denoted by t_{PTH}, t_{PCB}, and r_{via} respectively. Using the definition of conductive thermal resistance, we can calculate the R_{via} as:

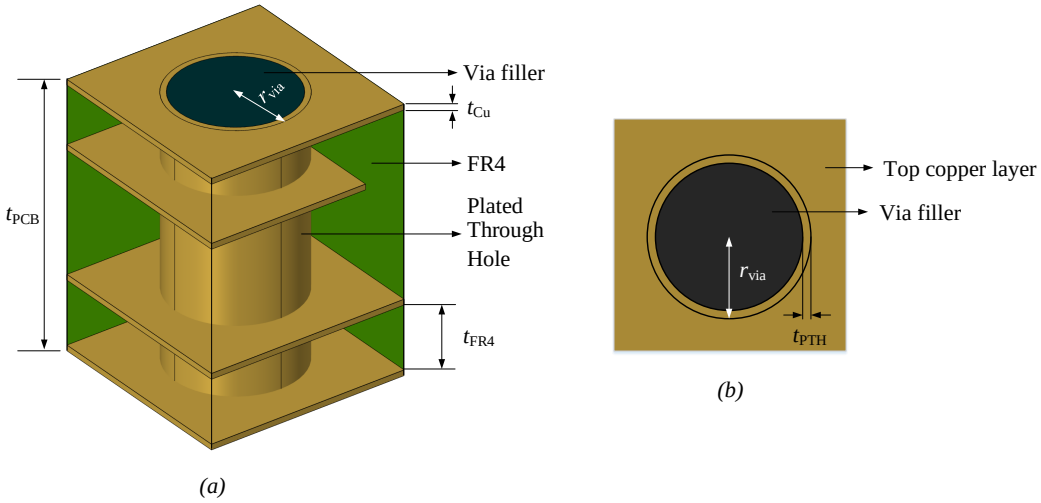


Figure 4.2: (a) 3D rendering of a plated through hole via showing all the internal layers it is connected to. We consider this section of the PCB to be a 'via cell' (b) Top view of the single via cell

$$R_{\text{via}} = \frac{t_{\text{PCB}}}{\kappa_{\text{Cu}} A_{\text{PTH}}} \tag{4.1}$$

A_{PTH} is the surface area of the via in the xy plane and computed from:

$$A_{\text{PTH}} = \pi\, r_{\text{via}}^2 - \pi\, (r_{\text{via}} - t_{\text{PTH}})^2 \tag{4.2}$$

Obviously, a via does not exist in isolation as stated here but instead is part of a small section of the PCB, whose possible construction details are shown in Fig 4.3 for a four-layer PCB. In this setup, R_{via} is part of a parallel resistor network consisting of the interleaved copper/FR4 path, whose total resistance is shown in the figure as the series combination $(n_z - 1)R_{\text{FR4}} + n_z R_{\text{Cu}}$, and the filler material, whose resistance is represented in the figure as R_{filler} at the centre of the via.

Assuming that all the copper and FR4 layers are of equal thickness respectively, we can compute the thermal resistance of the PCB subsection containing the via. We let n_z be the number of layers of copper, which in this case is 4. We also assume that all the copper layers have an equal cross-sectional surface area when viewed looking down onto the board. While this assumption does not apply to the general case

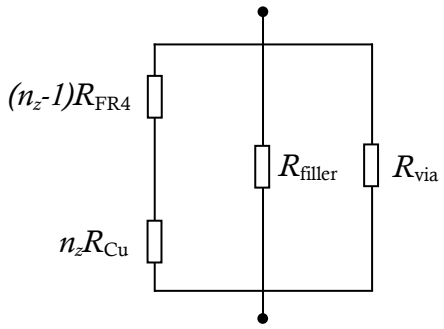


Figure 4.3: Resistor network for a representative via cell

where one or more of the copper layers will be patterned, it simplifies the analysis without losing generality. The thermal resistance of the thermal path is given by:

$$R_{\text{Cu/FR4}} = \left(\frac{1}{A_{\text{Cu/FR4}}}\right)\left(\frac{(n_z t_{\text{Cu}})}{\kappa_{\text{Cu}}} + \frac{t_{\text{PCB}} - n_z t_{\text{Cu}}}{\kappa_{\text{FR4}}}\right) \tag{4.3}$$

where $A_{\text{Cu/FR4}} = l_x l_y - \pi r_{\text{via}}^2$. Consequently, we can write the thermal resistance of this structure:

$$R_{\text{via,cell}} = \left(\frac{1}{R_{\text{Cu/FR4}}} + \frac{1}{R_{\text{filler}}} + \frac{1}{R_{\text{via}}}\right)^{-1} \tag{4.4}$$

The thermal resistance of the filler portion of the via is given by:

$$R_{\text{filler}} = \frac{t_{\text{PCB}}}{\kappa_{\text{filler}} A_{\text{filler}}} \tag{4.5}$$

The method presented here can be generalized to an array with more than one via (see Fig 4.8). Similar analysis can be performed for each via subsection in the array and the overall thermal resistance of the array of vias can then be constructed from the parallel resistor network composed of these tiny via sections.

4.2.3 Pixel counting approach to approximate dimensions of circular features on a raster image

Computing thermal parameters of vias from downsampled images

It is mandatory that the downsampling process must not result in thermal short-circuiting, which occurs when adjacent sections of copper separated by thin strips

of FR4 are bridged in the process of generating a lower-resolution image. In this context, we define a tile, $\mathbf{D}$, as a 2D matrix of pixels representing a small subregion of the input image, whose entries are combined to form a single pixel in the downsampled image. It follows then that the goal of downsampling is to generate a simpler, lower-resolution image with larger pixels on each layer. These larger pixels must be computed to encapsulate the properties of all the features that exist on the layer accurately. As a result, features that are common to all layers such as vias, cutouts and fillers contribute to the properties of the pixels they coincide with on all layers, and must maintain a consistent spatial mapping so that their thermal resistance contribution in the vertical direction can be modelled correctly.

Consequently, a tile, $\mathbf{D}$, may only be assigned the thermal properties of copper (or a value of 1) after downsampling only if it was completely filled with copper, to begin with. Also, any pixel in the downsampled image that is not assigned the thermal conductivity of copper, plating through, or filler material is assumed to be FR4. Any pixels on the plated via cylinder are also assumed to be mutually exclusive from those aligned with the filler material. Consequently, a pixel is either exclusively part of a copper pour, or a mixture of plating-through material, filler material or FR4.

Let the binary input image of vias be $\mathbf{V}$, and the binary input image of the filler material be $\mathbf{F}$, both with size $n_x \times n_y$. Built-in image-processing routines in MATLAB can be used to find the inner edges of all the vias (i.e. circular features in $\mathbf{V}$) and drawing a one-pixel perimeter along it. Similar processing can be performed on $\mathbf{F}$, yielding updated matrices $\mathbf{V}$ and $\mathbf{F}$ which contain the locations of the via perimeter and via filler respectively at the same resolution as the original input images.

The objective is now to downsample $\mathbf{V}$ and $\mathbf{F}$ into smaller (lower resolution) matrices, $\mathbf{V}_\mathrm{d}$ and $\mathbf{F}_\mathrm{d}$ of sizes $\left\lfloor \frac{n_x}{\mathrm{sf}} \right\rfloor \times \left\lfloor \frac{n_y}{\mathrm{sf}} \right\rfloor$, where sf is the scale factor used to resize the image. Each pixel in the downsampled image, $\mathbf{V}_\mathrm{d}(i, j)$, is derived from a tile, $\mathbf{D}$, which is a small $\mathrm{sf} \times \mathrm{sf}$ binary matrix obtained from $\mathbf{V}$. Note that we discard pixels that would result from non-square tiles which can occur at the right and bottom

edges of $\mathbf{V}_{\mathrm{d}}$ and $\mathbf{F}_{\mathrm{d}}$. In practice, this is a reasonable step provided that the original input images are sufficiently oversized on the right and bottom edges such that no information within the PCB boundary image is lost in the downsampling step.

To model the vias, our objective is to find the arc length of the via that is present in each tile. This can be achieved by counting non-zeros within the tile such that the value of each pixel in the down-sampled image is given by:

$$\mathbf{V}_{\mathrm{d(i,j)}} = \sum \mathbf{D} \tag{4.6}$$

and then normalizing by dividing by sf^2.

Note that, while $\mathbf{V}$ is binary, $\mathbf{V}_{\mathrm{d}}$ is a smaller grayscale version where the value of each entry in the image matrix represents the fractional via fill. Consequently, $\mathbf{V}_{\mathrm{d}}(i,j)$ can exist in the range $0 \leq \mathbf{V}_{\mathrm{d}}(i,j) \leq \pi$. The two extreme cases are when there are no via pixels in a given tile, in which case $\mathbf{V}_{\mathrm{d}}(i,j) = 0$, and the other is when an entire via is enclosed in a tile of size $\mathrm{sf} \times \mathrm{sf}$. In that case, the perimeter of the via within the tile in pixels is given by $p = \pi \times \mathrm{sf}$ meaning that the perimeter is given by π pixels.

4.2.4 An analysis of circle rasterization using the Bresenham algorithm

Discretizing a circle using pixels carries with it an inherent discretization error, which must be accounted for and corrected in the calculation of the discrete arc length of the via presented above. To both study and illustrate the nature of this discretization error, it is possible to calculate both the area and perimeter predicted of a discrete circle created using Bresenham's algorithm and compare these values to those obtained from analytical calculations. Examples of circles drawn using the Bresenham algorithm at two resolutions are shown in Fig 4.4.

Fig 4.5 shows the variation of the error in the area and perimeter of circles drawn using Bresenham's algorithm for a range of pixels on the diameter of the

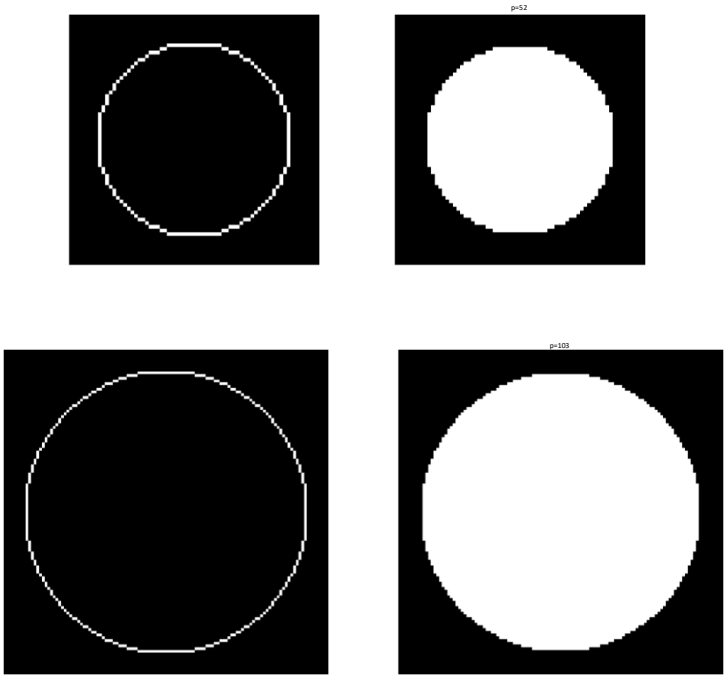


Figure 4.4: Discrete models of a via perimeter and via filler

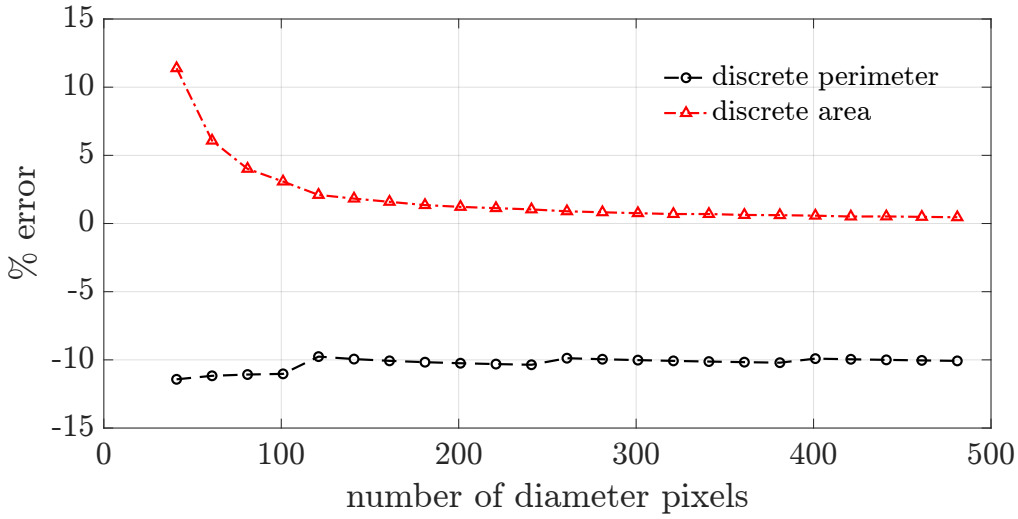


Figure 4.5: Error in the area and perimeter of a circle drawn using Bresenham's algorithm and the vector representation of the same circle

circle. As can be seen from the figure, estimating the perimeter of the circle by counting pixels has an inherent discretization error of approximately 10% at all resolutions considered, whereas the error in the area of the circle has an asymptote of 0 % as the number of pixels along its diameter increases. The region of interest for our purposes is below 50 pixels along the diameter of the circle (which implies pixels of at most 14 μm for a PCB with via diameters of 0.7 mm).

Perimeter

The Bresenham algorithm works by drawing the pixels along the circumference in the first octant of the circle and completing the other portions by symmetry [124]. The objective of the algorithm is to maximize the sum $x^2 + y^2$ subject to $x^2 + y^2 \leq r^2$ where r is the vector radius of the circle. At each iteration step, the algorithm increments x and then decide whether y should also be incremented. The given constraint is violated when $x = y$, and the algorithm terminates. In general, the error term in the computation is given by:

$$\epsilon = x^2 + y^2 - r^2$$

Let the total number of pixels within a circle drawn by the Bresenham algorithm be n_c, given by:

$$n_c = 4 \left\lceil \sqrt{2} \left(\frac{n_d - 1}{2} \right) \right\rceil \tag{4.7}$$

where n_d is the number of pixels along the diameter of the discrete circle. n_c is a geometric series that results from counting the number of pixels along the circumference of the circle $n_d = 6, 8, 10, \ldots$.

Therefore, comparing the pixel-based circumference computed above using the number of pixels n_c to the actual perimeter ($C = \pi d$), we can deduce the error, and thus the correction factor in the perimeter estimation for any via segment. The relationship is plotted below for the range [3, 1000] in Fig 4.6.

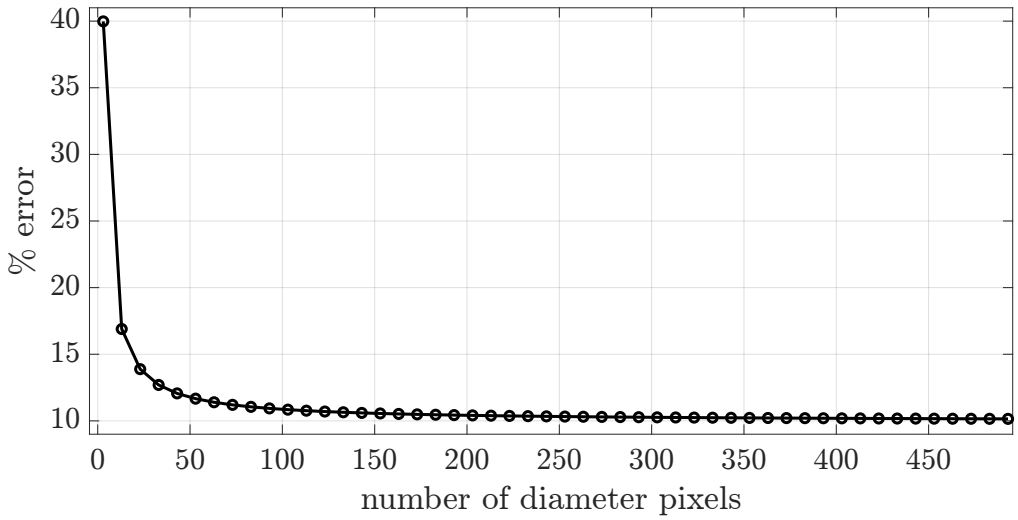


Figure 4.6: Correction factors needed for discretizing circles with pixels

As shown in 4.6, the error approaches a minimum of 10% , which is consistent with numerical experiments conducted using MATLAB's circle drawing routines `bwdist`. It follows that the arc length of the via computed for each tile must be corrected using a factor of ≈ 1.1 because the algorithm tends to underestimate the actual perimeter of the via.

Area

The number of lattice points covered by a circle that encloses a grid of equally spaced points can be deduced by following Gauss's Circle Problem solving approach and available on the Online Encyclopedia of Integer Sequences [5]. If the number of pixels along the diameter of the discrete circle is odd, then it is given by:

$$N_{px}(n_d) = 1 + 4\sum_{i=0}^{\infty}\left\lfloor\frac{n_d^2}{4i+1}\right\rfloor - \left\lfloor\frac{n_d^2}{4i+3}\right\rfloor \tag{4.8}$$

In the case above, the centre of the circle is located within a pixel. If, on the other hand, it is located at the intersection of four pixels, that is the circle has an odd diameter, then the number of lattice points enclosed is given by:

$$N_{px}(n_d) = 4\sum_{i=0}^{\infty}\left\lfloor\frac{1}{2} - \sqrt{\frac{n_d^2}{4} - (i - \frac{n_d^2}{4i+3})}\right\rfloor \tag{4.9}$$

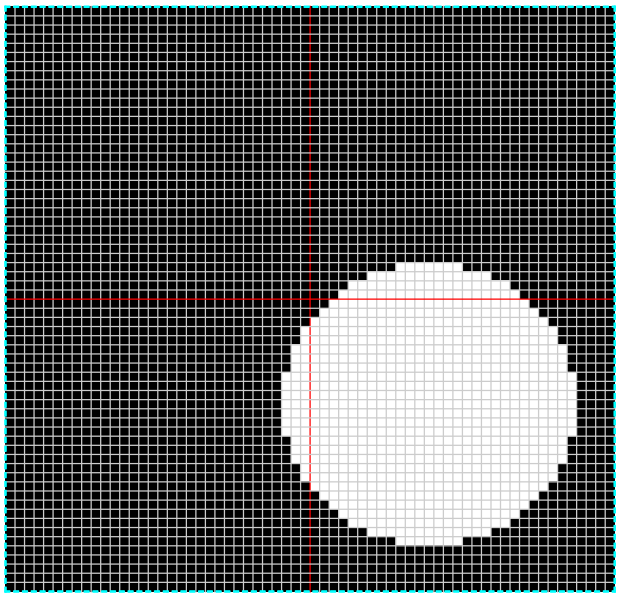


Figure 4.7: Via filler distributed among 4 tiles. The tile boundaries are drawn in red for clarity. A 1 pixel perimeter drawn around the filler material can be used to compute the arc length of the via in each tile. In this example sf = 36

Both (4.8) and (4.9) have an asymptote of 0 as the number of pixels along the diameter of the circle increases, which is consistent behaviour observed in Fig 4.5.

4.2.5 Applying the pixel-counting method to compute the thermal conductance of a layer in the vertical direction

In order to calculate the overall vertical thermal resistance of a layer, three downsampled images are required; namely - $\mathbf{V}_\mathrm{d}$, (the greyscale image obtained from via processing where each entry represents the arc length of a via present in the pixel) $\mathbf{F}_\mathrm{d}$, (the greyscale matrix of filler material) and $\mathbf{L}$, the binary image with ones where copper is present on a given layer. If, for a pixel (i, j), both $\mathbf{V}_\mathrm{d} = 1$ and $\mathbf{L}(i, j) = 1$ the pixel is considered to be exclusively copper. The same argument is made for pixels with a value of 1 in the $\mathbf{F}_\mathrm{d}$ matrix.

We can calculate the area of the pixel covered by the plating through material from the arc length matrix, the thickness of the via cylinder and the length of the pixel:

$$\mathbf{A}_{\mathrm{PTH}} = t_{\mathrm{PTH}} l_x \mathbf{V}_{\mathrm{d}} \tag{4.10}$$

The regions of the layer covered by FR4 are then computed from

$$\mathbf{L}_{\mathrm{FR4}} = \mathbf{1} - \mathbf{L} - \mathbf{F}_{\mathrm{d}} - \mathbf{A}_{\mathrm{PTH}} \tag{4.11}$$

where $\mathbf{1}$ is a matrix of ones with the same size as $\mathbf{L}$. It is possible for some nodes to have $\mathbf{L}_{\mathrm{FR4}} < 0$ indicating overfill, *i.e.* a pixel that contains some plating material but is also full of filler material. These nodes are assigned to be made of FR4.

We can now compute the conductance in the vertical direction as

$$\mathbf{G}_z = \kappa_{\mathrm{Cu}} \mathbf{L} + \kappa_{\mathrm{FR4}} \mathbf{L}_{\mathrm{FR4}} + \kappa_{\mathrm{PTH}} \mathbf{A}_{\mathrm{PTH}} + \kappa_{\mathrm{filler}} \mathbf{F}_{\mathrm{d}} \tag{4.12}$$

It then follows that the half-pixel resistance is given by:

$$\mathbf{R}_z = \frac{l_z}{2 l_x^2} \odot \frac{1}{\mathbf{G}_z} \tag{4.13}$$

where $\odot$ indicates elementwise multiplication.

4.2.6 Validation and verification of the pixel-derived via model

Standalone array of vias

In this section, we evaluate the accuracy of the pixel-derived model for a standalone array of vias commonly used in power electronics applications. Fig 4.8 illustrates a model with nine vias that serves as our test case. Our objective is to compare the results of the pixel-derived model against an analytical model presented earlier in this chapter.

