

© 2017 Gene Shiue

LINEAR-TIME 3-D THERMAL METHODS FOR TRANSIENT
ELECTRO-THERMAL SIMULATIONS

BY

GENE SHIUE

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2017

Urbana, Illinois

Adviser:

Professor José E. Schutt-Ainé

ABSTRACT

Both users and industries demand devices and systems with higher performances and better reliabilities. To achieve both, an electro-thermal simulator is needed as the thermal aspect of a device, such as the temperature, can play a big role on its reliability. By doing just the electrical simulation, an engineer cannot possibly determine the temperature of operation, and thus cannot know whether the design is reliable.

The approach of this work is to separate an electro-thermal simulator into two components, one takes care of the electrical part while the other one takes care of the thermal part. The main focus of this work is the thermal simulator. Two thermal simulators are discussed: the latency insertion method (LIM) and the Douglass-Gunn method (DGM). These two methods are chosen because they have linear complexity, which is valuable when doing a simulation on a large system from the simulation time perspective. In-depth formulations are covered for these two methods.

The problem of interest is a large metal-oxide semiconductor field-effect transistor (MOSFET). Simulation results of both LIM and DGM are provided and validated using Ansys Icepak, a commercially available thermal analysis tool.

Lastly, some comparisons and future work are provided, to improve results and take further steps from this work.

To my family, for their love and support.

ACKNOWLEDGMENTS

Foremost, I would like to thank my family for their love, support, and guidance. Without them, getting an Electrical and Computer Engineering degree at University of Illinois at Urbana-Champaign would not be possible.

I would like to thank my advisor, Professor José Schutt-Ainé, for providing suggestions and research directions that allowed me to expand my knowledge in this field of study. I appreciate all of the help and opportunities that were given to me.

Also, I would like to thank my friends in the research office. Thanks to Xu Chen, Xiao Ma, and Thong Nguyen not only for the help and motivation toward finishing the MS degree, but also for making this graduate studies journey enjoyable.

Finally, I thank all of the people that I have met so far for their assistance with this thesis. They helped my personal and academic development and made contributions to everything that I do.

TABLE OF CONTENTS

LIST OF TABLES	vi
LIST OF FIGURES	vii
LIST OF ABBREVIATIONS	viii
CHAPTER 1 INTRODUCTION	1
1.1 Motivation	1
1.2 Electrical Simulator	2
1.3 Thermal Simulator	2
1.4 Outline	2
CHAPTER 2 LATENCY INSERTION METHOD	4
2.1 Formulation	4
2.2 Stability of LIM	6
2.3 Thermal LIM	7
2.4 Boundary Conditions	9
CHAPTER 3 DOUGLAS-GUNN METHOD	12
3.1 Formulation	12
3.2 Boundary Conditions	15
CHAPTER 4 EXAMPLES AND RESULTS	16
4.1 Structure of Interest	16
4.2 LIM Validation	17
4.3 DGM Validation	22
4.4 Effect of Fictitious Inductance	26
4.5 LIM and DGM Comparison	29
CHAPTER 5 CONCLUSION AND FUTURE WORK	31
5.1 Conclusion	31
5.2 Future Work	31
REFERENCES	33

LIST OF TABLES

4.1	Material Properties for MOSFET Model	17
4.2	Dimensions of Simulated MOSFET Example	17
4.3	Error of DG for Different Δt	22

LIST OF FIGURES

2.1	Branch and node models for LIM simulation.	5
2.2	Analogy between electrical and thermal domain for LIM.	8
2.3	Equivalent RC model in the electrical domain for heat conduction problems.	10
2.4	Locations of equivalent resistances in the electrical domain for heat conduction problems and equivalent convective boundary condition.	11
4.1	The MOSFET structure for simulation examples.	17
4.2	Temperature of the MOSFET structure using LIM and Ansys Icepak.	19
4.3	Differences of temperature from LIM and Ansys Icepak.	20
4.4	Percentage errors of temperature from LIM and Ansys Icepak.	21
4.5	Temperature of the MOSFET structure using DGM and Ansys Icepak.	23
4.6	Differences of temperature from DGM and Ansys Icepak.	24
4.7	Percentage errors of temperature from DGM and Ansys Icepak.	25
4.8	A LIM simulation with low fictitious inductance.	27
4.9	A LIM simulation with high fictitious inductance.	27
4.10	A DGM simulation with small time step.	28
4.11	A DGM simulation with large time step.	28
4.12	Temperature at the bottom of transistor simulated by LIM, Ansys Icepak, and DGM.	29
4.13	Temperature at the top of transistor simulated by LIM, Ansys Icepak, and DGM.	30
4.14	Temperature at the top of transistor simulated by LIM, Ansys Icepak, and DGM, zoomed in.	30

LIST OF ABBREVIATIONS

ADI	Alternating Direction Implicit
DGM	Douglas-Gunn Method
LIM	Latency Insertion Method
MOSFET	Metal-Oxide Semiconductor Field-Effect Transistor
SiO ₂	Silicon Dioxide
SPICE	Simulation Program with Integrated Circuit Emphasis

CHAPTER 1

INTRODUCTION

1.1 Motivation

As the reliability requirements for devices become more strict, the heat generated by the system can become an important aspect to investigate. For example, a small device cannot contain as much heat due to smaller volume and cannot be cooled as fast with natural convection due to a smaller convective area. Another example is that if a device operates at high voltage, then it probably also produces a lot of heat, and its electrical properties could change when operating at high temperature. In the extreme case where the temperature of a device is too high, the device may be permanently damaged. Even if the device can withstand the high temperature, the electrical properties may change and the electrical performances may be altered so that it no longer meets the reliability requirements. Therefore, thermal analysis is needed to ensure the reliabilities of devices in addition to electrical simulations. A simulator that can capture both thermal and electrical phenomena can help to assess the performance of engineers' designs more accurately while operating in the physical world.

There are several steps of building such a simulator that can do electro-thermal simulations. First, one must choose the desired electrical and thermal simulation methods. Second, some connections between those simulation methods need to be established.

There will be a brief discussion of electrical simulators. However, the main focus of this thesis is on the thermal simulators.

1.2 Electrical Simulator

There are many ways to perform electrical simulations. The Simulation Program with Integrated Circuit Emphasis (SPICE) [1] is widely accepted by industries. SPICE uses the modified nodal analysis which involves inverting a matrix to get simulation results. This matrix inversion has a complexity of $n^{1.x}$ where n is the number of nodes. Another way to do such an electrical simulation is to use the latency insertion method (LIM) [2]. The LIM can produce simulation results with complexity proportional to the number of nodes. As the number of nodes increases, the linear time solver can produce results much faster than a super-linear time solver. Therefore, the LIM allows the electrical simulation to be done comparatively faster.

Another advantage of the LIM is that the method can do transient analysis by its natural formulations. In other words, it is capable of doing transient analysis without any modifications. To do an electro-thermal transient analysis, the electrical simulator should be able to produce transient results.

Due to LIM's advantages, it has been chosen as the electrical simulator portion of the electro-thermal simulator. To develop an electro-thermal simulator for steady state analysis, the LIM has demonstrated the abilities to do steady state analysis as well [3].

1.3 Thermal Simulator

In a similar manner to electrical simulations, there are many ways to perform thermal simulations. This thesis mainly focuses on two 3-D finite difference methods in performing thermal simulations that are both in linear time: LIM and the Douglas-Gunn method (DGM) [4]. Being able to solve problems in linear time is important, because when the system gets large, a superlinearity time method may result in a much longer runtime.

1.4 Outline

The two thermal simulation methods, LIM and DGM, will be covered extensively in this thesis.

Chapter 2 discusses the formulation, stability analysis, and boundary conditions of the LIM. This includes the requirement of the LIM models and discusses the capabilities of the method to simulate in both electrical and thermal domains.

Chapter 3 discusses the formulation and boundary conditions of the DGM. This is a variation of the Crank-Nicolson method that is favored for 3-D thermal analysis due to its speed and stability.

Chapter 4 provides simulation results of the problem of interest: a large metal-oxide semiconductor field-effect transistor (MOSFET). Both results for LIM and DGM are presented. Some comparisons of the two methods are made. The two methods are validated with Ansys Icepak, which is a commercially available product for conducting thermal analysis.

Chapter 5 contains the conclusion and future work of the thesis. Some steps are suggested to improve the accuracy of the thermal simulation methods and to finally completing an electro-thermal simulator.

CHAPTER 2

LATENCY INSERTION METHOD

2.1 Formulation

The LIM [2] is a finite-difference numerical technique to simulate circuits that can be modeled as various nodes and interconnecting branches between the nodes with resistors, inductors, and capacitors. As stated previously, this method is fast and produces results with linear complexity with respect to the number of nodes in the model.

The models of branches and nodes are shown in Fig. 2.1 [5]. Each branch (from node i to node j) is modeled as a series of a resistor R_{ij} and an inductor L_{ij} , and each node is modeled as a parallel combination of a conductance G_i , a capacitance C_i , and a current source H_i to ground.

By applying Ohm's law for the branch model on the left of Fig. 2.1, the following expression between node voltages and the branch current can be established as

$$V_i - V_j = I_{ij}R_{ij} + L_{ij}\frac{\partial I_{ij}}{\partial t}. \quad (2.1)$$

Similarly, by applying Kirchhoff's current law (KCL) for the node model on the right of Fig. 2.1, the following expression between branch currents and the node voltage can be established as

$$-\sum_{k=1}^{B_i} I_{ik} = C_i\frac{\partial V}{\partial t} + G_iV_i - H_i, \quad (2.2)$$

where B_i is the total number of branches that are connected to node i .

However, since numerical methods cannot work with continuous equations like Equations (2.1) and (2.2), these two equations need to be discretized. By substituting in the central difference formula of the finite difference method

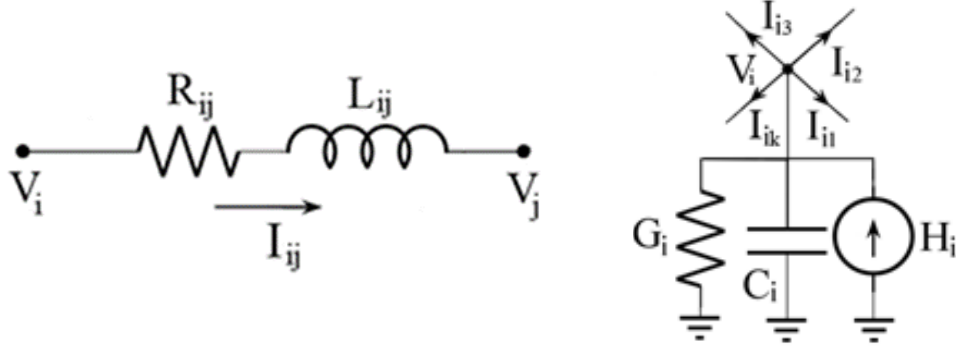


Figure 2.1: Branch and node models for LIM simulation.

which is second-order accurate [6], given by

$$\frac{df}{dx} = \frac{f(x + \Delta x) - f(x - \Delta x)}{2\Delta x}, \quad (2.3)$$

Equations (2.1) and (2.2) can be rewritten as:

$$V_i^{n+1/2} - V_j^{n+1/2} = R_{ij} \frac{I_{ij}^{n+1} + I_{ij}^n}{2} + L_{ij} \frac{I_{ij}^{n+1} - I_{ij}^n}{\Delta t} \quad (2.4)$$

and

$$-\sum_{k=1}^{B_i} I_{ik}^n = C_i \frac{V_i^{n+1/2} - V_i^{n-1/2}}{\Delta t} + G_i \frac{V_i^{n+1/2} + V_i^{n-1/2}}{2} - H_i^n, \quad (2.5)$$

where the superscripts denote discrete time. One thing to note is that similar to Yee's finite-difference time-domain scheme [7], voltages are computed at half time steps ($t = (n + 1/2)\Delta t$) and currents are computed at whole time steps ($t = n\Delta t$). This way, LIM can be viewed as a “leapfrog” scheme that solves voltages and currents in the alternating fashion. Also notable is that the I_{ij} term in Equation (2.1) is taken to be the average of I_{ij}^{n+1} and I_{ij}^n , and V_i in Equation (2.2) is taken to be the average of $V_i^{n+1/2}$ and $V_i^{n-1/2}$. This is part of the semi-implicit LIM formulation [8].

