UNIVERSITY OF THESSALY
SCHOOL OF ENGINEERING
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Order reduction of very large thermal models using Krylov subspace techniques

Diploma Thesis

Olympia-Kerasia Axelou

Supervisor: George Stamoulis

Volos 2021

UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Order reduction of very large thermal models using Krylov subspace techniques

Diploma Thesis

Olympia-Kerasia Axelou

Supervisor: George Stamoulis

Volos 2021

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Μείωση τάξης θερμικών μοντέλων πολύ μεγάλης κλίμακας με τεχνικές υποχώρων Krylov

Διπλωματική Εργασία

Ολυμπία-Κερασία Αξελού

Επιβλέπων: Γεώργιος Σταμούλης

Βόλος 2021

Approved by the Examination Committee:

Supervisor **George Stamoulis**
Professor, Department of Electrical and Computer Engineering, University of Thessaly

Member **Nestor Evmorfopoulos**
Associate Professor, Department of Electrical and Computer Engineering, University of Thessaly

Member **Gerasimos Potamianos**
Associate Professor, Department of Electrical and Computer Engineering, University of Thessaly

Date of approval: 18-6-2021

Acknowledgements

First and foremost, I would like to express my sincere appreciation to my advisor, Dr. George Floros, for his continuous and unwavering support and guidance along the journey of my M.Eng. studies. His enthusiasm and immense knowledge on the subject of model order reduction has inspired me to delve into the research.

Besides my advisor, I would like to express my deepest sense of gratitude and thanks to my supervisor, Prof. George Stamoulis, who introduced me to the people that constitute the Electronics lab of University of Thessaly which gave me the chance to collaborate with some incredible people in remarkable research projects, hence the opportunity for the R&D project with Huawei Technologies on which this Thesis is based.

I also want to thank Prof. Nestor Evfomorfopoulos along with my labmates and coworkers for their hard work on the project: Ph.D. candidate Pavlos Stoikos, M.Sc. student Konstantinos Moschos and especially Dr. Dimitrios Garyfallou who was leading our team. His dedication to the project and his eagerness to help each one of us improve ourselves are beyond measure and I cannot thank him enough for all the lessons that I have learnt during our collaboration.

Last but certainly not least, I am extremely greateful to my parents for their love and caring throughout all the phases of my life and for the way they brought my brother and me up. Their knowledge, their sacrifices, and above all their support made us who we are now. On top of that, I would like to acknowledge with gratitude the support and the love of my brother, Christos, my friends and my life companion, Theodoros, for all the help and understanding.

DISCLAIMER ON ACADEMIC ETHICS
AND INTELLECTUAL PROPERTY RIGHTS

«Being fully aware of the implications of copyright laws, I expressly state that this diploma thesis, as well as the electronic files and source codes developed or modified in the course of this thesis, are solely the product of my personal work and do not infringe any rights of intellectual property, personality and personal data of third parties, do not contain work / contributions of third parties for which the permission of the authors / beneficiaries is required and are not a product of partial or complete plagiarism, while the sources used are limited to the bibliographic references only and meet the rules of scientific citing. The points where I have used ideas, text, files and / or sources of other authors are clearly mentioned in the text with the appropriate citation and the relevant complete reference is included in the bibliographic references section. I fully, individually and personally undertake all legal and administrative consequences that may arise in the event that it is proven, in the course of time, that this thesis or part of it does not belong to me because it is a product of plagiarism».

The declarant

Olympia-Kerasia Axelou

18-6-2021

Abstract

The arising packing density of very large scale integrated (VLSI) circuits has caused the temperature to become a major issue since it has a strong impact on microelectronic designs, like the negative effect on the device life and the package reliability, making thermal analysis crucial for the proper functionality of the device.

Numerical thermal analysis methods require the solution of linear systems of equations that have extremely long simulation times. Although the construction of the problem formulation is easily done by applying a thermal RC equivalent circuit, the corresponding 3D equations network involves an undesirably time consuming numerical simulation over many time-steps.

However, in most cases, temperature does not need to be monitored across the whole device, but only at some pre-defined hotspots. In these cases, the very large thermal model can be substituted by a much smaller model with similar behavior at pre-specified points by applying model order reduction (MOR) techniques.

In this Thesis, and in collaboration with Huawei Technologies and the Electronics Research Lab of University of Thessaly, we constructed a parametric reduced-order model (ROM) generation tool that can handle large thermal models and generate the corresponding ROM matrices in parametric format. More specifically, we implemented a computationally efficient multi-point moment-matching technique that takes as input realistic discretized thermal models and can produce parametric reduced matrices with great accuracy and performance, using state-of-the-art sparse direct solvers and parallel techniques. Moreover, the tool can optimize the ROM for all possible parameter values of the parametric parts of the device, given a user-defined range. Experimental results demonstrate that the tool may achieve 99.99% order reduction and error deviation less than 5% for a model of 5 million nodes.

Περίληψη

Η αυξανόμενη πυκνότητα κυκλωμάτων μεγάλης κλίμακας (VLSI) έχει ως συνέπεια τη δημιουργία προβλημάτων που αφορούν τη θερμοκρασία, η οποία με τη σειρά της οδηγεί στη μείωση της διάρκειας ζωής των συσκευών και της αξιοπιστίας τους. Αυτό οδηγεί στην ανάγκη για θερμική ανάλυση.

Οι αριθμητικές μέθοδοι θερμικής ανάλυσης απαιτούν την επίλυση γραμμικών συστημάτων εξισώσεων που έχουν εξαιρετικά μεγάλους χρόνους προσομοίωσης. Παρόλο που η κατασκευή του θερμικού συστήματος του προβλήματος γίνεται εύκολα εφαρμόζοντας ένα RC ισοδύναμο κύκλωμα, το αντίστοιχο δίκτυο εξισώσεων τριών διαστάσεων περιλαμβάνει μία ανεπιθύμητα χρονοβόρα αριθμητική προσομοίωση στη διάρκεια πολλών χρονικών βημάτων.

Ωστόσο, στις περισσότερες περιπτώσεις, η θερμοκρασία δε χρειάζεται να μετρηθεί σε ολόκληρη τη συσκευή, παρά μόνο σε ορισμένα προκαθορισμένα σημεία έντονου ενδιαφέροντος (hotspots). Σε αυτές τις περιπτώσεις, το πολύ μεγάλο θερμικό μοντέλο μπορεί να αντικατασταθεί από ένα πολύ μικρότερο μοντέλο με παρόμοια συμπεριφορά στα συγκεκριμένα σημεία, εφαρμόζοντας τεχνικές μείωσης τάξης μοντέλου (MOR).

Σε αυτή τη διατριβή, και σε συνεργασία με τη Huawei Technologies και το εργαστηρίου ηλεκτρονικής του Τμήματος, κατασκευάσαμε ένα εργαλείο παραγωγής παραμετρικού μοντέλου μειωμένης τάξης (ROM), το οποίο μπορεί να διαχειριστεί μεγάλα θερμικά μοντέλα και να παράγει τους πίνακες του ROM σε παραμετρική μορφή. Συγκεκριμένα, υλοποιήσαμε μία ρουτίνα που δέχεται ως είσοδο πραγματικά διακριτοποιημένα μοντέλα και παρουσιάζει εξαιρετική απόδοση, χρησιμοποιώντας σύγχρονους παράλληλους επιλυτές. Τέλος, τα πειραματικά αποτελέσματα έδειξαν πως μπορεί να επιτευχθεί μείωση τάξης κατά 99.99% με απόκλιση σφάλματος λιγότερη του 5% για ένα μοντέλο 5 εκατομμυρίων κόμβων.

Table of contents

List of figures

List of tables

Abbreviations

3D	Three-dimensional
BiCD	Biconjugate gradient
BT	Balanced Trunaction
CSV	Comma-separated values
DC	Direct current
FDM	Finite-difference methods
FEM	Finite-element methods
HSV	Hankel singular values
HTC	Heat transfer coefficients
IC	Integrated Circuit
MM	Moment matching
MNA	Modified nodal analysis
MOR	Model order reduction
MPMM	Multi-point moment matching
ODE	Ordinary differential equation
PCG	Preconditioned conjugate gradient
PDE	Partial differential equation
PINN	Physics Informed Neural Networks
PMOR	Model order reduction
RC	Resistance, Capacitance
ROM	Reduced-order model
VLSI	Very large scale integrated

Chapter 1

Introduction

1.1 Motivation

For almost half a century, the semiconductor industry has been following Moore's Law [4] in terms of integrated circuit (IC) planning and setting targets for research and development. Figure 1.1 shows the trends in the device making from 1970 till the mid-2010s [1]. For the first two decades, trends follow a Dennard-like scaling pattern [5], followed by an acceleration in the rate of growth from 1990s until around mid-2000s, where it seemingly "hit a wall" and the rate of improvement in physical dimensions began to decrease. Nonetheless, the future of the semiconductor industry promises even more improvements on the downscaling and the package density, following - or even going beyond - Moore's law [6, 7, 8, 9].

This means that the technology downscaling will endure and the number of transistors in future ICs will continue to increase. For example, TSMC[10] has announced that 5nm technology node delivers ~1.8x times higher logic density to its previous 7nm generation [11]. This corresponds to higher power densities and die temperature. Figure 1.2 shows the power density of Intel products over the last 12 years [1]. The trend from 2014 till today for larger power density is, to a certain extent, due to the remodeling of the ICs into 3D architectures and the demanding packaging methods that follow, [12, 13, 14, 15, 16], which leads to higher die temperature and local hotspots.

Rising temperatures have become a major issue in the semiconductor industry as they contribute to the degradation of the chip's reliability since they can cause malfunction or even the destruction of the chip at extreme temperatures. More specifically, some of the problems that arise are slower devices because of the degradation of carrier mobility as it depends on how high the temperature is [17], shorter device life times and unreliability since elevated temperatures have a strong impact on the hazard rate [18]. And last but not least, temperature-dependent subthreshold current may lead to increased leakage power [19, 20].