The array of vias is assumed to be placed underneath a surface-mount MOSFET on a printed circuit board (PCB) converter. To calculate the equivalent vertical parallel thermal resistance for one via cell, we refer to the values provided in Table 4.1. Using the values provided in Table 4.1, the equivalent vertical parallel thermal

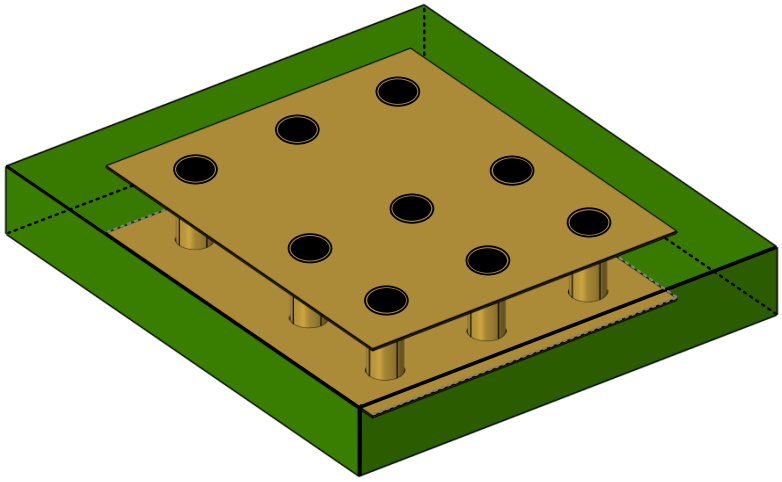


Figure 4.8: Via array used to validate the thermal model for vias. It shows a section of a two-layer PCB with vias filled using solder (black). The dielectric is shown in green

resistance for the 9 vias is computed based on the analytical formulas presented in section 4.1.2. Using the images shown in Fig 4.9 as input to the pixel-based via thermal model proposed in this chapter and applying a discretization correction factor of 1.11, the vertical parallel thermal resistance is also computed using the discrete thermal model. There is excellent agreement between the analytically computed thermal resistance (4.902 K/W) and thermal resistance generated from the discrete via thermal resistance model, such that the later value is within 0.2% of the analytical value at all considered resolutions. This shows that the pixel-derived model of vias is insensitive to the resolution of the downsampled image, and can be used to represent the vertical thermal conduction path of a tile occupied by a via.

Array of vias as part of a simple PCB

The same array of 9 vias is modelled as part of a small PCB in COMSOL and using the bare-PCB thermal model. The images used to build the FDM thermal model are shown in Fig 4.10.

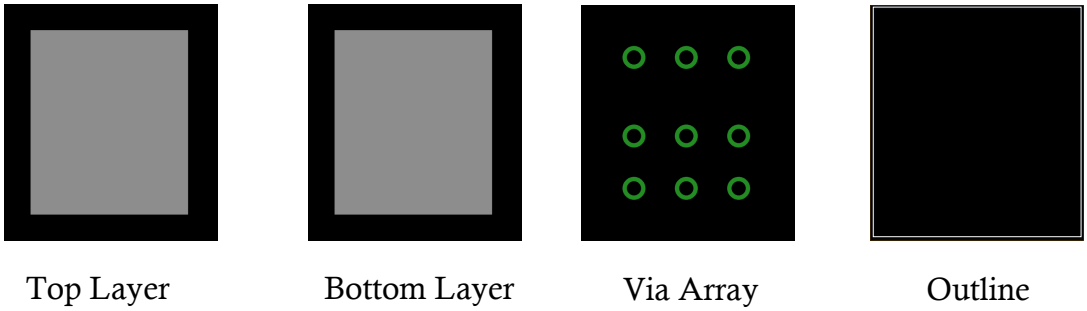


Figure 4.9: Input images to FDM for via verification. Images: 830 × 916, dpi: 2032

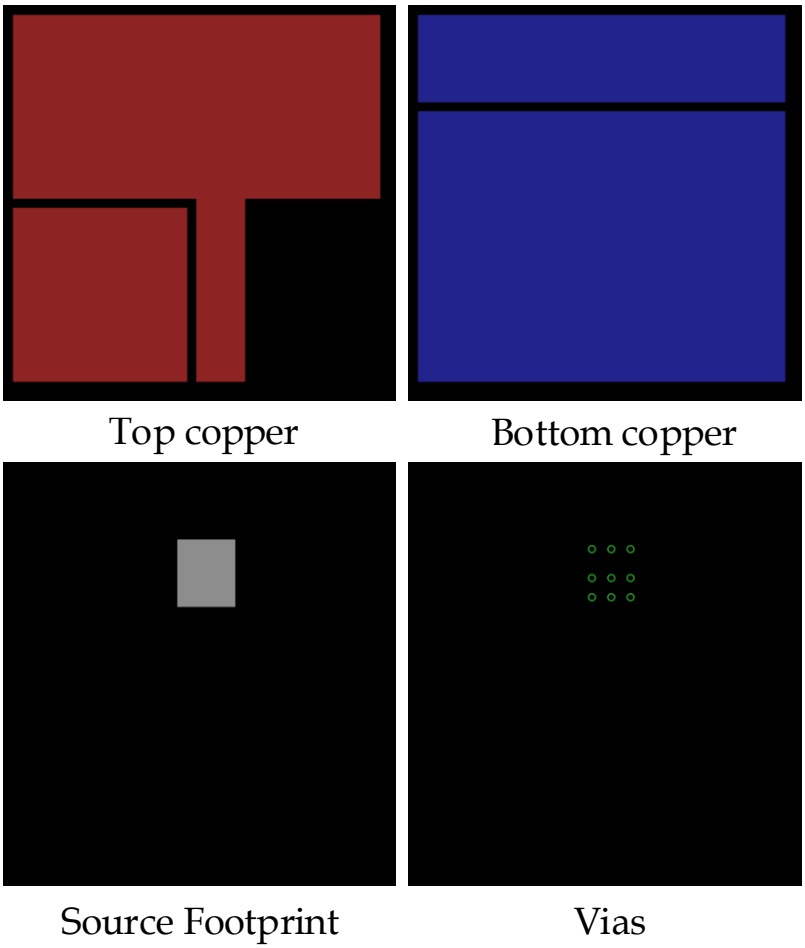


Figure 4.10: Input images to the FDM discretizer for verifying the via model PCB

Property	Value	Units
κ_{Cu}	400.00	$\mathrm{W/m^2K}$
κ_{FR4}	0.30	$\mathrm{W/m^2K}$
l_{cell}	50.80	mm
t_{PTH}	35.00	μm
t_{Cu}	35.00	μm
t_{PCB}	1.60	mm
l	8.9	mm
w	7.6	mm
d_{via}	0.70	mm

Table 4.1: Analytical model parameters

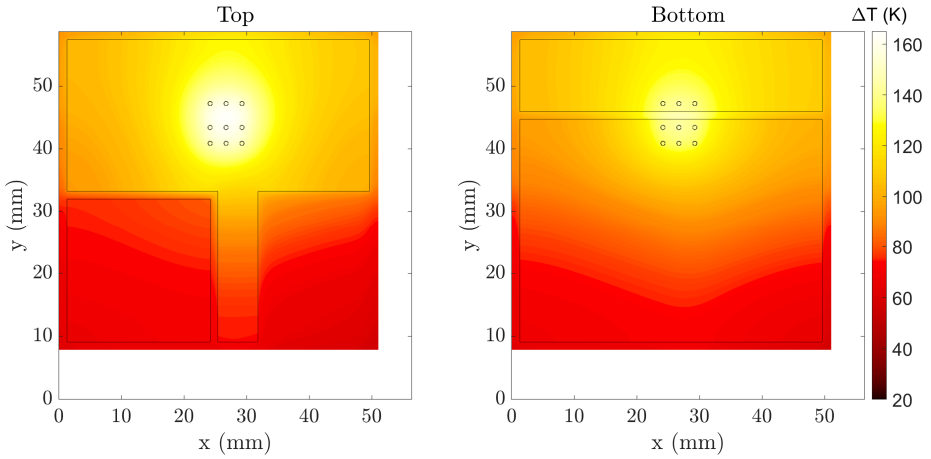


Figure 4.11: FDM top and bottom layer thermal profiles with vias modelled under the source

The same model is constructed in COMSOL and FDM without vias, with identical board dimensions, input parameters, heat source shape and location, and copper pour pattern. The model without vias is used to investigate the impact of vias on the average temperature of the heat source. Fig 4.11 and Fig 4.12 show the top and bottom layer temperature profiles obtained from the FDM code and COMSOL respectively. The thermal maps are near identical on all regions of the board.

Fig 4.13 shows the average error in the temperature of the heat source obtained from the two pixel-derived FDM models compared against identical models solved using COMSOL. Interestingly, the error in the average temperature of the heat

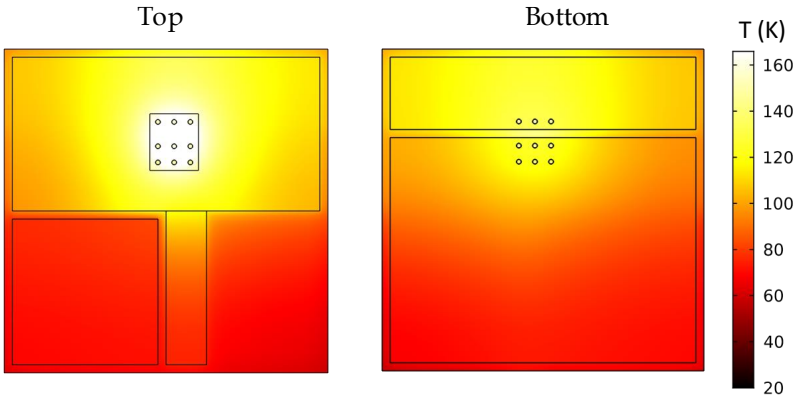


Figure 4.12: COMSOL generated top and bottom layer thermal profiles with vias modelled under the source

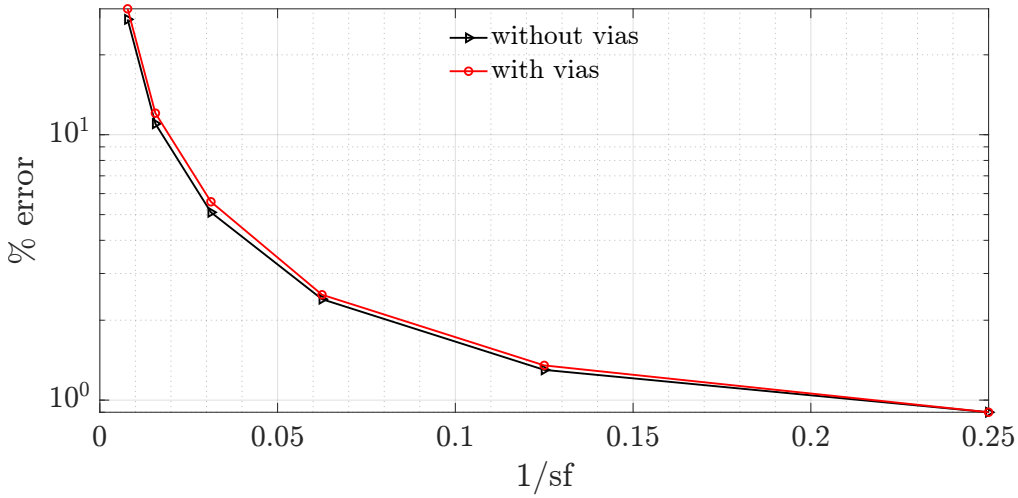


Figure 4.13: Error in the average heat source temperature for two nearly identical FDM models (one with and another without vias) compared against temperature calculated using COMSOL

source is consistent, regardless of the presence or absence of vias. Therefore, the error in the peak temperature of the board at different scale factors is primarily influenced by the resolution of the downsampled image used in the FDM model and not the vias. As the dpi and resolution of the input images are increased the error in the average temperature of the heat source is nearly identical.

To further assess the accuracy of the developed thermal model for vias, a comparative analysis focused on the root mean square error (RMSE) normalized to

the data range of the temperature profiles was conducted. The relevant equations are given below. The RMSE for a system with N nodes is calculated as:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(T_{\text{FDM},i} - T_{\text{COMSOL},i}\right)^2} \tag{4.14}$$

and then normalized using:

$$\text{Normalized RMSE} = \frac{\text{RMSE}}{\text{Data Range}} \tag{4.15}$$

where the Data Rage is given by:

$$\text{Data Range} = \max\left(T_{\text{FDM}}^{\max}, T_{\text{COMSOL}}^{\max}\right) - \min\left(T_{\text{FDM}}^{\min}, T_{\text{COMSOL}}^{\min}\right) \tag{4.16}$$

When utilizing 100% of the computational elements for the top layer, the temperature data spanned a range of 107.7349 K. The normalized RMSE between the COMSOL and FDM results was 0.0099, accounting for 0.99% of the data range. This low percentage indicates a high degree of concordance between the two computational approaches. Specifically, the RMSE at the top surface of the via was calculated to be 1.0669 K.

For the bottom layer the temperature data had a range of 92.6674 K. The normalized RMSE modestly was found to be to 0.0120, representing 1.20% of the data range. The RMSE at the bottom surface of the via was determined to be 1.1151 K.

These findings demonstrate that the via thermal model maintains high accuracy relative to COMSOL benchmarks, with normalized RMSE values remaining close to 1% of the temperature data range in both causes.

The temperature differences between FDM and Comsol are depicted in Fig. 4.14 and Fig. 4.14 respectively. While the majority of the temperature nodes are

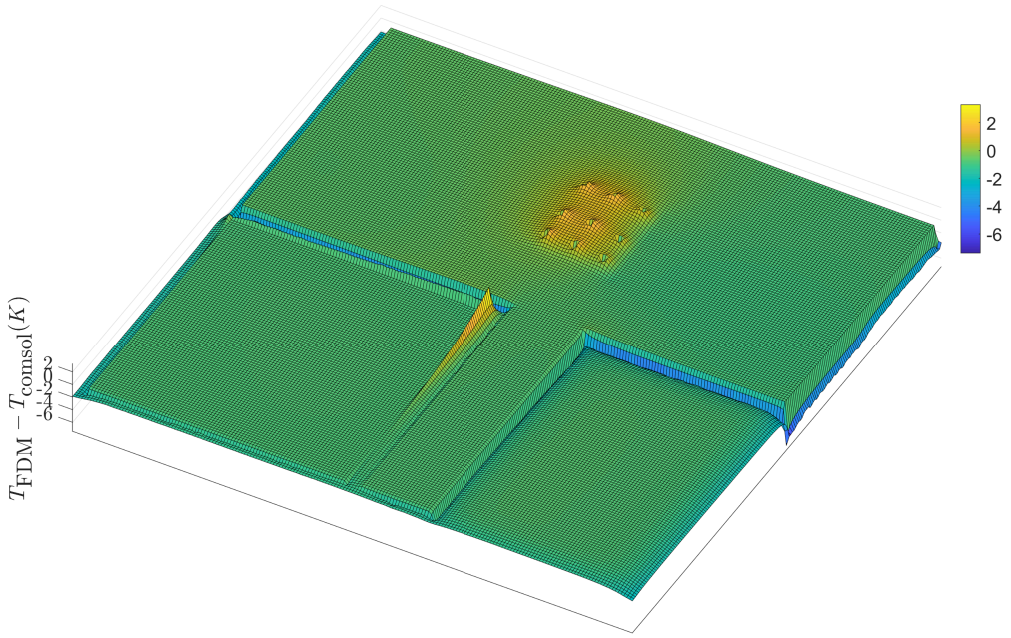


Figure 4.14: Plot of the difference between the top layer temperature profile computed using Comsol and FDM with via model

identical, there are a few outlier nodes concentrated at the FR4-copper interfaces and at the edges of the board. These discrepancies are due to a failure to align the edges exactly at the boundaries where COMSOL placed its edges. The misalignment arises from the use of a scale factor and dpi settings in the FDM model that do not result in the edges falling at their exact vector locations. Even modest deviations in edge placement can lead to dramatic differences in the temperatures computed for the nodes at these edges. However, this discrepancy does not affect the accuracy of the temperatures computed for large regions of copper or FR4, nor does it impact the accuracy of the average source temperature, which is typically the target when investigating potential hotspots on power electronics PCBs.

Notably, the temperatures computed for the nodes at the vias themselves exhibit a slight discrepancy due to the averaging required to map the highly precise

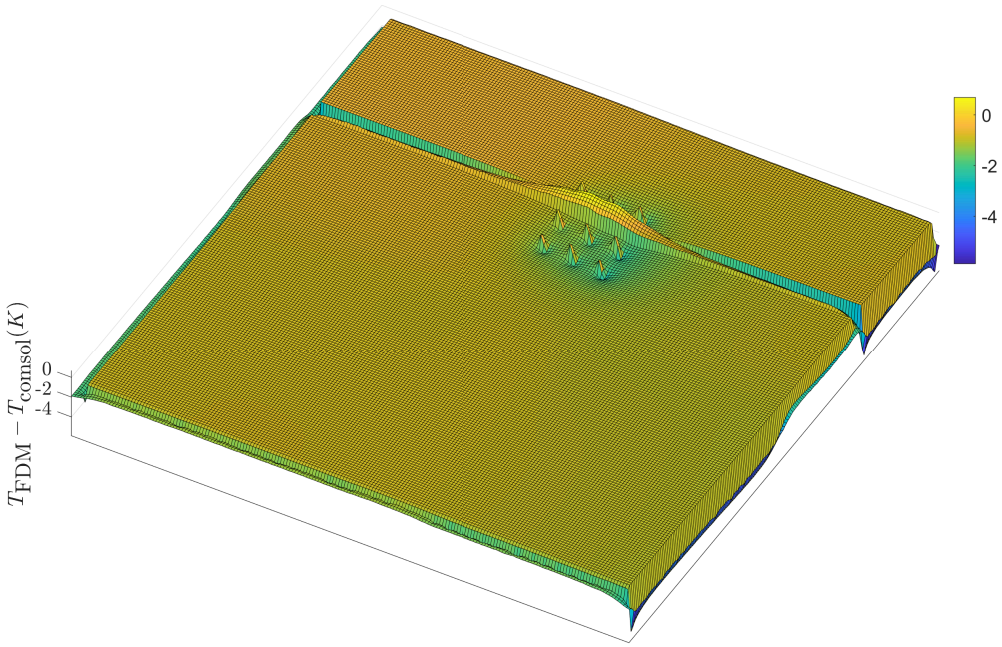


Figure 4.15: Plot of the difference between the bottom layer temperature profile computed using Comsol and FDM with via model

tetrahedral mesh from COMSOL onto the coarse rectilinear grid used to represent the via nodes in the FDM formulation.

4.3 Heat Sources

4.3.1 Motivation for an improved heat source model

In the model of a heat source presented in the last chapter, we assumed that the heat source is distributed equally among all the pixels within the boundaries of its footprint. Consequently, the magnitude of the dissipated power was divided equally among all participating nodes when constructing the **q** vector, while all other nodes were assigned zeros. While this model is simple and suitable as a first approximation, it has several shortcomings when used to model heat sources with complicated footprints such as a MOSFET with disconnected pads and pins for the source and drain of the device. The uniform heat dissipation assumed here

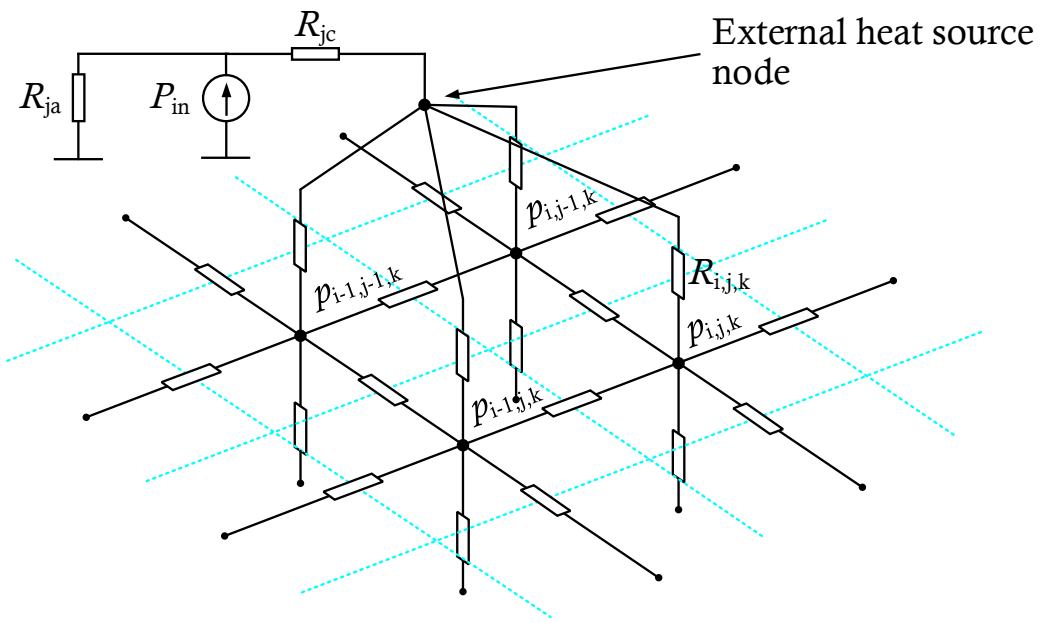


Figure 4.16: Lumped heat source model showing all the pixels occupied by the heat source having a parallel connection to an isothermal 'super-node'

does not hold for some of the smaller pads which may not be connected to large thermally conductive copper pours on the layer. They will tend to overheat as the heat flow would be forced to follow an unrealistic path.

The model also fails to take into account known thermal parameters associated with devices such as junction-to-case and junction-to-ambient resistances. These values are typically readily available in manufacturer data sheets and incorporating them in the thermal model can provide a more realistic representation of heat transfer within devices and across the board. Due to being based on a thermal resistor network, the bare-PCB model proposed in this thesis can be connected to other resistor network thermal models external to the PCB without changing the model's behaviour.

4.3.2 Overview of a detached node heat source thermal model

To address the identified limitations, we propose to model a heat source as a single node that is external to the PCB and has power injected into it. We can think of this node as representing a component with infinite thermal conductivity such that it has a uniform temperature. As shown in Fig 4.16 (a) the node is connected to the pixels associated with the heat source through source-to-node resistances, $R_{i,j,k}$. Each of these resistances represents a lumped thermal path consisting of two parallel thermal paths shown in Fig 4.16 (b). The junction to ambient thermal resistance of the source is parallel to all the nodal resistors and added in blue to the diagram.

It is convenient to describe the connections of all the participating thermal paths from the heat source area to the ambient environment in terms of thermal conductances since they are in parallel, and can be combined by simple addition. For example, the path associated with node (i, j, k) originating from a source pixel $p_{i,j,k}$, consist of the parallel combination of thermal conductances associated with the junction-to-ambient, junction-to-case, vertical half-pixel resistances, and surface-to-ambient convective resistances *i.e.* $1/R_{\text{ja}}$, $1/R_{\text{jc}}$, $1/R_{\text{zc}}$ and $1/R_{\text{hz}}$ masked by the heat source footprint.

4.3.3 Calculating the conductances from heat source to 'detached nodes'

The details for computing all the relevant conductances from the top or bottom layer to all external supernodes follow. To simplify the calculation, we focus on the top layer and assume it has n heat sources. Focusing on a single heat source i, the objective is to compute $\mathbf{J}_i$, the conductance matrix of the i^{th} source. We assume that the following details are known/available:

- the binary indicators of the source area $\mathbf{A}^{[s]}$. Here, $\mathbf{A}_i^{[s]}$ is an $n_x \times n_y$ matrix where all non-zero entries lie within the area of the i^{th} source that overlaps with the corresponding active pixels in the layer,

- $\mathbf{R}_z$, an $n_x \times n_y$ matrix of the half-pixel resistances in the z direction for the layer.