Finally, the updating equations are obtained by rearranging the current or the voltage term of the next time step in Equations (2.4) and (2.5) as the only term on the left-hand side, namely

$$I_{ij}^{n+1} = 2\Delta t \frac{V_i^{n+1/2} - V_j^{n+1/2}}{2L_{ij} + R\Delta t} + I_{ij}^n \frac{2L_{ij} - R\Delta t}{2L_{ij} + R\Delta t} \quad (2.6)$$

and

$$V_i^{n+1/2} = V_i^{n-1/2} \frac{2C_i - G_i \Delta t}{2C_i + G_i \Delta t} - 2\Delta t \frac{\sum_{k=1}^{B_i} I_{ik}^n - H_i^n}{2C_i + G_i \Delta t}. \quad (2.7)$$

During each time step, an iteration over all the nodes and branches is performed. The complexity of the LIM is therefore linear with respect to the number of nodes.

2.2 Stability of LIM

One way of evaluating the stability of LIM is suggested in [8]. Equations (2.6) and (2.7) can be written in the matrix form

$$\mathbf{V}^{n+1/2} = \left(\frac{\mathbf{C}}{\Delta t} + \frac{\mathbf{G}}{2} \right)^{-1} \left[\left(\frac{\mathbf{C}}{\Delta t} - \frac{\mathbf{G}}{2} \right) \mathbf{V}^{n-1/2} + \mathbf{H}^n - \mathbf{M} \mathbf{I}^n \right] \quad (2.8)$$

and

$$\mathbf{I}^{n+1} = \left(\frac{\mathbf{L}}{\Delta t} + \frac{\mathbf{R}}{2} \right)^{-1} \left[\left(\frac{\mathbf{L}}{\Delta t} - \frac{\mathbf{R}}{2} \right) \mathbf{I}^n + \mathbf{M}^T \mathbf{V}^{n+1/2} \right], \quad (2.9)$$

where \mathbf{V} and \mathbf{I} are voltage and current vectors at different times according to the superscript, \mathbf{C} , \mathbf{G} , \mathbf{L} , and \mathbf{R} are diagonal matrices of RLGC values at each node, \mathbf{H} is a diagonal matrix of the current source going into each node, and \mathbf{M} defined as

$M_{ij} = 1$ if branch j incidents on node i while the current flows away from the same node.

$M_{ij} = -1$ if branch j incidents on node i while the current flows toward the same node.

$M_{ij} = 0$ if branch j does not incident on node i .

Equations (2.8) and (2.9) can be rewritten into a discrete linear time invariant system representation with source terms omitted:

$$\begin{bmatrix} \mathbf{V}^{n+1/2} \\ \mathbf{I}^{n+1} \end{bmatrix} = \mathbf{A} \begin{bmatrix} \mathbf{V}^{n-1/2} \\ \mathbf{I}^n \end{bmatrix}, \quad (2.10)$$

where

$$\mathbf{A} = \begin{bmatrix} \mathbf{P}_+\mathbf{P}_- & -\mathbf{P}_+\mathbf{P} \\ \mathbf{Q}_+\mathbf{M}^T\mathbf{P}_+\mathbf{P}_- & \mathbf{Q}_+\mathbf{Q}_- - \mathbf{Q}_+\mathbf{M}^T\mathbf{P}_+\mathbf{M} \end{bmatrix} \quad (2.11)$$

and

$$\begin{aligned} \mathbf{P}_+ &= \left(\frac{\mathbf{C}}{\Delta t} + \frac{\mathbf{G}}{2} \right)^{-1} & \mathbf{P}_- &= \left(\frac{\mathbf{C}}{\Delta t} - \frac{\mathbf{G}}{2} \right) \\ \mathbf{Q}_+ &= \left(\frac{\mathbf{L}}{\Delta t} + \frac{\mathbf{R}}{2} \right)^{-1} & \mathbf{Q}_- &= \left(\frac{\mathbf{L}}{\Delta t} - \frac{\mathbf{R}}{2} \right). \end{aligned} \quad (2.12)$$

For a system like Equation (2.10) to be asymptotically stable, the spectral radius of \mathbf{A} must be less than 1. One observation is that the spectral radius of \mathbf{A} depends only on the circuit to be solved and Δt . Therefore, for every problem, there is a maximum Δt one can choose for the scheme to be stable.

Even though the above method can get the absolute upper bound of Δt by sweeping through the possible range of Δt 's, finding the eigenvalues of matrix \mathbf{A} can be computationally expensive. An alternative way using the energy method is proposed in [9]. Using the method, the upper bound of Δt is given by

$$\Delta t < \sqrt{2} \min_{i=1}^{N_n} \left(\sqrt{\frac{C_i}{N_b^i} \min_{p=1}^{N_b^i} L_{i,p}} \right), \quad (2.13)$$

where N_n and N_b are the total numbers of nodes and branches of the system, C_i is the capacitance to ground at node i , N_b^i is the number of branches incident to node i , and $L_{i,p}$ is the inductance for the p th branch that incidents to node i .

This shows that LIM requires latency in the circuit. If there is no capacitance to ground for a node, or if there is no inductance for a branch, a fictitious value should be added to enable the method.

2.3 Thermal LIM

The analysis for LIM so far is for electrical simulation. However, in general, the governing equation for a heat conduction problem is

$$\frac{\partial u}{\partial t} = \frac{k}{c_p \rho} \nabla^2 u + \frac{g}{c_p \rho}, \quad (2.14)$$

<u>Thermal</u>			<u>Electrical</u>	
Temperature	T [K]	\Leftrightarrow	Voltage	V [V]
Heat	Q [J]	\Leftrightarrow	Charge	Q [C]
Heat transfer rate	q [W]	\Leftrightarrow	Current	i [A]
Thermal resistance	R_T [K/W]	\Leftrightarrow	Electrical resistance	R [V/A]
Thermal capacitance	C_T [J/K]	\Leftrightarrow	Electrical capacitance	C [C/V]
Governing equations				
<u>Steady-state condition</u>				
Temperature rise		\Leftrightarrow	Voltage difference	
$\Delta T = q R_T$			$\Delta V = i R$	
<u>Transient condition</u>				
Heat diffusion		\Leftrightarrow	RC transmission line	
$\nabla^2 T = R_T C_T \frac{\partial T}{\partial t}$			$\nabla^2 V = R C \frac{\partial V}{\partial t}$	

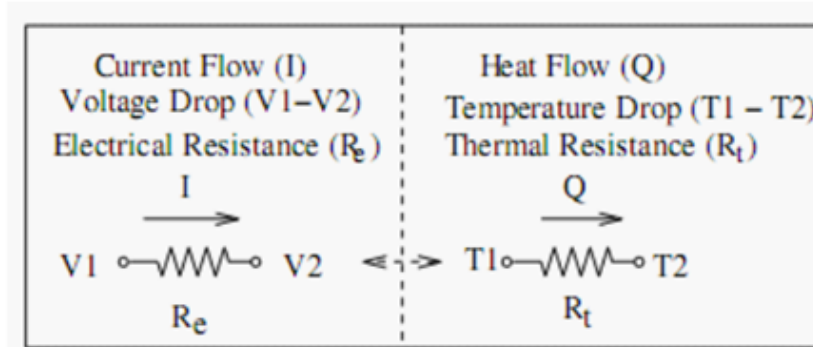


Figure 2.2: Analogy between electrical and thermal domain for LIM.

where u is the temperature, k is the thermal conductivity, c_p is the specific heat capacity, ρ is the mass density, and g is the volumetric heat flux. There is an analogy between an electrical problem and a heat conduction problem, as shown in Fig. 2.2 [5]. Therefore, solving Equation (2.14) with the LIM formulation is possible once the equivalent model in the electrical domain is established.

The equivalent thermal resistance and thermal capacitance for a rectan-

gular grid is given by [5]

$$\begin{aligned} R_x &= \frac{\Delta x}{k\Delta y\Delta z} \\ R_y &= \frac{\Delta y}{k\Delta x\Delta z} \\ R_z &= \frac{\Delta z}{k\Delta x\Delta y} \end{aligned} \tag{2.15}$$

and

$$C_{eq} = \rho c_p \Delta x \Delta y \Delta z. \tag{2.16}$$

Note that this is a model that has only resistances for branches and capacitance to ground for nodes.

2.4 Boundary Conditions

The boundary conditions used for the heat conduction problem focused on here is the convective boundary condition

$$k \frac{\partial T}{\partial n} = h(T - T_\infty), \tag{2.17}$$

where n is normal to the convective boundary, h is the heat transfer coefficient, and T_∞ is the ambient temperature of the boundary. The heat transfer coefficient is also defined as

$$h = \frac{\text{heat flux density}}{\Delta T} = \frac{q}{A_{eff}\Delta T}, \tag{2.18}$$

where A_{eff} is the area that belongs to the node. Equation (2.18) can also be written as

$$\Delta T = q \frac{1}{hA_{eff}} \leftrightarrow \Delta V = iR_h, \tag{2.19}$$

where

$$R_h = \frac{1}{hA_{eff}}. \tag{2.20}$$

This suggests that the boundary nodes with convective boundary condition can be modeled as having another branch with an equivalent convective resistance to a node that is held at the ambient temperature, as shown in Fig.