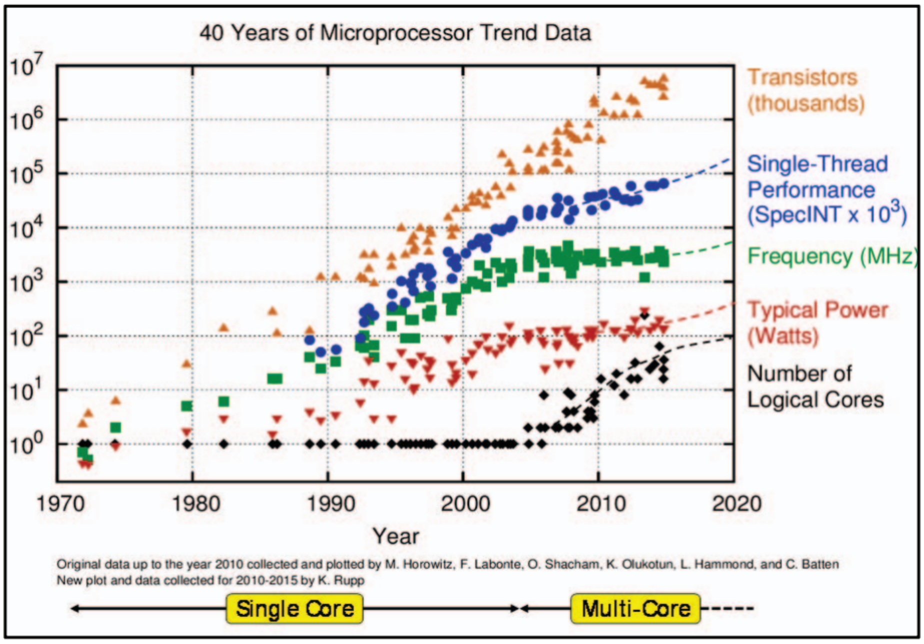

Figure 1.1: Microprocessor trends over the last 40 years [1].

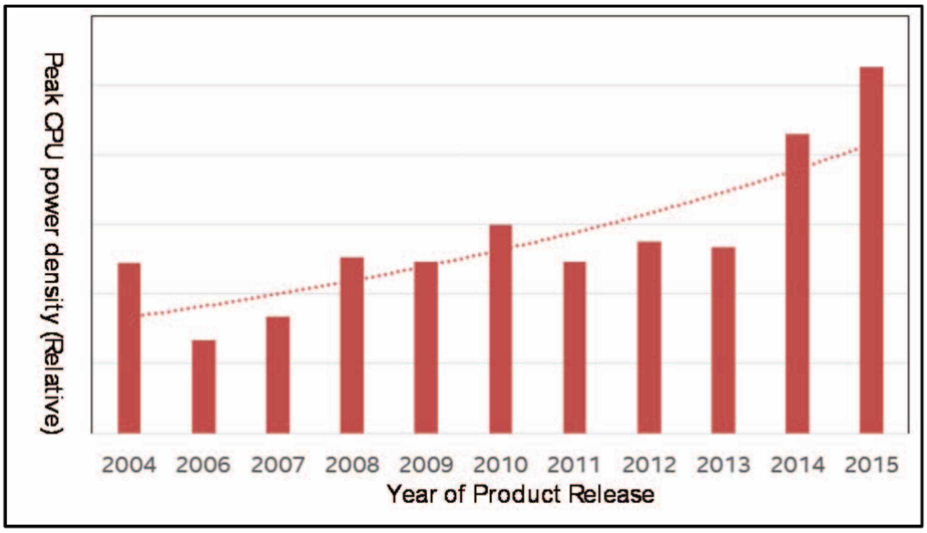

Figure 1.2: Power density of Intel products released over the period 2004-2015 [1].

Consequently, thermal modeling is of critical importance since knowing the temperature distribution across the chip can help prevent the issues mentioned above. Huang et al. [2, 21] suggest that a well-planned thermal model can help complete the leakage power calculation flowchart and play a significant role in developing a reliable and high-performance IC, as shown in Figure 1.3.

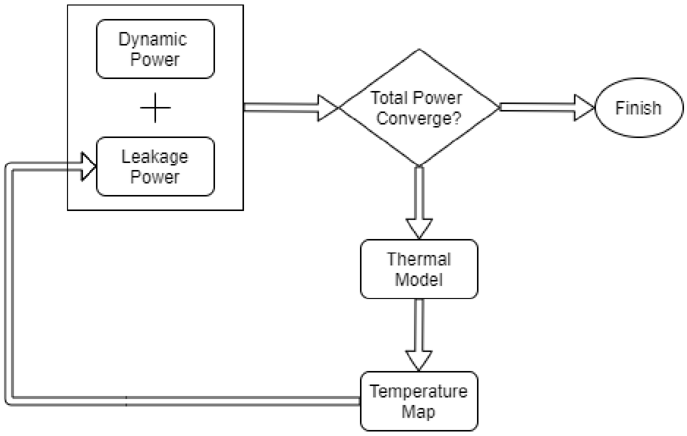

Figure 1.3: Thermal modeling in power integrity analysis [2].

Most of the past work on thermal analysis and simulation is focused on the discretization of the partial differential equation (PDE) of the heat conduction equation [22], transforming it into a system of ordinary differential equations (ODE) (e.g. [3, 23, 24, 25]). Then, the resulting discretized equations of the thermal model are solved by corresponding them to an electrical equivalent circuit [26]. The problem with these methods is that the result of the discretization could lead to extremely large matrices, leading to particularly long simulation times.

However, on most occasions, there is no need to perform thermal analysis across the whole device, but at some pre-defined monitor points. In these cases, the high-order system of equations can be transformed into a much smaller simplified system, that's an approximation of the original one, through a Model Order Reduction (MOR) approach.

The two main categories of MOR are the system theoretic techniques methods (such as Balanced Truncation, first occurrence in [27]) and the moment-matching methods (mainly Krylov subspace methods, first occurrence in [28]). The first one demonstrates better performance due to the fact that it preserves its stability. For the same reason, it offers an a-priori error bound [29, 30]. However, methods of this category appear to have the highest cost, which results from the fact that one has to solve two Lyapunov equations of size the same as the original. This severely impedes the amenability of these methods on systems of high order. On the other hand, moment-matching techniques are an efficient alternative since they only require the computation of the reduction subspace [31]. The downside to this type of methods is that they do not offer boundary approximation errors and that the quality of the ROM approximation depends only on the produced subspace. Overall, MM methods are well established due to their computational efficiency in producing reduced-order models with acceptable accuracy.

1.2 Contributions

In this thesis, we introduce a parametric ROM generation tool that uses MM techniques, and in particular Krylov Subspaces methods, in order to scale down the order of very large thermal models in low execution times while maintaining great accuracy. It should be noted that this research has been conducted in the Electronics Research Lab [1], in collaboration with Huawei Technologies. The contributions of this thesis are summarized below:

1. We created a parametric ROM tool that handles large industrial thermal models, consisting of several million elements.

2. The paramtric reduced-order matrices produced by our tool exhibit not only very small dimension but also great accuracy for all different parameter values within a prespecified range.

3. We integrated state-of-the-art C++ parallel sparse direct and iterative solvers, like the Pardiso and a preconditioned conjugate gradient (PCG), to achieve extremely good execution times.

4. Experimental results on very large scale thermal models consisting of millions of elements indicates significantly fast reduction time while achieving very well accuracy.

[1] Electronics Research Lab: `https://erl.e-ce.uth.gr/`

1.3 Thesis Organization

The rest of the Thesis is organized as follows. Chapter 2 provides a detailed description of the on-device thermal modeling, including the heat conduction and the analogy between thermal and electrical circuits, as well as the heat transfer from the surface to the environment. Chapter 3 introduces MOR, emphasizing on MM techniques. Chapter 4 presents our parametric ROM generation tool along with some efficient implementation choices. In Chapter 5, we present the experimental results and analyze the accuracy and the performance of the tool. Finally,we draw some conclusions and give future directions in Chapter 6.

Chapter 2

State-Space Model

2.1 Thermal Modeling

2.1.1 On-device thermal modeling

Thermal conduction is the main mechanism of heat transfer. It is primarily evaluated in terms of Fourier's Law for heat conduction:

$$\mathbf{q}(\mathbf{r}, t) = -k_t \nabla T(\mathbf{r}, t) \tag{2.1}$$

which shows that the local heat flux density $\mathbf{q}$ is equal to the product of thermal conductivity k of the material and the negative local temperdature gradient $-\nabla T$ over an area $\mathbf{r} = [x, y, z]^T$. In other words, the heat flux density is the amount of energy that flows through a unit area per unit time. Also, according to the conservation of energy, the rate of change of the heat flux $\mathbf{q}$ is equal to the difference between the power generated by the heat sources and the rate of change of temperature:

$$\nabla \cdot \mathbf{q}(\mathbf{r}, t) = g(\mathbf{r}, t) - \rho c_\rho \frac{\partial T(\mathbf{r}, t)}{\partial t} \tag{2.2}$$

where $g(\mathbf{r}, t)$ is the power density of the heat sources, and ρ, c_ρ are the density and the heat capacity of the material. So, by replacing $\mathbf{q}$ from 2.1 in 2.2, we result in:

$$-k_t \nabla^2 T(\mathbf{r}, t) = g(\mathbf{r}, t) - \rho c_\rho \frac{\partial T(\mathbf{r}, t)}{\partial t} \tag{2.3}$$

which can be writen as the following Partial Differential Equation (PDE):

$$\begin{aligned} \rho c_\rho \frac{\partial T(\mathbf{r}, t)}{\partial t} &= k_t \nabla^2 T(\mathbf{r}, t) + g(\mathbf{r}, t) \\ &= k_t (\frac{\partial^2 T(\mathbf{r}, t)}{\partial x^2} + \frac{\partial^2 T(\mathbf{r}, t)}{\partial y^2} + \frac{\partial^2 T(\mathbf{r}, t)}{\partial z^2}) + g(\mathbf{r}, t) \end{aligned} \tag{2.4}$$

followed by the corresponding boundary conditions, of which we are going to discuss at the end of this section.

The next step is to discretize the 3D space with the corresponding steps Δx, Δy, Δz, transforming the second-order derivatives into finite difference approximations. Applying these and multiplying the equation 2.4 by $\Delta x \Delta y \Delta z$, it becomes:

$$\begin{aligned}\rho c_\rho(\Delta x \Delta y \Delta z)\frac{dT(\mathbf{r},t)}{dt} &- k_t\frac{\Delta y \Delta z}{\Delta x}(T_{i+1,j,k} - 2T_{i,j,k} + T_{i-1,j,k}) \\ &- k_t\frac{\Delta x \Delta z}{\Delta y}(T_{i,j+1,k} - 2T_{i,j,k} + T_{i,j-1,k}) \\ &- k_t\frac{\Delta x \Delta y}{\Delta z}(T_{i,j,k+1} - 2T_{i,j,k} + T_{i,j,k-1}) \\ &= g_{i,j,k}\end{aligned} \tag{2.5}$$

2.1.2 Electrical analogy of heat conduction

In solids, heat conduction is analogous to the conduction of electricity in electrical conductors. The flow of heat is proportional to the difference in temperature, as in a conductor, the flow of electricity is proportional to a potential difference. The temperature corresponds to the voltage, the heat flow corresponds to the current, etc., as shown in table 2.1.