- An $n_x \times n_y$ binary mask, $\mathbf{M}$ indicating which pixels in the layer are within the boundaries of the PCB, computed as described in the image processing section of Chapter 3.

Our goal is to compute the node matrices needed to augment $\mathbf{G}$, the system matrix developed so far, such that it includes the heat source model proposed above.

The first step is to calculate the thermal resistance of the gap pad or source footprint to the i^{th} source, $R_{\text{g},i}$. The calculation depends on whether or not $R_{\text{jc},i}$, the junction to case resistance of the source is available from the data sheet, in which case $R_{\text{g},i}$ is computed using:

$$R_{\text{g},i} = R_{\text{jc},i} \sum_{j=1}^{m} \sum_{k=1}^{m} \mathbf{A}_{i,j,k}^{[s]}, \quad \text{where } i = 1, 2, \ldots, n$$

where j and k are used as indices into the binary indicator matrix $\mathbf{A}^{[s]}$. If instead, the resistivity of the pad, ρ_{pad} is available, from the datasheet, the resistance per pixel is obtained by dividing the resistance per unit area by the square of the size of the pixel (j, k):

$$R_{\text{g},i} = \frac{\rho_{\text{pad},i}}{l_x^2}, \quad \text{where } i = 1, 2, \ldots, n \tag{4.17}$$

Using the value obtained above, we can now compute $\mathbf{J}_i$, an $n_x \times n_y$ matrix of conductances of the i^{th} source into each pixel in the layer:

$$\mathbf{J}_i = \frac{\mathbf{A}_i^{[s]} \odot \mathbf{M}}{\mathbf{R}_z + \mathbf{R}_{\text{g},i}} \tag{4.18}$$

where $\odot$ is the element-wise multiplication operator.

After computing $\mathbf{J}$ for all n sources on the layer, we can now construct $\mathbf{E}$, an $n \times n$ diagonal matrix where the i^{th} diagonal entry $\mathbf{J}_i$ is the conductance of that source into the layer under analysis.

$$\mathbf{E} = \begin{bmatrix} \mathbf{J}_1 & 0 & \dots & 0 \\ 0 & \mathbf{J}_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{J}_n \end{bmatrix} \tag{4.19}$$

Next, we compute $\mathbf{S}_i$, which is a vector obtained by vectorizing $\mathbf{J}_i$, and represents the i^{th} source node stitch matrix

$$\mathbf{S}_i = \text{vec}(\mathbf{J}_i) \tag{4.20}$$

By summing all the elements in $\mathbf{J}_i$, we also compute d_i, the conductance out of the i^{th} source node

$$d_i = \sum_{j=1}^{n} \mathbf{J}_{i,j} \tag{4.21}$$

Finally, we now have the additional nodal matrices needed to augment $\mathbf{G}$ to incorporate the external heat source node and the thermal paths connecting to it. Concisely, these values are given by the $\mathbf{S}$ and $\mathbf{d}$ vectors below.

$$\mathbf{S} = [\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_n] \tag{4.22}$$

$$d = [d_1, d_2, \dots, d_n] \tag{4.23}$$

The next section discusses the details involved in creating the overall system matrix.

4.3.4 Adding the nodes to G

Once the matrices and vectors required to modify the $\mathbf{G}$ matrix are obtained, the nodes they represent must be added to the system matrix. Let $\mathbf{G}$ be the $n_x n_y \times n_x n_y$ system matrix derived in Chapter 3, such that that $\mathbf{G}_{i,j}$ is an $n_x \times n_y$ conductance matrix for a given layer. Assuming that there are n sources on the layer, thus n

nodes to be added to $\mathbf{G}$, $\mathbf{S}$ is an $n \times 1$ stitch vector computed in the preceding section and d is an $n \times 1$ vector of conductances out of the source.

Now, assuming $\mathcal{N}$ is a set of indices such that $\mathcal{N} = \{n_z + j : j = 1, ..., n\}$ we have,

$$\mathbf{G}_{i,j} = \begin{cases} d_j & \text{if } i = j \in \mathcal{N} \\ -\mathbf{S}_j & \text{if } i = n_c \in \mathcal{N}, j \in \mathcal{N} \\ \mathbf{0}_{n_n \times 1} & \text{if } i < j \leq n_z \\ 0 & \text{if } i < j > n_z \\ \text{unchanged} & \text{otherwise} \end{cases} \tag{4.24}$$

where n_c is the index on the layer to be connected.

4.4 Heat Sinks

An improved model of the heat sink is also required to address the limitations of the simple pixel-distributed heat sink presented in the last chapter. In this section, we propose a new model that, like the heat source model presented previously, includes data sheet thermal resistance values and accommodates an arbitrary number of arbitrarily shaped heat sink footprints attached to the PCB.

We formulate a heat sink as a supernode using similar reasoning as was done for heat sources. In this model, the heat sink is represented as a node that is external to the PCB resistor network with various thermal paths originating from pixels that lie within its footprint on the board. For example, one such thermal path can be made from a parallel combination of thermal conductances representing the thermal interface material, all the half-pixel resistances of the portion of the PCB layer directly underneath the heat sink, and the data sheet provided thermal resistance of the heat sink itself.

The analysis presented for heat sources can be repeated to deduce the nodal matrices needed to augment $\mathbf{G}$ by adding an arbitrary number of nodes representing

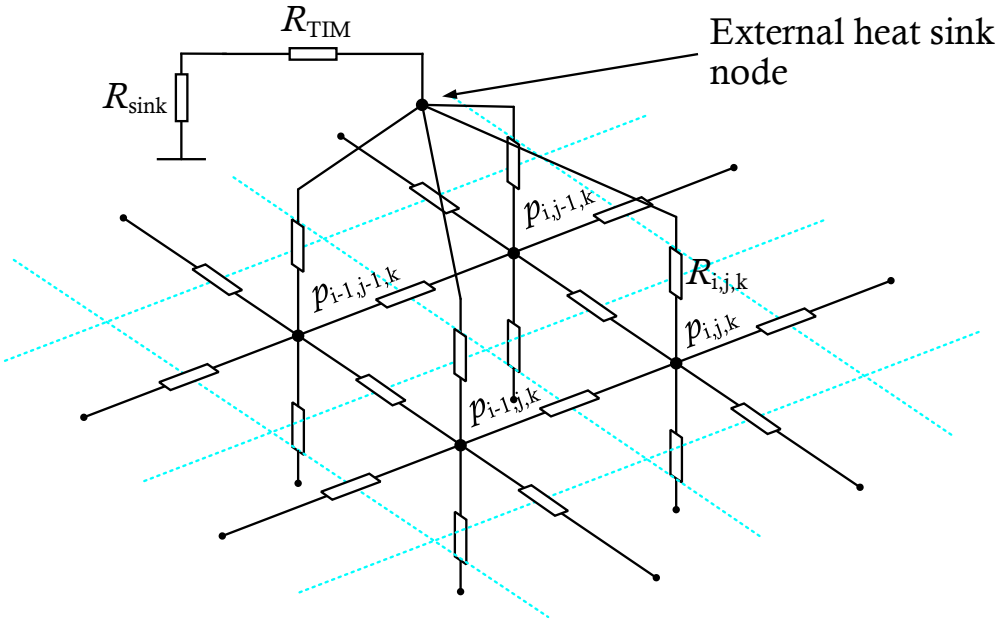


Figure 4.17: Lumped heat sink model showing all the pixels occupied by the heat source having a parallel connection to an isothermal 'detached-node'. The placement of the heat sink and TIM is shown as lumped resistances obtainable from manufacturer data sheets following validation by the supplier

heat sinks on either the top or bottom sides of the PCB. As before, the stitch vector (called **H** for a heat sink) and the total conductances to the super node (also assigned to d) can be calculated by following the steps described in the last section.

4.5 Assembling G

The system conductance matrix obtained in Chapter 3 can now be modified by adding the heat source and heat sink nodes. The resulting system has the format **GT=q** where the conductance coefficient matrix **G**, with additional heat source and heat sink nodes, now has the form:

$$
\mathbf{G} = \begin{bmatrix}
\mathbf{G}_{1,1} & -\mathbf{M}_{1,2} & & & & & & -\mathbf{S}_1 & \cdots & -\mathbf{S}_n & \\
-\mathbf{M}_{1,2} & \mathbf{L}_2 + \mathbf{M}_{1,2} + \mathbf{M}_{2,3} & -\mathbf{M}_{2,3} & & & & & & & & \\
 & -\mathbf{M}_{2,3} & \mathbf{L}_3 + \mathbf{M}_{2,3} + \mathbf{M}_{3,4} & -\mathbf{M}_{3,4} & & & & & & & \\
 & & \ddots & \ddots & \ddots & & & & & & \\
 & & & -\mathbf{M}_{k-1,k} & \mathbf{G}_{k,k} & & d_1 & & & & -\mathbf{H}_{\mathrm{k}}^T \\
 & & & & & & & \ddots & & & \\
 & & & & & & & & d_n & & \\
 & & & & & -\mathbf{H}_{\mathrm{k}}^T & & & & d_k & \\
-\mathbf{S}_1^T & & & & & & & & & & \\
\cdots & & & & & & & & & & \\
-\mathbf{S}_n^T & & & & & & & & & &
\end{bmatrix} \tag{4.25}
$$

where $\mathbf{S}_1 \ldots \mathbf{S}_n$ and $\mathbf{H}_k$ are sparse conductance vectors whose nonzero entries represent nodes corresponding to pixels that are connected to the source and sink nodes respectively. The number of nonzero entries in each column of $\mathbf{S}_1$ and $\mathbf{S}_n$ is equal to the number of pixels under the footprint of the heat source. The size of this matrix is $n_x n_y \times 1$. $\mathbf{H}_k$ is an $n_x n_y \times 1$ sparse vector. There are n_h such vectors in a system where the number of heat sinks on layer n_h. For simplicity, we have assumed that heat sinks are present on the bottom (*i.e.* k^{th}) layer only.

As before, $\mathbf{G}$ is symmetric and positive semi-definite, and can thus be solved using an iterative solver. However, these modifications increase both the bandwidth and condition number of $\mathbf{G}$ meaning that a preconditioner has the potential to greatly improve the convergence rate of the iterative solver. This fact motivates the choice of solver, in this case, MATLAB's pcg iterative solver. It indeed performs better than the direct methods available through the backslash operator.

4.6 Validation of the improved thermal models

4.6.1 Description of Case Study PCB

A four-layer PCB designed for a power electronics application is chosen as a case study. The input images are shown in Fig 4.18, with the images of the top, bottom and middle layers, heat sink, vias and pins (which include filled vias) shown. Fig 4.19 shows the heat source footprints modelled for the board. They correspond to pads for surface mount MOSFETs and pads corresponding to two ICs on the top layer. Other passive components such as surface mount capacitors are regarded as having negligible heat dissipation, which is consistent with contemporary engineering practice.

The input images have a resolution of 5501×5572 pixels at 2032 dpi. The parameters used in the model are listed in Table 4.2. The dissipation and junction

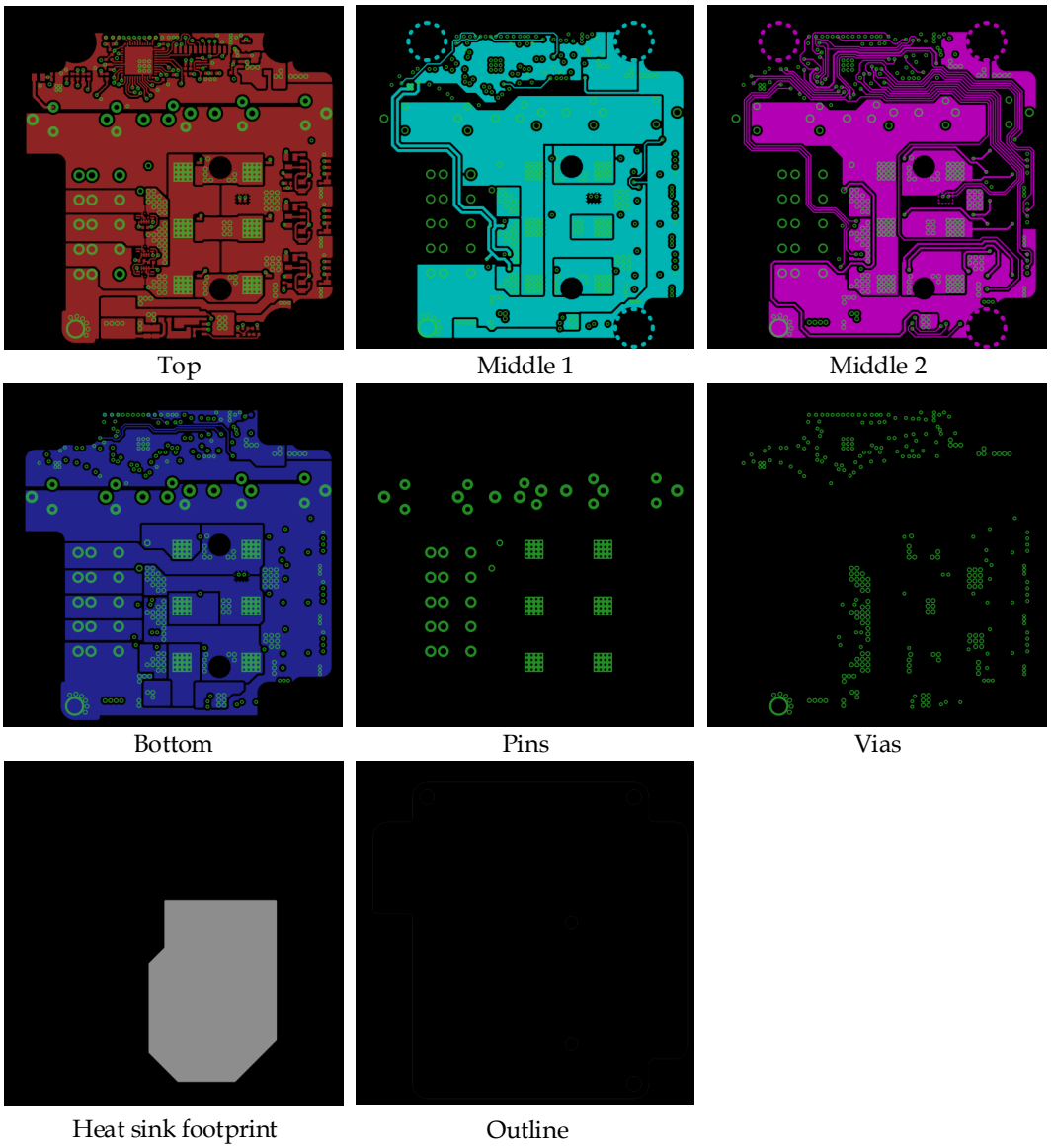


Figure 4.18: Input images of the layers for modelling the PCB. Courtesy of Prof. D Rogers

Quantity	Value	Units
κ_{Cu}	400	W/mK
κ_{FR4}	0.3	W/mK
h_{ambient}	10	W/Km^2
h_{sink}	15	W/Km^2
ρ_{pad}	1.11×10^4	W/mK
dpi	2032	-
n_z	19	-

Table 4.2: Parameters for testing the full-featured board

thermal resistances associated with all the heat sources are given in Table 4.3. The heat source $R_{\theta,\mathrm{jc}}$ are obtained from the data sheets of the devices.

Heat Source	Device	Power (W)	R_{jc} (K/W)
1	Power MOSFET	1.50	0.10
2	Power MOSFET	1.50	0.10
3	Power MOSFET	1.50	0.10
4	Power MOSFET	2.25	0.10
5	Power MOSFET	2.25	0.10
6	Power MOSFET	2.25	0.10
7	IC	0.50	3.00
8	IC	0.50	3.00

Table 4.3: Results from modelling the thermal profiles at various scale factors

4.6.2 Obtaining the ground truth benchmark temperature

This full-featured PCB is both too large and too complex to model in COMSOL in a reasonable time. Therefore in order to assess the performance of the full-featured model, we obtained the ground truth benchmark reference temperature profile by solving the problem at the highest resolution possible using the pixel-generated thermal model that met the following conditions:

1. The smallest feature was represented by at least two pixels

2. All gaps between FR4 and copper were maintained

Treating this as the ground truth benchmark temperature is reasonable because we have shown excellent agreement between COMSOL and FDM at small-scale factors. The two-layer PCBs used to validate the FDM model in Chapter 3 and the pixel-derived via thermal model in this chapter are representative of the features of interest on a power electronics PCB, with errors in the average temperature of the heat source not exceeding 1 % when the board is modelled in sufficient detail. The benchmark temperature profile is shown in Fig 4.20, for the problem solved at a scale factor of 3, computed in 20 minutes on an Intel core i7 64-bit

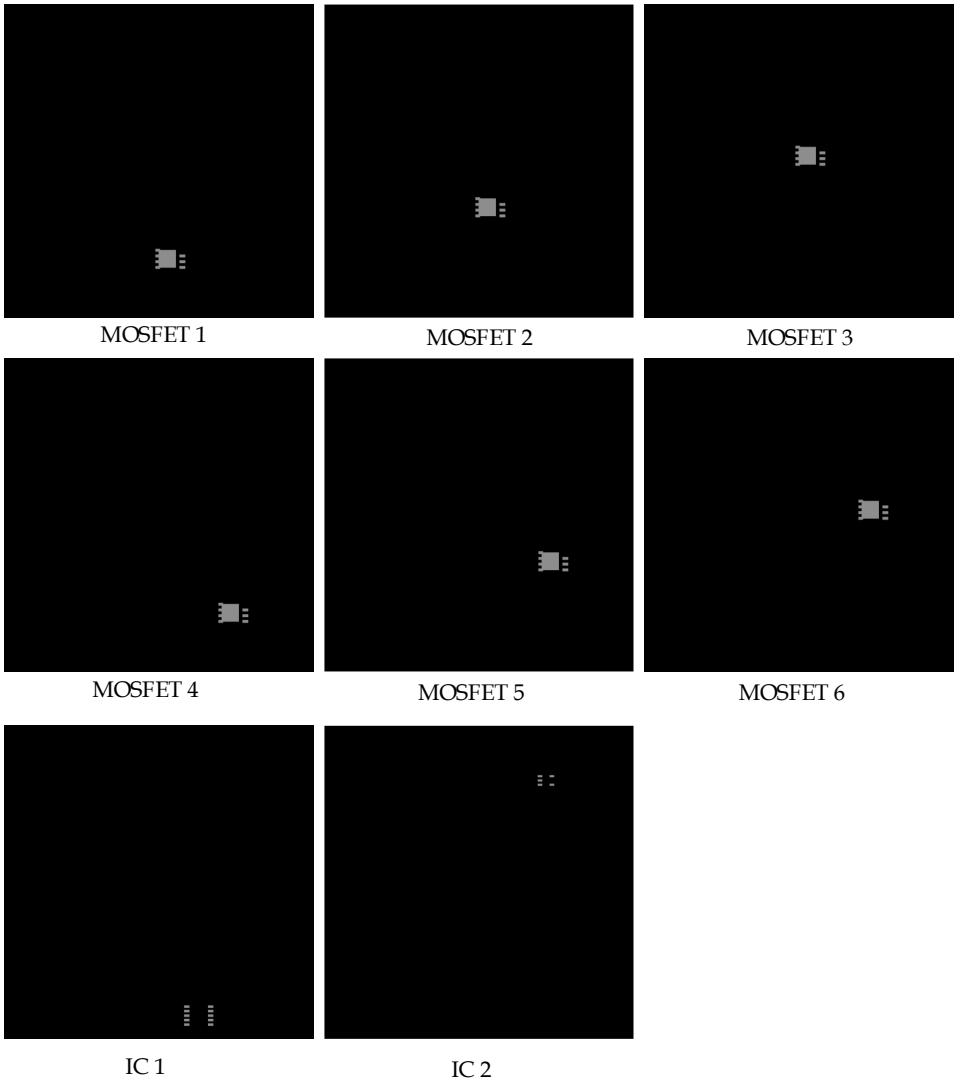


Figure 4.19: Input images of the heat sources present on the PCB

desktop. The target average temperature change of each heat source is shown in Table 4.4. Heat source 8 experiences the largest ΔT of 78.01 K, which is reasonable for the dissipation considered.

