Chip Level Thermal Profile Estimation Using On-chip Temperature Sensors

Yufu Zhang, Ankur Srivastava and Mohamed Zahran*
*University of Maryland, *City University of New York*
*{yufuzh, ankurs}@umd.edu, *mzahran@ccny.cuny.edu*

***Abstract*—This paper addresses the problem of chip level thermal profile estimation using runtime temperature sensor readings. We address the challenges of a) availability of only a few thermal sensors with constrained locations (sensors cannot be placed just anywhere) b) random on-chip power density characteristics due to unpredictable workloads and fabrication variability. Firstly we model the random power density as a probability density function. Given this random characteristic and runtime thermal sensor readings, we exploit the correlation between power dissipation of different chip modules to estimate the expected value of temperature at each chip location. Our methods are optimal if the underlying power density has Gaussian nature. We also present a heuristic to generate the chip level thermal profile estimates when the underlying randomness is non-Gaussian. Experimental results indicate that our method generates highly accurate thermal profile estimates of the entire chip at runtime using only a few thermal sensors.**

I. INTRODUCTION

Ever increasing power dissipation has inspired several research directions that essentially focus on efficient estimation of on-chip thermal profiles [1,3,4,5,6]. However, due to unpredictable workloads, fabrication and environmental randomness, on-chip power densities become random as well, thereby making design time estimation of thermal profiles a very hard problem.

In order to address this randomness associated with thermal profile estimation, several researchers have investigated the possibility of placing thermal sensors on-chip [7,8,9]. The readings of these sensors are used to perform dynamic thermal/power management. Although thermal sensors are being placed on-chip, there is no general methodology to recreate the chip level thermal profile from sensor readings. Hence sensor readings are generally used in a very ad-hoc way for performing dynamic thermal/power management.

In this paper, we propose a statistical methodology for recreating the entire chip-level thermal profile during runtime by using only a few on-chip thermal sensors (3 or 4). There are several challenges that have been addressed in this paper including a) the number of on-chip sensors are fairly limited due to area/power limitations b) sensors placement is constrained to areas where there is enough spatial slack c) underlying chip power density is highly random due to unpredictable workload and fabrication randomness. Our methodology addresses these challenges by first understanding the statistical behavior of the power density characteristics (mean, variance, covariance etc.) and then modeling the power density randomness as a probability density function. Given the runtime sensor observations and these random power characteristics (which can be extracted apriori using simulations etc.), our method tries to estimate the temperature at each chip location using methods inspired from signal detection and estimation theory. Essentially our techniques use both the sensor observations and also the fact that there exists a high degree of correlation in power dissipation between different chip modules. By exploiting this correlation, even a few thermal sensors can be used to generate accurate chip level thermal profile estimates. Our technique is optimal (no other methods can generate a better estimation) if the randomness associated with the power density is jointly Gaussian. We also develop an effective heuristic if the power density exhibits non-Gaussian nature.

We experimented with the design of an aggressive out of order processor and found that our technique resulted in extremely accurate estimates of the chip-level thermal profile using only 3 or 4 sensors. In fact our estimates were as much as 100x more accurate compared to performing simple polynomial regression on sensor data. Plus, our methods take only a few minutes to generate results for the entire chip, making it practical for real time thermal/power management applications.

II. PRELIMINARIES: THERMAL PROFILE ESTIMATION

In this section, we briefly outline the steady state thermal profile estimation problem, when given the power density characteristics of the design. The typical structure of an integrated chip includes the silicon on which the circuit is built and the heat sink associated with the silicon within the package (Fig.1).

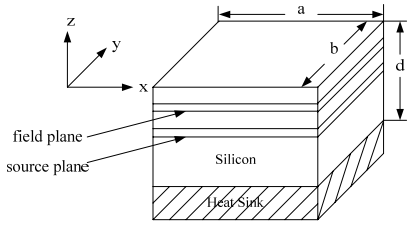

Fig.1 3-D schematic of a silicon chip.

