VarSim: A Fast and Accurate Variability and Leakage Aware Thermal Simulator

Hameedah Sultan
School of Information Technology
Indian Institute of Technology, New Delhi, India
Email: hameedah@cse.iitd.ac.in

Smruti R. Sarangi
Computer Science and Engineering
Indian Institute of Technology, New Delhi, India
Email: srsarangi@cse.iitd.ac.in

***Abstract*—Some of the fastest thermal estimation techniques at the architectural level are based on Green's functions (impulse response of a unit power source). The resultant temperature profile can be easily obtained by computing a convolution of the Green's function and the power profile. Sadly, existing approaches do not take process and temperature variation into account, which are integral aspects of today's technologies. This problem is still *open*. In this paper, we provide a closed-form solution for the Green's function after taking process, temperature, and thermal conductivity variation into account. Moreover, during the process of computing the thermal map, we reduce the amount of redundant work by identifying similar regions in the chip using an unsupervised learning-based approach. We were able to obtain a 700,000X speedup over state-of-the-art proposals with a mean absolute error limited to 0.7°C (1.5%).**

I. Introduction

Temperature simulation has become one of the most important steps in the overall semiconductor design flow. As the device dimensions keep decreasing, the impact of on-chip variability becomes significant [1]. Parameter variations inclusive of process and temperature variations lead to a deviation in the electrical and thermal parameters of transistors. It has been shown that process variation can result in up to a 20X variation in leakage power, and 30% frequency variation [1]. With the dimensions scaling from 350 nm to 45 nm, the yield was shown to go down from 90% to 30%, partly because of the effects of process variation [2]. It has been estimated that process variation can undo the complete performance gain obtained by a new technology generation [2].

A thermal simulation that considers the effect of *variation* (process, temperature, thermal conductivity) is important in all design stages. There are many architectural techniques to compensate for the adverse effects of process variation such as functional unit level body biasing, and retiming. To assess the impact of such schemes, researchers often generate a set of random variation maps and subsequently perform architectural simulations. Similarly, while making placement decisions, it is also necessary to generate a large number of variation maps and evaluate the efficacy of different placement strategies. The common feature of all of these approaches is that we need to perform a large number of thermal simulations on dies that have a significant amount of process, conductivity and temperature variation – hence, there is a need for a *fast* thermal simulator in this space.

Unfortunately, existing thermal simulators have failed to factor in the effects of variation. Researchers have shown that a chip designed by ignoring process variation may fail after fabrication [3]. In addition, most simulators also ignore the temperature dependence of conductivity, leading to significant errors in thermal estimation. It was shown by Yang et al. [4] that ignoring the temperature dependence of conductivity can lead to a 5°C error in temperature estimation. Furthermore, state-of-the-art simulators are very slow since they rely on the costly finite element and finite difference based methods, limiting the scope of design space exploration. To make problems worse, most existing thermal simulators consider the effects of leakage power by iterating through the leakage-temperature feedback loop. This further reduces their speed significantly.

This work has been sponsored in part by the Semiconductor Research Corporation (SRC).

Consequently, fast thermal estimation accounting for variability has hitherto remained an open problem. To overcome these problems, we propose *VarSim* in this paper. Our main contributions can be summarized as follows:

1) *VarSim* considers the impact of variability as well as the dependence of conductivity on temperature. To the best of our knowledge, there is no existing technique that achieves this.
2) We identify a new way of characterizing the variation-aware ambient leakage power. We use k-means clustering to cluster the variation maps into regions with similar properties. We then construct a dictionary of the clusters that can be used to characterize any variability map. This step is offline, and is done for a chip only once.
3) We derive a novel modified leakage aware Green's function to consider the impact of temperature-dependent conductivity and leakage power. This Green's function can directly be convolved with the power profile at runtime to obtain the temperature profile accounting for the two feedback effects. Where aggregate statistics are needed, we run our algorithm multiple times with a range of leakage power profiles to get the range of expected temperatures.

Since our approach is semi-analytical and is based on Green's functions, we obtain a several orders of magnitude speedup over state-of-the-art approaches, with the maximum error within 4%.

Section II provides the relevant background and Section III describes our modeling methodology. We evaluate our method in Section IV and conclude in Section V.

II. Theoretical Preliminaries

A. Fourier Equation

The fundamental equation governing heat transfer is the *heat equation*, given by:

$$\nabla.\left(\kappa\nabla T\right)+\dot{q}=\rho C_v\frac{\partial T}{\partial t}, \quad (1)$$

where ρ is the density of the material, C_v is the volumetric specific heat, T is the temperature, $\dot{q}$ is the thermal energy generation rate inside the volume and κ is the thermal conductivity. A solution of the heat equation yields the temperature profile of the chip. However, since the heat equation is too complex to solve analytically, most

thermal simulators rely on the finite difference methods (FDMs) or finite element methods (FEMs) for getting fast numerical solutions.