4.6.3 Numerical experiments results

With the benchmark temperature profiles established, the model is then solved at different scale factors to study the accuracy of the computed temperature profiles as large pixels are used. Table 4.5 shows the solution times, number of elements

Heat Source	ΔT (K)
1	47.95
2	48.36
3	46.59
4	55.45
5	55.31
6	54.25
7	63.62
8	70.18

Table 4.4: Average temperature rise of the heat source

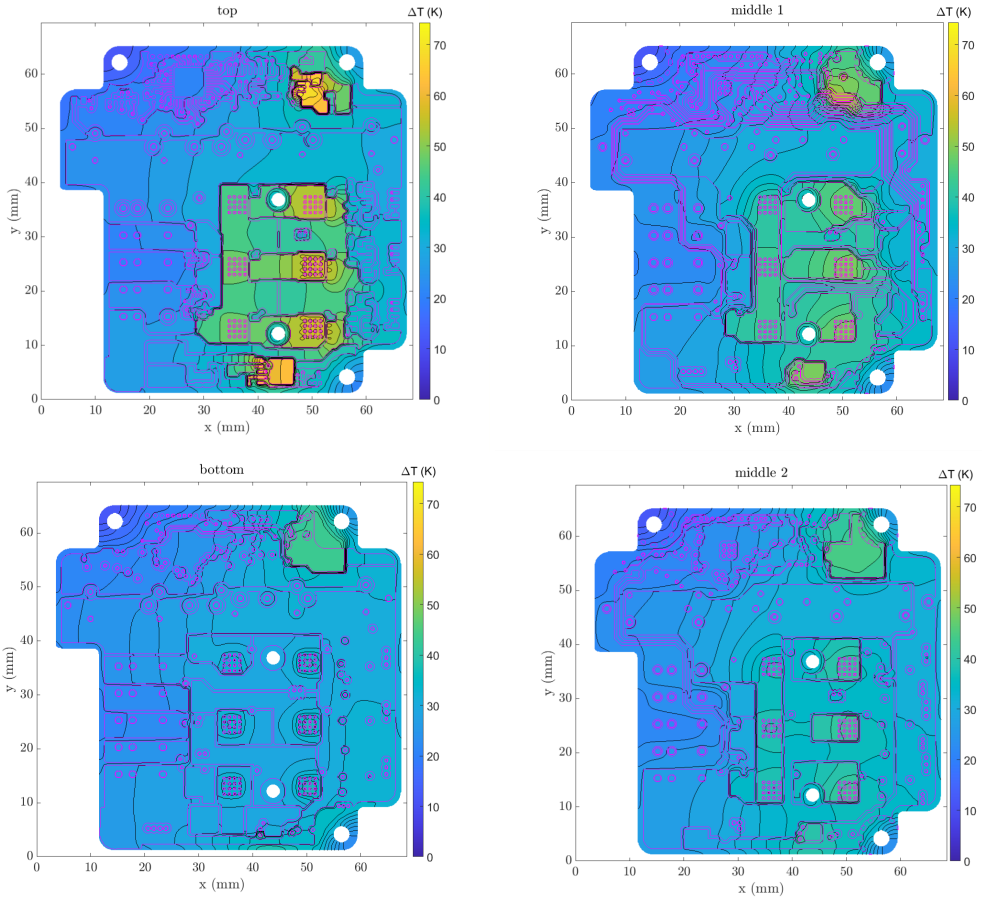


Figure 4.20: Top, middle and bottom layers thermal profiles for a four-layer power electronics board

Scale Factor	Number of Elements ($\times 10^6$)	Solutions Time (s)	Pixel Size (mm)
3	48.20	1678.0	0.0375
4	27.10	807.1	0.0500
6	12.10	282.7	0.0750
8	6.79	133.8	0.1000
16	1.71	24.1	0.2000

Table 4.5: Results of the full-featured PCB simulation at different scale factors

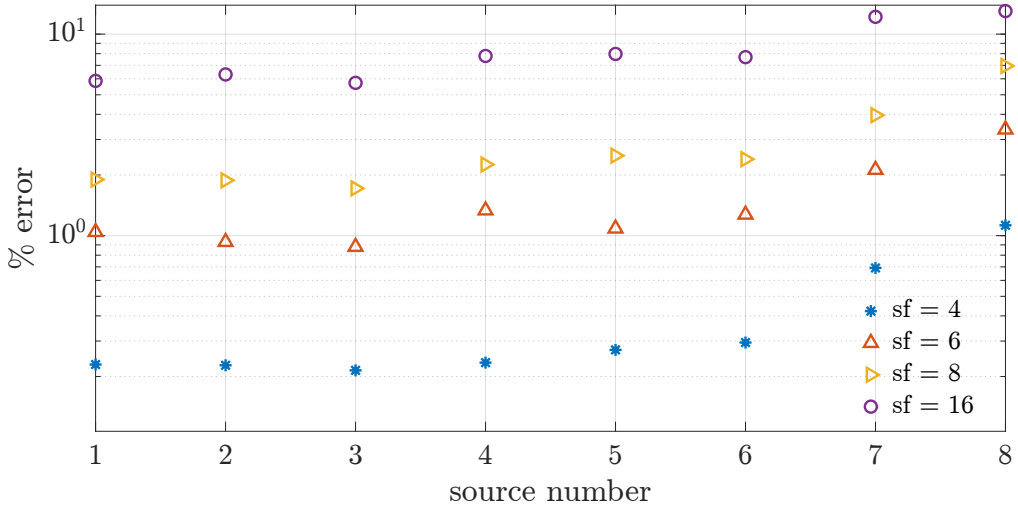


Figure 4.21: Errors associated with sources

and pixel sizes observed at the selected scale factors. The smallest scale factor used is 3, beyond which a top-end desktop PC (64-bit Intel core i7) runs out of memory and does not finish solving the problem. The average heat source temperature rise at each resolution is compared to the value predicted by the solution obtained using a scale factor of 3.

As can be seen in the table solving the problem using lower-resolution discretizations of the board leads to shorter solve times and a reduction in the number of elements accompanied by a sacrifice in the accuracy of the solution. Fig 4.21 illustrates this result and shows the error in the average temperature of each heat source at different scale factors. The average temperature of each heat source ranges from below 1 % at a scale factor of 4, and increases to a maximum of 12% when the scale factor is increased to 16. Consequently, a PCB layout engineer can gain insight into the thermal impact of their design choices in near real-time. Tolerances

of $\pm$ 5 % are normally acceptable in thermal design of power electronics. The choice of the acceptable tolerance is somewhat application dependant. A tight tolerance (for example $<$ 1 %) is not typically expected in idealized numerical models such as the one considered here which simplifies the representation of of the PCB. For example, details of airflow over the board could easily change the temperature rise by more than 5% or more, which we do not model in this context.

4.7 Conclusions

In this chapter, we have proposed a method for obtaining a thermal model for vias, which are typically challenging to represent accurately in numerical simulations due to their small size. This is achieved without resorting to the use of excessively high numbers of nodes in the grid. We have demonstrated that a discretization correction factor is necessary when calculating the perimeter and area of circular features from pixels and provided an analytical expression for computing the correction factor based on the number of pixels along the diameter of the via in the input image.

We proposed 'detached node' models for heat sources and heat sinks, that enable the addition of thermal resistances available in manufacturer data sheets to the bare-PCB thermal model. The ideas presented in this chapter are suitable for inclusion as a step in a PCB iterative design loop where the thermal response produced in near real-time can be used to inform the new layout strategy of the PCB. The speed of the method is attractive for informing the PCB layout following changes in every iteration. Such an electro-thermal co-design loop would be costly to model the board using mainstream FEA tools.

5

Improving the performance of pixel-derived thermal models through downsampling by tiling

5.1 Introduction

So far binary downsampling has been used to reduce the number of pixels in the input layer image, and thus reduce the computational effort required to solve the resulting thermal model by FDM. In this approach, an appropriate scale factor is used to generate a binary image of the conducting layer where each pixel is assigned homogeneous thermal properties. Methods such as the 'box' algorithm available in MATLAB's `imresize` routine have been used to generate downsampled images where each pixel assumes the thermal properties of the most abundant material in the tile from which this pixel is built. In this context, a tile is a 2D array of pixels in the high-resolution input image that is downsampled into a single pixel whose properties must approximate the thermal path represented by the tile. It follows that the original input layer image is then composed of a 2D array of tiles, the number and size of which are chosen in order to closely approximate the properties and layout of the original image.

When using binary images to generate FDM discretizations, it is necessary to maintain the general layout of each layer, by ascertaining that there are clear segments between the dielectric and conductive sections in order to prevent thermal short-circuiting. If this condition is met, then the overall approximate thermal paths on the board are enforced at the lower image resolution. This constraint means that the minimum size of a pixel in a thermally accurate downsampled binary image maintains a pixel size that is at least half the size of the minimum feature length on the board.

It is common for gaps between copper and FR4 sections on a layer of power electronics PCB to be as small as 0.2 mm (not considering the plating thickness of vias). As a result, in order to adhere to this constraint, pixel sizes of 0.1 mm are required in order to maintain the gaps between FR4 and copper. For example, a square board with a length of 5 cm and a minimum feature size of 0.2 mm requires 5 cm/0.1 mm pixels to represent the length of one side in a thermally accurate manner. The PCB thus requires 250,000 nodes per layer or 1,750,000 nodes if the system is discretized using 7 vertical layers, leading to a system matrix that can be inverted in less than 30 seconds on a modern CPU. If smaller features are present, for example, thin copper tracks or more vertical layers are required in the vertical discretization, then such a board will require a large number of elements to be modelled accurately.

This discretization approach stops being attractive if downsampled images with large pixel sizes are required in order to improve computational efficiency. In this scenario, the pixels in the downsampled image are larger than the smallest feature size on the original layer which, in turn, means the image generated using binary downsampling will have poorly defined or non-existent boundaries between FR4 and copper, thus thermal short circuits. As a result, representing a tile that contains a mixture of materials as having the thermal properties of the material with the greatest fraction of the composition leads to an inaccurate geometry.

This chapter considers the case where a tile of pixels is treated as having directional thermal conductivities (κ_x, κ_y, and κ_z) that depend on the thermal paths present in all three orthogonal directions within the tile. An analytical method for obtaining these thermal conductivity values is proposed. A downsampled image that accurately represents the anisotropic thermal behaviour of a layer at low resolutions results, enabling the average thermal paths available within a pixel to be modelled accurately.

The chapter begins by reviewing the methods of obtaining directional thermal conductivity available in the literature and identifying areas of improvement for robust computation of these parameters. A mathematically rigorous approach of computing the parameters from the dimensions of the tile and the configuration of the pixels is proposed and verified to be numerically accurate.

5.2 Tiling PCBs in the literature

The idea of replacing highly anisotropic thermal conductivities of a complex multi-material geometry with simpler analytically derived values has been an attractive proposition in PCB thermal modelling literature. Between the 1990s and early 2000s, the concept of using an "effective thermal conductivity" for PCB thermal modelling was dominant and driven by the limits of the available computational power at the time which dictated that only small models could be solved [17] [59] [99]. The basic notion was that a multi-material geometry such as a PCB can be modelled as a lumped material whose orthotropic thermal conductivity can be calculated by averaging ratios of the materials present within the domain. One of the most common formulas for averaging the thermal conductivity of the PCB, popularised in the 1990s is:

$$\kappa_{\text{eff}} = \frac{\sum_{n=1}^{N} \kappa_{\text{Cu}} f_n t_n + \sum_{n=1}^{N} \kappa_{\text{FR4}} (1 - f_n) t_n + \sum_{m=1}^{N} \kappa_{\text{FR4}} t_m}{\sum_{n=1}^{N} t_n + \sum_{m=1}^{N} t_m} \tag{5.1}$$

where f_n is the fraction of copper in the board, and the other variables have their usual meaning.

In [97], Sampson showed that this simplistic approach is wrong and proposed using bitmap files of the layers of a PCB to compute the properties of the board layer by layer. To date, Sampson's approach is perhaps the most mathematically rigorous method available in the literature on image-based thermal modelling of PCBs. By accounting for both the distribution and quantity of copper using KVL circuit analysis, his approach was more accurate and a departure from the use of κ_{eff}. Numerical results within 5% of empirical observations were achieved, a massive improvement from the errors in the temperature profiles that were being reported at the time (greater than 50%). However, the author used only one node per vertical layer which is known to be too coarse a discretization and insufficient to capture the heat flow where heat spreading and large thermal gradients are present, for example on regions directly under a heat source. The approach also did not offer clarity on how the thermal conductivity in the vertical direction could be calculated with vias present and did not provide a model for heat sources. Lastly, holes, solder and other materials were also not considered. These features are known to impact the temperature profile of a PCB significantly.

Although the idea of effective thermal conductivity has been shown to be inaccurate, it has still appeared in a number of publications over the years, including in some recent papers, for example in [13] [54] [79] [103].

Building on the ideas in [97], the authors of [7] proposed an image-based approach of computing the thermal conductivity map of a PCB based on processing tiles to generate a model that was then solved using the finite volume method. The authors demonstrated a method of extracting black-and-white layer images from EDA tools and decomposing them into a tiled array. The local directional thermal conductivity of each tile was then computed by counting pixels and used to generate a tessellated thermal conductivity map of a layer. Using the percentage

of copper bands within a tile, the author proposed a heuristic that determines a weighting of the computed directional thermal conductivity. An area-weighted average of the thermal conductivity of the conductor was then used to assign the thermal conductivity of a given tile.

Because this approach has gained acceptance in contemporary literature on the topic of computing thermal resistances from tiled images, for example, as cited recently in [13] and [54], we review it in detail to identify its shortcomings. First, as a processing step for a tile, the authors assume that each horizontal row of pixels is thermally in parallel with other rows and that the pixels within a row are in series. A similar argument is made for the columns of pixels within the tile. While a possibly useful heuristic, this assumption results in an overly simplistic and mathematically inaccurate thermal resistor network from which the thermal conductivity of the tile is computed in the x and y directions.

To illustrate, consider an $m \times n$ tile where the i^{th} row contains n_c copper-covered pixels. The paper proposes that the thermal resistance of this row must be computed as follows:

$$R_i = \frac{n - n_c}{\kappa_{\text{FR4}} A_i} + \frac{n_c}{\kappa_{\text{Cu}} A_i} \tag{5.2}$$

where A_i is the total area of all the pixels in the row, given by $A_i = n l_x l_y$. Assuming that the rows of pixels are in parallel, their combined parallel thermal conductance is then:

$$\frac{1}{R_x} = \sum_{i=1}^{m} \frac{1}{R_i} \tag{5.3}$$

and the effective thermal conductivity in the x direction is:

$$\kappa_x = \frac{l_x n_c}{A_{\text{tile}}} \sum_{i=1}^{m} \frac{1}{R_x} \tag{5.4}$$

Although simple and easy to compute, obtaining the thermal resistance in this way is not rigorous because the assumption that rows of pixels (and the accompanying

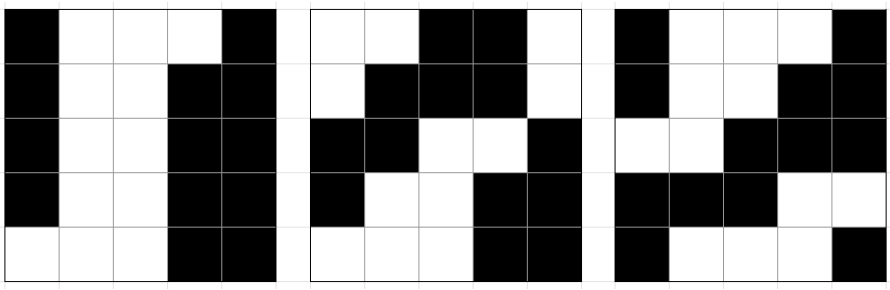


Figure 5.1: Pixel grid examples showing different patterns that lead to the same thermal resistance using Blackmore's calculations

half-pixel resistors) are thermally in parallel does not correspond to reality. In fact, the heat flux will tend to follow the path of least thermal resistance, regardless of whether the pixels forming this path are in adjacent rows of pixels. Therefore, there is no strong electrical equivalency for the assertion.

Secondly, using the assumptions presented in [7], pixel patterns that represent drastically different physical thermal paths can generate equivalent orthotropic thermal resistances in the xy plane. For example, consider the three pixel grids shown in Fig 5.1. Each grid is a 5×5 tile with 12 white pixels and 13 black pixels configured in three different ways that are plausible for traces on a real PCB.

Using the calculation process of [7], all the tiles have the same R_x values. Consequently, while determining the equivalent thermal resistance in the x direction this way takes into account the volume fraction of the copper content, it fails to account for the orientation or the physical spatial distribution of the copper within the layer, which is the true determinant of the thermal path and temperature behaviour.

We propose a rigorous method of calculating the in-plane orthotropic thermal conductivity using the FDM that is capable of modelling thermal paths more accurately. Using our method, the copper patterns shown in Fig 5.1 have drastically different thermal properties that reflect the true thermal resistance in both the x and y directions.

Moreover, our approach models vias better than the standard approaches in the literature, as it extends the tiled modelling of vias presented in Chapter 4,

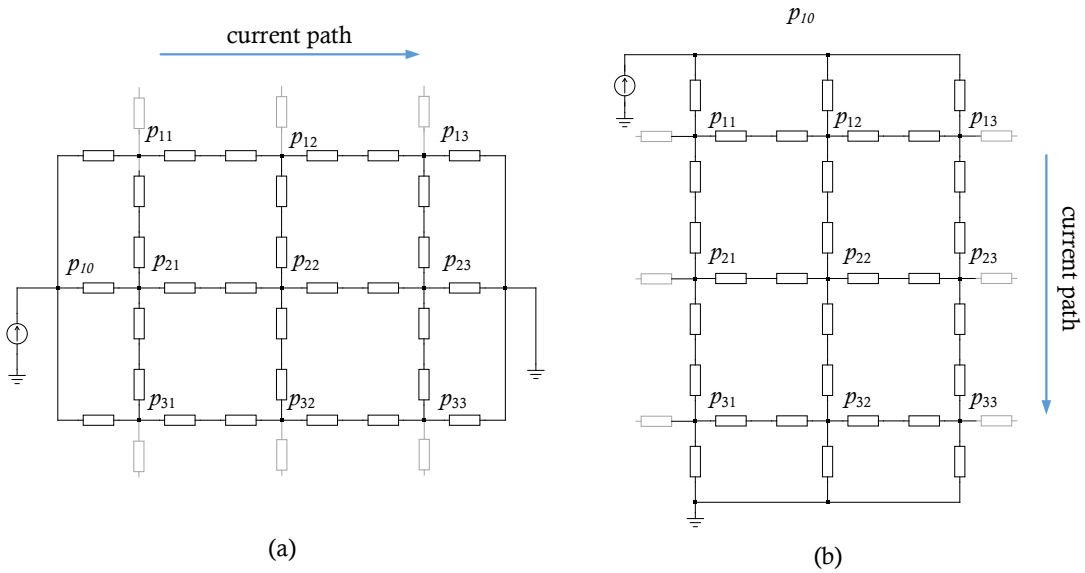


Figure 5.2: Equivalent half-pixel resistor networks for a 3×3 pixel grid used to compute the directional thermal conductivity. In this context, thermal ground refers to an arbitrary known reference temperature. Notice the additional node due to the current injection (a) Power flow path used in the computation of the equivalent $R_{\theta,x}$ resistance (b) Power flow path used in the computation of the equivalent $R_{\theta,y}$ resistance

where the arc length of a via was calculated for each tile, and used to determine the thermal resistance of the tile in all three directions.

In the remainder of this chapter, we present the mathematical derivation of the directional thermal resistances ($R_{\theta,x}$, $R_{\theta,y}$ and $R_{\theta,z}$) and by extension, reduce the number of dpi in an image by using pixels that are larger than the minimum feature size without greatly reducing the accuracy of the thermal model.

5.3 Analytical computation of in-plane directional thermal conductivity

Using the resistor network model of a pixel introduced in chapter 3, a 2D pixel array leads to a complicated thermal resistor network whose resistance in the x and y directions can be determined using nodal analysis. The thermal resistance can be obtained by injecting thermal power into one node and attaching the other to

thermal ground, then measuring the temperature drop between the two points. For example, consider a 3×3 pixel grid, represented by the half-pixel resistor networks shown in Fig 5.2. The thermal resistance of the network in the x direction, $R_{\theta,x}$, can be computed by injecting known thermal power into the leftmost node while keeping the rightmost node grounded as shown in Fig 5.2 (a). It is then possible to calculate the temperature drop between the two. $R_{\theta,y}$ (Fig 5.2. (b)) can be computed using a similar approach *i.e* injecting a known power into the top node, grounding the bottom one and calculating the temperature drop. The half-pixel resistors that would be aligned the y direction at the top and bottom of the network in Fig 5.2 (a) are assumed to be 'hanging', thus dropped from the calculation, since they effectively provide infinite thermal resistance to power flowing in the network. The thermal power can flow freely in either the x or y direction through any of the non-hanging resistors it encounters. Unlike previous methods in the literature, this approach of computing the thermal resistance imposes no artificial limits on the path that the heat flow can follow, thus more closely resembles reality.

While a number of methods of solving this system exist, for example by factoring the matrix using the single value decomposition, we solve it using the finite difference method, which is both consistent with the mathematical methods we have adopted so far. Using FDM also allows us to continue thinking in terms of the resistor networks that electrical engineers are accustomed to. As before, we write a nodal equation at each node that imposes a power balance, such that the power entering a node is equated to the power leaving it. Performing this systematically for each node yields a system of simultaneous equations similar to those derived for a 2D layer in chapter 3, which can also be assembled into a sparse system. The only important difference is that the sparse system of conductances for a tile $\mathbf{D}$ has two additional rows representing the injected current node and the grounded node (which we drop because its temperature is zero). It follows that for each tile considered we obtain a system of the form:

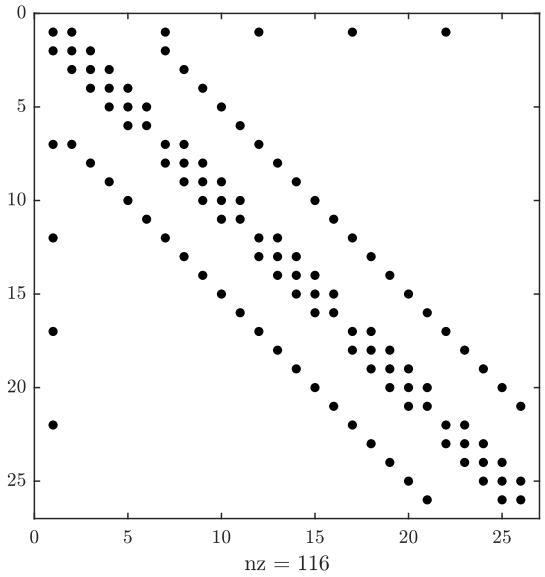


Figure 5.3: Spy plot showing the sparsity pattern of the conductance matrix used in computing orthogonal thermal conductivities. Note that it maintains the banded nature of the general layer matrix starting from a node $(1, 1)$ but with additional entries in the first row and column corresponding to all nodes that share a path with the added power injection node. This has the impact of increasing the bandwidth of the matrix

$$\mathbf{G}_{\text{tile}}\mathbf{T}_{\text{tile}} = \mathbf{q}_{\text{tile}} \tag{5.5}$$

where $\mathbf{G}_{\text{tile}}$ is the sf $\times$ sf conductance matrix of the tile, $\mathbf{T}_{\text{tile}}$ is the vector of unknown temperatures, and $\mathbf{q}_{\text{tile}}$ is a vector of known powers injected into the nodes (consisting of 0 entries except for the single node where a known power is injected).

$\mathbf{G}_{\text{tile}}$ is a higher bandwidth matrix than the one obtained for the PCB layer in chapter 3, and has a sparsity pattern shown in Fig. 5.3, for a 5×5 grid. Therefore, in order to create a downsampled layer image with a resolution of $n_{xd} \times n_{yd}$ we must invert $\mathbf{G}_{\text{tile}}$ $n_{xd} \times n_{yd}$ times.

5.4 Computing through-plane directional thermal conductivity

To calculate the thermal conductivity in the z-direction (κ_z), we first assume that the tile is at a uniform temperature. This enables us to consider all the R_z components in this portion of the thermal network to be in parallel, effectively representing the tile as a single node of uniform temperature that connects to all other thermal resistances in the vertical plane. This is a reasonable assumption, given that in the final calculation, the tile is reduced to a single grayscale pixel, which can only have one associated constant temperature.

To calculate the vertical thermal resistance under the tile, we can follow the same steps outlined in the computation of via thermal properties.

5.5 Analysis of theoretical computational performance

The size of $\mathbf{D}$, and thus the scale factor, must be selected to ensure that the tiling downsampling step does not introduce a time penalty to the total computation time of the overall 3D thermal model. Because $\mathbf{D}$ is generally small, up to 64×64 for the problems studied in this thesis, inverting $\mathbf{G}_{\text{tile}}$ is fast. The Gaussian elimination method is used to solve the small system resulting from the tile discretisation because of its modest memory requirements at each step. In general, the backslash operator in MATLAB is faster than PCG for small problems like the one considered in the tiling problem [73].