For the sake of brevity in description, we use a simplified model where the silicon has a uniform thermal conductivity. The steady state temperature distribution inside the chip is governed by the following Poisson's equation [1]

$$\nabla^2 T(r) = -\frac{P_d'(r)}{k_s} \quad (1)$$

and subject to the following boundary conditions

$$\begin{cases} \left.\frac{\partial T(r)}{\partial x}\right|_{x=0,a} = \left.\frac{\partial T(r)}{\partial y}\right|_{y=0,b} = \left.\frac{\partial T(r)}{\partial z}\right|_{z=0} = 0 \\ k_S \left.\frac{\partial T(r)}{\partial z}\right|_{z=-d} = h\left(T(r)\big|_{z=-d} - T_0\right) \end{cases}$$

where $\boldsymbol{r} = (x,y,z)$ and $T(\boldsymbol{r})$ is the temperature distribution (in ℃) in the silicon, $P_d'(\boldsymbol{r})$ is the volume power density (in W/m^3) and k_s is the thermal conductivity of silicon (in W/(m·℃)). The vertical and top surfaces are assumed to be adiabatic whereas the interface between the silicon and the heat sink is assumed to be convective with an effective heat transfer coefficient h (in W/(m^2·℃)). The ambient temperature is denoted by T_0 [1]. Note that the boundary conditions highlighted above are specific to the package design (Fig.1). Although different packages with varying heat sink properties would change the boundary conditions, the general nature of the solution to the Poisson's equation will not change.

The partial differential equation (abbr. PDE) (1) can be solved using the method of Green's function. For brevity we do not go into the details of the derivation of this solution which can be found in [1,2,3]. We only show that the solution can be expressed as follows

$$T(r) = T_0 + \int_0^a \int_0^b \int_{-d}^0 G(r,r')P_d'(r')dx'dy'dz'$$

$$G(r,r') = \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} Z'_{mn}(z,z')\cos\left(\frac{m\pi x}{a}\right)\cos\left(\frac{n\pi y}{b}\right) \times \cos\left(\frac{m\pi x'}{a}\right)\cos\left(\frac{n\pi y'}{b}\right)$$

where $G(\boldsymbol{r},\boldsymbol{r'})$ is the Green's function. Here $\boldsymbol{r}=(x,y,z)$ is an arbitrary point **on the field plane where temperature is being estimated** and $\boldsymbol{r'}=(x',y',z')$ is the coordinate of an arbitrary point power source **(located on the source plane)** whose power density is denoted by $P_d(\boldsymbol{r'})$. Also, Z'_{mn} is a function of z and z' and can be obtained from the boundary conditions [1,2,3] (details omitted). For any specific values of z and z', the above solution can be simplified to the 2-dimensional form as follows

$$T(x,y) = T_0 + \int_0^a \int_0^b G(x,y,x',y')P_d(x',y')dx'dy' \quad (2)$$

$$G(x,y,x',y') = \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} C_{mn}\cos\left(\frac{m\pi x}{a}\right)\cos\left(\frac{n\pi y}{b}\right) \times \cos\left(\frac{m\pi x'}{a}\right)\cos\left(\frac{n\pi y'}{b}\right)$$

Here $T(x,y)$ is the temperature at an arbitrary point on the field plane and $P_d(x',y')$is the 2-D power density function (in W/m^2) for the source plane. Constants C_{mn} can be derived from $Z'_{mn}(z, z')$. Note that the only unknown in these equations is the power density function P_d which depends on the layout, device dimensions, switching activity etc. Several researchers have investigated the problem of efficient thermal simulation at design time using prior knowledge of the power density of a design [1,4,5,6]. Here we rely on the method proposed in [1] which we describe briefly now.

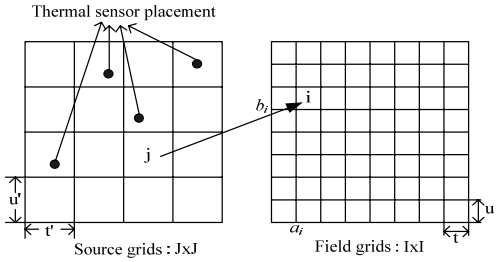

Fig.2 Source grids and field grids.

The source region (source of power dissipation) and field region (surface whose thermal profile is of interest) are split into JxJ and IxI grids respectively (Fig.2). Each source grid j is assumed to have a constant power density $P_d(j)$ [1] (note that these grids could be arbitrarily small).