B. Green's Functions

A fast approach to solving the heat equation is to obtain the impulse response of the chip (also called the Green's function) by applying a unit power source to the center of the chip, and convolving this impulse response with the power profile to obtain the full-chip temperature profile. This approach is analytical, and much faster than FDM and FEM based approaches since the entire heat transfer path is not modeled; rather only the power dissipating layers and the boundary conditions are considered [5], [6]. Using the Green's function, the temperature profile can be calculated as :

$$T = f_{sp} \star P \tag{2}$$

where f_{sp} is the Green's function, $\star$ is the convolution operator, P is the power profile and T is the temperature profile.

C. Process and Temperature Variation (or simply Variation)

The limitations of the manufacturing process of ICs impact the physical properties of the chip. As a result, the properties of the chip deviate from their nominal values. The parameters most susceptible are threshold voltage, oxide thickness, gate width, and channel length. These variations are classified into three categories: wafer-to-wafer, die-to-die and within-die variations. The first two effects are constant for a given die. These effects used to be more important for older technology generations, and can be handled by simple methods such as *clock binning*. For newer technology generations, within-die variation dominates and requires more complicated management strategies [2]. Within-die variation is further classified as:

1) **Systematic variations:** These are introduced because of lithographic aberrations. As a result, proximate regions on the die have similar values of parameters. It is modeled by a multivariate Gaussian distribution [7] with a covariance matrix having a spherical correlation.

2) **Random variations:** These are caused by random dopant fluctuations (RDF) and are modeled as a zero-mean Gaussian random variable. These variations do not exhibit any spatial correlation.

There are two variables in the heat equation that are affected by parameter variation: leakage power and thermal conductivity.

1) Leakage power: The variability in leakage power arises because of both systematic and random variations. However, the effects of random variation tend to get averaged out at the architectural level when considering temperature. The subthreshold leakage current, I_{leak} is given by Equation 3.

$$I_{leak} \propto v_T^2 * e^{\frac{V_{GS}-V_{th}-V_{off}}{\eta * v_T}}(1 - e^{\frac{-V_{DS}}{v_T}}) \tag{3}$$

where, v_T is the thermal voltage (kT/q), V_{th} is the threshold voltage, V_{off} is the offset voltage in the sub-threshold region and η is a constant. Because of variability, the oxide thickness and gate length change, which result in a change in the threshold voltage. The temperature dependence of I_{leak} can be modeled by a linear equation [5], [6], [8]. Equation 3 can then be approximated as:

$$I_{leak} \propto (1 + \beta\Delta T)e^{\beta_L \Delta L + \beta_{t_{ox}} \Delta t_{ox}} \tag{4}$$

where β represents the change in leakage power with temperature, $\beta_{t_{ox}}$ is a constant representing the variability in the oxide thickness t_{ox} and β_L represents the variability in the gate length, L. Thus we arrive at:

$$P_{leak} = (1 + \beta\Delta T)P_{leak_0}^{var}, \tag{5}$$

where $P_{leak_0}^{var}$ is the leakage power at ambient temperature after considering the impact of variability. For improved accuracy, we use a piece-wise linear leakage model, which provides an accuracy of over 99% [9].

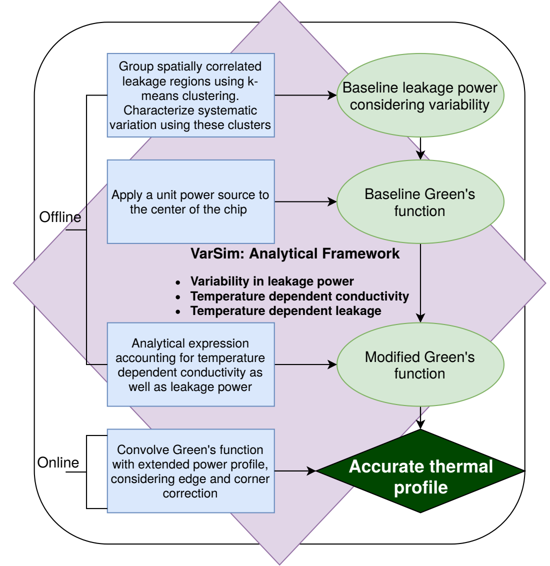


Fig. 1: Overview of our approach

2) Conductivity: We study the effect of variability in the conductivity of silicon as well. This is because the Fourier equation dictates that temperature is dependent on power as well as the conductivity of the material. Because of random dopant fluctuations (RDF), the doping profile varies, leading to significant variations in conductivity. We consider variability in conductivity to be a Gaussian random variable, K. To calculate its variance, we look at the variation in doping profiles because of RDF from existing works [10], and translate these dopant densities to the conductivity of silicon by obtaining the corresponding values from the literature [11].

D. Transforms used in this work

Thermal problems are often easier to solve in the transform domain. We use two types of transforms in this work in addition to the widely known *Fourier transform*.