Theoretical computational complexity of computing $\mathrm{G}_{\mathbf{tile}}$

Let us assume that we start with an image of size $n_x \times n_y$ pixels exported from an EDA tool. We can downsample this image into $n_{xd} \times n_{yd}$ tiles, each of size sf $\times$ sf. $\mathbf{G}_{\text{tile}}$ is an $n \times n$ sparse matrix where $n = \text{sf}^2$. It is a special type of banded matrix, called an 'outrigger' matrix whereby the outer diagonals are separated from the

main diagonals by bands of zeros [85]. These matrices naturally arise from PDE discretization using FEM and FDM in two or more dimensions. Since $\mathbf{G}_{\text{tile}}$ results from a 2D discretization of a tile, it has five bands and its bandwidth is given by:

$$m_b = \sqrt{n} \tag{5.6}$$

In general, if a banded sparse matrix is assembled using natural ordering, it can be factored using Gaussian elimination into an upper triangular matrix in $\mathcal{O}(n(m_b+1)^2)$ operations. The unknowns can then be obtained using systematic back substitution in $2n(m_b+1)$ operations. When analysing computational complexity, we can ignore low-order terms and constants because their impact is nullified by the higher-order term as the problem size grows. Thus, the computational complexity of calculating the thermal resistance in either the x or y direction is dominated by the number of operations required to perform the Gaussian elimination step: $\mathcal{O}(n(m_b+1)^2)$. Substituting $\sqrt{n}$ for m_b, it becomes $\mathcal{O}(2n^2) \approx \mathcal{O}(2\text{sf}^4)$.

The total number of tiles in the system is $n_{xd}n_{yd}$, such that the total number of operations required to compute both $R_{\theta,x}$ and $R_{\theta,y}$ is $\mathcal{O}(2n_{xd}n_{yd}\text{sf}^4)$.

Recalling that

$$n_{xd} \approx \frac{n_x}{\text{sf}} \tag{5.7}$$

and

$$n_{yd} \approx \frac{n_y}{\text{sf}} \tag{5.8}$$

we can now represent the complexity of obtaining the directional thermal conductivity for all layers in the system in terms of the import resolution and scale factor as:

$$\mathcal{O}(2n_c n_x n_y \text{sf}^2) \approx \mathcal{O}(n_x n_y \text{sf}^2) \tag{5.9}$$

since $2n_c \ll n_x$. (Note that the number of copper layers n_c is effectively a small constant that can be dropped since n_c is typically in the range 2-8, while n_x and n_y are typically > 1000.

Theoretical computational complexity of obtaining G

G is an outrigger system obtained from a 3D FDM discretization and has 7 bands. It has a high bandwidth due to the addition of heat source and heat sink nodes to its far-right columns. For a 3D outrigger system, the bandwidth is equal to the product of the number of grid points in two coordinate directions *i.e* $m_b = (n^{1/3})^2 = n^{2/3}$ according to [86]. Therefore the number of operations required to factor the system by Gaussian elimination is

$$\mathcal{O}(n(m_b + 1))^2 \approx \mathcal{O}(n^{7/3}) \tag{5.10}$$

Because **G** is large, square, symmetric, and positive semi-definite, an iterative method can yield better performance. Using the PCG algorithm for this problem type has a computational cost of $\mathcal{O}(n^2)$. In this case n is given by:

$$n = n_{xd} n_{yd} n_z \tag{5.11}$$

Expressing n_{xd} and n_{yd} in terms of n_x, n_y and sf by substituting (5.7) and (5.8) into (5.11) and replacing n in (5.10) by this new value leads to:

$$\mathcal{O}\left(\frac{n_x^2 n_y^2}{\mathrm{sf}^4} n_z^2\right) \tag{5.12}$$

From this discussion, it follows that the relationship between the computational complexity of finding the solution to $\mathbf{G}_{\text{tile}}$ and finding the solution to **G** very strongly depends on the choice of n_x, n_y and sf. Although n_z is raised to the same order as the in-plane resolutions, it plays a small role in the computational cost calculation because $n_z \ll n_x$ and $n_z \ll n_y$. Fig. 5.4 show that the computational complexity of processing many small $\mathbf{G}_{\text{tile}}$ matrices is lower than that required for **G** for all combinations of n_x, n_y and sf that are sensible for thermal modelling of the PCB. The figures also show that the cost of solving for $\mathbf{G}_{\text{tile}}$ becomes non-negligible as sf increases and also at larger values of n_x and n_y.

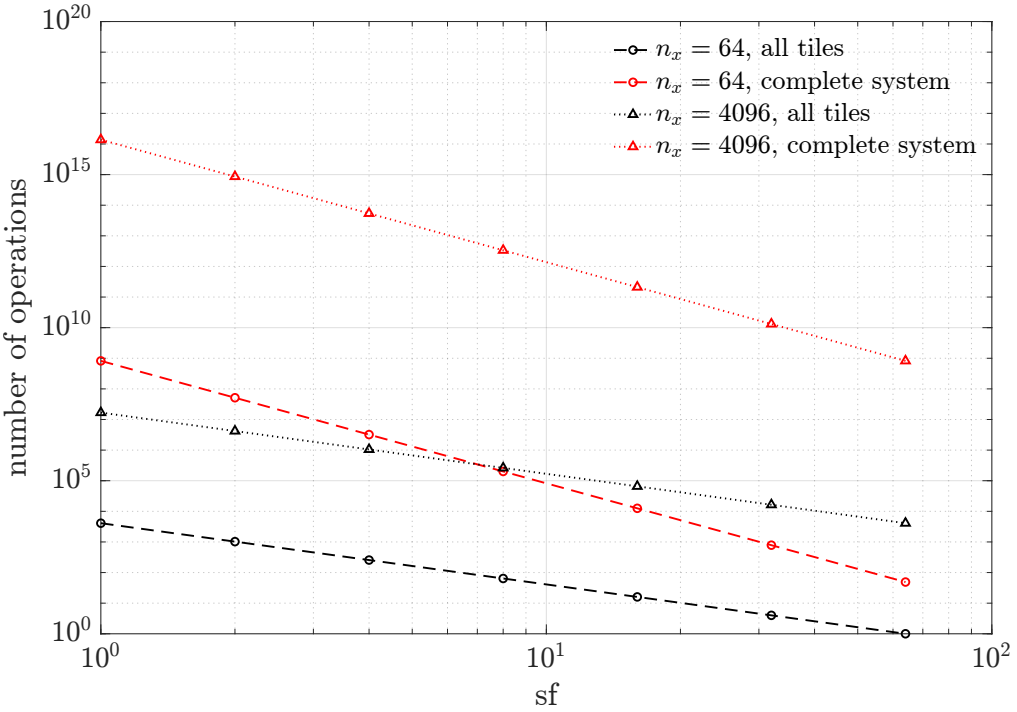


Figure 5.4: Scale factor vs number of operations for $n_x = 4096$ and $n_y = 4096$

Also, it is not necessary to invert $\mathbf{G}_{\text{tile}}$ for tiles composed of uniform material, requiring less than $n_{xd} \times n_{yd}$ inversions, especially for layers with large copper pours or uninterrupted FR4 regions.

Therefore, the main tradeoff centers around the scale factor (sf) and the grid dimensions (n_x, n_y). As sf increases, the cost of solving $\mathbf{G}_{\text{tile}}$ becomes more significant, but it remains lower than solving the full $\mathbf{G}$ matrix for practical PCB modeling scenarios. The choice of n_x, n_y, and sf greatly influences the relative computational costs. While solving $\mathbf{G}_{\text{tile}}$ is generally more efficient, its advantage diminishes with larger scale factors and grid sizes. Additionally, for uniform material tiles, inversions can be reduced, further improving efficiency in certain cases.

5.6 Choice of n_z

The best-performing discretization scheme for n_z is the decaying symmetric spacing method where the largest segment is placed at the centre of the FR4 layer, with adjacent layers doubled at each discretization step as we move away from the middle

section. This results in the finest grid between the FR4-Copper and Copper-FR4 transitions, where thermal gradients are greatest as was shown in Chapter 3.

5.7 Modelling 'fuzzy' heat sources from grayscale downsampled images

The 'detached' heat source models used so far have assumed that the heat source footprint is covered by binary pixels. However, a 'fuzzy' heat source results from tiling downsampling, motivating the need to update the heat source model.

In this context, a 'fuzzy' heat source results from the fact that edges of the heat source cannot be guaranteed to be perfectly aligned with the pixels in the downsampled heat source images particularly if tiles are large. As such, portions of the heat source may be partially contained in pixels located adjacent to the source, *i.e.* some tiles forming the heat source footprint contain pixels that are not part of the heat source. Therefore, naively distributing the heat source among all the pixels in a downsampled source image will lead to inaccurate results. This problem is naturally addressed by the detached 'node' heat source model because the thermal path from of all the heat source pixels from the heat source footprint to the external node is composed of vertical thermal resistances computed by considering the distribution and quantity of copper in all tiles.

5.8 Model verification

Before solving a full-scale PCB system, we turn to a simple example problem on which we perform numerical experiments to demonstrate the robustness and performance gains achievable using a model with grayscale downsampled images created using the approach described in the preceding sections.

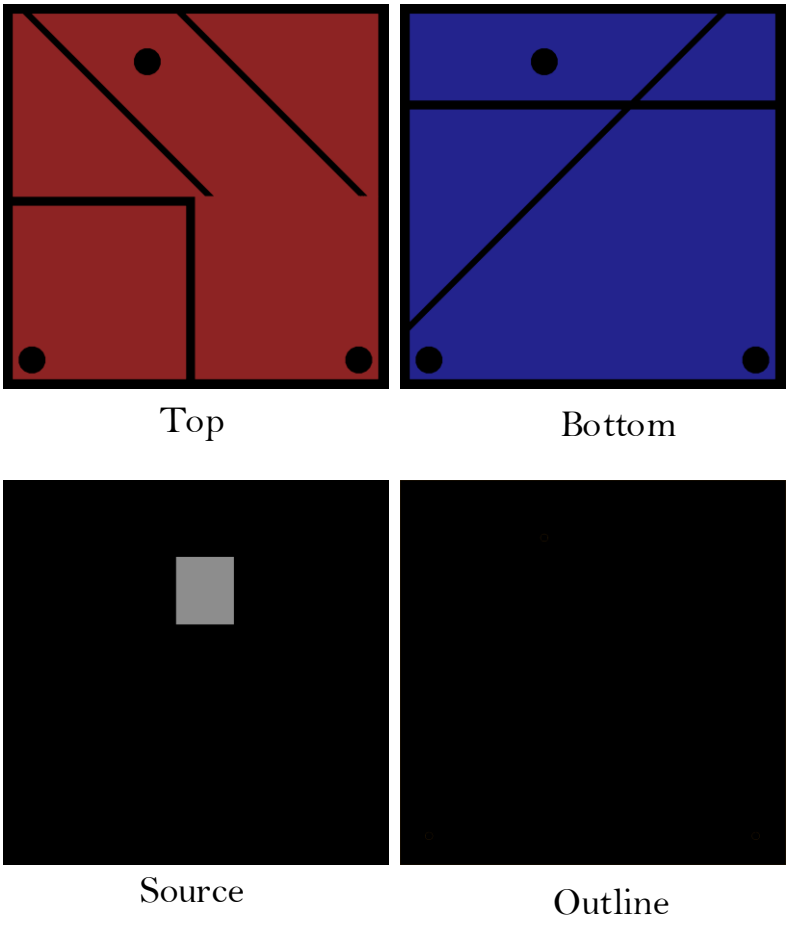


Figure 5.5: Input images used to validate the performance of grayscale downsampling. The slanted features and holes are included to ensure that the model can handle slanted features, tiles from which directional thermal paths are important

5.8.1 Choice of test case

A two-layer PCB with holes, a single heat source on one layer as well as slanted PCB features as shown in Fig 5.6. Such a system captures the general complexities present on a larger scale on a full-scale PCB in proportions one may find on a power electronics PCB while being simple enough to enable us to investigate the accuracy of the model. The model is also simple enough to be constructed and solved in a reasonable time in COMSOL, thus producing benchmark results that we use for comparison. The input images to the FDM approach are shown in Fig. 5.5.

Quantity	Value	Units
dpi	2048	-
l_{pcb}	50.8	mm
w_{pcb}	50.8	mm
P	5	W
h	10	W/Km^2
κ_{Cu}	400	W/mK
k_{FR4}	0.3	W/mK

Table 5.1: Parameters used for modelling the PCB using tiling

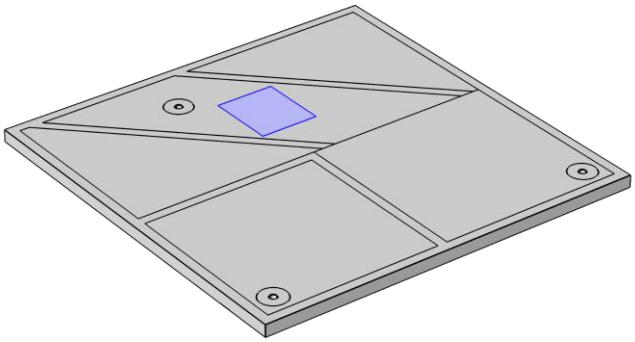


Figure 5.6: COMSOL geometry of the test case

The model is defined to have the parameters shown in Table 5.1.

The problem is solved using three approaches: FDM using binary downsampled images and isotropic thermal properties, FDM grayscale downsampled images with anisotropic thermal properties computed using the methods discussed in this chapter, and COMSOL, which generates the reference temperature profiles. As before, we choose the average heat source temperature as the metric of comparison for the error analysis performed. The pixel-distributed heat source model of chapter 3 is used for reference, again because this can be fully modelled in COMSOL.

The thermal profiles generated for the top and bottom layers using a tiling downsampling approach are shown in Fig. 5.7. The average heat source temperature is computed in each case and compared with the values obtained from COMSOL.

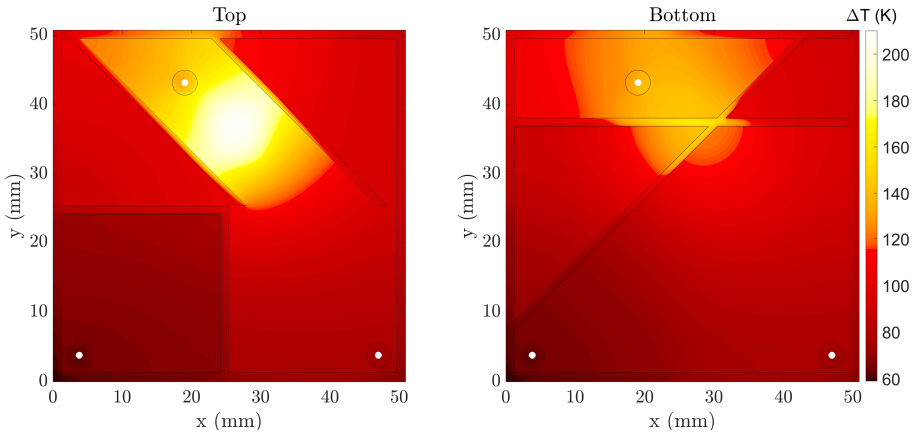


Figure 5.7: (a) Thermal profile from tiling (i) Top (ii) Bottom

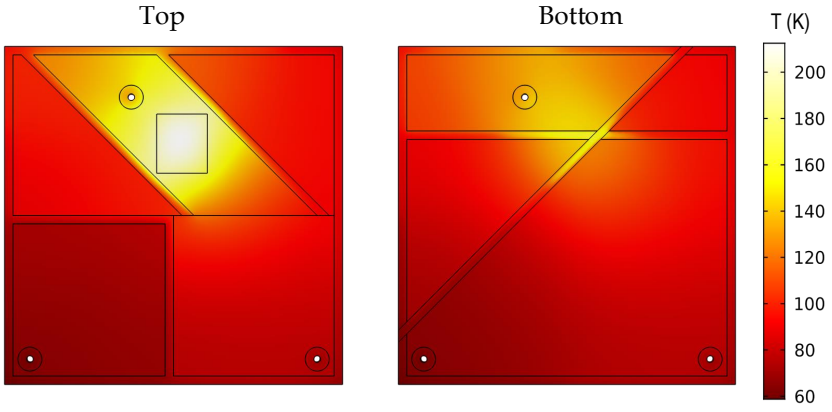


Figure 5.8: (a) Reference thermal profile computed in 167 seconds using COMSOL

Results

From the results figures we notice that while the computation time is *comparable* between the tiled and binary layout, the tiled layout is more accurate at lower resolutions. This is a very significant reduction in computational time for a given accuracy and it means that we can obtain the same level of performance that requires a high element count for binary images by using low-resolution greyscale images that contain information about the orthotropic thermal properties of the layers. The temperature profiles generated from the tiled downsampling model and the COMSOL model are shown in Figures 5.7 and 5.8 respectively, where a scale factor of 4 is used in this plot for the FDM results. The top and bottom layer thermal maps

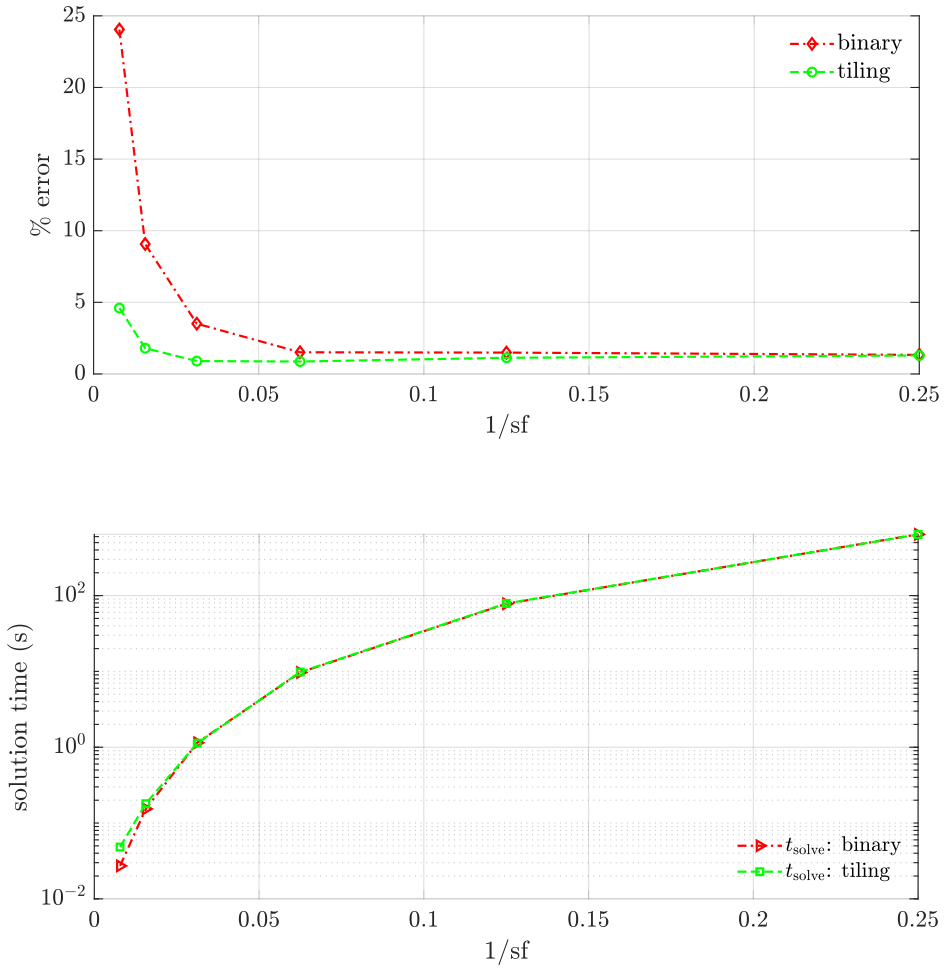


Figure 5.9: Top: Average heat source error computed against scale factor. Bottom: Comparison of solution time vs scaling factor for FDM solutions using tiled and binary layouts

are identical. Fig 5.8. shows the error in the average temperature of the heat source for the two FDM approaches when compared to the result obtained from COMSOL.

Fig. 5.9 shows the reciprocal of the scale factor plotted against the computation time. Interestingly, even though the computational time is nearly identical for the problem (especially at low scale factor, and thus large problem sizes since under these conditions the inversion of the 3D FDM problem dominates the computational effort), the tiled downsampling approach takes slightly more time than the binary downsampling approach. This is because at large scale factors, the size of D is

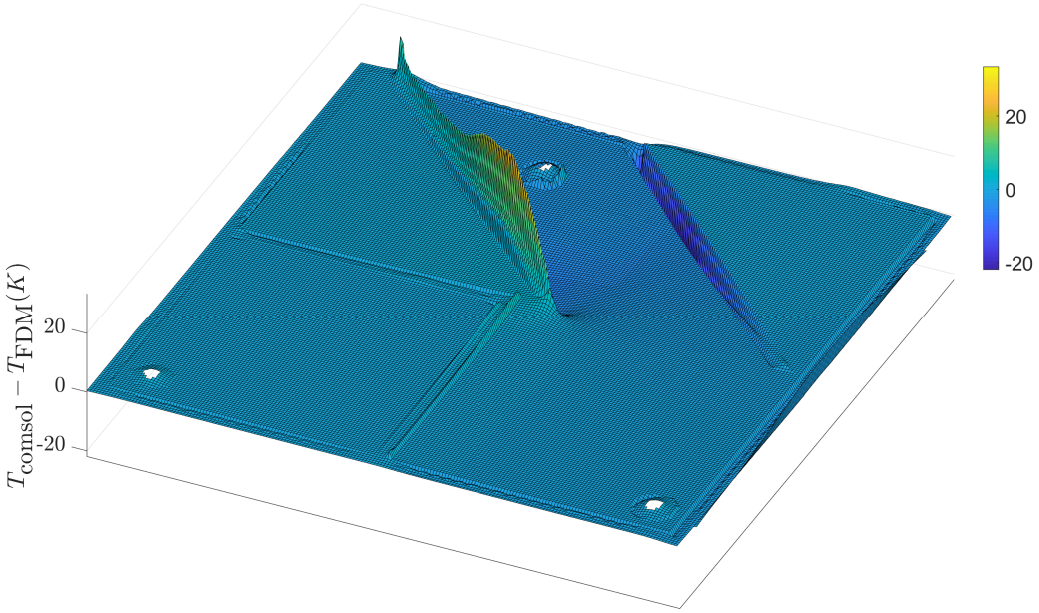


Figure 5.10: Plot of the difference between the top layer temperature profile computed using comsol and FDM with tiling

large enough for the overhead incurred by the intermediate inversion of $\mathbf{G}_{\text{tile}}$ to have an impact on the overall computation time of the problem.