Table 2.1: The analogous elements of thermal and electrical conduction

Electrical Circuit	**Thermal Circuit**
Charge	Heat
Current	Heat Flow
Voltage	Temperature
Electrical Resistance	Thermal Resistance
Electrical Capacitance	Thermal Capacitance
Electrical Conductance	Thermal Conductance

So, bearing this in mind, the analogous discretized electrical circuit has a node at every discrete point in the thermal grid, as shown in Figure 2.1. The conductances connecting two neighboring nodes in the directions of x, y, z respectively are:

$$G_x \equiv \frac{k_t \Delta y \Delta z}{\Delta x}, G_y \equiv \frac{k_t \Delta x \Delta z}{\Delta y}, G_z \equiv \frac{k_t \Delta x \Delta y}{\Delta z} \tag{2.6}$$

The capacitance at each node to ground is:

$$C \equiv \rho c_p(\Delta x \Delta y \Delta z) \tag{2.7}$$

Finally, the current sources are:

$$I_{i,j,k} \equiv g_{i,j,k}(\Delta x \Delta y \Delta z) \tag{2.8}$$

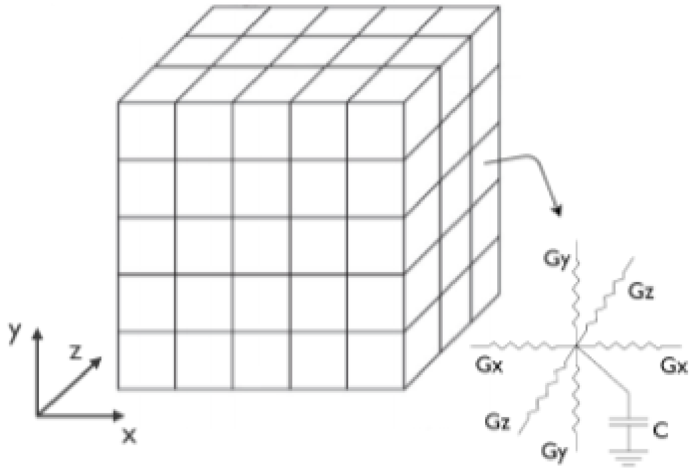

Figure 2.1: Discretization of space & electrical equivalent circuit [3].

So, having defined G_x, G_y, G_z, C, and $I_{i,j,k}$, and using the Modified Nodal Analysis method (MNA), our system becomes:

$$\mathbf{G}\mathbf{x}(t) + \mathbf{C}\frac{d\mathbf{x}(t)}{dt} = \mathbf{B}\mathbf{u}(t) \tag{2.9}$$

where,

$n \in \mathbb{R}$ is the number of nodes of the system,

$p \in \mathbb{R}$ is the number of ports of the system,

$\mathbf{G} \in \mathbb{R}^{nxn}$ is a symmetric positive-definite (SPD) matrix of the conductances,

$\mathbf{C} \in \mathbb{R}^{nxn}$ is a diagonal matrix of the cell capacitances,

$\mathbf{x} \in \mathbb{R}^{n}$ is the vector of the unknown temperatures,

$\mathbf{B} \in \mathbb{R}^{nxp}$ is the input-to-state connectivity (i.e., power distribution) matrix,

$\mathbf{u} \in \mathbb{R}^{p}$ is the vector of the input excitations from the current sources.

At this point, we have formulated a way to obtain the temperature at every node of the grid. However, in many cases, the temperature does not need to be monitored accross the whole device but only at some pre-defined points, which are denoted as monitor points. The part of $\mathbf{x}(t)$ corresponding to the monitor points can be computed by the following:

$$\mathbf{y}(t) = \mathbf{L}\mathbf{x}(t) \tag{2.10}$$

where,

$q \in \mathbb{R}$ is the number of the monitor points,

$\mathbf{y} \in \mathbb{R}^q$ is the output temperatures vector,

$\mathbf{L} \in \mathbb{R}^{qxn}$ is the state-to-outpt connectivity matrix.

So, the state-space form of our system is:

$$\begin{aligned} \mathbf{G}\mathbf{x}(t) + \mathbf{C}\frac{d\mathbf{x}(t)}{dt} &= \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{L}\mathbf{x}(t) \end{aligned} \tag{2.11}$$

2.1.3 Heat transfer coefficients

In order to solve the system of differential equations in 2.11, it must be accompanied by boundary conditions. These boundary conditions have a physical meaning as well, since they describe the heat flow between the device and the environment.

The radiative boundary condition, at a specific point, is modeled as:

$$\frac{dT}{dn} = h(T - T_o) \tag{2.12}$$

where,

$h \in \mathbb{R}$ is the heat transfer coefficient,

$T_o \in \mathbb{R}$ is the ambient temperature.

The equation 2.12 shows that the difference between the surface temperature and the ambient temperature is proportional to the heat flow between the surface and the environment. So, if we apply FDM, we obtain:

$$\frac{T_{i,j,k} - T_{i-1,j,k}}{\Delta x} = h(T_{i,j,k} - T_o) \tag{2.13}$$

By applying the discretized equation boundary condition of the HTCs (eq. 2.13) on the state-space model (eq. 2.11), it becomes:

$$\begin{aligned} (\mathbf{G} + \sum_i h_i \mathbf{A}_i)\mathbf{x}(t) + \mathbf{C}\frac{d\mathbf{x}(t)}{dt} &= \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{L}\mathbf{x}(t) + T_o \end{aligned} \tag{2.14}$$

where, $A \in \mathbb{R}^{\mathbf{nxn}}$ is a diagonal matrix that contains the surface nodes from which heat flows towards the environment. This way, the HTCs are added to the diagonal elements of the conductance matrix $\mathbf{G}$.

From this point until the end of the Thesis, although we keep the notation of the equation 2.11 for the state-space model, $\mathbf{G}$ is considered to include the HTCs.

2.2 Parametric Thermal Modeling

Parametrizing thermal models is crucial for the design and testing of ICs [32, 33, 34], as small changes of the geometry and the physical properties of the device may affect its response. However, time restrictions do not allow the re-calculation of the model for each variation of the circuit's properties de novo.

So, there is a need for a parametric discretized system that can handle the variation of such parameters so that the process of thermal modeling do not have to be implemented anew for all possible values, which can be essential for real-time applications and simulations.

With that being said, to construct the parametric model, the system in equation 2.11 should be transformed to allow the access and the modification of the parameter values inside the same representation, while avoiding re-calculating the system's equations and performing a new reduction. As a result, the state-space model can be written as:

$$\begin{aligned}\mathbf{G}(\lambda_1, ..., \lambda_m)\mathbf{x}(t, \lambda_1, ..., \lambda_m) + \mathbf{C}\frac{d\mathbf{x}(t, \lambda_1, ..., \lambda_m)}{dt} &= \mathbf{Bu}(t) \\ \mathbf{y}(t, \lambda_1, ..., \lambda_m) &= \mathbf{Lx}(t, \lambda_1, ..., \lambda_m)\end{aligned} \tag{2.15}$$

where, $\mathbf{G}, \mathbf{C} \in \mathbb{R}^{n\times n}$ are again the conductance and capacitance matrices, $\mathbf{B} \in \mathbb{R}^{n\times p}$ is the power distribution (input-to-state connectivity) matrix, and $\mathbf{L} \in \mathbb{R}^{q\times n}$ is the state-to-output connectivity matrix. The elements of $\mathbf{G}$ and $\mathbf{C}$, and as a result the states of the system $\mathbf{x}$, depend on a set of parameters $\lambda = [\lambda_1, \dots, \lambda_m]$ which model the effect of different materials that are going to be analyzed for each part of the device during thermal simulation. More specifically, each parameter λ_i corresponds to a specific conductance or capacitance scaling factor.

Chapter 3

Model Order Reduction

The complexity and the large size of some contemporary systems, render these systems' equations impossible to solve. Model order reduction (MOR) aims to lower the sheer complexity of these problems so that they can be dealt within a reasonable period of time. The methods used are divided into two main categories, system theoretic methods and the moment-matching techniques.

The system theoretic techniques, with the most common being the Balanced Truncation (BT) [27, 35, 36, 37], tend to have better accyracy since they preserve the system's stability [38]. For the same reason, it provides a global a-priori error between the transfer function of the reduced and the original model [29, 30]. The main idea behind BT is to truncate the least controllable and obervable states associated with the smallest Hankel singular values (HSVs) of the system [39, 40].

The downside of these methods is that, in order to obtain the controllability and observability matrices needed for the computation of the HSVs, one has to solve two Lyapunov equations [41, 42] of size equal to the original, which are very computationally demanding. They also involve storage of dense matrices [43, 44, 45], even if the system matrices are sparse, which is not amenable to our case, since our matrices are typically of very high order, e.g., hundrends of thousands or even millions of elements.

On the other hand, the main goal of the moment-matching methods, i.e., the Krylov subspace approaches, is to create a Krylov subspace of a much smaller order than the original system's order and then to project the system onto the specific produced subspace. The creation of the projection matrix that describes the subspace to be produced uses the moments of the original transfer functions in order to approximate the reduced ones and involves simple decompositions and linear solvers which makes the MM methods a better fit for our problem since they tend to be more time efficient.

These methods do not offer an a-priori error and the efficiency of the reduced matrices depends exclusively on the quality of the Krylov subspace produced. This is why some heuristic error bounds have been proposed in order to measure the error between the transfer functions of the original and the reduced model [46, 47, 48]. Furthermore, although they may not preserve some important proporties of the system such as the stability and the passivity, there are approaches that indeed lead to stable systems, ensuring an acceptable accuracy [49].