1) *Discrete Cosine Transform (DCT):* The DCT transforms a signal from the spatial domain to the frequency domain. It uses the cosine functions as its basis. It is purely real and concentrates energy well into the low-frequency coefficients.

2) *Hankel Transform:* It is equivalent to the 2-D Fourier transform of a radially symmetric function. The Hankel transform is defined as:

$$\mathcal{H}(f(r)) = H(s) = \int_0^\infty f(r)\mathcal{J}_0(sr)rdr, \tag{6}$$

where $\mathcal{J}_0$ is the Bessel function of the first kind of order 0.

III. Thermal Estimation Considering Variability

A. Overview

We begin by providing an overview of our approach (Figure 1). Steps ❶ and ❷ are the offline components, and step ❸ is the online component. Table I shows a glossary of all the terms used in our derivations.

❶ We analytically derive the equation for a novel leakage aware Green's function accounting for variability and the dependence of leakage power and conductivity on temperature.

❷ The expression derived for the modified Green's function in step ❶ requires the variation aware leakage power map at ambient temperature, $P_{leak_0}^{var}$. We do not want to recompute the Green's

functions every time we consider a new $P^{var}_{leak_0}$ – this is too time taking. We thus use a learning-based approach – we superimpose a grid of 64×64 blocks, and cluster the blocks based on their systematic variation component. We construct a dictionary of such clusters. The advantage of this approach is that if we get a new leakage power map, we need not recompute the Green's functions for each region, we can simply look up the dictionary, and find the corresponding Green's functions.

❸ We calculate the full-chip variability-aware thermal profile in the presence of leakage as well as dynamic power by convolving the modified Green's functions with the dynamic power map.

TABLE I: Glossary

Symbol	Meaning
$P^{var}_{leak_0}$	Leakage power at ambient temperature considering variability
β	Temperature dependence of leakage power
α	Temperature dependence of conductivity
κ	Conductivity of silicon
T	Temperature
$\mathcal{T}$	Temperature rise above ambient temperature
f_{sp_0}	Green's function without considering leakage and temperature-dependent conductivity = $f_{silic_0} + \phi$
$\mathbf{E}(X)$	Expected value of X
$\mathcal{F}$	Fourier transform operator
$\mathcal{H}$	Hankel transform operator
x, y	Spatial coordinates
u, v	Fourier frequency domain variables
h	Hankel variable
t	Time
C	Thermal capacitance
f^k_{leaksp}	Leakage aware Green's functions considering temperature dependence of conductivity

B. Variation of the Thermal Conductivity with Temperature

The conductivity of silicon has a strong temperature dependence, given by [12]:

$$\kappa = k_0 \left(\frac{T}{300}\right)^{-\eta}, \tag{7}$$

where k_0 is the conductivity of silicon at 300K, T is the temperature in Kelvin, and η is a material-dependent constant. In the operating range of ICs ($40-100°$C), we can linearize Equation 7:

$$\kappa(T) = {k_0}'(1 - c\Delta T), \tag{8}$$

where k'_0 is the nominal conductivity of silicon at the ambient temperature, and c is a constant.

Next, we study the variation of the Green's function, f_{sp}, with the change in conductivity and obtain the following empirical relation (using HotSpot [13]):

$$f_{sp}(\kappa) = f_{sp_0}(1 - c'(\kappa - k'_0)), \tag{9}$$

where, f_{sp_0} is the Green's function without considering the variation of κ with temperature (conductivity variation because of dopant density variations is captured in f_{sp_0}), and c' is another constant. Combining Equations 8 and 9, we obtain a temperature-dependent Green's function to capture the dependence of conductivity on temperature.

$$f_{sp}(T) = f_{sp_0}(1 + \alpha\Delta T), \tag{10}$$

where α is a constant representing the variation of the Green's function with temperature.

C. Derivation of the Modified Green's Functions

We follow the basic approach used by Sarangi et al. [5]. However, their work does not include the temperature dependence of conductivity or the effects of variability.

The power profile (P) is the sum of the dynamic power (P_{dyn}) and the leakage power (P_{leak}). Using Equation 5, we get:

$$P = P_{dyn} + P^{var}_{leak_0}(1 + \beta\Delta T). \tag{11}$$

Let us assume that initially no dynamic power is applied to the chip. The entire temperature rise is because of leakage. From Equation 2, we have:

$$T_0 = f_{sp_0} \star P^{var}_{leak_0} \tag{12}$$

Now, let us apply a unit impulse (Dirac delta) source at the center of the chip as the dynamic power. Using Equations 10 and 11 in Equation 2, we get:

$$T_f = f_{sp_0}(1 + \alpha\Delta T) \star \left(\delta(x, y) + P^{var}_{leak_0}(1 + \beta\Delta T)\right), \tag{13}$$

where x, y are the spatial coordinates. Subtracting T_0 from T_f, we arrive at the rise in temperature because of the impulse source, in the presence of variability in leakage power and temperature-dependent conductivity:

$$\begin{aligned}\mathcal{T} &= T_f - T_0 \\ &= f_{sp_0}(1 + \alpha\mathcal{T}) + f_{sp_0}(1 + \alpha\mathcal{T}) \star P^{var}_{leak_0}(1 + \beta\mathcal{T}) \\ &- f_{sp_0} \star P^{var}_{leak_0}\end{aligned} \tag{14}$$