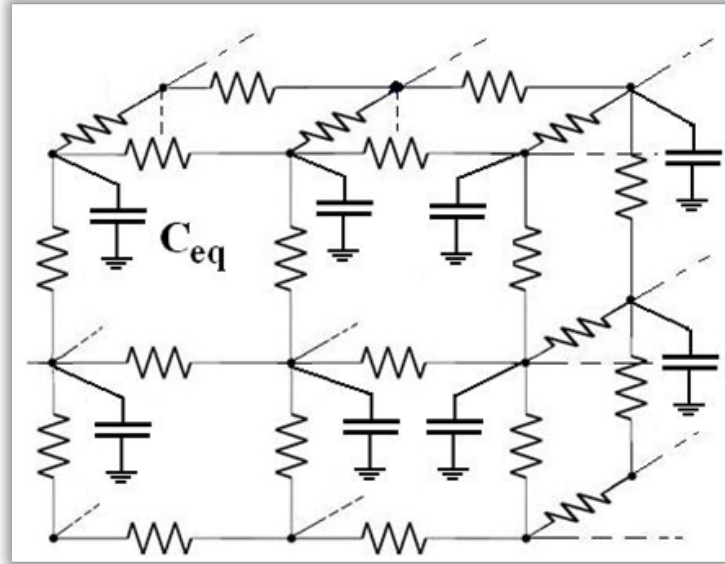


Figure 2.3: Equivalent RC model in the electrical domain for heat conduction problems.

2.4 [5].

One thing to note is that the equivalent convective resistances for nodes at the edges need some special scaling since they do not occupy full A_{eff} as other nodes on the boundary.

With Equations (2.15) and (2.16), a heat conduction problem can be modeled as an RC electrical problem as shown in Fig. 2.3. The locations of equivalent thermal resistances and current sources modeling the source terms are shown in Fig. 2.4. The convective boundary conditions are modeled as resistances R_h to nodes with voltage V where the value of the voltage is equal to the ambient temperature. As stated previously, since the RC equivalent model does not have any inductance, fictitious inductance must be added to every branch between two nodes.

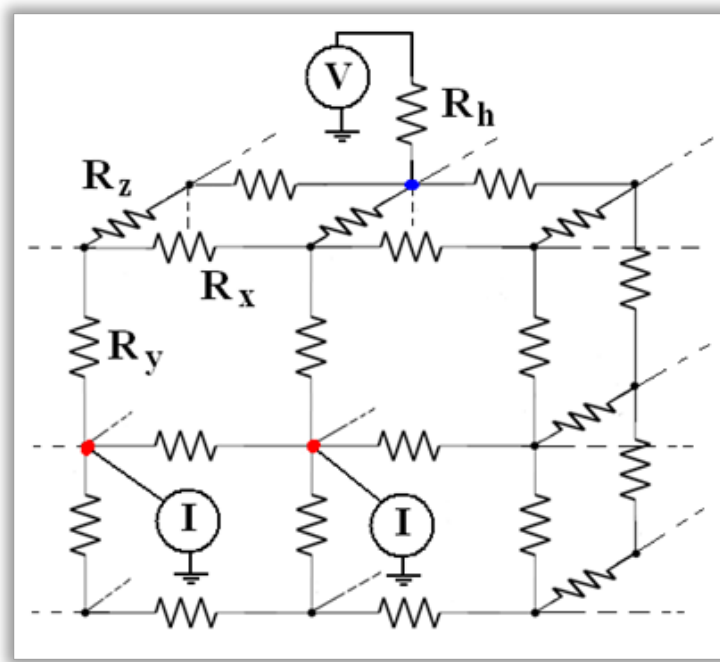


Figure 2.4: Locations of equivalent resistances in the electrical domain for heat conduction problems and equivalent convective boundary condition.

CHAPTER 3

DOUGLAS-GUNN METHOD

3.1 Formulation

This section focuses on another linear-time time-difference method, the Douglas-Gunn method (DGM). This method is a variation of the Crank-Nicolson method. The DGM uses an alternating direction implicit (ADI) scheme that is unconditionally stable and has second-order accuracy for a 3-D problem [4].

Equation (2.14), the heat conduction equation, for a 3-D problem can be discretized using the central difference formula [10] as

$$\frac{T^{n+1} - T^n}{\Delta t} = \frac{k}{c_p \rho} \left[\frac{\delta_x^2 T^{n+1} + \delta_x^2 T^n}{2(\Delta x)^2} + \frac{\delta_y^2 T^{n+1} + \delta_y^2 T^n}{2(\Delta y)^2} + \frac{\delta_z^2 T^{n+1} + \delta_z^2 T^n}{2(\Delta z)^2} \right] + \frac{g}{c_p \rho}, \quad (3.1)$$

where

$$\begin{aligned} \frac{\delta_x^2 T^n}{2(\Delta x)^2} &= \frac{T_{i+1,j,k}^n - 2T_{i,j,k}^n + T_{i-1,j,k}^n}{(\Delta x)^2} \\ &\approx \frac{T_{i+1,j,k}^n - 2T_{i,j,k}^n + T_{i-1,j,k}^n}{(\Delta x)^2} + O(\Delta x)^2 = \frac{\partial^2 T}{\partial x^2} \Big|_{i,j,k}^n. \end{aligned} \quad (3.2)$$

Similarly, the approximation is applied for y and z directions.

Using Equation (3.2), Equation (3.1) can then be rewritten as

$$\begin{aligned} T^{n+1} - T^n &= r_x \frac{\delta_x^2}{2} (T^{n+1} - T^n) + r_y \frac{\delta_y^2}{2} (T^{n+1} - T^n) \\ &\quad + r_z \frac{\delta_z^2}{2} (T^{n+1} - T^n) + \frac{\Delta t}{c_p \rho} g, \end{aligned} \quad (3.3)$$

where $r_x = k\Delta t/c_p\rho(\Delta x)^2$, $r_y = k\Delta t/c_p\rho(\Delta y)^2$, and $r_z = k\Delta t/c_p\rho(\Delta z)^2$.

So far, the formulation is identical to the Crank-Nicolson method. To solve Equation (3.3), a matrix of size $N \times N$ needs to be solved where N is the number of nodes. This matrix is big and computationally expensive to solve for a large system, just like what was discussed about the modified nodal analysis for SPICE. This is why a variation on the Crank-Nicolson method is needed, which evolves to the DGM.

Following [4], Equation (3.3) is broken up into three smaller time steps, given by

$$T^{n+1/3} - T^n = \frac{r_x \delta_x^2}{2} (T^{n+1/3} + T^n) + r_y \delta_y^2 T^n + r_z \delta_z^2 T^n + \frac{\Delta t}{c_p \rho} g, \quad (3.4)$$

$$T^{n+2/3} - T^n = \frac{r_x \delta_x^2}{2} (T^{n+1/3} + T^n) + \frac{r_y \delta_y^2}{2} (T^{n+2/3} + T^n) + r_z \delta_z^2 T^n + \frac{\Delta t}{c_p \rho} g, \quad (3.5)$$

and

$$T^{n+1} - T^n = \frac{r_x \delta_x^2}{2} (T^{n+1/3} + T^n) + \frac{r_y \delta_y^2}{2} (T^{n+2/3} + T^n) + \frac{r_z \delta_z^2}{2} (T^{n+1} + T^n) + \frac{\Delta t}{c_p \rho} g. \quad (3.6)$$

Intuitively, the first sub-time step tries to approximate the solution by assuming only the x-direction is implicit, the second sub-time step refines the approximation by taking the y-direction as implicit, and the last sub-time step further refines the approximation by taking the z-direction as implicit. By combining these three smaller time steps, the result of a full time step is obtained.

Equations (3.4)-(3.6), which are continuous equations, can be expanded

into the finite-difference equations

$$\begin{aligned}
& r_x T_{i-1,j,k}^{n+1/3} + 2(1+r_x)T_{i,j,k}^{n+1/3} - r_x T_{i+1,j,k}^{n+1/3} \\
&= r_x T_{i-1,j,k}^n + 2r_y T_{i,j-1,k}^n \\
&\quad + 2r_z T_{i,j,k-1}^n \\
&\quad + 2(1-r_x-2r_y-2r_z)T_{i,j,k}^n \\
&\quad + r_x T_{i+1,j,k}^n + 2r_y T_{i,j+1,k}^n \\
&\quad + 2r_z T_{i,j,k+1}^n + \frac{2\Delta t}{c_p \rho} g_{i,j,k},
\end{aligned} \tag{3.7}$$

$$\begin{aligned}
& r_y T_{i,j-1,k}^{n+2/3} + 2(1+r_y)T_{i,j,k}^{n+2/3} - r_y T_{i,j+1,k}^{n+2/3} \\
&= \left\{ r_x T_{i-1,j,k}^{n+1/3} - 2r_x T_{i,j,k}^{n+1/3} \right. \\
&\quad \left. + r_x T_{i+1,j,k}^{n+1/3} \right\} + \left\{ r_x T_{i-1,j,k}^n \right. \\
&\quad + r_y T_{i,j-1,k}^n + 2r_z T_{i,j,k-1}^n \\
&\quad + 2(1-r_x-r_y-2r_z)T_{i,j,k}^n \\
&\quad + r_x T_{i+1,j,k}^n + r_y T_{i,j+1,k}^n \\
&\quad \left. + 2r_z T_{i,j,k+1}^n \right\} + \frac{2\Delta t}{c_p \rho} g_{i,j,k},
\end{aligned} \tag{3.8}$$

and

$$\begin{aligned}
& r_z T_{i,j,k-1}^{n+1} + 2(1+r_z)T_{i,j,k}^{n+1} - r_z T_{i,j,k+1}^{n+1} \\
&= \left\{ r_x T_{i-1,j,k}^{n+1/3} - 2r_x T_{i,j,k}^{n+1/3} \right. \\
&\quad \left. + r_x T_{i+1,j,k}^{n+1/3} \right\} + \left\{ r_y T_{i,j-1,k}^{n+2/3} \right. \\
&\quad \left. - 2r_y T_{i,j,k}^{n+2/3} + r_y T_{i,j+1,k}^{n+2/3} \right\} \\
&\quad + \left\{ r_x T_{i-1,j,k}^n + r_y T_{i,j-1,k}^n \right. \\
&\quad + r_z T_{i,j,k-1}^n \\
&\quad + 2(1-r_x-r_y-r_z)T_{i,j,k}^n \\
&\quad + r_x T_{i+1,j,k}^n + r_y T_{i,j+1,k}^n \\
&\quad \left. + r_z T_{i,j,k+1}^n \right\} + \frac{2\Delta t}{c_p \rho} g_{i,j,k}.
\end{aligned} \tag{3.9}$$

These difference equations form a tridiagonal matrix A for the system of equations $Ax = b$. The tridiagonal matrix A of size $N_l \times N_l$ can be inverted

in linear time with respect to $O(N_l)$ using the Thomas algorithm [10] where l is either x, y, or z depending on the sub-time step. Since the DGM needs to repeat the first sub-time step for all y and z values, the second sub-time step for all x and z values, and the third sub-time step for all x and y values, the time complexity is $O(N_x) \times N_y N_z + O(N_y) \times N_x N_z + O(N_z) \times N_x N_y = O(N_x N_y N_z) = O(N)$, which means it has linear complexity with respect to the total number of nodes in the model.

The DGM can be proven to be unconditionally stable by applying a discrete Fourier transform of Equations (3.4)-(3.6) [10].