However, as Fig. 5.8 shows, achieving an average heat source error below 2% using tiled downsampling requires two orders of magnitude less time than using the alternative approach. These time savings are also realised using more complex full-featured boards as the next section discusses.

A comparative analysis of temperature profiles generated using the FDM with tiling and COMSOL reveals a remarkably high degree of similarity between these two approaches as evidenced by the low root mean square error (RMSE) values for both the top and bottom layers of the temperature profile (see Table 5.2).

For the top layer, the RMSE between the FDM and COMSOL results was determined to be 2.2195 K, set against a data range of 153.7236 K. This yields a normalized RMSE of 0.0144, indicating that the discrepancy constitutes merely

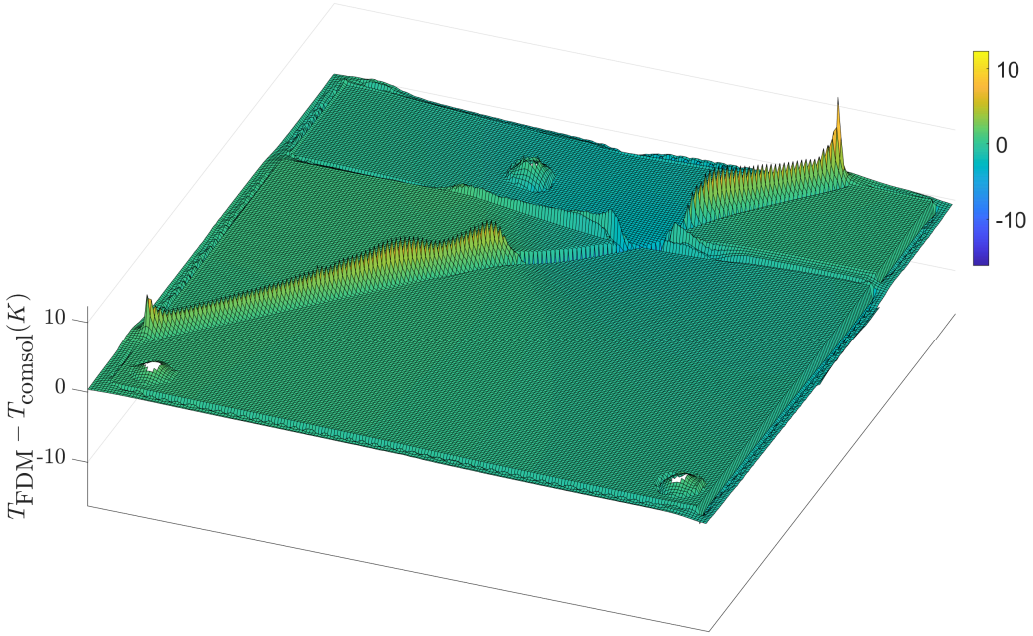


Figure 5.11: Plot of the difference between the bottom layer temperature profile computed using comsol and FDM with tiling

Metric	Top Layer	Bottom Layer
RMSE	2.2195	1.0326
Data Range	153.7236	104.7839
Normalized RMSE	0.0144	0.0099
Normalized RMSE (% of data range)	1.44%	0.99%

Table 5.2: Comparison of FDM and COMSOL results for temperature profile layers

1.44% of the data range. In the bottom layer, an even lower RMSE of 1.0326 was observed, with a data range of 104.7839 K, resulting in a normalized RMSE of 0.0099, or 0.99% of the data range.

These normalized RMSE values, both well below 1.5%, signify that the temperature profiles produced by FDM and COMSOL are virtually identical within the context of the data variability. The minimal differences suggest that any variations between the two methods are negligible for practical purposes.

Figures 5.10 and 5.11 show the differences between the temperatures computed

using the FDM with tiling and those computed using COMSOL. The temperatures are virtually identical for most regions of the PCB, especially in areas with large copper pours or uninterrupted FR-4 blocks. The main differences occur in regions where there is a transition between copper and FR-4, particularly at slanted edges. This is an inherent limitation of the FDM approach because the material properties (directional thermal conductivity) computed in these slanted regions represent copper-FR4 nodes and thermal paths that do not accurately reflect the physical boundary. Since FDM uses square pixels, it cannot accurately model slanted straight lines. Instead FDM uses a staircase-like boundary to approximate the slanted edge. COMSOL, on the other hand, adaptively meshes the PCB using a combination of triangular and tetrahedral elements, which more closely model the physical boundary of the FR-4 and copper interface.

5.9 Verification of the full model

To validate the full model, we revisit the case study full-featured PCB solved in Chapter 4. The model is constructed in FDM using the parameters presented in Table 4.2 and downsampled by tiling.

An example of the results obtained from the model downsampled by tiling and downsampled using binary elements is shown in Fig. 5.12. The temperature profiles generated using binary images downsampled at a scale factor of 4 are shown on the left-hand side for the top, middle and bottom layers. As is evident in the figure, the temperature contours follow largely similar patterns for both binary and grayscale downsampling even though the resolution of the tiled downsample is so low that small tracks and gaps are not visible on the layers.

Table 5.3 shows a summary of the solution times of the model at various scale factors for both binary and grayscale downsampling. While the solution times are roughly equivalent and largely depend on the overall number of elements in the

Scale Factor	Number of Elements ($\times 10^6$)	Binary: t_{solve} (s)	Tiling: t_{solve} (s)
3	48.20	1678.0	1676.0
4	27.10	807.1	810.3
6	12.10	282.7	283.0
8	6.79	138.6	138.1
16	1.71	30.2	33.1
24	0.0531	9.8	14.6

Table 5.3: Results of the full-featured PCB simulation at different scale factors

Heat Source	% Error (sf=4)		% Error (sf=8)		% Error (sf=16)	
	binary	tiling	binary	tiling	binary	tiling
1	0.23	0.23	1.19	0.25	7.86	1.76
2	0.24	0.24	1.88	0.26	8.31	1.56
3	0.22	0.22	1.72	0.25	7.73	1.69
4	0.23	0.23	2.25	0.24	9.79	1.78
5	0.27	0.29	2.49	0.29	9.97	1.58
6	0.29	0.26	2.39	0.26	9.68	1.89
7	0.69	0.67	3.96	0.82	12.18	1.92
8	1.12	1.10	6.95	1.46	13.01	2.10

Table 5.4: Comparison of error in the average source temperature for binary and tiling downsampling at selected scale factors

grid, it is slightly worse for the method that relies on tiles to compute the thermal conductivity for tiles where the scale factor is 16 or 24 because the overhead of inverting the tile matrices when computing the orthotropic thermal conductivities of the layer is non-negligible.

Table 5.4 shows the relative error between the average heat source temperature for the 8 sources atop the board at selected scale factors. Evident in these results are the lower errors in the average source temperature at lower resolutions (sf =16) when using tiled downsampling compared to binary downsampling. This verifies that the directional thermal conductivities computed using the method proposed in this chapter preserve the thermal paths within the layers of the PCB, even when the features (*i.e.* tracks and conductor-dielectric gaps) are no longer visible (*i.e* copper-FR4-copper gaps are bridged).

Solving a thermal model for a board of the size and complexity studied in this

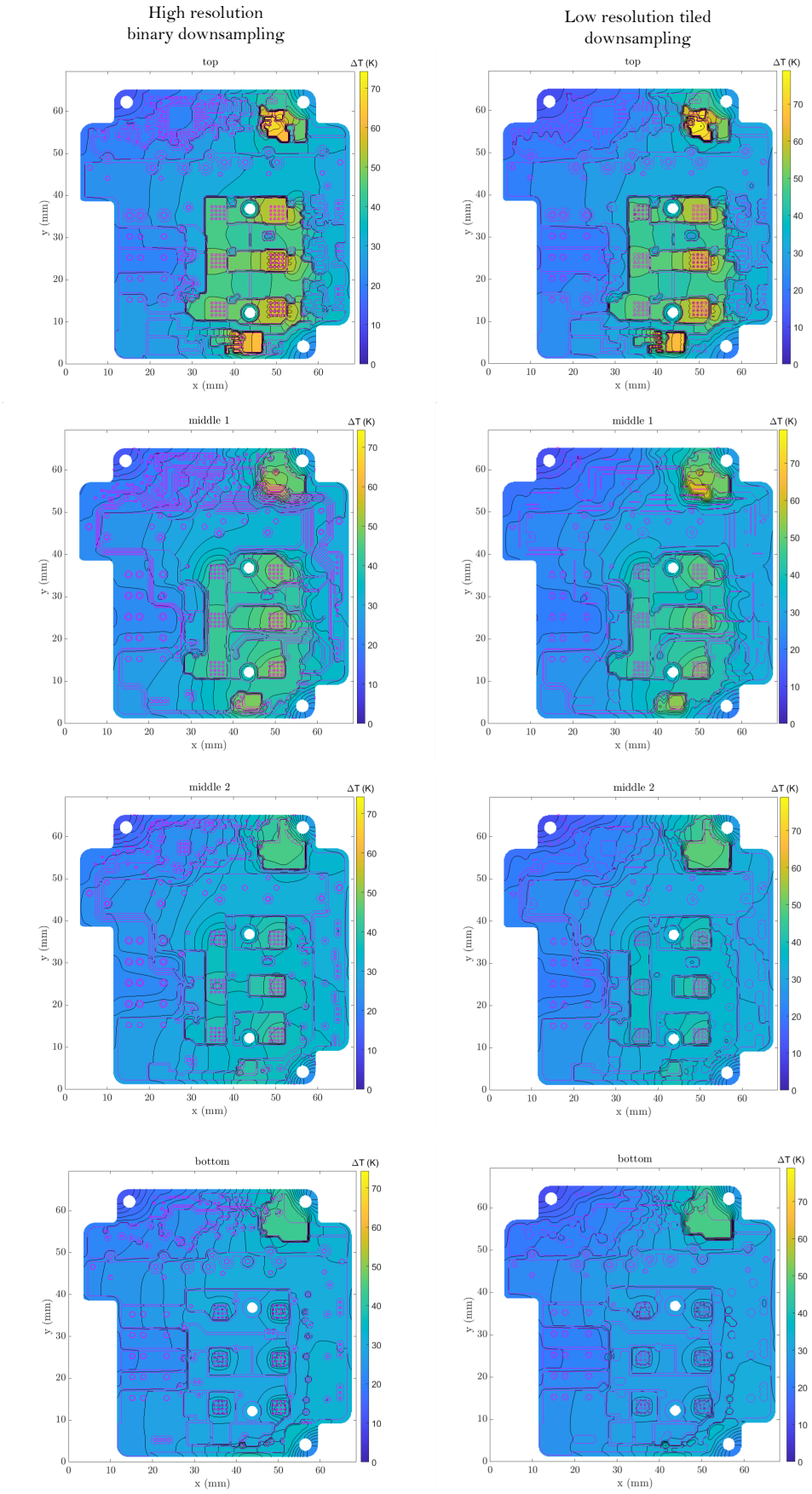


Figure 5.12: Comparison of temperature profiles obtained from a high-resolution binary downsampling (left) and tiled downsampling (right)

chapter at this level of fidelity, where temperature profiles for internal layers are also calculated in difficult and intractable using off-the-shelf numerical computing tools.

This is a complex, fully featured PCB with all the aspects of a real PCB that is intractable to solve using COMSOL on a desktop PC.

5.10 Choosing thermal modelling parameters

In order to model power electronics PCBs accurately using the ideas presented in this thesis, one must choose the input parameters sensibly. The model relies on many physics-based parameters such as thermal conductivities, boundary conditions and power dissipation. In many cases, these values depend on the application, design requirements and target ambient conditions for the PCB. However, the speed and accuracy of the solution also depend on the thermal designer's choice of scaling factor (sf) and dots per inch (dpi).

Fig. 5.4 and the corresponding asymptotic computational complexity equations (5.9) and (5.10).

5.11 Case Study: Modelling a bridge rectifier to compare FDM against Thermal Risk Management (TRM) software

TRM (Thermal Risk Management) is a PCB thermal analysis software developed by Adam Research [2]. It constructs PCB thermal models from Gerber files, component data, and drill information, also obtained from EDA layout tools. We use TRM as a reference tool for validating our thermal model due to its reported accuracy, which is within ±5% of infrared thermography measurements when given correct input parameters.

For our comparative study, we analyze a bridge rectifier PCB design, designed for high power AC applications, in both TRM and FDM. The board features

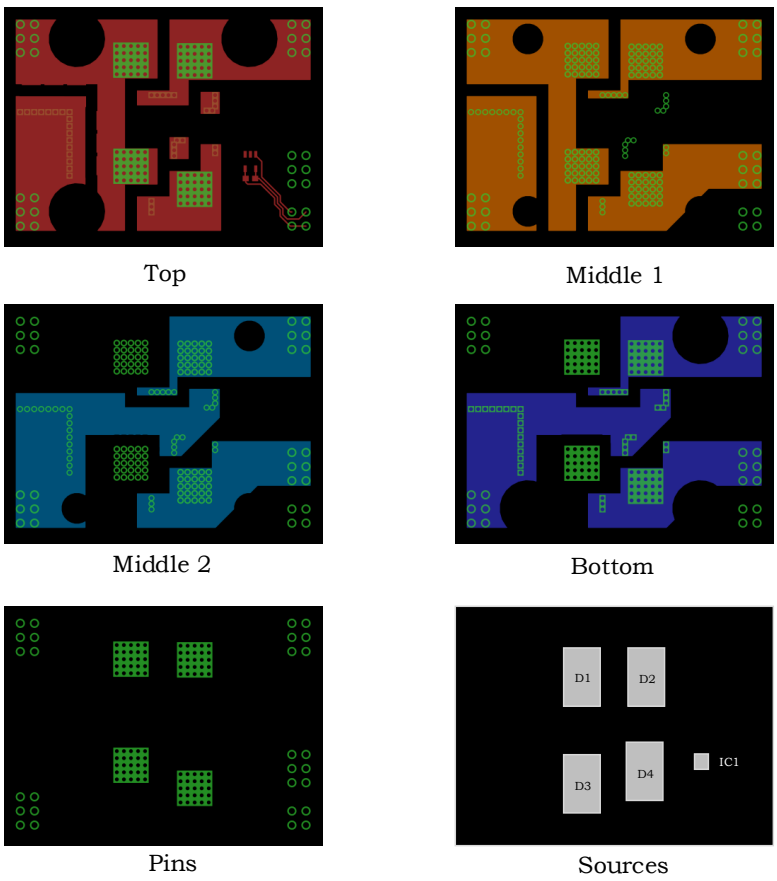


Figure 5.13: FDM input images for the comparative study, showing key layers and features.

four layers with characteristic large copper pours, typical in power electronics for enhanced heat dissipation and heat spreading. Four diodes and an IC atop the board are designated as the hear sources that dissipate during operation. The study parameters are as depicted in Table 5.5 and the input images used in FDM are shown in 5.13. The PCB is imported into TRM from Eagle by means of Gerber files, and defeatured by removing component blocks that TRM generates, exposing a bare PCB. The images in 5.13 are exported from the same EDA tool and used to construct a bare-PCB thermal model in FDM.

This setup allows us to compare construct the thermal models to be as identical as possible, enabling a comparison between FDM's temperature predictions with

Parameter	Value
Number of PCB layers	4
h	12.0 W/(m^2 · K)
T_∞	20.0 °C
t_{Cu}	70 μm
t_{PCB}	1.78 mm °C
Diode 1 dissipation	0.75 W
Diode 2 dissipation	0.5 W
Diode 3 dissipation	0.5 W
Diode 4 dissipation	1.0 W
IC 1 dissipation	0.1 W

Table 5.5: PCB and environmental parameters for TRM validation study

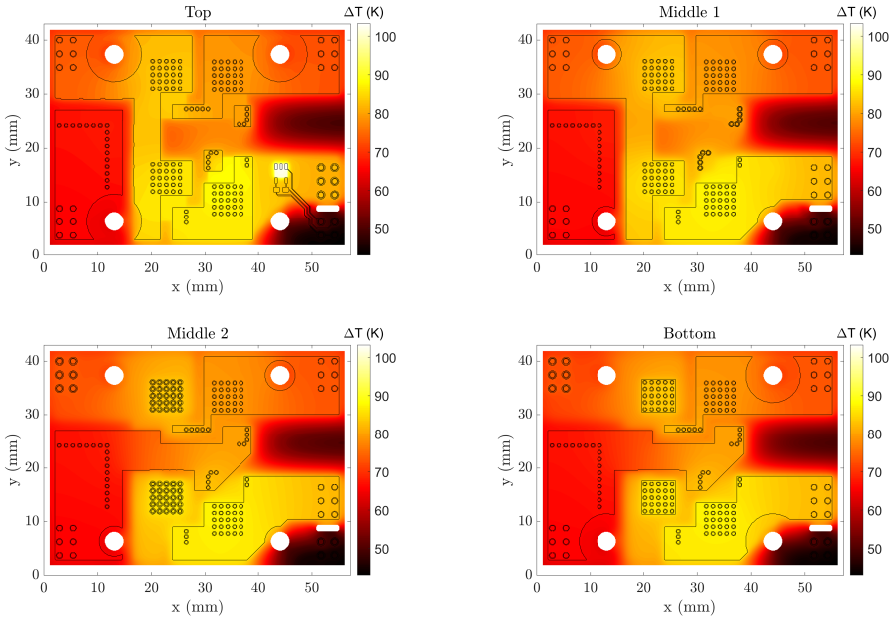


Figure 5.14: Temperature profile for full rectifier computed using the FDM method

TRM's results for a realistic power electronics PCB. The use of different power dissipation values for the diodes enables us to assess our model's accuracy across varying heat loads.

5.11.1 Results and discussion

Fig 5.14 shows the temperature profile generated by the FDM model for all four layers, while Fig 5.15 shows the equivalent thermal maps generated from the TRM

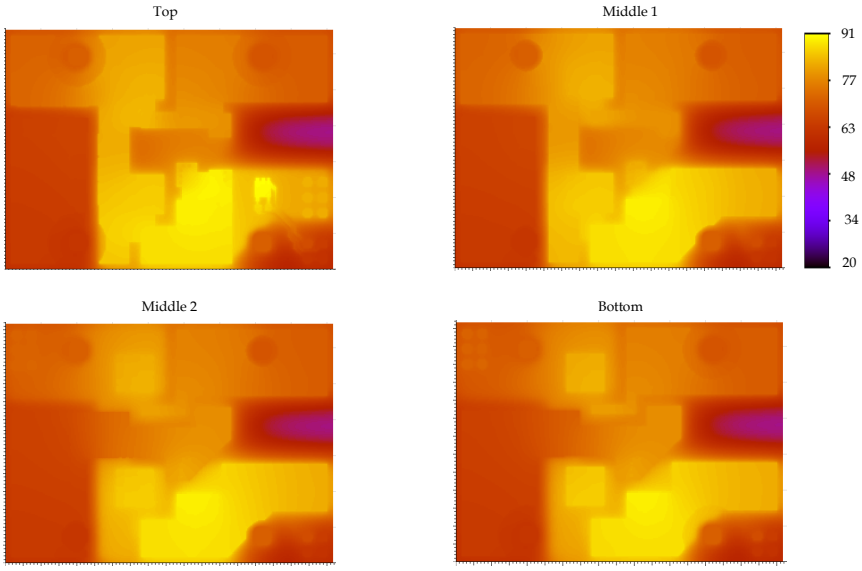


Figure 5.15: Temperature profile for full rectifier computed using TRM

software. There is generally good agreement in the steady state temperature distribution, and location of hotspots on the board.

Fig. 5.16 compares the percentage differences in the mean heat source error computed using either TRM or FDM.

Analysis of diode temperatures

The Mean Percent Error of -3.31% indicates a generally good overall agreement between TRM and FDM, with TRM typically producing slightly lower peak temperature predictions. A possible explanation for this difference is that, as can be observed when comparing the temperature profiles of Figures 5.14 and 5.15, the TRM thermal profile indicates that the solver meshed the holes and computed a temperature value for these regions whereas FDM did not. This difference means that TRM provided more heat spreading than FDM, leading to generally lower peak average source temperatures.

This result also demonstrates that the detached node heat source model functions as expected and mimics the lumped resistor network heat source model of Chapter 3 when the device is modeled with infinite junction-to-ambient resistance. The heat

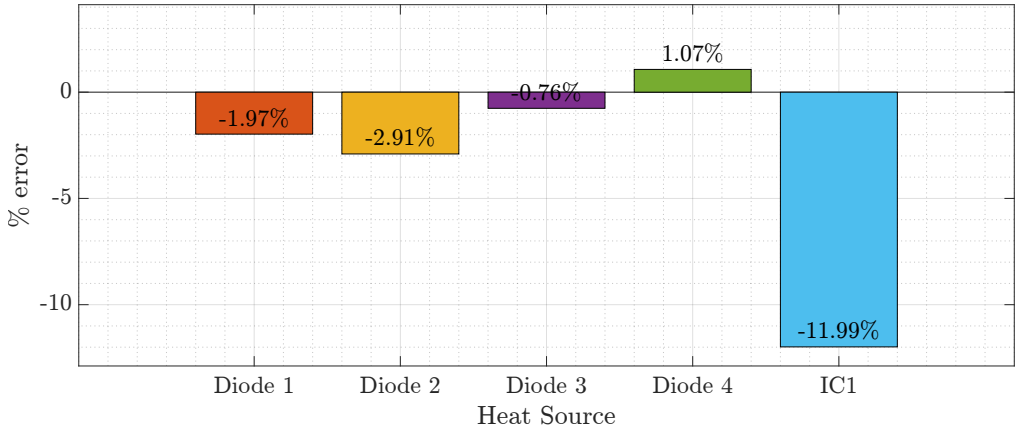


Figure 5.16: Percentage error between TRM and FDM heat source temperatures.

source model used in the TRM simulation consists of a uniformly dissipating pad with the same dimensions as the heat source footprint, as implemented in Chapter 3.

Discrepancy between FDM and TRM calculated mean source temperature for IC1

Although IC1 is dissipating the least heat, it generates the hottest hotspot on the PCB. The reason for the presence of this hotspot is threefold - resulting both from the absence of a direct copper pour in contact with the IC for heat speading, and the low via count under the device as well as the small surface area of the IC exposed to the ambient environment for heat loss by convection.

The Maximum Absolute Percent Error of 11.99% for IC1 is significantly higher than the discrepancy in temperature values computed for the diodes. This is due to the absence of a well defined pad connected to a copper pour under the device footprint, resulting in poor conduction of the heat by the thermal path composed of a combination of copper and FR4 under the device.