We are interested in measuring the average temperature in the *i-th* grid of the field plane. Let a_i and b_i be the (x, y) locations for the bottom left corner of the *i-th* grid (origin being the bottom left corner of the chip). Then the average temperature in the *i-th* grid is given by

$$\overline{T}(i) = \frac{1}{(t)(u)} \int_{a_i}^{a_i+t} \int_{b_i}^{b_i+u} T(x,y)dxdy$$

where t and u are the width and height of each grid in the field plane ($t = a/I$, $u = b/I$). Note that a, b are the chip dimensions. Since the power density in the source plane is assumed to be a constant $P_d(j)$ in each grid j, we can substitute (2) into the above equation and simplify it to

$$\overline{T}(i) = T_0 + \sum_j \frac{P_d(j)}{(t)(u)} \int_{a_i}^{a_i+t}\int_{b_i}^{b_i+u}\int_{a_j}^{a_j+t'}\int_{b_j}^{b_j+u'} G(x,y,x',y')dxdydx'dy' \quad (3)$$

Here t' and u' are the grid dimensions for the source plane (see Fig.2). G is the Green's function as shown before. This simplification (of approximating power density as piecewise constant in grids) could be used to speed up the chip level thermal simulation process (please see [1] for details).

III. Temperature Sensor Driven Chip Level Thermal Profile Estimation

A. Problem Description

Recently much attention has been provided on the issue of placing on-chip thermal sensors during design time and exploiting the sensed temperature information during runtime to perform dynamic thermal/power management [7,8,9]. The central motivation behind such approaches is the growing need for sophisticated runtime management techniques to control the detrimental effects of unpredictable thermal hotspots. A key challenge, which is also the focus of this paper, is how we can systematically reconstruct the chip-level thermal profile using the thermal sensor observations. Unfortunately, this problem is not trivial. The challenges and motivations are discussed below.

1) The total number of on-chip thermal sensors is limited

Table.1. Power Density Characteristics of Different Modules.

Correlation	branch predictor	rename	Instruction Window	load/store queue	register file	ALU	Icache	Dcache
Dcache	0.42	0.54	0.69	0.98	0.42	0.47	0.56	
Icache	0.66	0.97	0.89	0.61	0.58	0.68		
ALU	0.36	0.74	0.90	0.51	0.93			
register file	0.13	0.63	0.84	0.47				
load/store queue	0.45	0.58	0.74					
Instruction Window	0.54	0.91						
rename	0.61							
Mean ($x10^5$ W/m^2)	0.34	4.33	12.48	1.09	3.88	3.30	4.71	1.03
Std_Dev ($x10^5$ W/m^2)	0.18	1.68	4.21	0.42	1.49	1.33	1.87	0.41

and thus cannot cover all areas of the chip. Indeed, if we have the freedom to place infinite number of sensors at all locations, there would be no need for investigating the thermal profile estimation problem since the temperature at any location is accurately known. Unfortunately, since thermal sensors come with a cost in terms of area and power, we don't have the luxury to place infinite number of sensors under today's ever pushing design constraints. In addition, the sensors can not go into the pre-designed on-chip IP cores should that be the critical area of interest. The thermal profile estimation problem is thus highly constrained by the number and the locations of the thermal sensors. The key challenge here is to take the readings of only a few temperature sensors and re-create the chip level thermal profile as accurately as possible.

2) To effectively solve the thermal profile estimation problem one has to account for the underlying randomness in the power/thermal behavior. Let us consider equation (3) which is used to generate thermal profile of the chip while approximating the power density as piecewise constant $P_d(j)$ in each grid j. Assuming that the silicon thermal conductivity and heat transfer coefficient are constants, the only unknown in equation (3) is $P_d(j)$. If we knew $P_d(j)$ accurately during design time, then we would not need any thermal sensors at all since the entire thermal profile could be generated during design time. However, the power density itself is a strong function in the application workload (which is unpredictable until runtime), the device parameters (which are random due to fabrication variability), the environment (such as the ambient temperature change and power supply fluctuation). All these unpredictable factors make the power density a random quantity in real practice, hence the thermal profile becomes a random quantity as well. The key challenge here is to develop a probabilistic methodology to account for the random nature of power density and use it to effectively estimate the chip-level thermal profile, when given a few temperature sensor observations at runtime.