3.2 Boundary Conditions

The boundary conditions of interest are still the convective boundary condition, which is given by Equation (2.17). The boundary condition can again be discretized as

$$-k \frac{\partial T}{\partial x} \approx -k \frac{T_{1,j,k}^n - T_{-1,j,k}^n}{2\Delta x} = h_{x-}(T - T_\infty) \quad (3.10)$$

at surface $x = 0$. Then, the virtual point at $x = -1$ is

$$T_{-1,j,k}^n = T_{1,j,k}^n + \frac{2\Delta x h_{x-}}{k} (T_\infty - T_{0,j,k}^n). \quad (3.11)$$

Similar processes can be applied to other boundaries to obtain the corresponding virtual points.

To actually realize this, the simulator would have to keep the values for these virtual points that are outside of the solution domain. These values are used as given in Equation (3.10) and Equation (3.11) to calculate the nodes that are located at the boundaries of the domain.

CHAPTER 4

EXAMPLES AND RESULTS

4.1 Structure of Interest

The problem of interest is a circuit with the majority of its parts comprising a large MOSFET. Therefore, the thermal simulator techniques were tested on a MOSFET structure, shown in Fig. 4.1. For the model, the gate, source, and drain are made of polysilicon; the oxide layer is made of silicon dioxide (SiO_2); and the rest of the structure is made of silicon. The material properties used for those materials are shown in Table 4.1. The thermal capacitance in the table is equivalent to $c_p\rho$ in prior formulations. The heat source of the model is assumed to come from the MOSFET channel when the device is on. The boundaries are assumed to be insulated except the top and the bottom of the structure, because heat-exchange with the environment or cooling is most likely to occur at those spots.

In the rest of this chapter, the results of LIM and DGM are compared with results derived from Ansys Icepak, a commercial product for doing thermal analysis. Using comparisons with a commercially available product, some observations and comparisons can be made.

Table 4.2 shows the dimensions of different parts in the MOSFET structure, which are in the top portion of Fig. 4.1. In the table, substrate refers to the entire structure excluding the gate, source, drain, and oxide. The whole structure's thickness (the third dimension) is $2\ \mu\text{m}$.

One thing to note is that Ansys Icepak has its built-in non-uniform mesh builder, so the meshes for the LIM and the DGM are close to the mesh in Ansys Icepak, but not exactly the same.

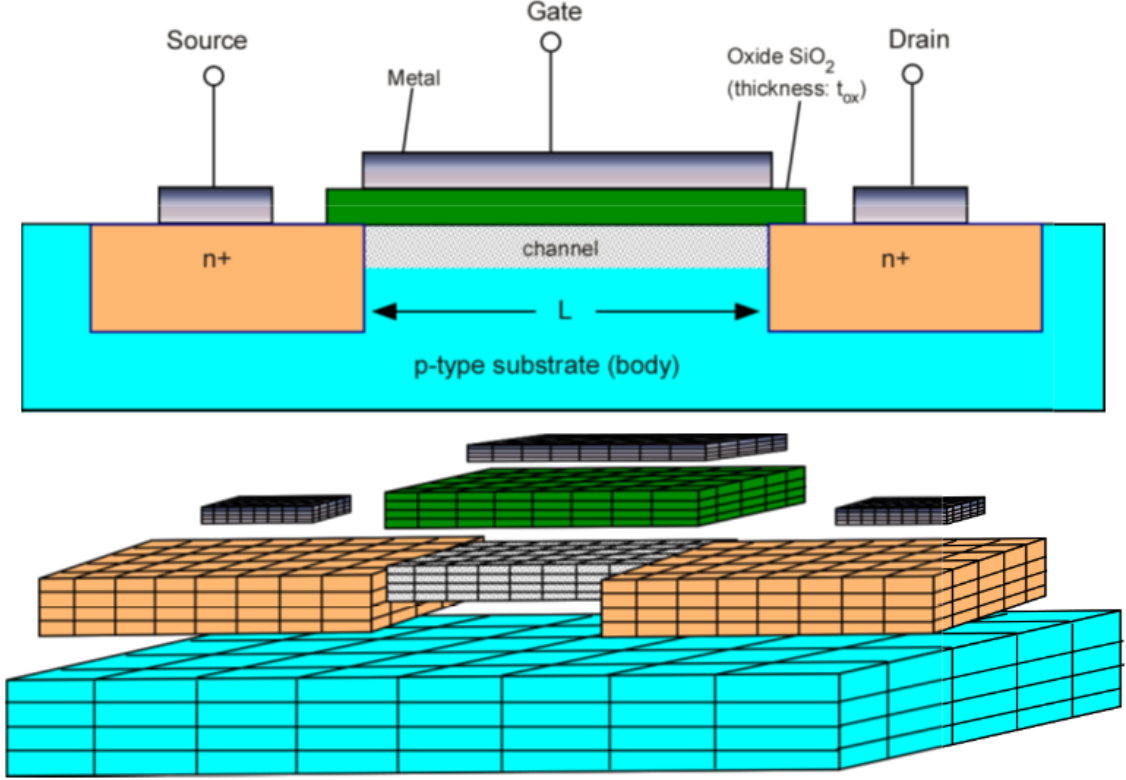


Figure 4.1: The MOSFET structure for simulation examples.

Table 4.1: Material Properties for MOSFET Model

	Thermal Conductivity (W/m K)	Thermal Capacitance (J/cm ³ K)
Silicon	180	1.63
SiO ₂	1.4	2.2
Polysilicon	125	1.755

Table 4.2: Dimensions of Simulated MOSFET Example

	Length (μm)	Width (μm)
Gate / Oxide	1.2	0.04
Source / Drain	0.2	0.04
Wells	0.5	3
Channel	1	2
Substrate	2.4	4.5

4.2 LIM Validation

For testing the thermal simulation using the LIM formulation, mesh sizes of the structure showed in Fig. 4.1 are $\Delta x = \Delta y = 1.33 \times 10^{-8}$ m and

$\Delta z = 6.66 \times 10^{-7}$ m. The effective heat transfer coefficients are $5 \text{ W/m}^2\text{K}$ at the top and $45 \text{ W/m}^2\text{K}$ at the bottom of the MOSFET structure in Fig. 4.1. The source is an ideal pulse that has a value of 0.3094 W (power of the entire channel) from $t = 0.02 \mu\text{s}$ to $t = 0.05 \mu\text{s}$, and the device is off otherwise. The chosen value of the fictitious inductances for the LIM is on the order of 10^{-8} H, and that results in a Δt on the order of 10^{-11} s from Equation (2.13). By having a similar mesh in Ansys Icepak and the same structural parameters, the simulation results are shown in Fig. 4.2.

Since LIM requires the insertion of fictitious inductance, but there is no equivalent thermal inductance, the rise and fall times of the LIM results are a bit different than those of Ansys Icepak results. More trials of the same simulation are studied using different fictitious inductance values in each branch, and the results are shown in Fig. 4.3 and Fig. 4.4. For the different fictitious inductance values, regular inductance is on the order of 10^{-6} H, lower inductance is on the order of 10^{-7} H, and lowest inductance is on the order of 10^{-8} H. The time step, Δt , of each run is chosen according to Equation (2.13). In general, the lower the fictitious inductance, the more accurate the results. However, by lowering the fictitious inductance, the time step has to be scaled down accordingly, making the simulation time longer.

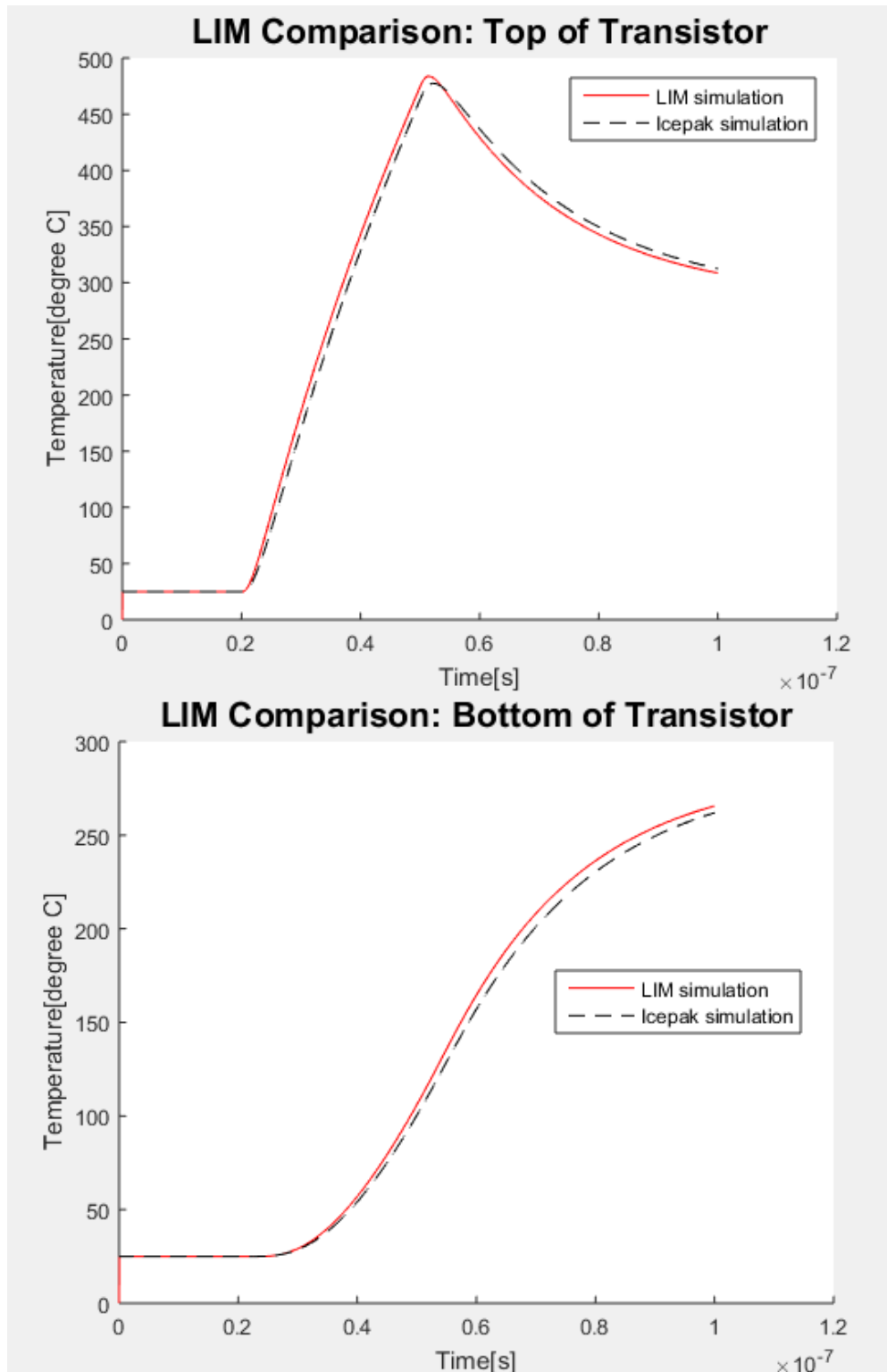


Figure 4.2: Temperature of the MOSFET structure using LIM and Ansys Icepak.