Additionally, the disagreement between FDM and TRM for this device points to an inherent limitation of the bare-PCB thermal model which requires well defined copper pours in order to discretize the region without thermal shortcircuiting. The lower temperature observed by TRM is likely due to thermal shortcuircuting effects, which can be visually observed as the smudging of the top layer temperature in

the region occupied by IC1 in Fig. 5.15. The detached node heat source model used by FDM for IC1 ensures that any thermal shortcuirts are avoided, with no single isolated pad footprint getting too hot.

5.12 Conclusions

In this chapter, a framework for generating directional thermal conductivities for a tile of pixels is proposed. Numerical experiments show that pixel-derived bare-PCB thermal models constructed from tiled layouts can achieve good quality results, that are not achievable from images downsampled in a binary fashion. There is a time penalty associated with multilayer boards, due to the repeated matrix inversions that must be performed for each layer. Skipping the inversion of uniform tiles, which are common for small sf, and on ground and power planes, has been shown to improve computational speed.

The likelihood of a large tile containing a mixture of materials is high. This increases the chances of having to invert a large $\mathbf{G}_{\text{tile}}$, which is computationally expensive. As a result, grayscale downsampling takes, on average longer, to solve than binary downsampling at the same resolution as sf increases (i.e. when the final images have low resolutions). However, the results from grayscale downsampling are more accurate at these resolutions and match the results obtained from larger resolutions using binary downsampling.

A comparative study using a bridge rectifier PCB examined the performance of the FDM against TRM software, a tool known for its accuracy within ±5%. Both models displayed good agreement in terms of hotspot locations and overall temperature distribution, with FDM achieving a Mean Percent Error of -3.31% for diode temperatures. However, discrepancies were noted, particularly for IC1, where the Maximum Absolute Percent Error reached 11.99%. These differences highlight

FDM's handling of heat sources and limitations in TRM's results, especially in regions lacking copper pours, where thermal short-circuiting can occur.

6

Conclusions and further research

In Chapters 3, 4, and 5, we have presented a variety of techniques for generating highly accurate thermal models of power electronics PCBs from rasterized images of PCB layers extracted from EDA tools in the order of a few minutes. While the circuits examined in this study are power electronic converters, the same techniques can be applied to other PCB-based circuits and planar integrated circuits to rapidly model their thermal behaviour. The techniques presented in this work can enable short PCB development cycles by improving the speed of thermal analysis.

Our work demonstrates that significant progress has been made in computing power over the past few decades, enabling the rapid solution of large discretized PDE systems through Gaussian matrix inversion techniques (such as the backslash operator in MATLAB) and iterative approaches (like PCG). This progress has not yet reached theoretical limits, thus presenting an opportunity to model the thermal behaviour of PCBs using desktop computing accurately.

6.1 Contributions of this thesis

6.1.1 Rapid discretization generation technique for the FDM method

We have proposed an innovative approach to generating FDM discretizations using image processing techniques. This method enables the efficient and accurate generation of thermal profiles during the PCB layout phase, providing designers with valuable insights into the thermal behaviour of the system.

The use of this technique has the potential to greatly accelerate virtual prototyping and reduce design timelines, as it eliminates the need for specialized numerical and multiphysics software. Furthermore, this approach is a low-cost design tool that can be integrated with existing PCB layout tools.

The results of our study demonstrate that the proposed method can accurately capture the thermal behaviour of PCBs, particularly hotspots on the board and the average temperature of heat sources, to a high degree of accuracy (for example, in Chapter 4, heat source errors below 5% were achievable in under one minute of solve time for a practical power electronics PCB). This contribution can be extended to other planar electronic systems, making it a valuable tool for researchers and practitioners in the field.

The results of a case study comparing FDM to Thermal Risk Management (TRM) software further validate the effectiveness of our model. For a high-power bridge rectifier PCB, FDM was able to predict heat source temperatures with a Mean Percent Error of -3.31% compared to TRM, indicating good agreement. FDM also accurately captured average and peak source temperatures and the overall steady-state temperature profile.

6.1.2 Framework for modelling complicated vias on the PCB from raster image data

Modelling vias in numerical simulations of heat transfer in PCBs using finite difference or finite element methods has traditionally been a challenge due to their small lateral thickness, which necessitates prohibitively large mesh sizes. We have developed a method of mapping the locations of vias on the PCB layers and generating a detailed model based on a simple pixel-counting algorithm. The via model was shown to be capable of sufficiently preserving the plated through-hole perimeter for the whole board at various image resolutions and to accurately model the heat transfer across the board.

While this model shows high accuracy relative to COMSOL benchmarks (with normalized RMSE values around 1% for both top and bottom layers), slight discrepancies in via node temperatures arise due to differences in grid alignment between COMSOL and FDM. These discrepancies were observed to be minor and localized, and should be accounted for in assessing the quality of results obtained from FDM.

6.1.3 Method for modelling lumped heat sources

To overcome the limitations of uniform volumetric heat generation in numerical simulations, we have developed a model that treats heat sources as nodes external to the PCB, with heat flow determined by the dynamic temperature gradients within the source. This approach more closely approximates the physical reality of heat generation and dissipation, and avoids inaccurate hotspots in regions with lower heat spreading capability.

This model improves upon the basic uniform dissipation model by taking into account the detailed thermal paths of heat dissipation, such as junction-to-case and junction-to-ambient resistances, which are commonly available in manufacturer data sheets. This external ‘detached-node’ approach, where heat source pixels

connect to a detached isothermal node, better represents complex devices with non-uniform thermal footprints, such as MOSFETs with separate source and drain pads. As a result, the model captures localized overheating in smaller pads, improving prediction accuracy over the simpler distributed models.

6.1.4 Method for modelling heat sinks using a single node

We have developed a more physically accurate model for heat sinks by treating them as single nodes external to the PCB, connected to pixels that map to the heat sink's footprint through thermal interface material and thermal resistance. This approach provides a more realistic representation of heat transfer in the presence of heat sinks, which is important for accurate thermal modelling of PCBs.

In this model, the heat sink is similarly treated as a detached node connected via lumped thermal resistances, including manufacturer-provided values for thermal interface material and heat sink resistance. The external node formulation simplifies the integration of data sheet parameters, and more accurately models the thermal dissipation for systems where multiple or irregularly shaped heat sinks are present. This makes it adaptable for a wider variety of practical PCB designs.

6.1.5 Framework for computing anisotropic thermal conductivity for the PCB for improved thermal performance

We have shown that one can improve the accuracy of the thermal properties of the thermal paths present on a PCB layer through grayscale downsampling, where orthotropic instead of isotropic thermal conductivities are computed for each pixel in the downsampled layer image. This was in contrast to binary downsampling whereby the properties of each downsampled pixel are assigned based on the most abundant material in the input tile, which leads to thermal short-circuiting.

We have shown that the accuracy of the solution at lower resolutions is greatly increased for the same speed or with a small time penalty that depends on the optimal size of the tile selected in the process.

In verifying the model, the use of grayscale downsampling demonstrated improved accuracy over binary downsampling in predicting temperature profiles, especially at lower resolutions. A comparison on a two-layer PCB test case showed that the tiled grayscale model yielded an average heat source error below 2% while significantly reducing computation time (Fig. 5.9) without sacrificing accuracy. The normalized RMSE remained below 1.5% compared to the reference COMSOL solution (Table 5.2). These results indicate that by leveraging anisotropic thermal conductivity, we can reduce the number of elements required for a given accuracy, leading to faster solve times with comparable precision.

6.2 Limitations of the research

This section summarizes a few key limitations identified in the developed the bare-PCB thermal model to provide context for its applicability.

Representation of complex, non-rectilinear geometries remains limited by the use of square pixels to generate the FDM discretization. Non-orthogonal features, such as slanted traces and circular regions, are approximated using zig-zag lines, which may lead to localized inaccuracies in temperature distribution, particularly in regions with high thermal gradients. This issue is mitigated by the grayscale downsampling approach for anisotropic thermal conductivity but does not entirely eliminate the limitations of using a rectangular grid to represent arbitrary geometries.

Our steady-state focus mean that our model is insufficient for applications requiring time-dependent analysis. This limits the model's ability to predict thermal performance during converter startup, shutdown, or rapidly changing operating conditions, which are often critical in real-world applications. However, the model's

strength in forecasting thermal behaviour in nominal operating conditions remains a valid and highly applicable proposition in industry practice.

The model's accuracy and performance depends on large copper pours dominating the PCB layout, which is typical for power electronics PCBs. However, other types of PCBs that feature many thin copper traces require large image resolutions to be modelled accurately, limiting the scope of application of the model.

Lastly although we have validated our model against industry software, such as Thermal Risk Management (TRM) and COMSOL, further **experimental validation** under varied real-world conditions can greatly improve the understanding of the model.

Despite these limitations, the thermal models developed in this thesis provide a robust framework for rapid, accurate PCB thermal analysis. These limitations present opportunities for future work which are discussed in the next section.

6.3 Future work

In this concluding section, we present interesting areas of research directly arising from this work.

6.3.1 Multigrid discretization for improved resolution in PCB modeling

Adaptive mesh refinement techniques, which increase mesh resolution in regions with large gradients, are known to improve the accuracy of finite element and finite difference simulations of partial differential equations. In the context of image-based FDM discretization of PCBs, coupling such techniques with a multigrid approach is a promising way to obtain higher fidelity results.

By allowing for elements of different sizes, multigrid discretization can be used to create higher-resolution meshes that preserve the geometry of fine features such as vias. Conversely, larger regions of copper pours, which have relatively uniform

temperature profiles, may be represented using coarser meshes to reduce the node count and computational effort.

Future research could automate the selection of regions for finer or coarser meshes based on image features and investigate how this approach impacts simulation time and accuracy.

6.3.2 Study the use of the modelling methodology for other planar electronic systems that resemble PCBs, for example, ICs

While this work has focused on the thermal modelling of printed circuit boards (PCBs) by leveraging their typical planar structures, other electronic systems with similar layouts may also benefit from the proposed methodology. For example, power electronic modules and integrated circuits (ICs) are other planar electronic systems that could potentially be modelled using the same ideas and techniques.

Power electronic modules typically have simpler geometric layouts than PCBs, which makes them suitable for thermal analysis using analytical methods, as demonstrated in the literature. However, image-driven finite difference method (FDM) could provide a fast and efficient alternative for their thermal analysis. Further investigation is required to explore the applicability of image-driven FDM to the thermal modelling of power electronic modules, ICs, and other similar structures.

6.3.3 Investigate the potential of using other numerical methods to solve the problem, for example, FEM

Finite element and finite volume methods have been extensively studied in the literature and are well-established numerical methods for solving thermal problems. However, their use in conjunction with image-based techniques is a promising area for further investigation. In particular, extending the pixel of an image into the third dimension to generate structured hexahedral meshes could provide a basis for performing finite element analysis.

Using image-based discretization in conjunction with finite element methods can simplify the mesh generation process, as the mesh can be directly derived from the image, and then refined using triangular elements to more closely represent circular and slanted geometric features.

Combining the strengths of FEM with image-based FDM could improve the accuracy of models for more complex geometries, but the computational costs must be carefully managed. Adaptive meshing techniques or hardware acceleration could help mitigate this.

6.3.4 Use of hardware acceleration to solve the problem faster

The steps involved in generating the 3D discretization of a PCB from raster images and solving the resulting system of equations are highly parallelizable. There is potential to significantly improve both the processing and solution time of the problem using GPUs, which are designed to natively perform computation on matrices/images. GPUs are likely going to become mainstream and accessible for general use in the future, and understanding how the thermal modelling scheme we have proposed would perform on such hardware is an important research direction.

Future research could explore the specific challenges associated with GPU parallelization, such as memory limitations and data access patterns, to fully leverage hardware acceleration for real-time thermal analysis.

6.4 Software availability

The software tools developed as part of this research will be made publicly accessible. Interested researchers, engineers, and practitioners can access these tools through the Power Electronics Group website at the University of Oxford:

`https://eng.ox.ac.uk/power-electronics-group/software/`

By making our software freely available, we aim to promote further research in this field, enable replication of our results, and encourage the wider adoption of rapid thermal modeling techniques in PCB design processes.

References

[1] *A Quick PCB Thermal Calculation for Power Electronic Devices with Exposed Pad Packages*. AND9596/D. Rev. 1.0. ON Semiconductor. Oct. 2017.

[2] ADAM Research. *ADAM Research*. Accessed on: 01/10/2023. 2024. URL: https://www.adam-research.de/en/.

[3] K. Azar and J.E. Graebner. "Experimental determination of thermal conductivity of printed wiring boards". In: *Twelfth Annual IEEE Semiconductor Thermal Measurement and Management Symposium. Proceedings*. 1996, pp. 169–182. DOI: 10.1109/STHERM.1996.545107.

[4] K.J. Bathe. *Finite Element Procedures*. Prentice Hall, 2006. ISBN: 9780979004902. URL: https://books.google.co.uk/books?id=rWvefGICfO8C.

[5] William A. Berry. *A124623 Number of unit squares having center within inscribed circle of an n X n integer square.* 2006. URL: https://oeis.org/A124623. (Accessed on: 15/11/2022).

[6] Swarup Bhunia and Mark Tehranipoor. "Chapter 4 - Printed Circuit Board (PCB): Design and Test". In: *Hardware Security*. Ed. by Swarup Bhunia and Mark Tehranipoor. Morgan Kaufmann, 2019, pp. 81–105. ISBN: 978-0-12-812477-2. DOI: 10.1016/B978-0-12-812477-2.00009-5.

[7] Byron Blackmore. "Validation and sensitivity analysis of an image processing technique to derive thermal conductivity variation within a printed circuit board". In: *2009 25th Annual IEEE Semiconductor Thermal Measurement and Management Symposium*. 2009, pp. 76–86. DOI: 10.1109/STHERM.2009.4810746.

[8] Bogdan et al. "Impact of Chip-Package-PCB Design on the Optimization of AC/DC Switched Mode Power Supply using Time Domain Analysis". In: *2021 33rd International Symposium on Power Semiconductor Devices and ICs (ISPSD)*. 2021, pp. 127–130. DOI: 10.23919/ISPSD50666.2021.9452222.

[9] Douglas Brooks and Johannes Adam. *PCB Design Guide to Via and Trace Currents and Temperatures*. 2021.

[10] Rémy Caillaud et al. "Thermal Considerations of a Power Converter With Components Embedded in Printed Circuit Boards". In: *IEEE Transactions on Components, Packaging and Manufacturing Technology* 10.2 (2020), pp. 230–239. DOI: 10.1109/TCPMT.2019.2939969.

[11] Antonio Pio Catalano et al. “Analytical Modeling of Through-PCB Thermal Vias and Heat-Sinks for Integrated Power Electronics”. In: *2019 25th International Workshop on Thermal Investigations of ICs and Systems (THERMINIC)*. 2019, pp. 1–6. DOI: 10.1109/THERMINIC.2019.8923386.

[12] Antonio Pio Catalano et al. “Numerical Simulation and Analytical Modeling of the Thermal Behavior of Single- and Double-Sided Cooled Power Modules”. In: *IEEE Transactions on Components, Packaging and Manufacturing Technology* 10.9 (2020), pp. 1446–1453. DOI: 10.1109/TCPMT.2020.3007146.

[13] Youngsang Cho et al. “Pattern Continuity Check Approach to Model Effective Thermal Conductivity of PCB”. In: *2020 19th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm)*. 2020, pp. 909–917. DOI: 10.1109/ITherm45881.2020.9190418.

[14] Daniel Christen, Milos Stojadinovic, and Juergen Biela. “Energy Efficient Heat Sink Design: Natural Versus Forced Convection Cooling”. In: *IEEE Transactions on Power Electronics* 32.11 (2017), pp. 8693–8704. DOI: 10.1109/TPEL.2016.2640454.

[15] Clyde F. Coombs Jr. and Happy T. Holden, eds. *Printed Circuits Handbook*. 7th ed. New York: McGraw-Hill Education, 2016.

[16] Luisa Coppola, Didier Cottet, and Franz Wildner. “Investigation on current density limits in power printed circuit boards”. In: *2008 Twenty-Third Annual IEEE Applied Power Electronics Conference and Exposition*. 2008, pp. 205–210. DOI: 10.1109/APEC.2008.4522723.

[17] J.R. Culham and M.M. Yovanovich. “Factors affecting the calculation of effective conductivity in printed circuit boards [thermal analysis]”. In: *ITherm’98. Sixth Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (Cat. No.98CH36208)*. 1998, pp. 460–467. DOI: 10.1109/ITHERM.1998.689603.

[18] I. Damian, C. Branescu, and V. Soare. “Solder paste optimization in thermal pad padstack”. In: *2012 IEEE 18th International Symposium for Design and Technology in Electronic Packaging (SIITME)*. 2012, pp. 125–128. DOI: 10.1109/SIITME.2012.6384360.

[19] Swagat Das et al. “Detection of aging related IGBT bond-wire lift-off using spread spectrum time domain reflectometry (SSTDR)”. In: *2017 IEEE Applied Power Electronics Conference and Exposition (APEC)*. 2017, pp. 789–794. DOI: 10.1109/APEC.2017.7930785.

[20] Jonathan Davidson, D.A. Stone, and Martin Foster. “Required Cauer network order for modelling of thermal transfer impedance”. In: *Electronics Letters* 50 (Feb. 2014), pp. 260–262. DOI: 10.1049/el.2013.3426.

[21] Timothy A. Davis. *Direct Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, 2006.

[22] Ercan M. Dede et al. “Design of Anisotropic Thermal Conductivity in Multilayer Printed Circuit Boards”. In: *IEEE Transactions on Components, Packaging and Manufacturing Technology* 5.12 (2015), pp. 1763–1774. DOI: 10.1109/TCPMT.2015.2473103.

[23] Ercan M. Dede et al. “Thermal Design, Optimization, and Packaging of Planar Magnetic Components”. In: *IEEE Transactions on Components, Packaging and Manufacturing Technology* 11.9 (2021), pp. 1480–1488. DOI: 10.1109/TCPMT.2021.3105003.

[24] Nicolas Degrenne and Stefan Mollov. “A low-cost and high-performance thermal interface assembly between printed circuit board and heat-sink”. In: *2017 19th European Conference on Power Electronics and Applications (EPE'17 ECCE Europe)*. 2017, P.1–P.6. DOI: 10.23919/EPE17ECCEEurope.2017.8098996.

[25] Mehmet Baris Dogruoz and Gokul Shankaran. “Spatial Variation of Temperature on Printed Circuit Boards: Effects of Anisotropic Thermal Conductivity and Joule Heating”. In: *IEEE Transactions on Components, Packaging and Manufacturing Technology* 2.10 (2012), pp. 1649–1658. DOI: 10.1109/TCPMT.2012.2205251.

[26] W.D. van Driel et al. “Influence of material combinations on delamination failures in a cavity-down TBGA package”. In: *IEEE Transactions on Components and Packaging Technologies* 27.4 (2004), pp. 651–658. DOI: 10.1109/TCAPT.2004.838859.

[27] U. Drofenik and J.W. Kolar. “Teaching thermal design of power electronic systems with web-based interactive educational software”. In: *Eighteenth Annual IEEE Applied Power Electronics Conference and Exposition, 2003. APEC '03.* Vol. 2. 2003, 1029–1036 vol.2. DOI: 10.1109/APEC.2003.1179343.

[28] Uwe Drofenik et al. “Computationally Efficient Integration of Complex Thermal Multi-Chip Power Module Models into Circuit Simulators”. In: *2007 Power Conversion Conference - Nagoya*. 2007, pp. 550–557. DOI: 10.1109/PCCON.2007.373020.

[29] G.N. Ellison. “Extensions of the closed form method for substrate thermal analyzers to include thermal resistances from source-to-substrate and source-to-ambient”. In: *IEEE Transactions on Components, Hybrids, and Manufacturing Technology* 15.5 (1992), pp. 658–666. DOI: 10.1109/33.180028.

[30] G.N. Ellison. “Thermal analysis of circuit boards and microelectronic components using an analytical solution to the heat conduction equation”. In: *Twelfth Annual IEEE Semiconductor Thermal Measurement and Management Symposium. Proceedings*. 1996, pp. 144–150. DOI: 10.1109/STHERM.1996.545104.

[31] Gordon N. Ellison. "Thermal analysis of microelectric packages and printed circuit boards using an analytic solution to the heat conduction equation". In: *Advances in Engineering Software* 22.2 (1995), pp. 99–111. ISSN: 0965-9978. DOI: https://doi.org/10.1016/0965-9978(94)00032-E. URL: https://www.sciencedirect.com/science/article/pii/096599789400032E.

[32] Nadezhda Evstatieva and Boris Evstatiev. "Modelling the Temperature Conditions of a Printed Circuit Board". In: *2021 12th International Symposium on Advanced Topics in Electrical Engineering (ATEE)*. 2021, pp. 1–4. DOI: 10.1109/ATEE52255.2021.9425281.

[33] G. Farkas et al. "Dynamic compact models of cooling mounts for fast board level design". In: *Ninteenth Annual IEEE Semiconductor Thermal Measurement and Management Symposium, 2003.* 2003, pp. 255–262. DOI: 10.1109/STHERM.2003.1194371.

[34] A.S. Fleischer, R.D. Weinstein, and S.A. Khobragade. "Forced convective cooling of electro-optical components maintained at different temperatures on a vertically oriented printed circuit board". In: *IEEE Transactions on Components and Packaging Technologies* 27.2 (2004), pp. 296–304. DOI: 10.1109/TCAPT.2004.828575.

[35] J.N. Funk et al. "A semi-analytical method to predict printed circuit board package temperatures". In: *IEEE Transactions on Components, Hybrids, and Manufacturing Technology* 15.5 (1992), pp. 675–684. DOI: 10.1109/33.180030.

[36] Deepak Gautam et al. "A review of thermal management in power converters with thermal vias". In: *2013 Twenty-Eighth Annual IEEE Applied Power Electronics Conference and Exposition (APEC)*. 2013, pp. 627–632. DOI: 10.1109/APEC.2013.6520276.

[37] Deepak S. Gautam et al. "A comparison of thermal vias patterns used for thermal management in power converter". In: *2013 IEEE Energy Conversion Congress and Exposition*. 2013, pp. 2214–2218. DOI: 10.1109/ECCE.2013.6646981.

[38] Alan George, Joseph Liu, and Esmond Ng. *Finite Difference Methods for Ordinary and Partial Differential Equations*. 1994.

[39] Ettore Scabeni Glitz and Martin Ordonez. "MOSFET Power Loss Estimation in LLC Resonant Converters: Time Interval Analysis". In: *IEEE Transactions on Power Electronics* 34.12 (2019), pp. 11964–11980. DOI: 10.1109/TPEL.2019.2909903.