Our goal here is to solve for the temperature rise, $\mathcal{T}$. To get rid of the convolution operation, we use the property that the Fourier transform of the convolution of two functions is equal to their product in the frequency domain. We compute the Fourier transform of both the sides of Equation 14 to arrive at Equation 15.

$$\begin{aligned}\mathcal{F}(\mathcal{T}) = &(\mathcal{F}(f_{sp_0}) + \alpha\mathcal{F}(f_{sp_0}\mathcal{T})) + (\mathcal{F}(f_{sp_0}) + \alpha\mathcal{F}(f_{sp_0}\mathcal{T})) \times \\ &(\mathcal{F}(P^{var}_{leak_0}) + \beta\mathcal{F}(P^{var}_{leak_0}\mathcal{T})) - \mathcal{F}(f_{sp_0})\mathcal{F}(P^{var}_{leak_0}) \\ = &\underbrace{\mathcal{F}(f_{sp_0})}_{I} + \underbrace{\alpha\mathcal{F}(f_{sp_0}\mathcal{T})}_{II} + \underbrace{\beta\mathcal{F}(f_{sp_0})\mathcal{F}(P^{var}_{leak_0}\mathcal{T})}_{III} \\ &+ \underbrace{\alpha\mathcal{F}(f_{sp_0}\mathcal{T})\mathcal{F}(P^{var}_{leak_0})}_{IV} + \underbrace{\alpha\beta\mathcal{F}(P^{var}_{leak_0}\mathcal{T})\mathcal{F}(f_{sp_0}\mathcal{T})}_{V}\end{aligned} \tag{15}$$

In Equation 15, term I is the baseline Green's function, term II is the additional increase in temperature because of temperature-dependent conductivity, term III represents the temperature rise because of the temperature-dependent leakage power, term IV represents the compounded effect of the baseline leakage power and the temperature-dependent Green's function because of the conductivity, and term V is the additional temperature rise arising from the compounded effect of the temperature-dependent Green's function and temperature-dependent leakage power. The last term will be small, since the two effects individually lead to a small increase in the temperature profile, and their compounded effect will not be very large. Hence we neglect this term. We also make another simplifying assumption to enable the calculation of $\mathcal{F}(P^{var}_{leak_0}\mathcal{T})$. Although $P^{var}_{leak_0}$ is a distribution, we assume it to be a constant equal to its expected value. Thus $\mathcal{F}(P^{var}_{leak_0}\mathcal{T}) = \mathbf{E}(P^{var}_{leak_0})\mathcal{F}(\mathcal{T})$, where $\mathbf{E}$ represents the expected value. All the assumptions are justified empirically in Section IV.

$$\begin{aligned}\mathcal{F}(\mathcal{T}) = &\mathcal{F}(f_{sp_0}) + \alpha\mathcal{F}(f_{sp_0}\mathcal{T})(1 + \mathbf{E}(P^{var}_{leak_0})) + \\ &\beta\mathbf{E}(P^{var}_{leak_0})\mathcal{F}(f_{sp_0})\mathcal{F}(\mathcal{T}) \\ = &\mathcal{F}(f_{sp_0}) + \alpha'\mathcal{F}(f_{sp_0}\mathcal{T}) + \beta'\mathcal{F}(f_{sp_0})\mathcal{F}(\mathcal{T}),\end{aligned} \tag{16}$$

where $\alpha' = \alpha(1 + \mathbf{E}(P^{var}_{leak_0}))$ and $\beta' = \beta\mathbf{E}(P^{var}_{leak_0})$.

The bottleneck in the above equation is the computation of $\mathcal{F}(f_{sp_0}\mathcal{T})$. Let $G(u,v) = \mathcal{F}(f_{sp_0}\mathcal{T})$. Let $f_{sp_0} = f_{sp_0}^{var} + f_{sp_0}^{det}$, where $f_{sp_0}^{det}$ is the deterministic component of the Green's function and $f_{sp_0}^{var}$ is the component of the Green's function due to variability.

$$\begin{aligned} G(u,v) &= \mathcal{F}((f_{sp_0}^{var} + f_{sp_0}^{det})\mathcal{T}) \\ &= \mathbf{E}(f_{sp_0}^{var})\mathcal{F}(\mathcal{T}) + \underbrace{\mathcal{F}(f_{sp_0}^{det}\mathcal{T})}_{\chi(u,v)} \end{aligned} \tag{17}$$

To compute the second term, we take advantage of the Hankel transform. The 2D Fourier transform in the above equation can be replaced by the zero-order Hankel transform, since it is a radially symmetric function, and then we apply integration by parts.