3.1 Moment-Matching

Consider the state-space model of equation 2.11:

$$\begin{aligned} \mathbf{C}\frac{d\mathbf{x}(t)}{dt} &= \mathbf{G}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{L}\mathbf{x}(t) \end{aligned} \tag{3.1}$$

The objective of MOR is to produce a reduced-order model:

$$\begin{aligned} \tilde{\mathbf{C}}\frac{d\tilde{\mathbf{x}}(t)}{dt} &= \tilde{\mathbf{G}}\tilde{\mathbf{x}}(t) + \tilde{\mathbf{B}}\mathbf{u}(\mathbf{t}), \\ \tilde{\mathbf{y}}(t) &= \tilde{\mathbf{L}}\tilde{\mathbf{x}}(t) \end{aligned} \tag{3.2}$$

with $\tilde{\mathbf{G}}, \tilde{\mathbf{C}} \in \mathbb{R}^{r\times r}$, $\tilde{\mathbf{B}} \in \mathbb{R}^{r\times p}$, $\tilde{\mathbf{L}} \in \mathbb{R}^{q\times r}$, where the order of the reduced model is $r << N$ and the output error $||\tilde{\mathbf{y}}(t) - \mathbf{y}(t)||_2$ is small. An equivalent metric of accuracy in the frequency domain (via Plancherel's theorem [50]) is the distance $||\tilde{\mathbf{H}}(s) - \mathbf{H}(s)||_\infty$, where

$$\begin{aligned} \mathbf{H}(s) &= \mathbf{L}(s\mathbf{C} - \mathbf{G})^{-1}\mathbf{B} \\ \tilde{\mathbf{H}}(s) &= \tilde{\mathbf{L}}(s\tilde{\mathbf{C}} - \tilde{\mathbf{G}})^{-1}\tilde{\mathbf{B}} \end{aligned}$$

are the transfer functions of the original and the reduced-order model, and $||.||_\infty$ is the induced $\mathcal{L}_2$ matrix norm (or the $\mathcal{H}_\infty$ norm of a rational transfer function).

The most important and successful MOR methods for linear systems are based on MM. They are very efficient in circuit simulation problems and are formulated in a way that has a direct application to the linear model of (3.1).

By applying the Laplace transform to (3.1), we obtain the s domain equations as:

$$\begin{aligned} s\mathbf{C}\mathbf{x}(s) - \mathbf{x}(0) &= \mathbf{G}\mathbf{x}(s) + \mathbf{B}\mathbf{u}(s) \\ \mathbf{y}(s) &= \mathbf{L}\mathbf{x}(s) \end{aligned} \tag{3.3}$$

Assuming that $\mathbf{x}(0) = 0$ and that a unit impulse is applied to $\mathbf{u}(s)$ (i.e., $\mathbf{u}(s) = 1$), then the above system of equations can be written as follows:

$$\begin{aligned}(s\mathbf{C} - \mathbf{G})\mathbf{x}(s) &= \mathbf{B} \\ \mathbf{y}(s) &= \mathbf{L}\mathbf{x}(s)\end{aligned} \tag{3.4}$$

and by expanding the Taylor series of $\mathbf{x}(s)$ around zero, we derive the following equation:

$$(s\mathbf{C} - \mathbf{G})(\mathbf{x}_0 + \mathbf{x}_1 s + \mathbf{x}_2 s^2 + \dots) = \mathbf{B} \tag{3.5}$$

The transfer function of (3.1) is a function of s, and can be expanded into a moment expansion around $s = 0$ as follows:

$$\mathbf{H}(s) = \mathbf{M}_0 + \mathbf{M}_1 s + \mathbf{M}_2 s^2 + \mathbf{M}_3 s^3 \dots \tag{3.6}$$

where, $\mathbf{M}_0$, $\mathbf{M}_1$, $\mathbf{M}_2$, $\mathbf{M}_3, \dots$ are the moments of the transfer function. Specifically, in circuit simulation problems, $\mathbf{M}_0$ is the DC solution of the linear system. This means that the inductors of the circuit are considered as short circuits and the capacitors as open circuits. Moreover, $\mathbf{M}_1$ is the Elmore delay of the linear model, which is defined as the time required for a signal at the input port to reach the output port. In general, $\mathbf{M}_i$ is related to the system matrices as:

$$\mathbf{M}_i = \mathbf{L}(\mathbf{G}^{-1}\mathbf{C})^i\mathbf{G}^{-1}\mathbf{B} \tag{3.7}$$

The goal of MM reduction techniques is the derivation of a reduced-order model where some moments $\tilde{\mathbf{M}}_i$ of the reduced-order transfer function $\tilde{\mathbf{H}}(s)$ match some moments of the original transfer function $\mathbf{H}(s)$.

3.1.1 Krylov subspaces

Let us now denote the two projection matrices onto a lower dimensional subspace as $\mathbf{V}_\ell, \mathbf{V}_r \in \mathbb{R}^{N\times r}$. These matrices can be derived from the associated moments using one or more expansion points. As a result, if we assume that $s = 0$, then the matrices $\mathbf{V}_\ell$ and $\mathbf{V}_r$ are defined as follows:

$$\begin{aligned}range(\mathbf{V}_r) &= span\{\mathbf{G}^{-1}\mathbf{B}, (\mathbf{G}^{-1}\mathbf{C})\mathbf{G}^{-1}\mathbf{B}, \dots, (\mathbf{G}^{-1}\mathbf{C})^{r-1}\mathbf{G}^{-1}\mathbf{B}\} \\ range(\mathbf{V}_\ell) &= span\{\mathbf{L}^T, (\mathbf{C}^T\mathbf{G}^{-T})\mathbf{L}^T, \dots, (\mathbf{C}^T\mathbf{G}^{-T})^{r-1}\mathbf{L}^T\}\end{aligned} \tag{3.8}$$

The computed reduced-order model matches the first $2r$ moments and is obtained by the following matrices:

$$\tilde{\mathbf{C}} = \mathbf{V}_\ell^T \mathbf{C} \mathbf{V}_r, \quad \tilde{\mathbf{G}} = \mathbf{V}_\ell^T \mathbf{G} \mathbf{V}_r, \quad \tilde{\mathbf{B}} = \mathbf{V}_\ell^T \mathbf{B}, \quad \tilde{\mathbf{L}} = \mathbf{L} \mathbf{V}_r \tag{3.9}$$

This reduced model provides a good approximation around the DC point. Finally, in case we employ a one-sided Krylov method, which is usually the case, the matrix $\mathbf{V}_\ell$ can be set equal to $\mathbf{V}_r$, an equality that also holds for symmetric systems.

3.1.2 ROM generation by the Arnoldi procedure

The Arnoldi procedure [51] that computes the projection matrix $\mathbf{V}$ begins with $\mathbf{B}_k$, and then iteratively generates a sequence of subspaces $\mathcal{K}_k(\mathbf{A}_k, \mathbf{B}_k)$ in order to compute the matrix $\mathbf{V} \in \mathbb{R}^{N \times r}$ and produce the ROM as described in (3.9).

The complete Arnoldi procedure is given in Algorithm 1, where we pass as arguments the $\mathbf{A}_k$ and $\mathbf{B}_k$ matrices. The matrix $\mathbf{B}_k$ depends on the expansion point around which we want to expand the Taylor series in order to produce the Krylov subspace. Finally, we also pass as arguments the desired order and the number of ports of the system, which are then used in step 4 in order to compute the number of moments used to match the trasnfer functions. This step can also be explained as “how many times the number of ports of the system p fits in the desired order r”.

Now that the arguments have been explained, let us analyze the Arnoldi procedure. At step 3, the initial projection matrix is computed for the first moment and the rest of the projection matrix $\mathbf{V}$ is iteratively computed for all the other moments in steps 5-12. At step 7, the projection matrix for each new moment is computed and then the function $orth\_wrt$, at step 8, performs an orthogonalization on $\mathbf{V}_1$ w.r.t. all the previous moments. Finally, in steps 9-10, the new $\mathbf{V}$ matrix is orthogonalized using a QR orthogonalization, which is then concatenated to the final $\mathbf{V}$ matrix.

Finally, before returning the produced projection matrix, we make sure at step 13 that the number of columns of this matrix matches the desired order r.

3.1.3 Multi-point moment-matching

Most popular approach towards the expansion point selection is using DC as the expansion point, i.e., expanding on zero. However, in some cases, there’s a need for multiple ex-

Algorithm 1: Arnoldi procedure for computing the projection matrix [51]

Input: $\mathbf{A}_k \equiv \mathbf{G^{-1}C}, \mathbf{B}_k \equiv (\mathbf{G+sC})^{-1}\mathbf{B}$ where s the expansion point, desired order r, #ports p

Output: $\mathbf{V}$

Function `compute_projection_matrix`($\mathbf{A}_k, \mathbf{B}_k, r, p$) **:**

  $j = 1$

  $\mathbf{V}^{(j)} = \mathtt{qr}([\mathbf{B}_E])$

  $k = \frac{r}{p}$

  while $(j < k)$ **do**

    $k_1 = p(j-1); k_2 = k_1 + p;$

    $\mathbf{V}_1 = [\mathbf{A}_k \mathbf{V}^{(j)}(:, k_1 + 1 : k_2)]$

    $\mathbf{V}_2 = \mathtt{orth\_wrt}(\mathbf{V}_1, \mathbf{V}^{(j)}, p)$

    $\mathbf{V}_3 = \mathtt{qr}(\mathbf{V}_2)$

    $\mathbf{V}^{(j+1)} = [\mathbf{V}^{(j)}, \mathbf{V}_3]$

    $j = j + 1$

  end

  $\mathbf{V} = \mathbf{V}(:, 1 : r)$

  return V

End Function

pansion points. Let $[s_1, ..., s_k]$ be a set of k distinct expansion points. There are many methodologies where complex and real expansion points are tested. However, for the purpose of this Thesis, all expansion points are real numbers. The subspace that makes use use of this set is:

$$
\begin{aligned}
range(\mathbf{V}_r) &= span\{(\mathbf{G} - s_1\mathbf{C})^{-1}\mathbf{B}, \dots, (\mathbf{G} - s_k\mathbf{C})^{-1}\mathbf{B}\} \\
range(\mathbf{V}_\ell) &= span\{(\mathbf{G} - s_1\mathbf{C})^{-T}\mathbf{L}^T, \dots, (\mathbf{G} - s_k\mathbf{C})^{-T}\mathbf{L}^T\}
\end{aligned}
\tag{3.10}
$$

The matrices $\mathbf{V}_l$ and $\mathbf{V}_r$ of Equation 3.10 can be implemented using a Rational Arnoldi method [52] which uses multiple expansion points in order to create the Krylov subspace.

Previous work (e.g [53, 54]), has made use of the poles of $\mathbf{A}_E \equiv \mathbf{G}^{-1}\mathbf{C}$ matrix in order to calculate the expansion points. Codecasa et al. [55] have built an algorithm that exploits the poles of matrix $\mathbf{A}_E$ and calculates the optimal expansion points. The essense of the expansion points algorithm is given in Algorithm 2. It should be pointed out that in step 10, $\mathtt{dn}and\mathtt{K}$ are the Jacobi elliptic function and the complete elliptic integral function respectively.