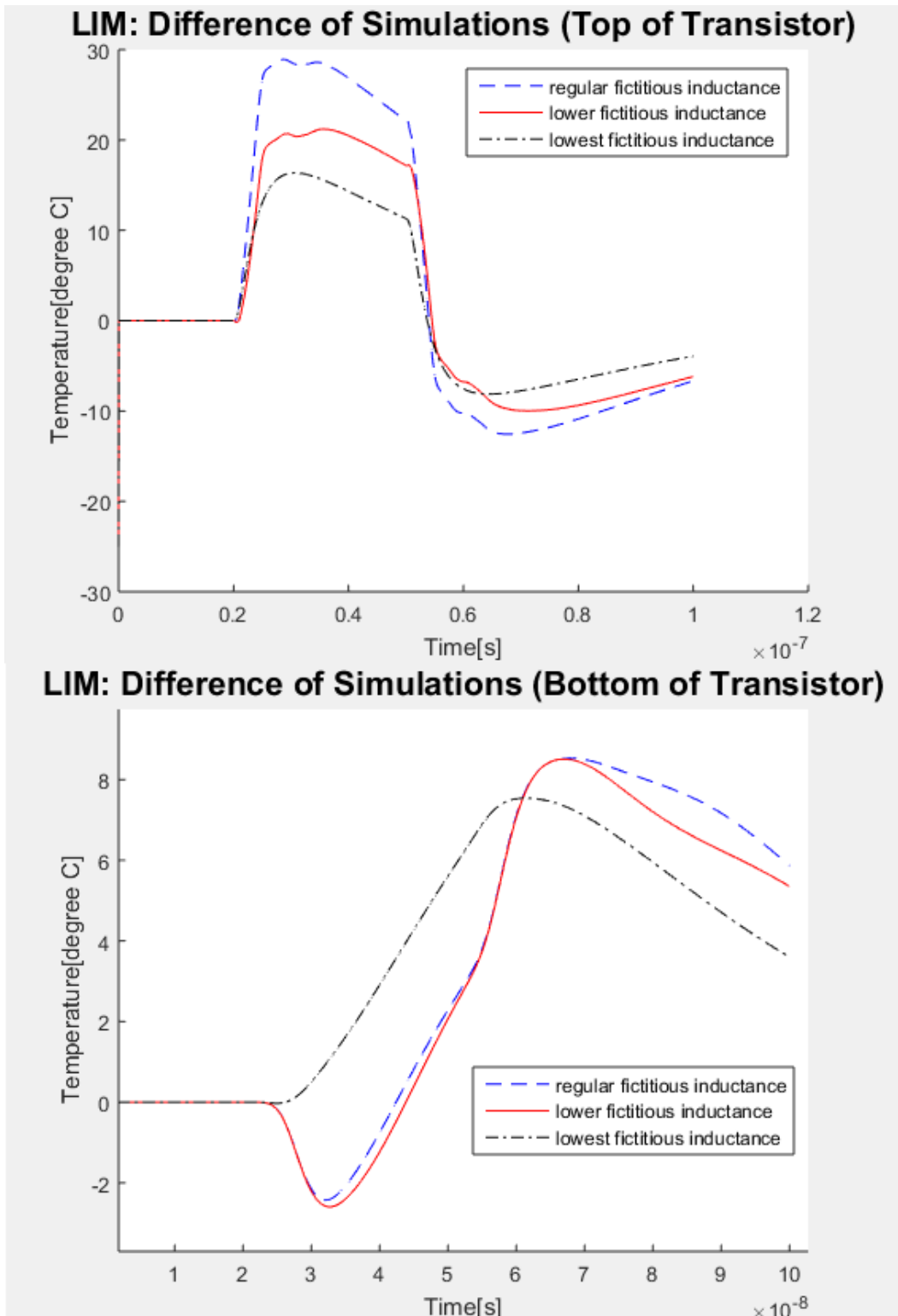


Figure 4.3: Differences of temperature from LIM and Ansys Icepak.

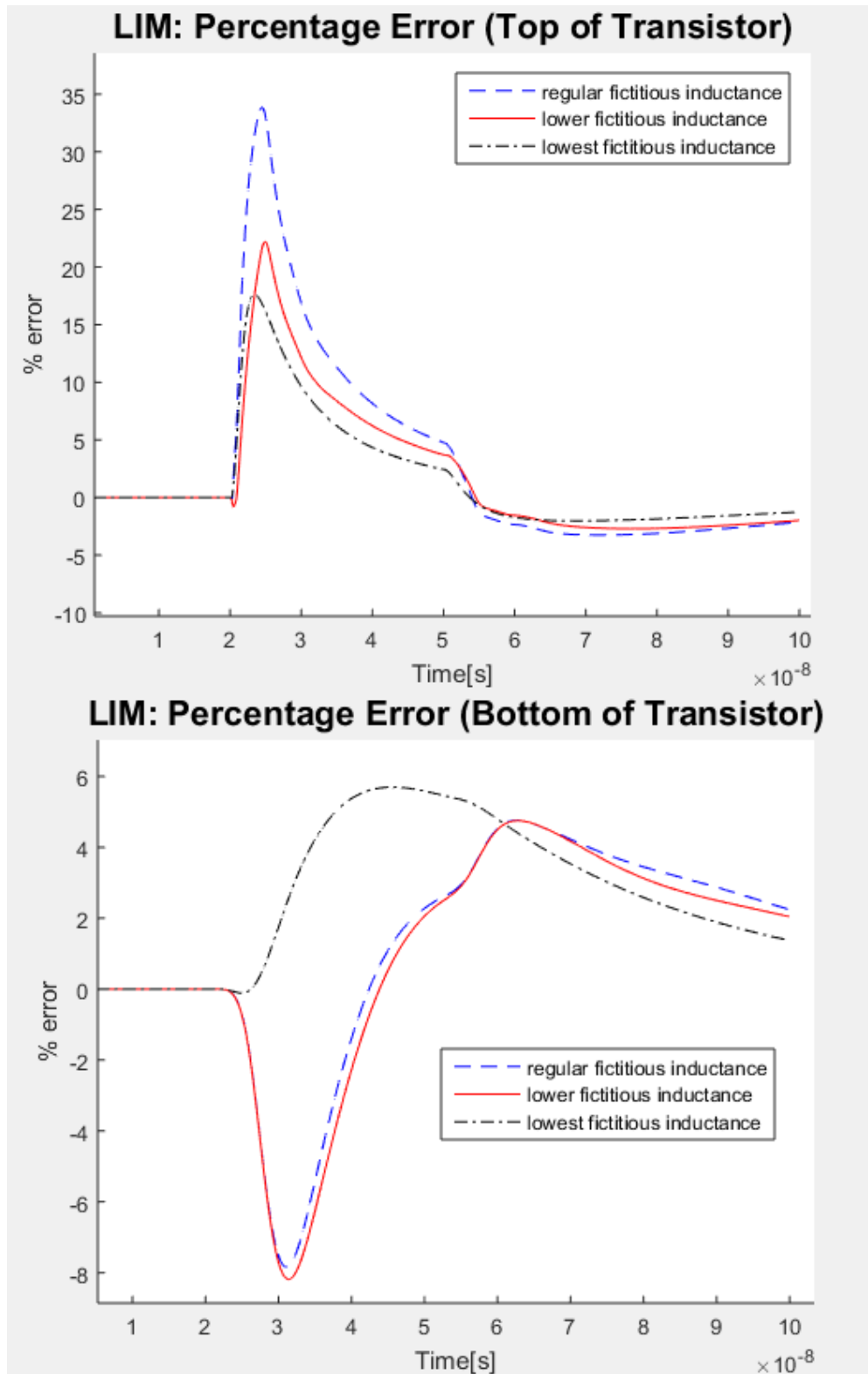


Figure 4.4: Percentage errors of temperature from LIM and Ansys Icepak.

Table 4.3: Error of DG for Different Δt

Δt (s)	Error at $t = 10^{-7}$ s
10^{-8}	$5.63 \times 10^{-2}\%$
10^{-9}	$6.95 \times 10^{-4}\%$
10^{-10}	$6.94 \times 10^{-6}\%$
10^{-11}	$9.11 \times 10^{-8}\%$

4.3 DGM Validation

For testing the DGM, minimum mesh sizes of the MOSFET structure in Fig. 4.1 are $\Delta x = 1.66 \times 10^{-8}$ m, $\Delta y = 1.33 \times 10^{-8}$ m and $\Delta z = 5 \times 10^{-7}$ m. The effective heat transfer coefficients are 5 W/m²K at the top and 45 W/m²K at the bottom. The source is an ideal pulse that has a value of 0.3094×10^{-3} W (power of the entire channel) from $t = 0$ s to 0.5 μ s, and the device is off otherwise. By having a similar mesh in Ansys Icepak and the structural parameters the same, the results are shown in Fig. 4.5.

Since the DGM does not require insertions of fictitious inductances, the timing of the transient results seems to match. The trial with $\Delta t = 10^{-9}$ s for DGM, implemented in Matlab, also runs twice as fast as Ansys Icepak, using the same Δt , with the same hardware. The differences and errors are shown in Fig. 4.6 and Fig. 4.7. For a self-consistency test, more trials of simulation for different Δt were conducted. By assuming the trial with $\Delta t = 10^{-12}$ s is the exact solution, errors of other trials with larger Δt are shown in Table 4.3. It can be seen that as Δt is reduced by 10 times, the error is reduced by 100 times. This demonstrates that DGM is second-order accurate with Δt , as discussed in Chapter 3.