$$\begin{aligned} \chi(u,v) = \mathcal{H}(f_{sp_0}^{det}\mathcal{T}) &= \int_0^\infty (f_{sp_0}^{det}\mathcal{T})J_0(hr)rdr \\ &= f_{sp_0}^{det}\int_0^\infty \mathcal{T}J_0(hr)rdr - \int_0^\infty f_{sp_0}^{det'}dr\int_0^\infty \mathcal{T}J_0(hr)rdr \\ &= f_{sp_0}^{det}\mathcal{H}(\mathcal{T}) - \int_0^\infty f_{sp_0}^{det'}dr \times \mathcal{H}(\mathcal{T}) \end{aligned} \tag{18}$$

We convert this equation back to Cartesian coordinates, into the Fourier domain and substitute this back in Equation 17:

$$\begin{aligned} G(u,v) &= \underbrace{\mathbf{E}(f_{sp_0}^{var})\mathcal{F}(\mathcal{T}) + f_{sp_0}^{det}\mathcal{F}(\mathcal{T})}_{\approx\mathcal{F}(\mathcal{T})f_{sp_0}} - \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{sp_0}^{det'}dxdy\mathcal{F}(\mathcal{T}) \\ &= \mathcal{F}(\mathcal{T})\left(f_{sp_0} - \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f'_{sp_0}dxdy\right) \end{aligned} \tag{19}$$

Using this back in Equation 16 and solving for $\mathcal{F}(\mathcal{T})$, we get:

$$\mathcal{F}(\mathcal{T}) = \frac{\mathcal{F}(f_{sp_0})}{1-\alpha'\left(f_{sp_0} - \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f'_{sp_0}dxdy\right) - \beta'\mathcal{F}(f_{sp_0})} \tag{20}$$

The Green's function f_{sp} can be broken down into a rapidly decaying function f_{silic} and a constant ϕ (established in prior work [5], [6], and also in our experiments).

$$\boxed{f_{leaksp}^{k} = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}(f_{silic_0}) + \phi\delta(u,v)}{\begin{aligned}1-\beta'\mathcal{F}(f_{silic_0}) - \beta'\phi\delta(u,v) - \alpha' f_{silic_0} \\ - \alpha'\phi + \alpha'\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f'_{sp_0}dxdy\end{aligned}}\right\}} \tag{21}$$

D. Leakage Power at Ambient Temperature: K-Means Clustering

Because of the systematic component of process variation, different regions of the chip have different values of the baseline leakage power, $P_{leak_0}^{var}$. This is an input to the final equation for the modified Green's function (Equation 21) (subsumed in parameters α' and β'). Whenever the variation map changes, we have to recompute the Green's function for each region. This is inefficient. We can leverage the fact that we have a large spatial correlation in the systematic variation component. This can be used to store a dictionary of pre-computed Green's functions. Whenever we have a new variation map, we can break it into a set of regions, look up the dictionary and find the corresponding Green's functions.

The first problem is to uniquely characterize the variation within a rectangular region on the die. Given that the systematic component primarily determines the mean of the distribution of the baseline leakage power, we use the mean as one of the parameters. However, it is possible that different distributions have the same mean, and this can lead to aliasing. To capture this effect, we propose to compute

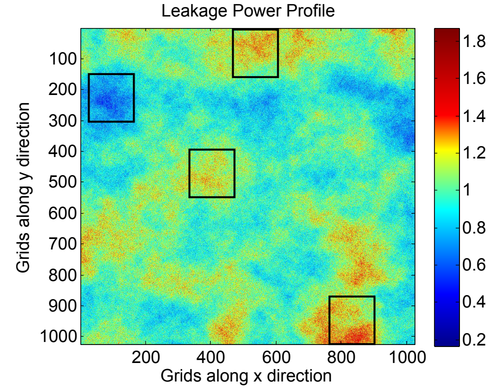


Fig. 2: Leakage power in the presence of variability. Note the spatial correlation in the leakage power values.

the Discrete Cosine Transform (DCT) of the baseline leakage power values in a block and filter out the high-frequency components. The mean of the remaining coefficients is used as the second feature.

Subsequently, we consider a large number of randomly generated variation maps, and cluster the regions using standard K-Means clustering – each block is represented by its two features (mean, mean of filtered DCT). For each block, we store the corresponding Green's function (computed using Equation 21). During the process of lookup, we compute the (mean,DCT mean) tuple for a given region and choose the entry in the dictionary that minimizes the Euclidean distance.

E. Thermal Estimation at the Edges and Corners

To compute the full-chip thermal profile, we convolve the modified Green's functions with the power profile. In this process, the edges and corners are special cases. Since the boundaries are adiabatic, we apply an analogy with the method of images from electromagnetics (also used in [14]). In this approach, we double the size of the area that we are considering with the die in the middle and zero-pad the regions outside the die. Then for each power source, we add a *mirror image source* that is on the *other* side of the die boundary (same distance). We obtain the temperature map by convolving the power profile with the respective Green's functions.