Algorithm 2: Computation of expansion points for MPMM [55]

Input: $\mathbf{G}, \mathbf{C}, \#exp\_points$

Output: exp_points

Function `get_exp_points` ($\mathbf{G}, \mathbf{C}, \#exp\_points$) **:**

  Compute inverse of $\mathbf{C}$: $\mathbf{C}_{inv} = 1./\mathbf{C}$

  $\lambda_{max} = 1/\texttt{min\_eigenvalue}(\mathbf{C}_{inv} * \mathbf{G})$

  $\lambda_{min} = 1/\texttt{max\_eigenvalue}(\mathbf{C}_{inv} * \mathbf{G})$

  $k' = \frac{\lambda_{min}}{\lambda_{max}}$

  $k = \sqrt{1 - k'^2}$

  Determine smallest integer m such that $4\texttt{exp}(-m\pi^2/log(4/k')) \leq \epsilon$

  $j = 0$

  while $j < m$ **do**

    $\sigma(j) = \lambda_{max} * \texttt{dn}((2j-1)/(2m)) * \texttt{K}(k)$

    $j = j + 1$

  end

  $exp\_points = \texttt{get\_dominant\_exp\_points}(\sigma, \#exp\_points)$

  return exp_points

End Function

3.2 Parametric Moment-Matching

Now we are going to take the problem of order reduction one step further by adding the issue of the parameters. As analyzed in Section 2.2, there is a need for parametric thermal modeling. However, standard MOR methods are not typically robust when there is a parametric model to be reduced, hence the necessity of techniques that take into consideration the variability of such parameters. Parametric Model Order Reduction (PMOR) [56] has been established for this exact reason, with first occurrences in [46, 57] .

Most of the previous work on parametric MOR has focused on moment-matching techniques since Krylov subspaces can be easily constructed w.r.t. independent parameters of the model [58, 59, 60, 61].

Similarly to (3.2), the parametric transfer function of the original model is computed as:

$$\mathbf{H}(s, \lambda_1, \ldots, \lambda_m) = \mathbf{L}(s\mathbf{C}(\lambda_1, \ldots, \lambda_m) - \mathbf{G}(\lambda_1, \ldots, \lambda_m))^{-1}\mathbf{B} \tag{3.11}$$

for which we seek to generate a ROM approximation, which will be able to accurately capture the input/output behavior of the system for any possible combination in the parameter space:

$$\tilde{\mathbf{H}}(s, \lambda_1, \ldots, \lambda_m) = \tilde{\mathbf{L}}(s\tilde{\mathbf{C}}(\lambda_1, \ldots, \lambda_m) - \tilde{\mathbf{G}}(\lambda_1, \ldots, \lambda_m))^{-1}\tilde{\mathbf{B}} \tag{3.12}$$

Generally, MOR methods try to generate a ROM whose structure is similar to the original one, i.e., exhibiting a similar parametric dependence. The most straightforward method for representing a parametric system is based on a Taylor series expansion with respect to the parameters:

$$\begin{aligned} (s(\mathbf{C}_0 + \mathbf{C}_1\lambda_1 + \cdots + \mathbf{C}_m\lambda_m) - (\mathbf{G}_0 + \mathbf{G}_1\lambda_1 + \cdots + \mathbf{G}_m\lambda_m))\mathbf{x}(s, \lambda) &= \mathbf{B}\mathbf{u}(s) \\ \mathbf{y}(s, \lambda) &= \mathbf{L}\mathbf{x}(s, \lambda) \end{aligned} \tag{3.13}$$

where, $\mathbf{C}_0$ and $\mathbf{G}_0$ contain the capacitance and conductance values corresponding to the non-parametric parts, while $\mathbf{C}_i$ and $\mathbf{G}_i$ include the capacitance and conductance values that are scaled with respect to the parameters. Using this kind of representation, with explicit parameter dependence, allows the tool to obtain a ROM, with similar representation when a projection matrix is applied

$$\begin{aligned} (s(\tilde{\mathbf{C}}_0 + \tilde{\mathbf{C}}_1\lambda_1 + \cdots + \tilde{\mathbf{C}}_m\lambda_m) - (\tilde{\mathbf{G}}_0 + \tilde{\mathbf{G}}_1\lambda_1 + \cdots + \tilde{\mathbf{G}}_m\lambda_m))\mathbf{x}(s, \lambda) &= \tilde{\mathbf{B}}\mathbf{u}(s) \\ \mathbf{y}(s, \lambda) &= \tilde{\mathbf{L}}\mathbf{x}(s, \lambda) \end{aligned} \tag{3.14}$$

where $\tilde{\mathbf{G}}_0, \tilde{\mathbf{G}}_1, \ldots, \tilde{\mathbf{G}}_m$, $\tilde{\mathbf{C}}_0, \tilde{\mathbf{C}}_1, \ldots, \tilde{\mathbf{C}}_m \in \mathbb{R}^{r \times r}$, $\tilde{\mathbf{B}} \in \mathbb{R}^{r \times p}$, and $\tilde{\mathbf{L}} \in \mathbb{R}^{q \times r}$ are the ROM matrices, with $r << N$.

Chapter 4

ROM Generation Tool

4.1 Constructing the State-Space Model

For the generation of the thermal equivalent circuit, we have parsed the corresponding comma-separated values (CSV) files, provided by Huawei, that contained the netlists, i.e., the conductance and capacitance elemtned of the model, needed. We constructed the corresponding $\mathbf{G}, \mathbf{C}$ matrices by parsing the netlists and by applying the modified nodal analysis (MNA) framework on these input files and described by a system of PDEs, as explained in equation 2.9 of Chapter 2:

$$\begin{aligned} \mathbf{G}\mathbf{x}(t) + \mathbf{C}\frac{d\mathbf{x}(t)}{dt} &= \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{L}\mathbf{x}(t) \end{aligned} \tag{4.1}$$

where

n denotes the order of the original system,

p denotes the number of power distribution sources,

$\mathbf{G} \in \mathbb{R}^{n \times n}$ is a Symmetric Positive Definite (SPD) matrix of the conductances,

$\mathbf{C} \in \mathbb{R}^{n \times n}$ is a diagonal matrix of cell capacitances,

$\mathbf{x} \in \mathbb{R}^{n}$ is the vector of unknown temperatures $T_{i,j,k}$ at all discretization points (i, j, k) (constituting the internal states of the system),

$\mathbf{B} \in \mathbb{R}^{n \times p}$ is the power distribution (input-to-state connectivity) matrix,

$\mathbf{u} \in \mathbb{R}^{p}$ is the vector of input excitations from the current sources $I_{i,j,k}$,

$\mathbf{L} \in \mathbb{R}^{q \times n}$ is the state-to-output connectivity matrix, where q is the number of monitor points,

T_0 is the ambient temperature,

$\mathbf{y} \in \mathbb{R}^q$ is the q-rows column vector of the final output temperatures.

In order to demonstrate statistics, such as sparsity and patterns, we have constructed several artificial benchmarks that represent simplified microprocessor designs with random control logic and datapath. Since in these designs, each node has a capacitor to ground and a resistance for each neighboring node, they have high sparsity ratio, e.g., $90 - 99\%$.

The sparsity paterns of the generated sparse matrices $\mathbf{G}$, $\mathbf{C}$ are shown in Figures 4.1 and 4.2. Matrix $\mathbf{G}$ is a five-diagonal matrix containing the conductances between two neighboring nodes of the circuit. Its non-zero values $3,455,074 \ll 500,000x500,000$. Matrix $\mathbf{C}$ is a diagonal matrix containing the capacitance values between each node and the ground. As one can imagine, the non-zero elements of this matrix are $500,000$. Both matrices are symmetric.

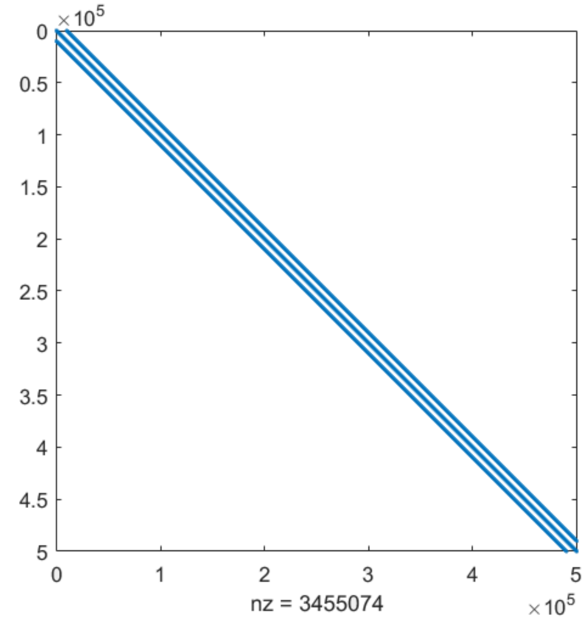

Figure 4.1: Sparsity pattern of a state-space model matrix $\mathbf{G} \in \mathbb{R}^{500Kx500K}$.

Now that the model is ready, we need to add the HTCs to the diagonal of the matrix $\mathbf{G}$. As explained in Section 2.1.3, the heat exchange through device interfaces is modeled with the radiative boundary condition (discretized boundary condition in Equation 2.13). In order to take into account the HTCs, they have to be added to the diagonal of matrix $\mathbf{G}$ and in particular only to the surface nodes on which there is a heat exchange with the environment:

$$(\mathbf{G} + \sum_i h_i \mathbf{A}_i)\mathbf{x}(t) + \mathbf{C}\frac{d\mathbf{x}(t)}{dt} = \mathbf{B}\mathbf{u}(t) \tag{4.2}$$

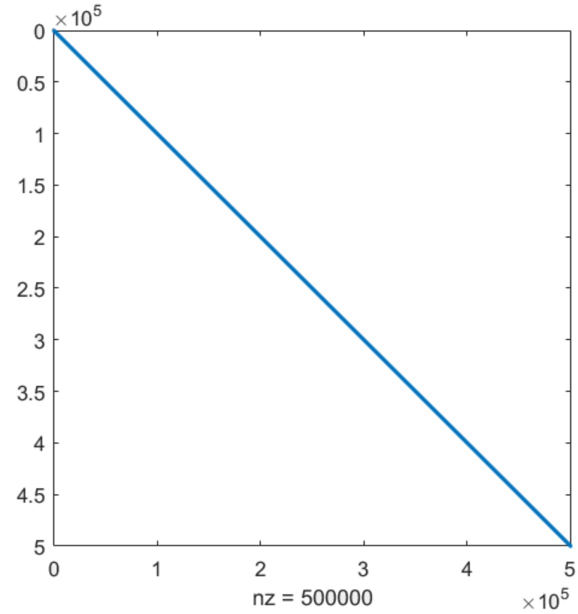

Figure 4.2: Sparsity pattern of a state-space model matrix $\mathbf{C} \in \mathbb{R}^{500Kx500K}$.

where, for each surface direction i, $\mathbf{A}_i \in R^{n\times n}$ is the diagonal matrix arising from the discretization of the convection boundary conditions on each surface node and h_i is the HTC value in each discretization area which describes the heat flow between the device and the environment.