3) As discussed above, the desired methodology is required to account for the fact that we only have a few sensors and the underlying power density is random. An important fact about this randomness is that different parts of the chip can exhibit highly correlated power behavior. In order to quantify this correlation we used Wattch [12] with alpha binary to generate the power consumption results for different parts of a processor. We simulated a high performance aggressive out-of-order processor with pipeline width of 8 instructions and an instruction window of 128 instructions. Level 1 caches (both instruction cache and data cache) are 32KB 4-way set associative. All the caches in the hierarchy are using LRU replacement policy and a block size of 64 bytes. For benchmarks, we simulated all the SPEC 2000 CPU benchmark suite [13] compiled with the default parameters provided with the suite. We bypassed the startup part, based on simpoint [14], and simulated a representative 250M instructions for each of the benchmarks.

The correlation in the power dissipation of different architectural components is presented in Table 1. It can be seen that different components have a high degree of correlation in power dissipation. For example, the correlation between instruction window and register rename modules is as high as 0.91. This means that for a given application running on the processor, if there is high power density in instruction window module then the power density for register rename module is very likely to be high as well.

The random fabrication parameters (like channel length, oxide thickness etc.) also exhibit strong spatial correlation. Hence it can be correctly argued that the random power density in different parts of the chip exhibits strong correlation.

Thus, in principle, power density in some parts of the chip can be used to predict the power density in other parts. For this reason, even if the total number of thermal sensors is small, reasonably accurate thermal profile estimation could be obtained by exploiting this correlation. It is reasonable to assume that the probabilistic characteristics (including mean, variance, covariance etc.) of the chip power density could be generated during design time. Note that these random characteristics would be a strong function of the device parameters, application workload etc.

All in all, the key challenge is to exploit the temperature readings of a few thermal sensors, along with knowledge of the random characteristics of chip power density to generate accurate thermal profile estimates. As would be highlighted in the experimental results, ignoring the correlation information leads to highly inaccurate estimates of thermal profile even with relatively more sensors whereas exploiting the correlation information enables one to generate high fidelity estimates with only a few thermal sensor readings.

B. Modeling Randomness in Power Density.

As highlighted in figure 2, we discretize the chip into JxJ grids each with an average power density $P_d(j)$.

$$P_d(j) = f\left[SW(j), V_{th}(j), L_{eff}(j), V_0(j)\cdots\right] \quad (4)$$

Equation (4) highlights the general dependence of the

power density variable in *j-th* grid upon its local switching activity $SW(j)$, device parameters like threshold voltage $V_{th}(j)$ and the effective channel length $L_{eff}(j)$, the environment factor like supply voltage $V_0(j)$ etc. Because of these random parameters $P_d(j)$ becomes random as well. Various simulation and analytical techniques can be carried out to extract the random characteristics of the power density variables including the mean, variance, covariance etc.

For example the part of the chip that corresponds to the instruction issue register will exhibit similar trends in power density change (high correlation) with the functional modules. Similarly, the fabrication variability of devices in close vicinity would be correlated as well thereby leading to correlation in their power density. Knowing the correlation between the various parameters in (4) would allow us to estimate the correlation between different JxJ $P_d(j)$ variables. Towards this purpose, extensive work has been carried out on the problem of modeling the correlated fabrication variability [10]. Extensive simulation could also be used to estimate the correlation between the switching activity, leakage etc. of different grids (similar to the simulation that we used to extract the correlation data shown in Table 1). All this could be used to extract the correlation between the various power density variables. We do not go any further into the details of how this modeling is done since the focus of this paper is different. But we assume that the JxJ random power density variables could be represented as a vector $\boldsymbol{P}$ with mean $\boldsymbol{\mu_p}$ and covariance matrix $\boldsymbol{\Sigma_{pp}}$ (*note that we do not assume the power densities follow any specific distribution, we just assume that the mean and covariance are known*).

Though the values obtained here may not be completely accurate, the key idea is that these approximate values, when combined with the runtime sensor observations, could significantly increase the accuracy of the thermal profile estimation.