IV. Evaluation

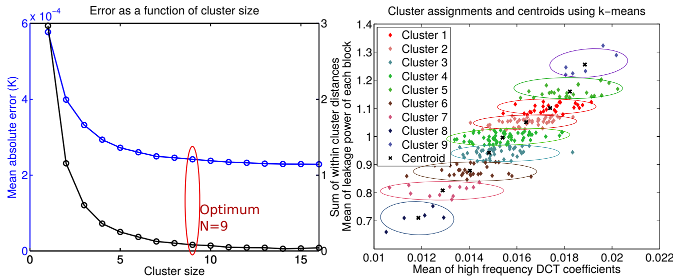
A. Setup

We augment the popular thermal modeling tool Hotspot [13] to carry out thermal simulations with variable leakage power as well as conductivity. Our HotSpot thermal simulation routines have been written in R. We run all our Hotspot simulations on an Intel i7-7700 4-core CPU running Ubuntu 16.04 with 16 GB of RAM. We implemented and tested our proposed algorithm in Matlab on a Windows 8 desktop with an Intel i7-2600S processor and 8 GB of RAM. We discretized the chip into a 64×64 grid.

1) Error Metric: We use the mean absolute error and the percentage error relative to the maximum temperature rise as the error metric. Other thermal modeling tools often report errors relative to the maximum temperature in the die, which under-represents the error.

B. Calibration of the Setup

To calibrate our HotSpot setup, we use the commercial CFD software Ansys Icepak. It is an industry-standard tool widely used for high-accuracy thermal simulations. We model an identical layout in HotSpot and Icepak, and compare the temperature values obtained from the two tools. We find that the normalized temperature values obtained using both of these tools conform well (within 1.5%).



(a) K-means clustering for different cluster sizes (b) K-means clustering to create dictionary of localized spatial regions

Fig. 3: K-means clustering

Simulator	Considering $\kappa(T)$	Without considering $\kappa(T)$
Hotspot[1]	18 $minutes$	4 s
3D-ICE	–	1.36 s
Icepak	15 $minutes$	15 $minutes$
Jaffari et al.[15]	–	158 s
VarSim	1.3 ms	1.3 ms

1. To model temperature-dependent conductivity, detailed thermal modeling is done in HotSpot, since the properties of each block are different.
2. HotSpot, 3D-ICE and Icepak do not consider variability

TABLE II: Speed of existing simulators

C. Results

1) Dictionary using K-Means clustering: To obtain the leakage power values in the presence of systematic and random variations, we use the widely used variation modeling tool, *Varius* [7]. We obtain the leakage power at a fine granularity (1024×1024) from Varius. We then discretize the chip into 64×64 blocks, each containing 16×16 elements. We construct a dictionary using the mean and low frequency DCT coefficients of each block as features. To select the optimum number of clusters, we vary the number of clusters until no significant gain in accuracy is obtained. The mean absolute error and the sum of within cluster distances is plotted in Figure 3(a). The cluster assignments for the optimal size are shown in Figure 3(b).

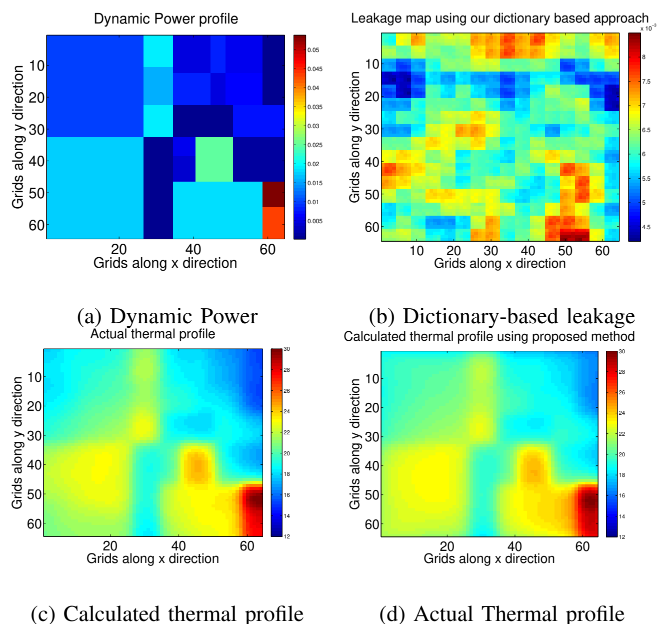
2) Modified Green's functions: We begin by applying a unit impulse power source to the center of the chip and obtain the baseline Green's function using HotSpot considering the variability in conductivity due to random dopant fluctuations. We then use Equation 21 to obtain the modified Green's function accounting for the effects of temperature-dependent conductivity and leakage power (captures the effect of the systematic component). Our approach takes 0.55 ms to compute the modified Green's function with an error limited to 3%. Next, we construct a dictionary of the Green's functions using the approach described in Section III-D. For a new variation map, we first obtain the corresponding Green's function by looking up the dictionary for each block, and finding the cluster with the maximum similarity.