In case of a parametric model, the procedure for the HTCs is mostly the same. The information, that we were provided, on the HTCs was the final value of the HTC to a specific direction, i.e., these values already encapsule the information of the material and it is not affected by the conductance and the capacitance factors. Also, as analyzed in Section 3.2, matrices $\mathbf{G}_0$ and $\mathbf{C}_0$ contain the non-parametric parts of the conductance and capacitance matrices, respectively, which means that they are not going to be multiplied by a factor like the rest of the parts. In conclusion, the values of the HTCs are added to the diagonal of matrix $\mathbf{G}_0$ and are dealt with the non-parametric parts of the devive.

4.2 Multi-Point Moment-Matching Procedure

In the process of developing the parametric ROM generation tool, we first created a multi-point moment matching (MPMM) procedure for non-parametric models. It is the basis of the parametric procedure that is going to be thoroughly explained in Section 4.3. The complete MPMM procedure is given in Algorithm 3, which employs Algorithm 2 for computing the optimal expansion points.

Algorithm 3: MPMM procedure for computing ROM $(\tilde{\mathbf{G}}, \tilde{\mathbf{C}}, \tilde{\mathbf{B}}, \tilde{\mathbf{L}})$

Input: $\mathbf{G}$, $\mathbf{C}$, $\mathbf{B}$, $\mathbf{L}$, desired_size r, #ports p, $moments\_per\_exp\_point$

Output: $\tilde{\mathbf{G}}, \tilde{\mathbf{C}}, \tilde{\mathbf{B}}, \tilde{\mathbf{L}}$

1 **Function** `mpmm_mor` $(\mathbf{G}, \mathbf{C}, \mathbf{B}, \mathbf{L}, r, p, moments\_per\_exp\_point)$ **:**

2 $\quad \#exp\_points = r/p$

3 $\quad exp\_points =$ `get_exp_points`$(\mathbf{C}, \mathbf{G}, \#exp\_points)$

4 $\quad \mathbf{V} = []$

5 $\quad \mathbf{A}_k = \mathbf{G}^{-1}\mathbf{C},$

6 $\quad$ **while** $i < \#exp\_points$ **do**

7 $\qquad \mathbf{B}_k = (\mathbf{G}{+}exp\_point(i){*}\mathbf{C})^{-1}\mathbf{B}$

8 $\qquad \mathbf{V}_i =$ `compute_projection_matrix`$(\mathbf{A}_k, \mathbf{B}_k, moments\_per\_exp\_point * p)$

9 $\qquad \mathbf{V} = [\mathbf{V}, \mathbf{V}_i]$

10 $\qquad i = i + 1$

11 $\quad$ **end**

12 $\quad \tilde{\mathbf{G}} = \mathbf{V}^T * \mathbf{G} * \mathbf{V}$

13 $\quad \tilde{\mathbf{C}} = \mathbf{V}^T * \mathbf{C} * \mathbf{V}$

14 $\quad \tilde{\mathbf{B}} = \mathbf{V}^T * \mathbf{B}$

15 $\quad \tilde{\mathbf{L}} = \mathbf{L} * \mathbf{V}$

16 $\quad$ **return** $\tilde{\mathbf{G}}, \tilde{\mathbf{C}}, \tilde{\mathbf{B}}, \tilde{\mathbf{L}}$

17 **End Function**

MPMM takes as input the matrices of the original model $(\mathbf{G}, \mathbf{C}, \mathbf{B}, \mathbf{L})$, the reduced size r, the number of ports p, and the number of moments $moments\_per\_exp\_point$ for each call to the Arnoldi procedure. As outputs, it returns the reduced matrices $\tilde{\mathbf{G}}, \tilde{\mathbf{C}}, \tilde{\mathbf{B}}, \tilde{\mathbf{L}}$.

The algorithm begins at step 2 by computing the number of the expansion points based on the desired reduced order and the number of ports of the system, and then it initializes the vector of the expansion points using Algorithm 2 for optimal expansion points selection (step 3). In steps 4-5, it initializes the total projection matrix $\mathbf{V}$ and the matrix $\mathbf{A}_k$ which is used in the Arnoldi procedure along with matrix $\mathbf{B}_k$.

In steps 6-10, the algorithm iteratively computes the final projection matrix, i.e., the Krylov subspace onto which the original matrices are then projected (in steps 12-15). The resulted matrices $\tilde{\mathbf{G}}, \tilde{\mathbf{C}}, \tilde{\mathbf{B}}, \tilde{\mathbf{L}}$ are the reduced order matrices, a.k.a. the output of the algorithm.

Chapter 5

Experimental Evaluation

5.1 Experimental Setup

For the experimental evaluation of our tool, we created three artificial benchmarks that represent real microprocessor designs. The characteristics of the constructed benchmarks are shown in Table 5.1. *Nodes* is the number of the discretized points on the grid, which means that the system of equations that the tool solves is of size *Nodes*, since $G, C \in \mathbb{R}^{NodesxNodes}$. *Conductances* is the number of the conductances, i.e. the connections between the nodes. *Capacitances* refer to the capacity to ground for each node, hence $Nodes = Capacitances$. *Power Sources* represents the number of ports of the circuit. Each port of the circuit is connected to a power source. *Monitor Points* is the number of nodes where we want to observe the temperature.

Table 5.1: Model Characteristics

Model	Nodes	Conductances	Capacitances	Power Sources	Monitor Points
model1	500,000	~1,500,000	500,000	10	100
model2	1,500,000	~4,000,000	1,500,000	10	100
model3	5,000,000	~15,000,000	5,000,000	10	100

The reduced models of these benchmarks were calculated using the Huawei's proprietary ROM generation tool that was implemented for the purpose of "Reduced order model generation for thermal simulation" project. The parametric ROM generation tool was tested with both procedures of DC approach and Multi-Point Moment Matching (MPMM) approach in order to evaluation their performances and compare them. In both methods, the projection matrix was built by the Arnoldi procedure, shown in Algorithm 1. In DC method, where the only expansion point is the DC (zero), for the cases where the reduced size is more than the number of ports, in the Arnoldi procedure, the number moments that the subspace is expanded are computed by $\#moments = reduced\_size/\ \#ports$. On the other hand, in MPMM approach, if the desired reduced size is set to a number larger than the number of ports, then the number of expansion points is set to $reduced\_size/\ \#ports$ and only one moment is used for the expansion of the Krylov subspace.

Both approaches were tested using *solver1* and *solver2*[1] and for three different reduced orders, 10, 20 and 70. For all cases, the tolerance of the error tolerance for both solvers, which were evalutated as well, was set to $\epsilon = 10^{-5}$. Since the solvers used are parallelizable, the tool was run with 40 threads. The total transient analysis time is 2000sec and the step is 200sec.

An example of the configuration file of the tool for model1 and desired reduced order 10 is shown in Figure 5.1. We first set the input files, then the number of threads used for the parallel solvers, the desired reduced size and finally the ambient temperature, in which case it is $273.15K$, i.e., $0°C$.

```
set_working_directory       C:/ROM_gen_tool/
set_output_directory        output/model1/

set_capacitance_file        input/model1/capacity.csv
set_conductance_file        input/model1/conductance.csv
set_power_distribution_file input/model1/power_distr.csv
set_power_table_file        input/model1/power_table.csv
set_monitor_points_file     input/model1/monitor.csv
set_htc_file                input/model1/htc.csv

//cmd spec: set_threads  <number of threads>
set_threads                 5

//cmd spec: set_reduced_size  <size> //should be a multiple of io ports
set_reduced_size      10

//cmd spec: set_ambient_temperature  <temperature> //in Kelvin
set_ambient_temperature 273.15
```

Figure 5.1: Configuration file for model1 & reduced order 10.

[1]The solvers used cannot be shown, as they are covered by an NDA agreement with Huawei Technologies

For the evaluation of the methods, we compared the responses of the transient analysis of both the original and the reduced model of the same characteristics at the monitor points. We used Backward Euler for the iterative approximation of the derivatives. Finally, the transient analysis end time was set to 2500sec and the timestep to 50sec.

The metrics we used in order to evaluate the above-mentioned approaches where the mean relative error(%), the max relative error(%) and the Reduction time (in sec). The time that was spent on transient analysis is not taken into account. All experiments were executed with C++11 on a Windows workstation, having an Intel Zeon Silver processor with 40 cores running at 2.2GHz and 128GB memory.

5.2 Experimental Results

The experimental results from the transient analysis are reported in Table 5.2, where the values in bold are the best time for the specific combination of benchmark and reduced size. The experiments that are compared to each other are all in the same line, i.e., the ones for the same model and the same desired reduced order. For example, the first experiment corresponds to testing the DC approach along with *solver1* with model1 and reduced order 10.

Table 5.2: Reduction results of transient analysis of the benchmarks

Model	Reduced Order	**DC Approach**						**MPMM Approach**					
		solver1			*solver2*			*solver1*			*solver2*		
		MRE (%)	Max RE (%)	Time (sec)	MRE (%)	Max RE (%)	Time (sec)	MRE (%)	Max RE (%)	Time (sec)	MRE (%)	Max RE (%)	Time (sec)
model1	10	0.00962	0.14998	24.390	0.00962	0.14998	**23.357**	0.00962	0.14998	24.519	0.00962	0.14998	23.853
	20	0.00019	0.00982	27.344	0.00019	0.00982	**26.993**	0.00019	0.00982	27.414	0.00019	0.00982	27.092
	70	0.00005	0.00022	130.564	0.00005	0.00022	**126.437**	0.00005	0.00022	129.677	0.00005	0.00022	126.475
model2	10	0.01264	0.18730	41.559	0.01264	0.18730	45.712	0.01264	0.18730	42.144	0.01263	0.18730	**39.713**
	20	0.00035	0.01635	**47.553**	0.00034	0.01634	50.414	0.00035	0.01633	50.263	0.00035	0.01634	49.647
	70	0.00005	0.00072	223.911	0.00005	0.00072	226.317	0.00004	0.00076	**222.006**	0.00004	0.00073	225.102
model3	10	0.01400	0.20776	331.324	0.01403	0.20776	317.412	0.01403	0.20777	326.497	0.01405	0.20778	**316.765**
	20	0.00043	0.02348	377.179	0.00047	0.02343	376.274	0.00043	0.02348	**374.590**	0.00045	0.02344	376.929
	70	0.00018	0.00091	1748.300	0.00015	0.00097	1731.650	0.00011	0.00092	**1682.150**	0.00011	0.00092	1742.110

By observing the table, one can immediately notice the exceptionally low errors, both for mean and max relative errors with acceptable reduction times. The significance of these results can become even greater considering that the reduction shown in this table is between 98.74% and 99.999813% for original models of sizes 500K and 5M with desired reduced orders of 70 and 10 respectively.