C. Formal Problem Statement

Let us suppose we have *n* on-chip temperature sensors with known locations (as illustrated in figure 2)

$$s_1 = (x_1, y_1),\ s_2 = (x_2, y_2),\ \ldots\ldots,\ s_n = (x_n, y_n)$$

which provide a vector of *n* temperature reading $\boldsymbol{T_s} = \{T_1, T_2, \ldots, T_n\}$. As mentioned in sub-section B we also assume that we have a prior knowledge of the random characteristics (mean $\boldsymbol{\mu_p}$ and covariance matrix $\boldsymbol{\Sigma_{pp}}$) of the JxJ power density variables. The problem is to find the conditional expectation of the temperature at all IxI grid locations given the thermal sensor readings and the random power density characteristics.

$$E(\vec{T} \mid \vec{T}_s, \vec{\mu}_p, \Sigma_{pp}) \tag{5}$$

Note that here we are exploiting both the sensor readings and the probabilistic characteristics of the power density variables to provide accurate estimation. Exploiting the latter would enable us to use less number of sensors while providing high fidelity chip level thermal estimates.

D. Estimation Methodology

D.1 Optimal Solution for jointly Gaussian Distribution

Now we present an algorithm that would allow us to solve the problem posed by (5). In this section we consider the case where the power density variables are jointly Gaussian in nature.

Let us consider (3) once again which could be rewritten as follows (by substituting the Green's function $G(x, y, x', y')$ into equation 3).

$$\overline{T}(i) = T_0 + \sum_j \frac{P_d(j)}{(t)(u)} \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} C_{mn} \int_{a_i}^{a_i+t} \cos\left(\frac{m\pi x}{a}\right) dx \times \int_{b_i}^{b_i+u} \cos\left(\frac{n\pi y}{b}\right) dy \int_{a_j}^{a_j+t'} \cos\left(\frac{m\pi x'}{a}\right) dx' \int_{b_j}^{b_j+u'} \cos\left(\frac{n\pi y'}{b}\right) dy'$$

The above equation can also be written as

$$T(i) = T_0 + \sum_j \alpha_{i,j} P_d(j) \tag{6}$$

where

$$\alpha_{i,j} = \frac{1}{(t)(u)} \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} C_{mn} \int_{a_i}^{a_i+t} \cos\left(\frac{m\pi x}{a}\right) dx \times \int_{b_i}^{b_i+u} \cos\left(\frac{n\pi y}{b}\right) dy \int_{a_j}^{a_j+t'} \cos\left(\frac{m\pi x'}{a}\right) dx' \int_{b_j}^{b_j+u'} \cos\left(\frac{n\pi y'}{b}\right) dy' \tag{7}$$

As illustrated in [1], the infinite summation in variables *m* and *n* can be well approximated by a finite summation where variables *m* and *n* are bounded by *M* and *N*. In [1], it was demonstrated that setting *M* and *N* to around 64 leads to satisfactory accuracy in thermal estimation. Hence parameters $\{\alpha_{i,j}\}$ are constants for a given grid *i* of thermal interest and a given power density grid *j* with power density $P_d(j)$.

Let $\boldsymbol{T} = \{T(1), T(2), \ldots T(i), \ldots\}$ be the vector that represents the average temperature at all IxI grid locations and $\boldsymbol{P} = \{P_d(1), P_d(2), \ldots P_d(j), \ldots\}$ be the vector of all power density variables at the JxJ grid locations (see figure 2), equation (6) can be rewritten in the matrix form.

$$\vec{T} = A\vec{P} \tag{8}$$

$$A = \begin{pmatrix} \alpha_{1,1} & \alpha_{1,2} & \cdots & \alpha_{1,J\times J} \\ \alpha_{2,1} & \alpha_{2,2} & \cdots & \alpha_{2,J\times J} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{I\times I,1} & \alpha_{I\times I,2} & \cdots & \alpha_{I\times I,J\times J} \end{pmatrix}$$

This simple transformation highlights that indeed the dependence between temperature and power density can be approximated as a linear transformation [1]. Since the $\boldsymbol{P}$ vector is random with mean $\boldsymbol{\mu_p}$ and covariance matrix $\boldsymbol{\Sigma_{pp}}$, the $\boldsymbol{T}$ vector is random too with mean and covariance [11]

$$\overrightarrow{\mu_T} = A\overrightarrow{\mu_P}, \quad \Sigma_{TT} = A\Sigma_{PP}A^T$$