3) Full-chip thermal simulations: At runtime, the sum of the dynamic and baseline leakage power profiles is mirrored and convolved with the calculated Green's functions. This step takes an additional 0.74 ms. Thus the total time taken by our algorithm is 1.3 ms, with the maximum error limited to 4%.

To validate our proposal, we adopt the following approach: the leakage power obtained from Varius is added to the dynamic power profile, and HotSpot is invoked iteratively. After each iteration, we update the leakage power and conductivity values on the basis of the current temperature. We keep iterating until the temperature values converge. HotSpot supports modeling of variable conductivity only when detailed 3D modeling is enabled, since different conductivity values for different blocks increase the complexity greatly. As a result, HotSpot requires 18 min to compute the final temperature. If we do not model variable conductivity, the simulation completes in 4s.

Test Case 1 [Real floorplan]: We validate our approach using the floorplan of the Alpha21264 processor. The power values are taken from the *ev6* test case of HotSpot. The leakage and dynamic power profile, and the corresponding temperature profiles are shown in Figure 4. We can see in Figure 4c, that the calculated thermal profile matches the actual thermal profile very well (within 2%).

Test Case 2 [Stress testing]: In this case, multiple dynamic power sources are applied to different locations on the chip. The total dynamic power is 8 W. Although the total power applied is lower than test case 1, the power density of the sources is much higher. Consequently, the maximum temperature in the chip is higher than test case 1. In this case too, the temperature obtained using our algorithm matches the actual value very well, with a mean absolute error limited to 0.7°C (1.2%).



(a) Dynamic Power (b) Dictionary-based leakage

(c) Calculated thermal profile (d) Actual Thermal profile

Fig. 4: Evaluation for Alpha21264

D. Analysis of Results

To further understand the importance of the individual effects we have modeled, we carry out multiple simulations with a few effects not modeled. Table III summarizes the error obtained in various scenarios. As seen in Table III, not accounting for any kind of variability leads to a temperature estimation error of up to 22%. If we ignore the temperature dependence of conductivity but consider variability and temperature dependence of leakage, the error varies from 2 to 7.5%. Ignoring variations in the conductivity profile leads to $< 1\%$ error in thermal estimation. Thus in thermal modeling, the effects of random variations in the conductivity profile can be safely ignored, but considering variability in leakage power, and the temperature dependence of leakage power and conductivity is *absolutely essential*.

Comparison with state-of-the-art approaches: Since there is no state-of-the-art work that considers variability as well as temperature-dependent leakage, we compare our results against the modified version of HotSpot. Our algorithm provides a 700,000X speedup over

TABLE III: Errors in various scenarios

Effects considered	Test Case 1 (Alpha21264)			Test Case 2		
	Max. Temp. (K)	Max. Deviation (K)	Percent Deviation	Max. Temp. (K)	Max. Deviation (K)	Percent Deviation
No effects	341.36	6.77	22.6	372.90	9.01	14.1
Rand.-cond., Cond.(T)	341.86	6.27	20.9	377.59	4.32	6.8
Leakage-var	344.04	4.09	13.6	375.29	6.62	11.6
Leakage(T)	344.30	3.83	12.8	374.56	7.35	12.2
Rand.-cond., Cond.(T), Leakage(T)	344.91	3.22	10.7	378.39	3.52	5.8
Leakage-var, Leakage(T)	347.43	0.70	2.33	377.16	4.75	7.5
Cond.(T), Leakage-var, Leakage(T)	348.12	0.01	0.03	381.36	0.55	0.86
Rand.-cond., Cond.(T), Leakage-var, Leakage(T)	348.13	–	–	381.91	–	–
VarSim	**347.58**	**0.55**	**1.8**	**379.31**	**2.60**	**4.1**

Leakage(T) = temperature-dep. leakage, Leakage-var = variability in leakage, Rand.-cond. = random conductivity, cond.(T) = temperature-dep. conductivity

HotSpot, while maintaining the error within 4%. Table II summarizes the simulation speed of various tools.

E. Related Work

While variability itself is a well-studied area, there are only a few works that consider its impact on temperature. A major limitation of the works that consider the effects of leakage power variability is that they assume the operating temperature to be equal to the ambient temperature [15], i.e. ignore the temperature dependence of leakage power. In our work, we observe that considering variability but ignoring temperature dependence of leakage can result in a 4 to 6°C error, which is completely unacceptable.