[40] W.M. Godfrey et al. "Interactive thermal modeling of electronic circuit boards". In: *IEEE Transactions on Components, Hybrids, and Manufacturing Technology* 16.8 (1993), pp. 978–985. DOI: 10.1109/33.273700.

[41] Pawel Gorecki and Krzysztof Gorecki. "Non-linear compact thermal model of IGBTs". In: *2017 21st European Microelectronics and Packaging Conference (EMPC) and Exhibition*. 2017, pp. 1–5. DOI: 10.23919/EMPC.2017.8346910.

[42] Krzysztof Górecki and Paweł Górecki. "A new form of the non-linear compact thermal model of the IGBT". In: *2018 IEEE 12th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG 2018)*. 2018, pp. 1–6. DOI: 10.1109/CPE.2018.8372563.

[43] Bruce Guenin. "Heat Spreading Calculations Using Thermal Circuit Elements". In: *Electronics cooling* (Aug. 1, 2008). URL: https://www.electronics-cooling.com/2008/08/heat-spreading-calculations-using-thermal-circuit-elements/. (Accessed on: 15/10/2022).

[44] Xiaoming Guo et al. "A general method for the connection of a component thermal model to a board". In: *IEEE Transactions on Advanced Packaging* 29.2 (2006), pp. 250–263. DOI: 10.1109/TADVP.2006.873136.

[45] D. Haller et al. "Integration of thermal compact models into finite element simulation of printed wiring assemblies". In: *Twentieth Annual IEEE Semiconductor Thermal Measurement and Management Symposium (IEEE Cat. No.04CH37545)*. 2004, pp. 271–277. DOI: 10.1109/STHERM.2004.1291334.

[46] C. Y. Ho, R. W. Powell, and P. E. Liley. "Thermal conductivity of the elements". In: *J. Phys. Chem. Ref. Data* 1.2 (Apr. 1972), pp. 279–421.

[47] Siow Ling Ho, Shailendra P. Joshi, and Andrew A. O. Tay. "Experiments and Three-Dimensional Modeling of Delamination in an Encapsulated Microelectronic Package Under Thermal Loading". In: *IEEE Transactions on Components, Packaging and Manufacturing Technology* 3.11 (2013), pp. 1859–1867. DOI: 10.1109/TCPMT.2013.2266406.

[48] E.C.W. de Jong, J.A. Ferreira, and P. Bauer. "Thermal design based on surface temperature mapping". In: *IEEE Power Electronics Letters* 3.4 (2005), pp. 125–129. DOI: 10.1109/LPEL.2005.860625.

[49] Eveliina Juntunen et al. "Thermal Performance Comparison of Thick-Film Insulated Aluminum Substrates With Metal Core PCBs for High-Power LED Modules". In: *IEEE Transactions on Components, Packaging and Manufacturing Technology* 2.12 (2012), pp. 1957–1964. DOI: 10.1109/TCPMT.2012.2206390.

[50] Kourosh Kalayeh et al. "Automated Method Using Finite Element Simulation to Identify Microvia Stacks at Risk of Separation in Complex PCB Designs". In: *2019 20th International Conference on Thermal, Mechanical and Multi-Physics Simulation and Experiments in Microelectronics and Microsystems (EuroSimE)*. 2019, pp. 1–4. DOI: 10.1109/EuroSimE.2019.8724519.

[51] Tejas Manohar Kesarkar and Nitesh Kumar Sardana. "How TIM impacts thermal performance of electronics: : A thermal point of view study to understand impact of Thermal Interface Material (TIM)". In: *2019 International Conference on Electronics Packaging (ICEP)*. 2019, pp. 200–206. DOI: 10.23919/ICEP.2019.8733441.

[52] Belal Al-Khamaiseh et al. “Thermal Resistance of a 3-D Flux Channel With Nonuniform Heat Convection in the Sink Plane”. In: *IEEE Transactions on Components, Packaging and Manufacturing Technology* 8.5 (2018), pp. 830–839. DOI: 10.1109/TCPMT.2017.2776601.

[53] Igor I. Kochegarov, Aleksandr Tynda, and Nikolay Goryachev. “The method of obtaining the thermal model of the printing circuit assembly”. In: *2016 XIX IEEE International Conference on Soft Computing and Measurements (SCM)*. 2016, pp. 183–185. DOI: 10.1109/SCM.2016.7519722.

[54] Daiji Kondo et al. “Thermal design method of printed circuit board with effective thermal conductivity”. In: *2013 8th International Microsystems, Packaging, Assembly and Circuits Technology Conference (IMPACT)*. 2013, pp. 71–73. DOI: 10.1109/IMPACT.2013.6706627.

[55] T.M. Krziwanek. “Low-CTE materials for printed wiring boards”. In: *Proceedings International Symposium on Advanced Packaging Materials Processes, Properties and Interfaces (IEEE Cat. No.01TH8562)*. 2001, pp. 175–179. DOI: 10.1109/ISAOM.2001.916571.

[56] S. Kummerl and H. Nguyen. *PowerPAD™ thermally enhanced package*. SLMA002H. Rev. 1.2. Texas Instruments. July 2018.

[57] Hans Petter Langtangen and Svein Linge. *Finite Difference Computing with PDEs - A Modern Software Approach*. Springer Open, 2016.

[58] S. Lee, T.F. Lemczyk, and M.M. Yovanovich. “Analysis of thermal vias in high density interconnect technology”. In: *[1992 Proceedings] Eighth Annual IEEE Semiconductor Thermal Measurement and Management Symposium*. 1992, pp. 55–61. DOI: 10.1109/STHERM.1992.172851.

[59] T.F. Lemczyk et al. “PCB trace thermal analysis and effective conductivity”. In: *1991 Proceedings, Seventh IEEE Semiconductor Thermal Measurement and Management Symposium*. 1991, pp. 15–22. DOI: 10.1109/STHERM.1991.152906.

[60] Randal J LeVeque. *Finite Difference Methods for Differential Equations*. R.J. LeVeque, 2005.

[61] Randall J. LeVeque. *Finite Difference Methods for Ordinary and Partial Differential Equations*. Society for Industrial and Applied Mathematics, 2007.

[62] Da Li et al. “Battery Thermal Runaway Fault Prognosis in Electric Vehicles Based on Abnormal Heat Generation and Deep Learning Algorithms”. In: *IEEE Transactions on Power Electronics* 37.7 (2022), pp. 8513–8525. DOI: 10.1109/TPEL.2022.3150026.

[63] R.S. Li. “Optimization of thermal via design parameters based on an analytical thermal resistance model”. In: *ITherm’98. Sixth Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (Cat. No.98CH36208)*. 1998, pp. 475–480. DOI: 10.1109/ITHERM.1998.689605.

[64] Jingcun Liu et al. "Gate Failure Physics of SiC MOSFETs Under Short-Circuit Stress". In: *IEEE Electron Device Letters* 41.1 (2020), pp. 103–106. DOI: 10.1109/LED.2019.2953235.

[65] Sean Shih-Ying Liu et al. "Fast Thermal Aware Placement With Accurate Thermal Analysis Based on Green Function". In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 22.6 (2014), pp. 1404–1415. DOI: 10.1109/TVLSI.2013.2268582.

[66] Zao Liu et al. "Compact Lateral Thermal Resistance Model of TSVs for Fast Finite-Difference Based Thermal Analysis of 3-D Stacked ICs". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 33.10 (2014), pp. 1490–1502. DOI: 10.1109/TCAD.2014.2334321.

[67] Prospero Lombardi et al. "Thermally Optimizing a High-Power PCB". In: *Cadence Whitepaper* (2020), pp. 1–12. URL: https://www.cadence.com/content/dam/cadence-www/global/en_US/documents/tools/system-analysis/celsius-pcb-with-st-wp.pdf.

[68] M. Ludwig et al. "PCB integrated inductors for low power DC/DC converter". In: *IEEE Transactions on Power Electronics* 18.4 (2003), pp. 937–945. DOI: 10.1109/TPEL.2003.813757.

[69] Allen M Lush. "Modeling Heat Conduction in Printed Circuit Boards Using Finite Element Analysis". In: *Electronics Cooling Magazine* ().

[70] Ke Ma, Jiayang Lin, and Ye Zhu. "Statistical Characterization for Loss Distributions of Power Semiconductor Devices". In: *IEEE Transactions on Power Electronics* 36.7 (2021), pp. 7384–7388. DOI: 10.1109/TPEL.2020.3045033.

[71] Ke Ma, Yongheng Yang, and Frede Blaabjerg. "Transient modelling of loss and thermal dynamics in power semiconductor devices". In: *2014 IEEE Energy Conversion Congress and Exposition (ECCE)*. 2014, pp. 5495–5501. DOI: 10.1109/ECCE.2014.6954154.

[72] Martin März and Paul Nance. "Thermal modeling of power electronic systems". In: *Infineon Technologies AG Munich* (2000), pp. 1–20.

[73] MathWorks. *mldivide: Left Matrix Division on MATLAB*. 2021. URL: https://uk.mathworks.com/help/matlab/ref/mldivide.html. (Accessed on: 12/04/2022).

[74] Hiroaki Matsumori et al. "A Practical Core Loss Model for Filter Inductors of Power Electronic Converters". In: *IEEE Journal of Emerging and Selected Topics in Power Electronics* 6.1 (2018), pp. 29–39. DOI: 10.1109/JESTPE.2017.2761127.

[75] Qiu Min, Er-Ping Li, and Jian-Ming Jin. "Lumped 3-D Equivalent Thermal Circuit Model for Transient Thermal Analysis of TSV Array". In: *2019 IEEE International Conference on Computational Electromagnetics (ICCEM)*. 2019, pp. 1–3. DOI: 10.1109/COMPEM.2019.8779007.

[76] Qiu Min et al. "Electrical-thermal co-analysis of through silicon via with equivalent circuit model". In: *2017 IEEE Electrical Design of Advanced Packaging and Systems Symposium (EDAPS)*. 2017, pp. 1–4. DOI: 10.1109/EDAPS.2017.8276944.

[77] Eric Monier-Vinard et al. "Analytical modeling of multi-layered Printed Circuit Board dedicated to electronic component thermal characterization". In: *Solid-State Electronics* 103 (2015), pp. 30–39. ISSN: 0038-1101. DOI: https://doi.org/10.1016/j.sse.2014.09.004.

[78] Y. S. Muzychka, J. R. Culham, and M. M. Yovanovich. "Thermal Spreading Resistance of Eccentric Heat Sources on Rectangular Flux Channels ". In: *Journal of Electronic Packaging* 125.2 (June 2003), pp. 178–185. ISSN: 1043-7398. DOI: 10.1115/1.1568125. URL: https://doi.org/10.1115/1.1568125.

[79] Minh-Nhat Nguyen et al. "Calculation limits of the homogeneous effective thermal conductivity approach in modeling of printed circuit board". In: *20th International Workshop on Thermal Investigations of ICs and Systems*. 2014, pp. 1–6. DOI: 10.1109/THERMINIC.2014.6972484.

[80] Dongkeun Oh, Charlie Chung Ping Chen, and Yu Hen Hu. "Efficient Thermal Simulation for 3-D IC With Thermal Through-Silicon Vias". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 31.11 (2012), pp. 1767–1771. DOI: 10.1109/TCAD.2012.2196435.

[81] David W. Hahn; M. Necati Özişik. *Heat Conduction*. John Wiley and Sons, Inc, 2012.

[82] Neelakantan Padmanabhan. "A Transient Thermal Model for Power Electronics Systems". In: *SoutheastCon 2024*. 2024, pp. 1294–1299. DOI: 10.1109/SoutheastCon52093.2024.10500091.

[83] Panteleimon Papamanolis et al. "Minimum Loss Operation and Optimal Design of High-Frequency Inductors for Defined Core and Litz Wire". In: *IEEE Open Journal of Power Electronics* 1 (2020), pp. 469–487. DOI: 10.1109/OJPEL.2020.3027452.

[84] Yongwan Park et al. "Electro-Thermal Co-Design of a Cooling System-Integrated High-Frequency Transformer". In: *2020 IEEE Transportation Electrification Conference Expo (ITEC)*. 2020, pp. 26–31. DOI: 10.1109/ITEC48692.2020.9161660.

[85] Masayuki Yano; James Douglass Penn; George Konidaris; Anthony T Patera. *Math, Numerics, and Programming (for Mechanical Engineers)*. Draft V1.2 Lecture Notes, 2012.

[86] Masayuki Yano; James Douglass Penn; George Konidaris; Anthony T Patera. *Math, Numerics, and Programming (for Mechanical Engineers)*. 2022. URL: https://eng.libretexts.org/Bookshelves/Mechanical_Engineering/Math_Numerics_and_Programming_(for_Mechanical_Engineers)/00%5C%3A_Front_Matter/03%5C%3A_Table_of_Contents. (Accessed on: 04/12/2023).

[87] "PCB design". In: *Computer-Aided Design* 13.2 (1981), p. 118. ISSN: 0010-4485. DOI: 10.1016/0010-4485(81)90101-9.

[88] *PCB Thermal Design Guide for GaN Enhancement Mode Power Transistors*. GN005. Rev. 2014-11-04. GAN Systems. Nov. 2014.

[89] Francisco Perdigones and José Manuel Quero. "Printed Circuit Boards: The Layers' Functions for Electronic and Biomedical Engineering". In: *Micromachines* 13.3 (2022). ISSN: 2072-666X. DOI: 10.3390/mi13030460. URL: https://www.mdpi.com/2072-666X/13/3/460.

[90] Francisco Perdigones and José Manuel Quero. "Printed Circuit Boards: The Layers' Functions for Electronic and Biomedical Engineering". In: *Micromachines* 13.3 (2022). ISSN: 2072-666X. DOI: 10.3390/mi13030460.

[91] K.A. Pervaiz et al. "Conduction and Convection Based Thermal Analysis of Printed Circuit Boards for Air-borne Applications". In: *2005 International Conference on Microelectronics*. 2005, pp. 350–354. DOI: 10.1109/ICM.2005.1590099.

[92] Salvatore Race et al. "Circuit-Based Electrothermal Modeling of SiC Power Modules With Nonlinear Thermal Models". In: *IEEE Transactions on Power Electronics* 37.7 (2022), pp. 7965–7976. DOI: 10.1109/TPEL.2022.3147688.

[93] *RC Thermal Models*. AN11261. Rev. 5.0. Nexperia. Mar. 2021.

[94] MÁrta Rencz, Vladimir Szekely, and AndrÁs Poppe. "A Methodology for the Co-Simulation of Dynamic Compact Models of Packages With the Detailed Models of Boards". In: *IEEE Transactions on Components and Packaging Technologies* 30.3 (2007), pp. 367–374. DOI: 10.1109/TCAPT.2007.901740.

[95] J.J. Rodriguez et al. "Thermal component models for electro thermal analysis of multichip power modules". In: *Conference Record of the 2002 IEEE Industry Applications Conference. 37th IAS Annual Meeting (Cat. No.02CH37344)*. Vol. 1. 2002, 234–241 vol.1. DOI: 10.1109/IAS.2002.1044094.

[96] Yousef Saad. *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, 2003.

[97] Richard L. Sampson. "Printed Circuit Board Thermal Modeling Without the Use of an Effective Thermal Conductivity". In: *International Electronic Packaging Technical Conference and Exhibition*. July 2005, pp. 19–25. DOI: 10.1115/IPACK2005-73013.

[98] Sanjeev Sathe. "Heat Conduction in Concentric Cylinders: An Application to Electronic Packages". In: *Journal of Electronic Packaging* 116.3 (Sept. 1994), pp. 230–233. ISSN: 1043-7398. DOI: 10.1115/1.2905690.

[99] Y. Shabany. "Component size and effective thermal conductivity of printed circuit boards". In: *ITherm 2002. Eighth Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (Cat. No.02CH37258)*. 2002, pp. 489–494. DOI: 10.1109/ITHERM.2002.1012496.

[100] G.V. Shankaran and R.K. Singh. "Selection of Appopriate Thermal Model For Printed Circut Boards in CFD Analysis". In: *Thermal and Thermomechanical Proceedings 10th Intersociety Conference on Phenomena in Electronics Systems, 2006. ITHERM 2006.* 2006, pp. 234–242. DOI: 10.1109/ITHERM.2006.1645348.

[101] Gokul V. Shankaran, M. Baris Dogruoz, and Daniel de Araujo. "Orthotropic thermal conductivity and Joule heating effects on the temperature distribution of printed circuit boards". In: *2010 12th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems*. 2010, pp. 1–9. DOI: 10.1109/ITHERM.2010.5501370.

[102] Yanfeng Shen et al. "Thermal Modeling and Design Optimization of PCB Vias and Pads". In: *IEEE Transactions on Power Electronics* 35.1 (2020), pp. 882–900. DOI: 10.1109/TPEL.2019.2915029.

[103] João V. T. Silveira et al. "Estimation of Heat Generation in Semiconductors Centrally Mounted on Printed Circuit Boards". In: *IEEE Transactions on Components, Packaging and Manufacturing Technology* 12.2 (2022), pp. 280–287. DOI: 10.1109/TCPMT.2022.3143507.

[104] V. Sodan et al. "Fast and Distributed Thermal Model for Thermal Modeling of GaN Power Devices". In: *IEEE Transactions on Components, Packaging and Manufacturing Technology* 8.10 (2018), pp. 1747–1755. DOI: 10.1109/TCPMT.2018.2808680.

[105] G. Swift, T.S. Molinski, and W. Lehn. "A fundamental approach to transformer thermal modeling. I. Theory and equivalent circuit". In: *IEEE Transactions on Power Delivery* 16.2 (2001), pp. 171–175. DOI: 10.1109/61.915478.

[106] Omri Tayyara et al. "Electro-Thermal Codesign Methodology of an On-Board Electric Vehicle Charger". In: *Journal of Electronic Packaging* 142.4 (June 2020). 041102. ISSN: 1043-7398. DOI: 10.1115/1.4047226. URL: https://doi.org/10.1115/1.4047226.

[107] Renato A. Torres et al. "Cooling Design of Integrated Motor Drives Using Analytical Thermal Model, Finite Element Analysis, and Computational Fluid Dynamics". In: *2021 IEEE Applied Power Electronics Conference and Exposition (APEC)*. 2021, pp. 1509–1509. DOI: 10.1109/APEC42165.2021.9487449.

[108] *Transient thermal measurements and thermal equivalent circuit models*. AN2015-10. Rev. 1.2. Infenion. Apr. 2020.

[109] P.L. Tu, Y.C. Chan, and J.K.L. Lai. "Effect of intermetallic compounds on the thermal fatigue of surface mount solder joints". In: *IEEE Transactions on Components, Packaging, and Manufacturing Technology: Part B* 20.1 (1997), pp. 87–93. DOI: 10.1109/96.554534.

[110] Marcus Miguel V. Vicedo, Febus Reidj G. Cruz, and Ramon G. Garcia. "Copper Coin Over Thermal VIA in PCB for Thermal Management of 12W". In: *2020 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)*. 2020, pp. 169–172. DOI: 10.1109/APCCAS50809.2020.9301674.

[111] Guochun Wan, Kaifeng Zhang, and Kang Fu. "A 3-D Finite Difference Method for Obtaining the Steady-State Temperature Field of a PCB Circuit". In: *IEEE Transactions on Power Electronics* 38.2 (2023), pp. 2032–2040. DOI: 10.1109/TPEL.2022.3210594.

[112] Baohua Wang and Pinaki Mazumder. "Accelerated Chip-Level Thermal Analysis Using Multilayer Green's Function". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 26.2 (2007), pp. 325–344. DOI: 10.1109/TCAD.2006.883919.

[113] Kangjia Wang and Zhongliang Pan. "An Analytical Model for Steady-State and Transient Temperature Fields in 3-D Integrated Circuits". In: *IEEE Transactions on Components, Packaging and Manufacturing Technology* 6.7 (2016), pp. 1026–1039. DOI: 10.1109/TCPMT.2016.2574897.

[114] Yunchan Wu et al. "Conduction Thermal Runaway of SiC MOSFET Under Natural Convection Heat Dissipation". In: *2023 IEEE 2nd International Power Electronics and Application Symposium (PEAS)*. 2023, pp. 125–129. DOI: 10.1109/PEAS58692.2023.10395386.

[115] N. Yamashita, N. Murakami, and T. Yachi. "Conduction power loss in MOSFET synchronous rectifier with parallel-connected Schottky barrier diode". In: *IEEE Transactions on Power Electronics* 13.4 (1998), pp. 667–673. DOI: 10.1109/63.704135.

[116] Shaoyong Yang et al. "Condition Monitoring for Device Reliability in Power Electronic Converters: A Review". In: *IEEE Transactions on Power Electronics* 25.11 (2010), pp. 2734–2752. DOI: 10.1109/TPEL.2010.2049377.

[117] Lian-Tuu Yeh. "Thermal performance evaluation of various heat sinks for air cooling". In: *13th InterSociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems*. 2012, pp. 446–449. DOI: 10.1109/ITHERM.2012.6231465.

[118] Myong-Jae Yoo et al. "Electric Field Assembled Anisotropic Dielectric Layer for Metal Core Printed Circuit Boards". In: *IEEE Electron Device Letters* 33.11 (2012), pp. 1607–1609. DOI: 10.1109/LED.2012.2212872.

[119] Yong Zhan and Sachin S. Sapatnekar. "High-Efficiency Green Function-Based Thermal Simulation Algorithms". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 26.9 (2007), pp. 1661–1675. DOI: 10.1109/TCAD.2007.895754.

[120] Jin Zhang et al. "A Rapid Thermal Analysis Approach for Power Module Design Considering Non-Linear Thermal Conductivity". In: *2024 25th International Conference on Electronic Packaging Technology (ICEPT)*. 2024, pp. 1–5. DOI: 10.1109/ICEPT63120.2024.10668789.

[121] Yabin Zhang. "Improved Numerical-Analytical Thermal Modeling Method of the PCB With Considering Radiation Heat Transfer and Calculation of Components' Temperature". In: *IEEE Access* 9 (2021), pp. 92925–92940. DOI: 10.1109/ACCESS.2021.3093098.

[122] Yabin Zhang and Paolo Emilio Bagnoli. "A modeling methodology for thermal analysis of the PCB structure". In: *Microelectronics Journal* 45.8 (2014), pp. 1033–1052. ISSN: 0026-2692. DOI: https://doi.org/10.1016/j.mejo.2014.04.042.

[123] Yabin Zhang, Alessio Pennatini, and Paolo Emilio Bagnoli. "A thermal model for the PCB structure including thermal contribution of traces and vias". In: *18th International Workshop on Thermal Investigation of ICs and Systems*. 2012, pp. 1–6.

[124] Alois Zingl. *A Rasterizing Algorithm for Drawing Curves*. 2012. URL: http://members.chello.at/~easyfilter/Bresenham.pdf. (Accessed on: 10/10/2022).