In regard to the errors between the various approaches and solvers, the differences are negligible, especially for the smaller benchmarks. For example, all the experiments concerning model1 and desired reduced order of 70 provide the same mean and max errors. This was expected, since we made use of established methods.

Finally, by noticing the pattern of the best times, the conclusion that can be made is that for relatively small benchmarks, the DC approach in combination with *solver2* is the best choice. On the other hand, for very large benchmarks, the MPMM approach is clearly the best choice. Also, the *solver1* solver seems to outrun *solver2* in most scenarios.

In general, the differences in performance and accuracy between the approaches are almost indistinguishable. The reduction percentage varies from 98.74% (model1 with reduced size 70, Figure 5.2 shows the transient analysis of a monitor for this experiment) to 99.999813% (model3 with reduced size 10, a transient plot of this experiment is shown in Figure 5.3) with the corresponding max relative errors 0.00005% and 0.20777% and reduction times 126.437sec and 316.765sec.

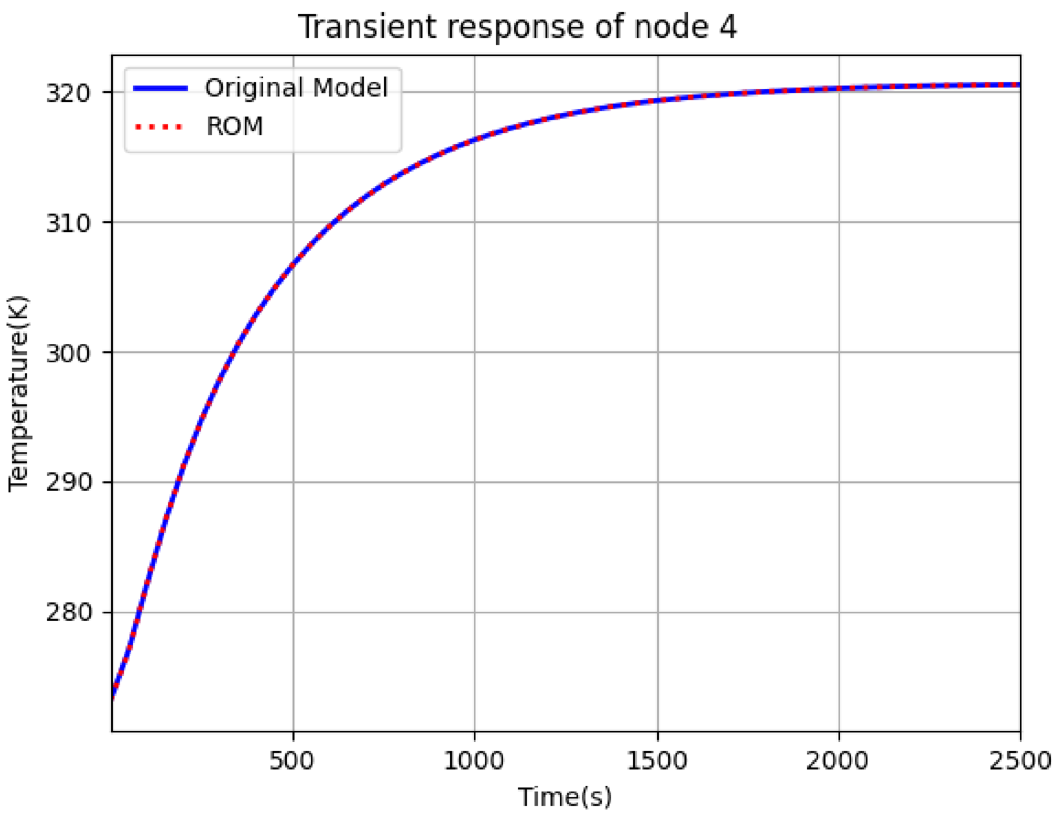

Figure 5.2: Transient analysis of model1 - desired order 70 - *solver2*.

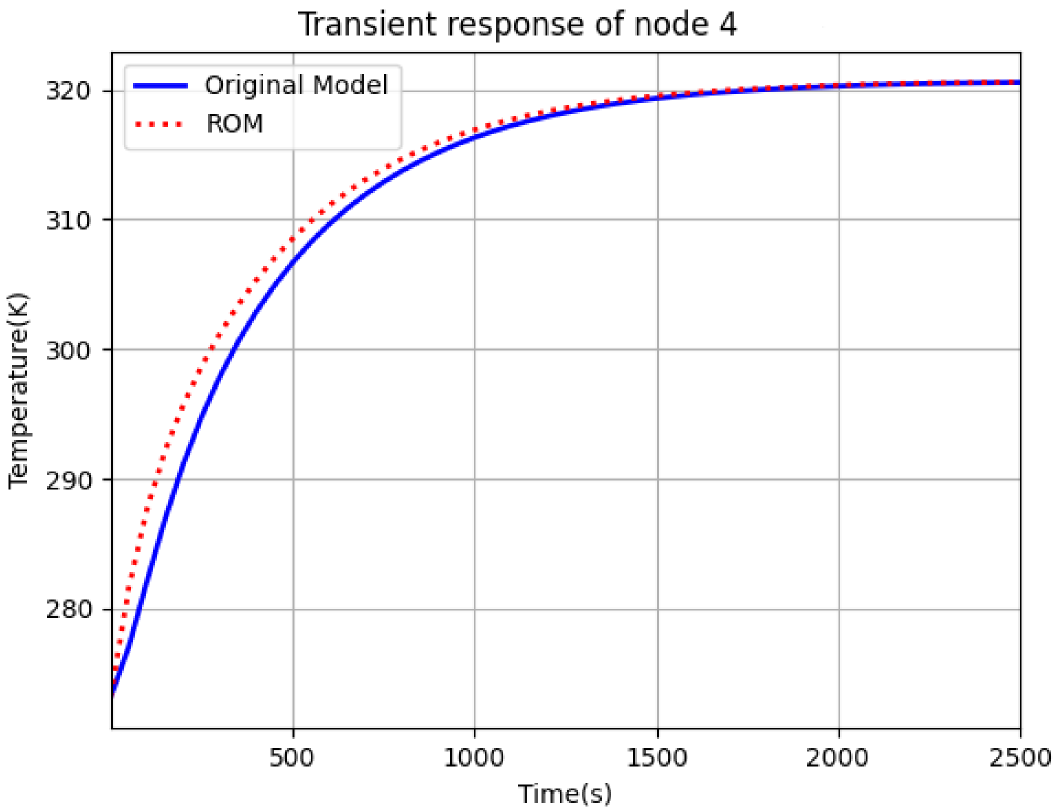

Figure 5.3: Transient analysis of model3 - desired order 10 - *solver2*.

Chapter 6

Conclusions and Future Work

In this Thesis, we presented a parametric ROM generation tool that handles real-world large thermal models, up to millions of elements. We implemeted two procedures for the expansion points selection, a multi-point moment matching (MPMM) and a single point moment matching (MM) procedure that were tested based on their accuracy and performance. Furthermore, the developed procedures have been optimized by exploiting state-of-the-art (direct and iterative) solvers.

Experimental results show that both approaches (MPMM and single point MM) have almost identical accuracy but with some small differences in performance. For relatively small models (e.g., of 500 thousands states), a single expansion point with multiple moments has better accuracy than multiple expansion points. For large models, it is evident that there is a need for more expansion points. Regarding the solvers, they have very small - almost negligible - differences in both accuracy and performance. Overall, our tool achieves a model reduction of 98.74% with 0.00005% Max Relative Error and reduction time of 126.537sec for a model of 500k nodes and desired reduced order 70.

In regard to future work, neural networks can be applied to our tool to improve its performance in reduction time, since the complexity of a prediction is $O(1)$. Moreover, according to recent developments, Physics Informed Neural Networks (PINN) [62] are very efficient in computing the temperature response of the transient analysis with acceptable accuracy. Finally, as an extension to our tool, it can be modified in order to handle industrial geometry description files (e.g., STEP) and construct the MNA system of equations.

References

[1] C. Prasad, S. Ramey, and L. Jiang. Self-heating in advanced cmos technologies. In *2017 IEEE International Reliability Physics Symposium (IRPS)*, pages 6A–4.1–6A–4.7, 2017.

[2] Wei Huang, Eric Humenay, Kevin Skadron, and Mircea R. Stan. The need for a full-chip and package thermal model for thermally optimized ic designs. In *Proceedings of the 2005 international symposium on Low power electronics and design - ISLPED '05*, ISLPED ’05, page 245–250, New York, NY, USA, 2005. Association for Computing Machinery.

[3] George Floros, Nestor Evmorfopoulos, and George Stamoulis. Efficient ic hotspot thermal analysis via low-rank model order reduction. *Integration*, 66:1–8, 2019.

[4] R.R. Schaller. Moore’s law: past, present and future. *IEEE Spectrum*, 34(6):52–59, 1997.

[5] Mark Bohr. A 30 year retrospective on dennard’s mosfet scaling paper. *IEEE Solid-State Circuits Society Newsletter*, 12(1):11–13, 2007.

[6] Saurabh Sinha, Xiaoqing Xu, Mudit Bhargava, Shidhartha Das, Brian Cline, and Greg Yeric. Stack up your chips: Betting on 3d integration to augment moore’s law scaling. 2020.

[7] Liming Xiu. Time moore: Exploiting moore’s law from the perspective of time. *IEEE Solid-State Circuits Magazine*, 11(1):39–55, 2019.

[8] Chris Edwards. Moore’s law: What comes next? *Commun. ACM*, 64(2):12–14, January 2021.

[9] John Shalf. The future of computing beyond moore's law. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 378(2166):20190061, Jan 2020.

[10] Taiwan semiconductor manufacturing company. `https://www.tsmc.com/`.