Jaffari and Anis [15] statistically calculate the expected value of temperature considering the impact of variability. They first obtain the leakage converged temperature iteratively without considering variation, and then statistically compute the effect of parameter variation. They use their technique to iteratively update power and temperature to estimate the fullchip power, temperature and the probability density function of the temperature. However, a major limitation of their technique is that it is iterative, making it extremely slow ($\approx 158s$ for a 50×50 grid), 121X slower than *VarSim*. Juan et al. [16] use a linear regression-based model to train and predict the temperature profile of a 3D IC in the presence of variability. They use measured values of leakage power for training. However, learning-based methods are very sensitive to input data and do not generalize well when test conditions change. Varipower [17] models power variability at the architectural functional unit level by performing circuit-level Monte Carlo simulations incorporating parameter variation. However, it does not model the effects of temperature effects of variability in power. Yang et al. [4] propose a temporally and spatially adaptive thermal analysis technique that accounts for the temperature dependence of conductivity. However, they do not consider leakage power. Ziabari et al. [18] consider the temperature-dependence of conductivity by using a lookup table to store Green's functions with different conductivities. At runtime, they iteratively update the Green's function until the temperature profile converges. They too have not modeled leakage. In comparison, our approach encompasses the effects of leakage power, variability in leakage as well as temperature dependent conductivity analytically. Process variation along with the variation of conductivity with temperature has never been considered before.

V. Conclusion

In this paper, we propose a fast leakage and variability-aware thermal estimation technique that also captures the temperature dependence of conductivity. We construct a dictionary of leakage power maps that can characterize any variation profile. We then derive a closed-form of the Green's function that captures the temperature dependence of leakage power as well as conductivity. Our approach is several orders of magnitude faster than the state-of-the-art while maintaining an error within 4%.

References

[1] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De, "Parameter variations and impact on circuits and microarchitecture," in *DAC'03*.

[2] S. Mittal, "A survey of architectural techniques for managing process variation," *CSUR*, vol. 48, no. 4, p. 54, 2016.

[3] Y. Zhan, B. Goplen, and S. Sapatnekar, "Electrothermal analysis and optimization techniques for nanoscale integrated circuits," in *ASPDAC'06*.

[4] Y. Yang, Z. Gu, C. Zhu, R. P. Dick, and L. Shang, "ISAC: Integrated space-and-time-adaptive chip-package thermal analysis," *IEEE TCAD*, vol. 26, no. 1, pp. 86–99, 2006.

[5] S. Sarangi, G. Ananthanarayanan, and M. Balakrishnan, "Lightsim: A leakage aware ultrafast temperature simulator," in *ASPDAC'14*.

[6] H. Sultan and S. Sarangi, "A fast leakage aware thermal simulator for 3D chips," in *DATE'17*.

[7] S. Sarangi, B. Greskamp, R. Teodorescu, J. Nakano, A. Tiwari, and J. Torrellas, "Varius: A model of process variation and resulting timing errors for microarchitects," *IEEE TSM*, vol. 21, no. 1, pp. 3–13, 2008.

[8] Y. Liu, R. P. Dick, L. Shang, and H. Yang, "Accurate temperature-dependent integrated circuit leakage power estimation is easy," in *DATE 2007*.

[9] H. Sultan, A. Chauhan, and S. R. Sarangi, "A survey of chip-level thermal simulators," *CSUR*, vol. 52, no. 2, pp. 1–35, 2019.

[10] G. Leung and C. O. Chui, "Variability impact of random dopant fluctuation on nanoscale junctionless FinFETs," *IEEE Electron Device Letters*, vol. 33, no. 6, pp. 767–769, 2012.

[11] M. G. Burzo, P. L. Komarov, and P. Raad, "Non-contact thermal conductivity measurements of p-doped and n-doped gold covered natural and isotopically-pure silicon and their oxides," in *EuroSimE'04*.

[12] Z. Yu, D. Yergeau, and R. W. Dutton, "Full chip thermal simulation," in *ISQED'00*.

[13] R. Zhang, M. R. Stan, and K. Skadron, "Hotspot 6.0: Validation, acceleration and extension," University of Virginia, Tech. Rep., 2015.

[14] A. Ziabari, J.-H. Park, E. K. Ardestani, J. Renau, S.-M. Kang, and A. Shakouri, "Power blurring: Fast static and transient thermal analysis method for packaged integrated circuits and power devices," *IEEE TVLSI*, vol. 22, no. 11, pp. 2366–2379, 2014.

[15] J. Jaffari and M. Anis, "Statistical thermal profile considering process variations: Analysis and applications," *IEEE TCAD*, vol. 27, no. 6, pp. 1027–1040, 2008.

[16] D.-C. Juan, S. Garg, and D. Marculescu, "Statistical thermal evaluation and mitigation techniques for 3D chip-multiprocessors in the presence of process variations," in *DATE'11*.

[17] K. Meng, F. Huebbers, R. Joseph, and Y. Ismail, "Modeling and characterizing power variability in multicore architectures," in *ISPASS'07*.

[18] A. Ziabari, Z. Bian, and A. Shakouri, "Adaptive power blurring techniques to calculate IC temperature profile under large temperature variations," *IMAPS'10*.