Now that we know the relationship between $\boldsymbol{T}$ and $\boldsymbol{P}$ as linear, estimating the conditional temperature vector can be done by estimating conditional power density vector, i.e.

$$E(\vec{T} \mid \overrightarrow{T_s}) = AE(\vec{P} \mid \overrightarrow{T_s}) \tag{9}$$

$$E(\vec{T} \mid \overrightarrow{T_s}) \xrightarrow{simplify} E(\vec{P} \mid \overrightarrow{T_s})$$

The above equation is easily obtained by rewriting equation

(8) using the expected values. Note for brevity we have omitted $\boldsymbol{\mu_p}$ and $\boldsymbol{\Sigma_{pp}}$ from the notation. Here the problem of estimating the thermal profile has been simplified to estimating the expected value of the power density vector given sensor readings and the power density correlation information. Thus, the problem posed by equation (5) can be solved optimally by finding the optimal estimator for the following.

$$E(\vec{P}\,|\,\vec{T}_s) \tag{10}$$

The advantage of first solving (10) and then (5) is that in general the power density correlation is easier to evaluate due to the reason mentioned in sub-section B. After finding out the expected value of the power density vector given sensor readings and the power density correlation information, it could then be used to generate the temperature at any location of interest in the chip using equation (9). Hence we don't need to generate the entire thermal profile if we don't want to and could focus only on the hotspot areas, leading to computationally efficient thermal estimation in critical areas. The general flow of our method is shown in Fig.3.

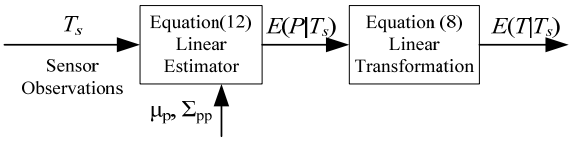

Fig.3 Estimation algorithm flow

Now we present an algorithm to solve (10) when the power density vector is jointly Gaussian. As mentioned earlier, $\boldsymbol{T_s}$ is the vector of sensor readings. Since we know the location of the sensors, we can always represent $\boldsymbol{T_s}$ as follows.

$$\vec{T_s} = A_s \vec{P} \qquad \overrightarrow{\mu_{Ts}} = A_s \overrightarrow{\mu_P} \tag{11}$$

where A_s is a matrix with elements $\{\alpha_{i,j}^{(s)}\}$ which can be derived similar to matrix A. More specifically, because we know the locations of all thermal sensors, we know their corresponding field grid and source grid, i.e. the indices i and j. By substituting these two values into equation (7), we can generate all elements $\{\alpha_{i,j}^{(s)}\}$ of matrix A_s.

Now because the power density vector $\boldsymbol{P}$ is random, $\boldsymbol{T_s}$ is a random vector also. Using the current sensor observations, the power density correlation information and the underlying relationship between these two posed by equation (11), we would like to estimate the expected value of the power density vector $\boldsymbol{P}$ (as shown in equation 10). For this we present a linear estimator of the $\boldsymbol{P}$ vector which is a linear function of the temperature sensor observations [11]. The expression for this estimator is as follows

$$E(\vec{P}\,|\,\vec{T_s}) = \overrightarrow{\mu_P} + \Sigma_{pp} A_s^T (A_s \Sigma_{pp} A_s^T)^{-1} (\vec{T_s} - A_s \overrightarrow{\mu_p}) \tag{12}$$

Given the sensor observation vector $\boldsymbol{T_s}$, matrix A_s and $\boldsymbol{\mu_p}$, $\boldsymbol{\Sigma_{pp}}$, we can use equation (12) to estimate the quantity in (10). Note that this is a linear estimator (linear function of observations $\boldsymbol{T_s}$).

Theorem-1: The estimator in equation (12) is an optimal linear estimator when the randomness associated with the power density variables is jointly Gaussian with mean $\boldsymbol{\mu_p}$ and covariance $\boldsymbol{\Sigma_{pp}}$.

Proof: Due to limited space we omit the proof. The interested reader is referred to [11] for details.

Theorem-1 is the crux of our work. It essentially states that for estimating the quantity of equation (10), no other linear estimator can perform better than the one in equation (12). In this sense our methodology gives the optimal solution for the thermal profile estimation problem for the jointly Gaussian case.