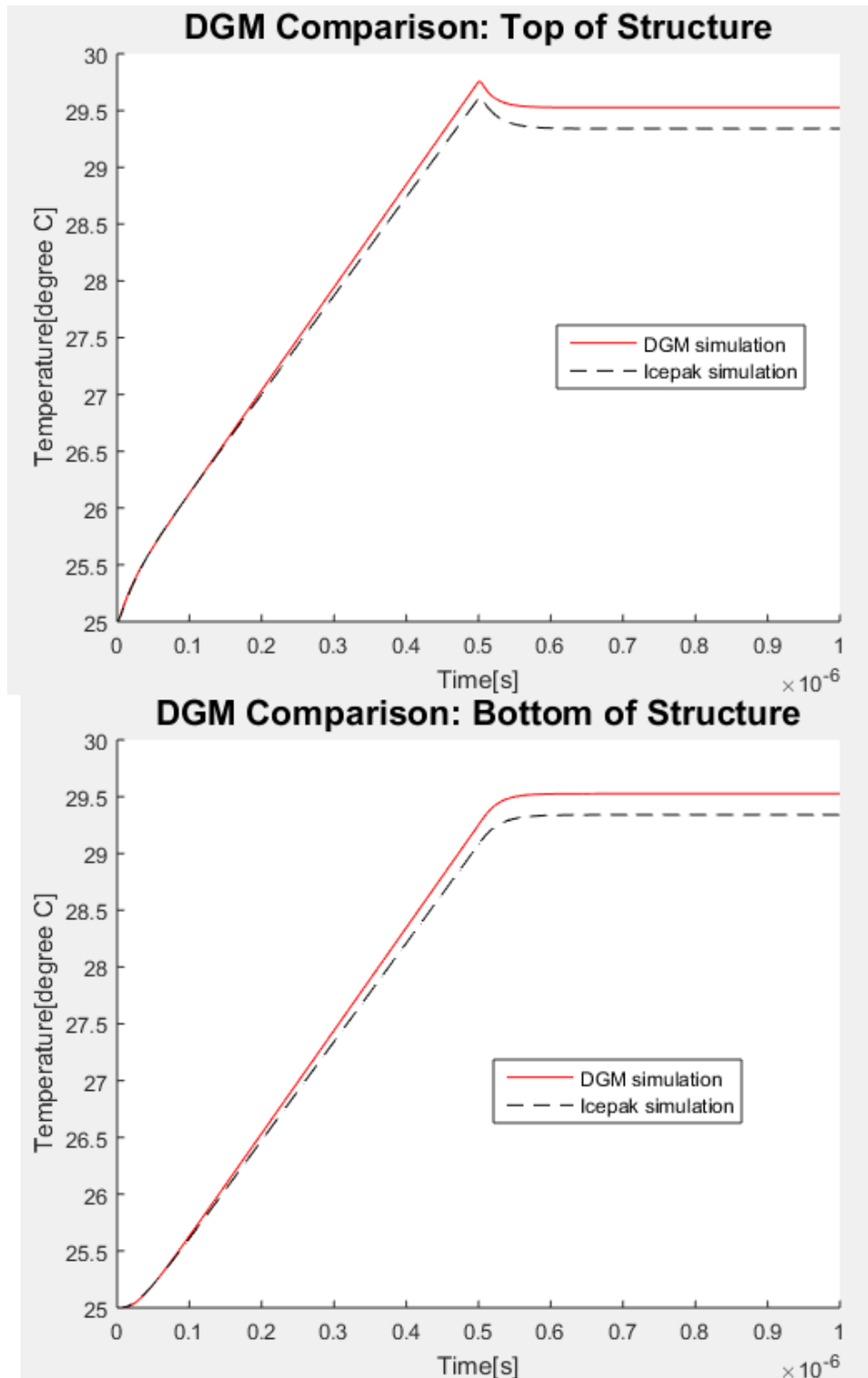


Figure 4.5: Temperature of the MOSFET structure using DGM and Ansys Icepak.

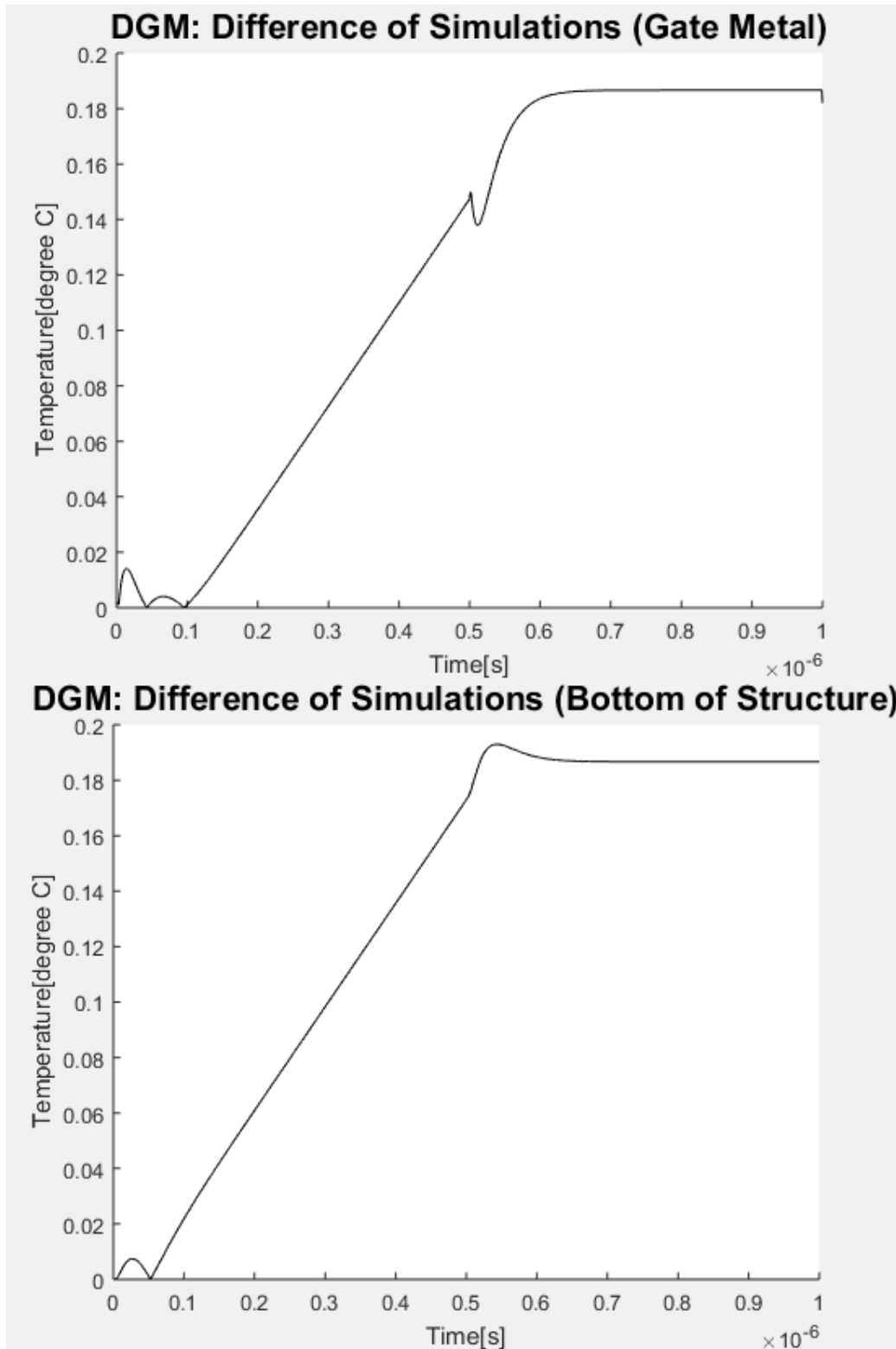


Figure 4.6: Differences of temperature from DGM and Ansys Icepak.

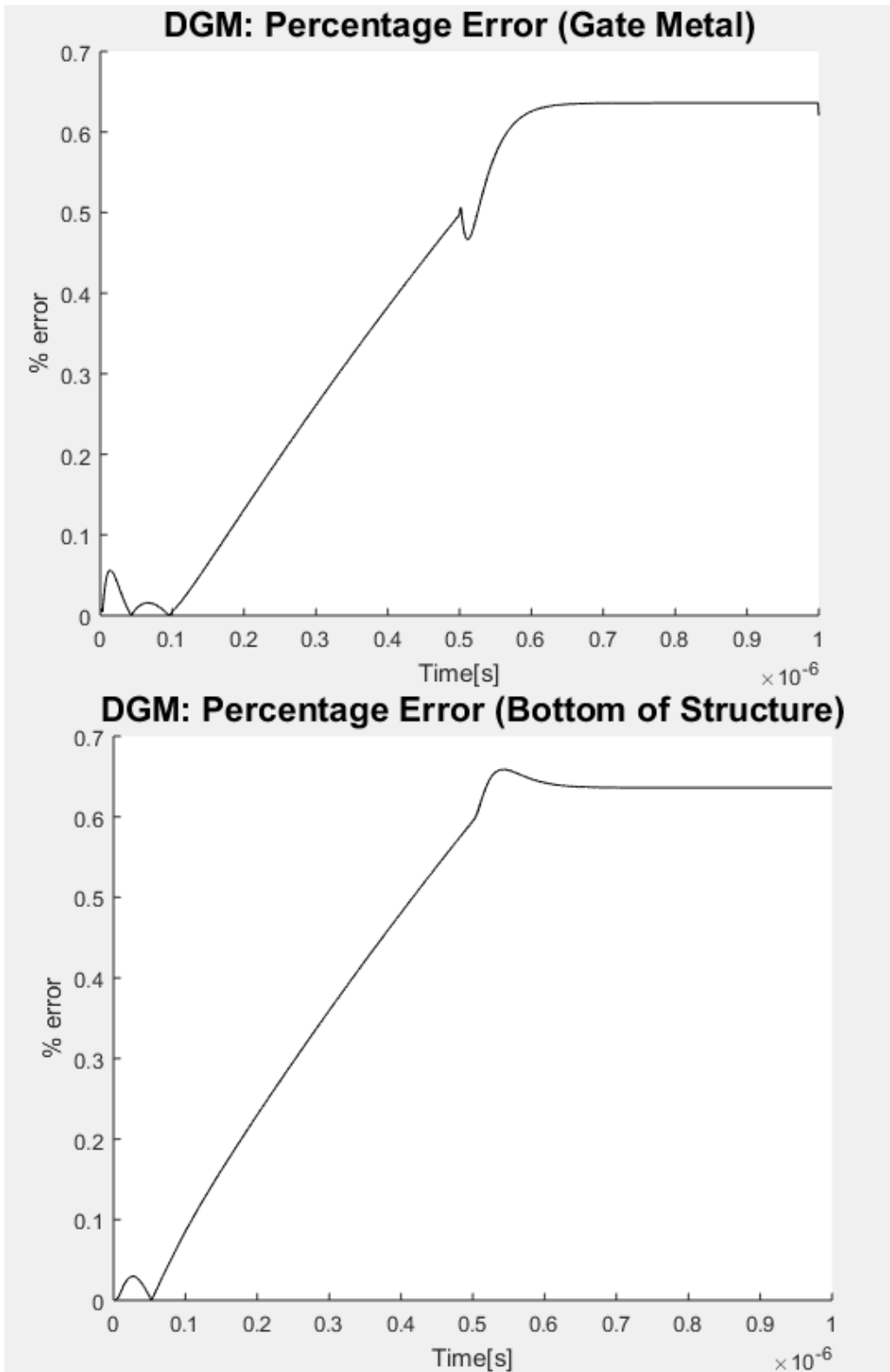


Figure 4.7: Percentage errors of temperature from DGM and Ansys Icepak.

4.4 Effect of Fictitious Inductance

In Chapter 2, the LIM requires the presence of inductances in each branch of the model. Those inductances introduce a delay in the information or voltage propagation between nodes. This enables the LIM to solve for voltage or temperature of each node separately from most of other nodes. While it is desirable that the method obtains linear complexity with fictitious inductance, the inserted elements introduce other effects to the simulation.

First, as discussed in Section 2.2, the time step has to be picked accordingly. The smaller the fictitious inductance, the smaller the time step and the longer the simulation time.

Second, the fictitious inductances allow oscillation of the voltage/temperature when combined with other capacitors in the model. Figure 4.8 and Fig. 4.9 show the difference between two simulations with different fictitious inductance values. Figure 4.8, with lower fictitious inductance, has less magnitude of oscillation than Fig. 4.9, with higher fictitious inductance. It can be seen that if the steady-state solution is the interested quantity, having a high fictitious inductance value will produce the solution faster, since the oscillation eventually diminishes after reaching a steady state. However, if the transient solution is the interested quantity, those oscillations make the simulated solution deviate from the actual solution. More importantly, there is no thermal inductance in the thermal domain. By having oscillations in temperature, the second law of thermal dynamics is violated.