[11] J.C. Liu, S. Mukhopadhyay, Amit Kundu, S.H. Chen, H.C. Wang, D.S. Huang, J.H. Lee, M.I. Wang, Ryan Lu, S.S. Lin, Y.M. Chen, H.L. Shang, P.W. Wang, H.C. Lin, Geoffrey Yeap, and Jun He. A reliability enhanced 5nm cmos technology featuring 5th generation finfet with fully-developed euv and high mobility channel for mobile soc and high performance computing application. In *2020 IEEE International Electron Devices Meeting (IEDM)*, pages 9.2.1–9.2.4, 2020.

[12] Johann Knechtel, Ozgur Sinanoglu, Ibrahim (Abe) Elfadel, Jens Lienig, and Cliff Sze. Large-scale 3d chips: Challenges and solutions for design automation, testing, and trustworthy integration. *IPSJ Transactions on System LSI Design Methodology*, 10:45–62, 08 2017.

[13] Dick James. 3d ICs in the real world. In *25th Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC 2014)*. IEEE, May 2014.

[14] Dong Hyuk Woo, Nak Hee Seong, Dean L. Lewis, and Hsien-Hsin S. Lee. An optimized 3d-stacked memory architecture by exploiting excessive, high-density tsv bandwidth. In *HPCA - 16 2010 The Sixteenth International Symposium on High-Performance Computer Architecture*, pages 1–12, 2010.

[15] Mottaqiallah Taouil, Said Hamdioui, Kees Beenakker, and Erik Jan Marinissen. Test impact on the overall die-to-wafer 3d stacked ic cost. *J. Electron. Test.*, 28(1):15–25, February 2012.

[16] Simon Wong, Abbas El-Gamal, Peter Griffin, Yoshio Nishi, Fabian Pease, and James Plummer. Monolithic 3d integrated circuits. In *2007 International Symposium on VLSI Technology, Systems and Applications (VLSI-TSA)*. IEEE, April 2007.

[17] Guo Wang. Effect of edge-hydrogen passivation and saturation on the carrier mobility of armchair graphene nanoribbons. *Chemical Physics Letters*, 533:74–77, 2012.

[18] Finn Jensen. Activation energies and the arrhenius equation. *Quality and Reliability Engineering International*, 1(1):13–17, 1985.

[19] K. Roy, S. Mukhopadhyay, and H. Mahmoodi-Meimand. Leakage current mechanisms and leakage reduction techniques in deep-submicrometer cmos circuits. *Proceedings of the IEEE*, 91(2):305–327, 2003.

[20] Z. Liu and V. Kursun. Leakage power characteristics of dynamic circuits in nanometer cmos technologies. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 53(8):692–696, 2006.

[21] Wei Huang, Mircea R. Stan, Kevin Skadron, Karthik Sankaranarayanan, Shougata Ghosh, and Sivakumar Velusam. Compact thermal modeling for temperature-aware design. In *Proceedings of the 41st Annual Design Automation Conference*, DAC '04, page 878–883, New York, NY, USA, 2004. Association for Computing Machinery.

[22] T. N. Narasimhan. Fourier's heat conduction equation: History, influence, and connections. *Reviews of Geophysics*, 37(1):151–172, 1999.

[23] Peng Li, L.T. Pileggi, M. Asheghi, and R. Chandra. Ic thermal simulation and modeling via efficient multigrid-based approaches. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 25(9):1763–1776, 2006.

[24] Scott Ladenheim, Yi-Chung Chen, Milan Mihajlović, and Vasilis Pavlidis. Ic thermal analyzer for versatile 3-d structures using multigrid preconditioned krylov methods. In *2016 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pages 1–8, 2016.

[25] Yong Zhan and Sachin S. Sapatnekar. High-efficiency green function-based thermal simulation algorithms. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 26(9):1661–1675, 2007.

[26] Gilles Fraisse, Christelle Viardot, Olivier Lafabrie, and Gilbert Achard. Development of a simplified and accurate building model based on electrical analogy. *Energy and Buildings*, 34(10):1017–1031, 2002.

[27] B. Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Transactions on Automatic Control*, 26(1):17–32, 1981.

[28] L.T. Pillage and R.A. Rohrer. Asymptotic waveform evaluation for timing analysis. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 9(4):352–366, 1990.

[29] Dale F. Enns. Model reduction with balanced realizations: An error bound and a frequency weighted generalization. In *The 23rd IEEE Conference on Decision and Control*, pages 127–132, 1984.

[30] Keith Glover. All optimal hankel-norm approximations of linear multivariable systems and their l, ∞ -error bounds†. *International Journal of Control*, 39(6):1115–1193, 1984.

[31] A. Odabasioglu, M. Celik, and L.T. Pileggi. Prima: passive reduced-order interconnect macromodeling algorithm. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 17(8):645–654, 1998.

[32] Saeed Farahany, Ali Ourdjini, Mohd Idris, and S.G. Shabestari. Computer-aided cooling curve thermal analysis of near eutectic al–si–cu–fe alloy. *Journal of Thermal Analysis and Calorimetry*, 114, 11 2013.

[33] Jaeho Lee, David W. Gerlach, and Yogendra K. Joshi. Parametric thermal modeling of heat transfer in handheld electronic devices. In *2008 11th Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems*, pages 604–609, 2008.

[34] Xiaojun Dong, Antonio Griffo, and Jiabin Wang. Multiparameter model order reduction for thermal modeling of power electronics. *IEEE Transactions on Power Electronics*, 35(8):8550–8558, 2020.

[35] Serkan Gugerciny and Athanasios Antoulasy. A survey of model reduction by balanced truncation and some new results. *International Journal of Control - INT J CONTR*, 77, 05 2004.

[36] C. Mullis and R. Roberts. Synthesis of minimum roundoff noise fixed point digital filters. *IEEE Transactions on Circuits and Systems*, 23(9):551–562, 1976.

[37] Peter Benner. *Advances in Balancing-Related Model Reduction for Circuit Simulation*, volume 14, pages 469–482. 01 2010.

[38] L. Pernebo and L. Silverman. Model reduction via balanced state space representations. *IEEE Transactions on Automatic Control*, 27(2):382–387, 1982.

[39] Chuili Sun and Juergen Hahn. Parameter reduction for stable dynamical systems based on hankel singular values and sensitivity analysis. *Chemical Engineering Science*, 61(16):5393–5403, 2006.

[40] A.C. Antoulas, D.C. Sorensen, and Y. Zhou. On the decay rate of hankel singular values and related issues. *Systems & Control Letters*, 46(5):323–342, 2002.

[41] V. Simoncini. A new iterative method for solving large-scale lyapunov matrix equations. *SIAM J. Sci. Comput.*, 29:1268–1288, 2007.

[42] Eric Chu, Hung-Yuan Fan, and Wen-Wei Lin. Projected generalized discrete-time periodic lyapunov equations and balanced realization of periodic descriptor systems. *SIAM J. Matrix Analysis Applications*, 29:982–1006, 01 2007.

[43] José Badía, Peter Benner, Rafael Mayo, and Enrique S Quintana-Orti. Solving large sparse lyapunov equations on parallel computers. 08 2002.

[44] Stefano Massei, Davide Palitta, and Leonardo Robol. Solving rank-structured sylvester and lyapunov equations. *SIAM Journal on Matrix Analysis and Applications*, 39, 11 2017.

[45] Maximilian Behr, Peter Benner, and Jan Heiland. Solution formulas for differential sylvester and lyapunov equations. *Calcolo*, 56, 12 2019.

[46] Eric Grimme. *Krylov projection methods for model reduction*. PhD thesis, 05 1997.

[47] Tamara Bechtold, Evgenii Rudnyi, and Jan Korvink. Error indicators for fully automatic extraction of heat-transfer macromodels for mems. *Journal of Micromechanics and Microengineering*, 15:430, 12 2004.

[48] Heiko Peuscher, Thomas Wolf, and Boris Lohmann. h_2 and h_∞ error bounds for model order reduction of second order systems by krylov subspace methods. pages 4484–4489, 07 2013.

[49] Athanasios Antoulas, Danny Sorensen, and Serkan Gugercin. A survey of model reduction methods for large systems. *Contemp. Math*, 280, 11 2006.

[50] Thomas Muirhead Flett. *Mathematical analysis*. McGraw-Hill, 1966.

[51] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, third edition, 1996.

[52] S. Gugercin, A. Antoulas, and C. Beattie. H2 model reduction for large-scale linear dynamical systems. *SIAM J. Matrix Anal. Appl.*, 30:609–638, 2008.

[53] Ngoc Kien Vu and Hong Quang Nguyen. Model order reduction algorithm based on preserving dominant poles. *International Journal of Control, Automation and Systems*, 19(6):2047–2058, Jun 2021.

[54] Moomal Bukhari, Mukhtar Ullah, Mukhtar Ali, and Jamshed Iqbal. Model order reduction of dynamical systems: An approach to investigate real and complex poles. In *2017 13th International Conference on Emerging Technologies (ICET)*, pages 1–6, 2017.

[55] Lorenzo Codecasa, Vincenzo d'Alessandro, Alessandro Magnani, Niccolo Rinaldi, and Peter J. Zampardi. Fast novel thermal analysis simulation tool for integrated circuits (FANTASTIC). In *20th International Workshop on Thermal Investigations of ICs and Systems*. IEEE, September 2014.

[56] Abid Bazaz, Mashuq Nabi, and Janardhanan Sivaramakrishnan. A review of parametric model order reduction techniques. 03 2012.

[57] D.S. Weile, E. Michielssen, E. Grimme, and K. Gallivan. A method for generating rational interpolant reduced order models of two-parameter linear systems. *Applied Mathematics Letters*, 12(5):93–102, 1999.

[58] Peter Benner and Lihong Feng. *A Robust Algorithm for Parametric Model Order Reduction Based on Implicit Moment Matching*, pages 159–185. 01 2014.

[59] R. Eid, B. Salimbahrami, B. Lohmann, E. Rudnyi, and J. Korvink. Parametric order reduction of proportionally damped second order systems. 2006.

[60] Alfred Leung and R. Khazaka. Parametric model order reduction technique for design optimization. volume 2, pages 1290 – 1293 Vol. 2, 06 2005.

[61] Lihong Feng, Evgenii Rudnyi, and Jan Korvink. Preserving the film coefficient as a parameter in the compact thermal model for fast electrothermal simulation. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 24:1838 – 1847, 01 2006.

[62] Shengze Cai, Zhicheng Wang, Sifan Wang, Paris Perdikaris, and George Em Karniadakis. Physics-informed neural networks for heat transfer problems. *Journal of Heat Transfer*, 143(6), Apr 2021. 060801.