D.2 Heuristic Solution for Non-Gaussian Distribution

When the underlying nature of the power density variables does not exhibit a Gaussian distribution, the estimator presented in equation (12) is not optimal anymore. However, for the non-Gaussian power density distribution, a closed form solution for the estimation problem is hard to find, sometimes even impossible. In general, real data is not far from a Gaussian distribution. Under these scenarios, we can approximate the actual joint probability of the power density vector with a Gaussian distribution. This enables us to use the estimator of equation (12) to approximate the result of equation (10). In order to do that the mean, variance and covariance of the power density vector could be matched with the respective mean, variance, covariance of a Gaussian joint density function (also known as moment matching). Following which the result of equation (12) could be directly applied as an approximate solution. The accuracy of this heuristic algorithm are shown in the next section.

IV. Experimental Results

In this section we present the results obtained using our estimation methodology. We simulated a high performance aggressive out-of-order processor with pipeline width of 8 instructions and an instruction window of 128 instructions. Level 1 caches (both instruction and data) are 32KB 4-way set associative. The shared level 2 cache is 1MB and 8-way set associative. All the caches in the hierarchy are using LRU replacement policy and a block size of 64 bytes. Firstly we simulated this architecture using Wattch [12] and all the SPEC 2000 CPU benchmark suite [13]. The power distribution data was generated for each functional module of the processor for different benchmarks. Using this data, we extracted the mean, variance, covariance etc. for different modules (data illustrated in table 1). Note that this variance in power dissipation occurs due to the fact that during design time we don't know exactly which mix of benchmarks we will end up executing on the processor. We used the techniques described in III.D.2 to approximate the data with a Gaussian distribution. We created an ad-hoc floorplan of the processor with dimensions 2mm×2mm×0.5mm. The thermal conductivity k_s was assumed as 148 W/(m·℃) and the effective heat transfer coefficient h is

set to be 8700 W/(m2·℃) (this value is consistent with the value used in [1]).

Then we tried to address the problem of estimating the thermal profile of the chip, given the random power density characteristics (that we learnt from the simulation) and a few sensor observations when a specific SPEC benchmark was executed. For this specific benchmark, we calculated the real power density map (through Wattch simulations) and therefore the real thermal profile which was used as a basis to compare results. Then we placed a few sensors in modules like "I-Cache", "Instruction Window" and "ALU" etc. arbitrarily and noted the temperature at these points. Then we used the presented theory to estimate the thermal profile. We also used regression to fit a bivariate 2nd order polynomial onto the sensor data. This fit was used to compare the quality of our estimate with that of polynomial regression on sensor data.

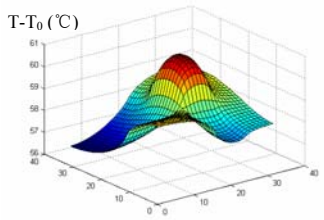

Fig.4 Real thermal Profile

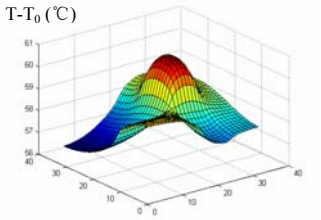

Fig.5 Our estimation

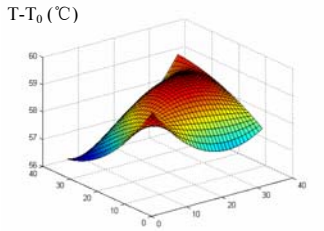

Fig.6 Polynomial fit

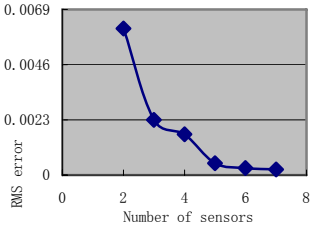

Fig.7 RMS error decreases when increasing the # of sensors

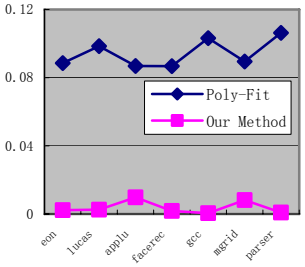

Fig.8 RMS error for different benchmarks

Figure 4,5,6 show the real thermal map, estimated thermal map (using our technique) and the one generated using polynomial regression for the EON benchmark. We used only three sensors in our technique whereas the regression based technique needed 16 sensors. It is obvious that even with much less sensors, the accuracy of our estimation methodology far exceeds that of the regression method. We also highlight the dependence between RMS error and number of sensors for our method (figure 7). As expected, as the number of sensors increases, the error decreases.