For the DGM, since there are no inductive elements in the model, it does not have oscillations in the simulated results. Figure 4.10 and Fig. 4.11 are results of DGM simulation of the same structure but with different time steps. Figure 4.11 has sharper transitions since there are fewer points, but does not have oscillations. With a smaller time step, the simulation result is more accurate. However, it may be rewarding to use a larger time step so the simulation takes less time, as long as the error is within the designated margin.

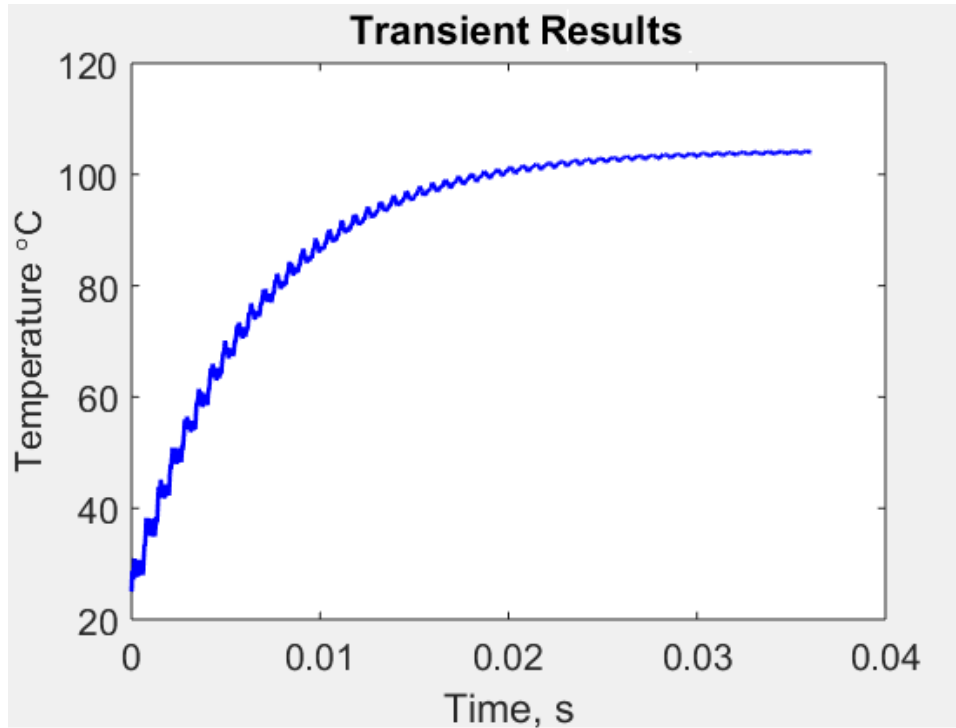


Figure 4.8: A LIM simulation with low fictitious inductance.

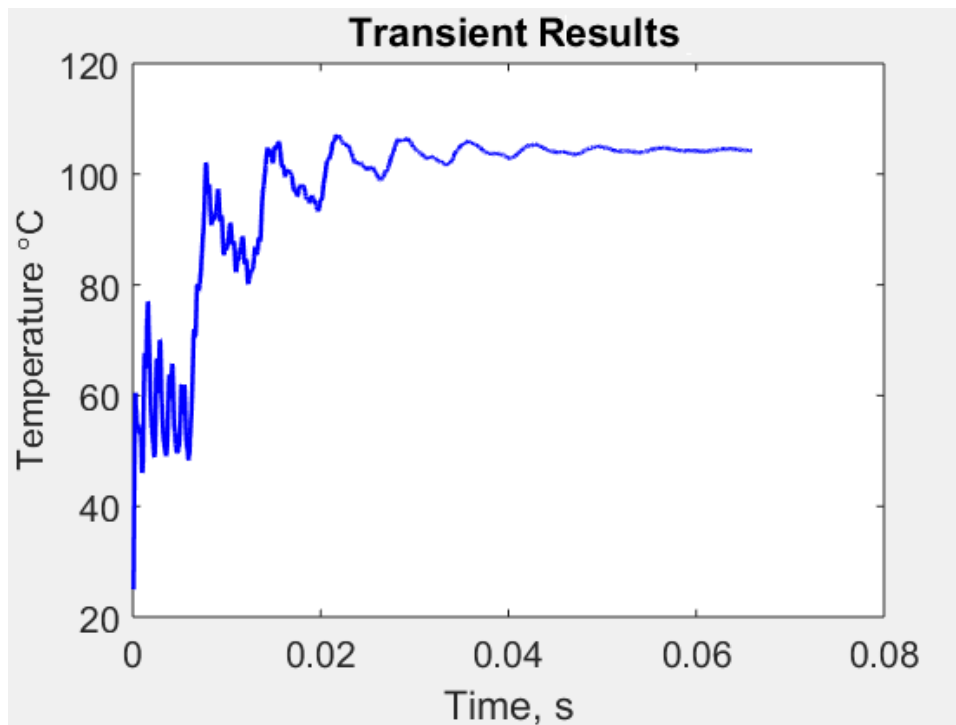


Figure 4.9: A LIM simulation with high fictitious inductance.

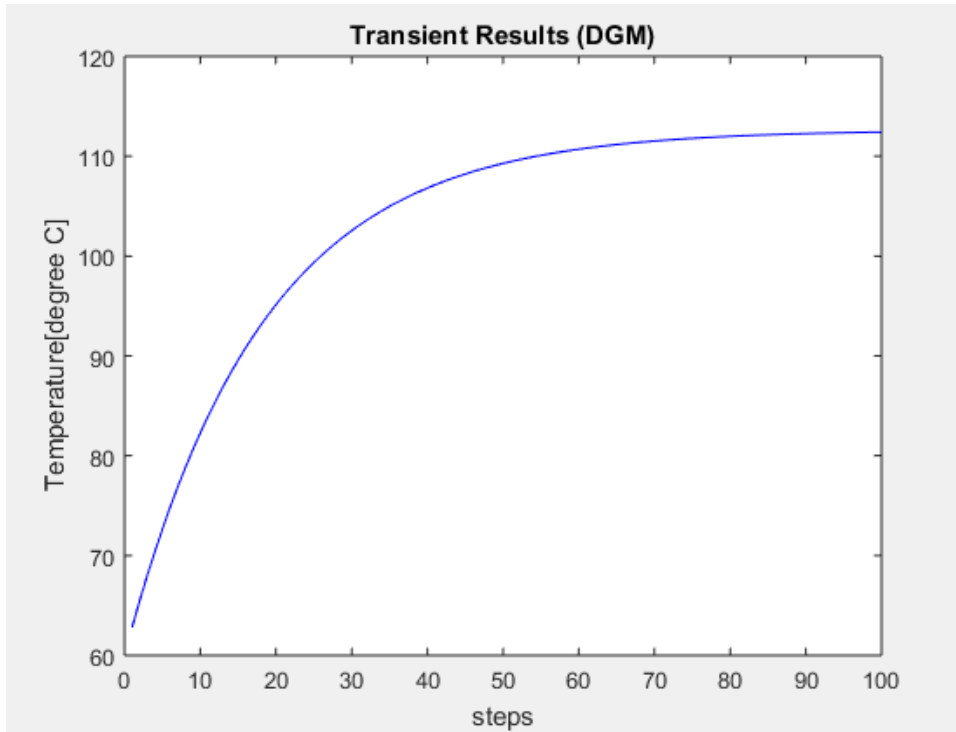


Figure 4.10: A DGM simulation with small time step.

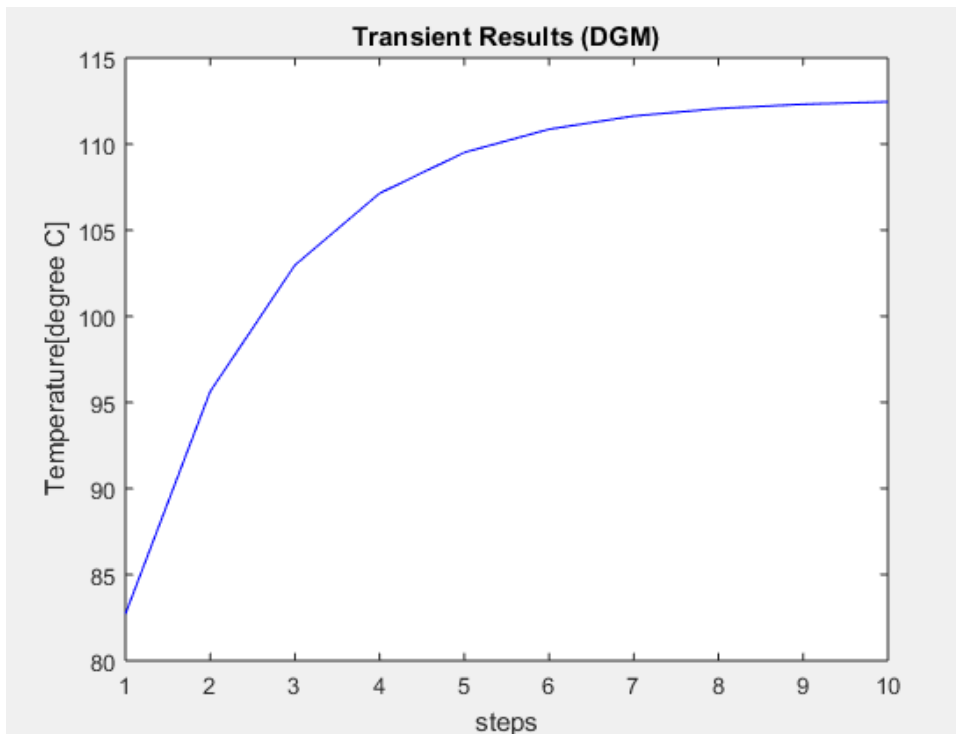


Figure 4.11: A DGM simulation with large time step.

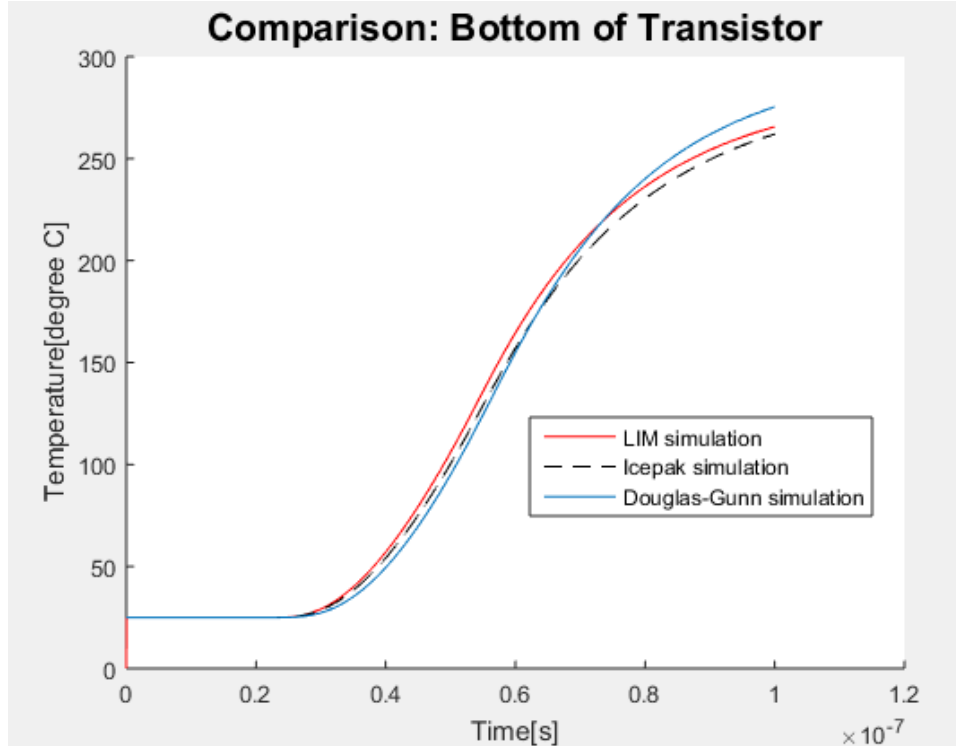


Figure 4.12: Temperature at the bottom of transistor simulated by LIM, Ansys Icepak, and DGM.

4.5 LIM and DGM Comparison

Since the LIM and the DGM are both capable of solving thermal problems, the example in Section 4.2 can also be simulated with the DGM. Figure 4.12 and Fig. 4.13 show the results for LIM, Ansys Icepak, and DGM. While the results of both LIM and DGM have good agreements with that of Ansys Icepak, the effect of fictitious inductances change the timing accuracy of LIM slightly. Figure 4.14 is the zoomed-in version of Fig. 4.13 at around $t = 0.5 \times 10^{-7}$ s. One can see that the timing of the DGM simulation matches that of the Ansys Icepak simulation, and that of the LIM simulation is slightly delayed. This effect is not introduced by the mesh difference, as that seems to only change the temperature at a given time but not the sign of the slope, but is introduced by the fictitious inductance.

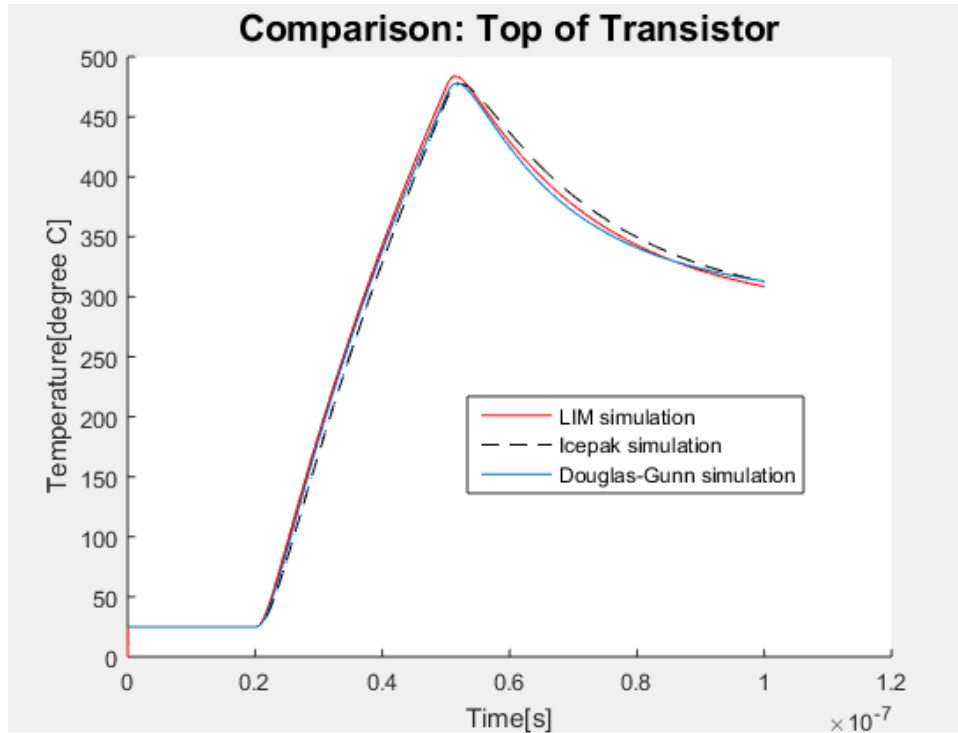


Figure 4.13: Temperature at the top of transistor simulated by LIM, Ansys Icepak, and DGM.

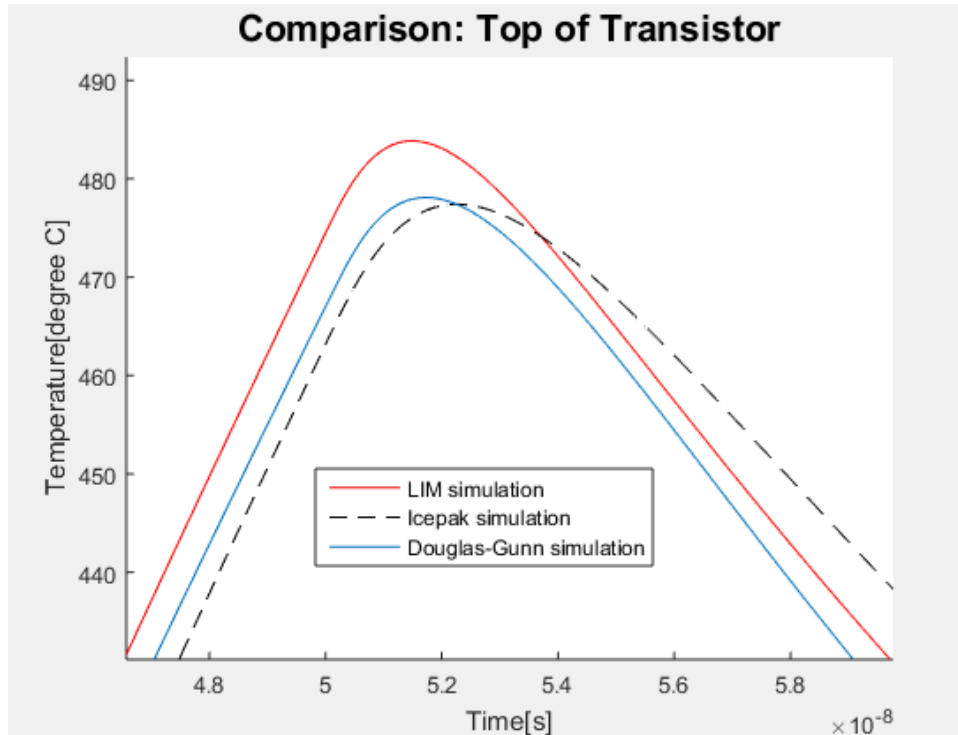


Figure 4.14: Temperature at the top of transistor simulated by LIM, Ansys Icepak, and DGM, zoomed in.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

In this thesis, two linear-time 3-D thermal methods for transient electro-thermal simulations, the LIM and the DGM, are discussed. Both are finite-difference methods that discretize the solution domain (the structure of interest) to find the solutions of the governing differential equations. In this case, the governing equation is the heat conduction equation. The linear complexity of these two methods is extremely valuable as is the speed of the simulation, comparatively, faster than that for the methods with super-linear complexity.

The problem of interest is the thermal simulation of a large MOSFET. Using a model from the physical structure of a MOSFET, some simulation results are produced using the LIM, the DGM, and Ansys Icepak. From the results in Chapter 4, it seems to be the case that the simulation by DGM, without fictitious elements, has a better timing agreement with that produced by Ansys Icepak. However, if one chooses to integrate a thermal method with the LIM as the electrical method, the LIM is also capable of doing thermal simulations with good accuracy without the need of implementing a second solver.

5.2 Future Work

Both LIM and DGM are shown to be able to perform accurate transient thermal simulations. However, there are some improvements and steps that could be implemented in future studies.

The effect of the fictitious inductances are experimented in Chapter 4.

Higher fictitious inductances allow faster simulation times but introduce more oscillations and delay in the simulation results. Currently there is no way to determine the value of the fictitious inductances that balances the accuracy versus simulation time. A good algorithm for choosing such a value not only enhances the LIM for thermal simulation, but is also appreciated when using the LIM for electrical simulations.

The model that is used in this work is a simple model of a MOSFET. A better MOSFET model can more accurately capture the heat production and propagation in an actual device, thus improving the quality of the simulation results.

Next, the meshes used in Ansys Icepak is different than the meshes used in LIM and DGM simulations in this work. Since Ansys Icepak is a commercially available tool, it is very likely that the meshing algorithm is more powerful and more accurate. Therefore, by having better meshes with LIM and DGM, they may produce even more accurate results than those presented in Chapter 4.

Finally, choosing a electrical simulator and a thermal simulator is half of the entire process of making an electro-thermal simulator. The connection of those two simulators in different domains must be made to perform accurate electro-thermal analysis of engineering designs.

Taking these steps further will produce an electro-thermal simulator that is capable of producing accurate simulation results. This is important because the thermal aspect of designs must be examined to ensure both good performance and reliability of devices.

REFERENCES

- [1] L. W. Nagel, *SPICE2, a computer program to simulate semiconductor circuits*, 1975. [Online]. Available: <https://www2.eecs.berkeley.edu/Pubs/TechRpts/1975/ERL-520.pdf>
- [2] J. E. Schutt-Ainé, “Latency insertion method (LIM) for the fast transient simulation of large networks,” *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 48, no. 1, pp. 81–89, Jan 2001.
- [3] D. Klokotov and J. Schutt-Ainé, “Latency insertion method for the analysis of steady state on-chip power distribution networks and transient simulation of lossy interconnects,” in *2008 Asia-Pacific Microwave Conference*, Dec 2008, pp. 1–4.
- [4] J. Douglas and J. E. Gunn, “A general formulation of alternating direction methods,” *Numerische Mathematik*, vol. 6, no. 1, pp. 428–453, 1964.
- [5] D. Klokotov and J. E. Schutt-Ainé, “Latency insertion method (LIM) for electro-thermal analysis of 3-d integrated systems at pre-layout design stages,” *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 3, no. 7, pp. 1138–1147, July 2013.
- [6] J.-M. Jin, *The Finite Difference Method*. Wiley-IEEE Press, 2010, pp. 295–341. [Online]. Available: <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5628381>
- [7] K. Yee, “Numerical solution of initial boundary value problems involving Maxwell’s equations in isotropic media,” *IEEE Transactions on Antennas and Propagation*, vol. 14, no. 3, pp. 302–307, May 1966.
- [8] J. Schutt-Ainé, “Stability analysis for semi-implicit LIM algorithm,” in *2009 Asia Pacific Microwave Conference*, Dec 2009, pp. 1270–1272.
- [9] S. N. Lalgudi, M. Swaminathan, and Y. Kretchmer, “On-chip power-grid simulation using latency insertion method,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 55, no. 3, pp. 914–931, April 2008.

- [10] T.-Y. Wang and C. C.-P. Chen, “3-d thermal-adi: A linear-time chip level transient thermal simulator,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 21, no. 12, pp. 1434–1445, Dec 2002.