Figure 8 compares the RMS error obtained using both methods for various SPEC benchmarks. In comparison to polynomial regression our approach has significantly smaller error with even fewer sensors. These results clearly highlight the effectiveness of our method.

V. Conclusion

In this paper, we addressed the problem of estimating the chip-level thermal profile at runtime using only a few on-chip sensor observations. The underlying random nature of the thermal/power characteristics are well accounted for. We proposed two methodologies for solving this problem: 1) When the probability density function governing the power density variables exhibits jointly Gaussian distribution, we presented an optimal estimator for recreating the chip-level thermal profile. 2) When such a prior probabilistic property does not exist we presented a heuristic algorithm to approximate the optimal solution. The experimental results demonstrated significant advantage (as much as 100x more accurate) of our method over a simple 2nd order polynomial regression strategy. Thus making our method an effective approach in guiding the dynamic thermal/power management.

* This work was partly supported by NSF grant CCF 0728969.

References

[1] Yong Zhan and Sachin S.Sapatnekar, "High-Effciency Green Function-Based Thermal Simulation Algorithms", *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 26, pp. 1661, Sept. 2007.

[2] R. Gharpurey and R. G. Meyer, "Modeling and analysis of substrate coupling in integrated circuits", *IEEE J. Solid-State Circuits*, vol. 31, pp. 344, Mar. 1996.

[3] A. M. Niknejad, R. Gharpurey and R. G. Meyer, "Numerically Stable Green Function for Modeling and Analysis of Substrate Coupling in Integrated Circuits", *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 17, pp. 305, Apr. 1998.

[4] T. Y. Wang and C. P. Chen, "3-D Thermal-ADI: A Linear-time Chip Level Transient Thermal Simulator", *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 21, pp. 1434, Dec. 2002.

[5] P. Li , L. T. Pileggi , M. Asheghi and R. Chandra, "Efficient Full-chip Thermal Modeling and Analysis", *Proc. IEEE/ACM Int. Conf. Comput.-Aided Des., Dig. Tech. Papers*, Nov. 2004, p. 319.

[6] B. Wang and P. Mazumder, "Fast Thermal Analysis for VLSI Circuits via Semi-analytical Green"s function in Multi-layer Materials", *Proc. IEEE Int. Symp. Circuits and Syst.*, pp. 409, May 2004.

[7] R. Mukherjee, S. Mondal and S. Ogrenci Memik, "A Sensor Distribution Algorithm for FPGAs with Minimal Dynamic Reconfiguration Overhead", *Proc. IEEE Int. Conf. on Engineering of Reconfigurable Systems and Algorithms*, pp. 437, Jun. 2006.

[8] Qikai Chen, M. Meterelliyoz and K. Roy, "A CMOS thermal sensor and its applications in temperature adaptive design", *Proc. IEEE Int. Symp. Quality Electronic Design,* pp. 6, March 2006.

[9] R. Mukherjee and S. Ogrenci Memik, "Systematic Temperature Sensor Allocation and Placement for Microprocessors", *Proc. IEEE Design Automation Conference,* pp. 542, Jul. 2006

[10] Hongliang Chang and Sachin S. Sapatnekar, "Statistical Timing Analysis Considering Spatial Correlations Using a Single Pert-Like Traversal", *Proc. IEEE Int. Conference on Computer Aided Design*, pp. 621, Mar.2003

[11] H. Vincent Poor, "An Introduction to Signal Detection and Estimation", 2nd Ed., Springer 1994.

[12] David Brooks, Vivek Tiwari, and Margaret Martonosi. Wattch, "A Framework for Architectural- Level Power Analysis and Optimizations", *27th International Symposium on Computer Architecture*, June, 2000.

[13] Greg Hamerly, Erez Perelman, Jeremy Lau, and Brad Calder, "SimPoint 3.0: Faster and More Flexible Program Analysis", *Workshop on Modeling, Benchmarking and Simulation*, June 2005.